



República Federativa do Brasil
Ministério da Economia
Instituto Nacional da Propriedade Industrial

(21) BR 112019019210-7 A2



(22) Data do Depósito: 22/03/2018

(43) Data da Publicação Nacional: 14/04/2020

(54) **Título:** INFORMAÇÃO DE VETOR DE MOVIMENTO DE RESTRIÇÃO DERIVADA POR DERIVAÇÃO DE VETOR DE MOVIMENTO DE LADO DE DECODIFICADOR

(51) **Int. Cl.:** H04N 19/577.

(30) **Prioridade Unionista:** 21/03/2018 US 15/927,854; 22/03/2017 US 62/475,177.

(71) **Depositante(es):** QUALCOMM INCORPORATED.

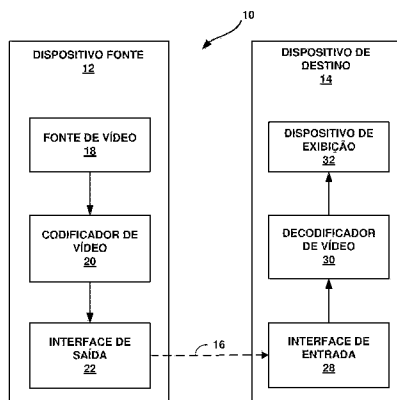
(72) **Inventor(es):** MARTA KARCZEWICZ; YI-WEN CHEN; WEI-JUNG CHIEN; HSIAO-CHIANG CHUANG; XIANG LI; JIANLE CHEN.

(86) **Pedido PCT:** PCT US2018023761 de 22/03/2018

(87) **Publicação PCT:** WO 2018/175720 de 27/09/2018

(85) **Data da Fase Nacional:** 16/09/2019

(57) **Resumo:** As técnicas relacionadas com a derivação de vetor de movimento de lado de decodificador (DMVD) são descritas. Por exemplo, essa descrição descreve as técnicas relacionadas com aplicar uma ou mais restrições à informação de movimento, tal como um vetor de movimento (MV) derivado pela DMVD, e/ou uma diferença de MV entre um MV inicial e um MV derivado pela DMVD. Quando a restrição é aplicada à DMVD, em determinados exemplos, apenas a informação de movimento derivado, que corresponde à restrição, é considerada como informação de movimento válida. As condições podem ser impostas às restrições.



"INFORMAÇÃO DE VETOR DE MOVIMENTO DE RESTRIÇÃO DERIVADA POR DERIVAÇÃO DE VETOR DE MOVIMENTO DE LADO DE DECODIFICADOR"

[01] Esse pedido reivindica os benefícios do pedido de patente provisório U.S. No. 62/475,177, depositado em 22 de março de 2017, que é incorporado aqui por referência em sua totalidade.

CAMPO TÉCNICO

[02] Essa descrição refere-se à codificação de vídeo.

FUNDAMENTOS

[03] Capacidades de vídeo digital podem ser incorporadas a uma ampla faixa de dispositivos, incluindo televisores digitais, sistemas de difusão direta digital, sistemas de difusão sem fio, assistentes digitais pessoais (PDAs), computadores laptop e desktop, computadores tablet, leitoras e-book, câmeras digitais, dispositivos de gravação digital, aparelhos de reprodução de mídia digital, dispositivos de jogos de vídeo, consoles de jogos de vídeo, telefones celulares ou de rádio via satélite, os chamados "smartphones", dispositivos de vídeo teleconferência, dispositivos de sequenciamento de vídeo, e similares. Os dispositivos de vídeo digital implementam as técnicas de codificação de vídeo, tal como as descritas nos padrões definidos por ITU-T H. 261, ISO/IEC MPEG-1 Visual, ITU-T H.262 ou ISO/IEC MPEG-2 Visual, ITU-T H.263, ITU-T H.264/MPEG-4, Parte 10, Codificação de Vídeo Avançada (AVC), ITU-T H.265/Codificação de Vídeo com Alta Eficiência (HEVC), e extensões de tais padrões, tal como extensões de Codificação de Vídeo Escalonável (SVC) e Codificação de Vídeo de Múltiplas Visualizações (MVC). Os dispositivos de

vídeo podem transmitir, receber, codificar, decodificar e/ou armazenar informação de vídeo digital de forma mais eficiente pela implementação de tais técnicas de codificação de vídeo.

[04] As técnicas de codificação de vídeo incluem previsão espacial (intraimagem) e/ou previsão temporal (interimagem) para reduzir ou remover a redundância inerente às sequências de vídeo. Para codificação de vídeo com base em bloco, uma fatia de vídeo (por exemplo, uma imagem de vídeo ou uma parte de uma imagem de vídeo) pode ser dividida em blocos de vídeo, que também podem ser referidos como unidades de árvore de codificação (CTUs), unidades de codificação (CUs) e/ou nós de codificação. Os blocos de vídeo em uma fatia intracodificada (I) de uma imagem são codificados utilizando-se previsão espacial com relação às amostras de referência nos blocos vizinhos na mesma imagem. Blocos de vídeo em uma fatia intercodificada (P ou B) de uma imagem podem utilizar a previsão espacial com relação às amostras de referência nos blocos vizinhos na mesma imagem ou previsão temporal com relação às amostras de referência em outras imagens de referência. As imagens podem ser referidas como quadros, e as imagens de referência podem ser referidas como quadros de referência.

[05] A previsão espacial ou temporal resulta em um bloco de previsão para um bloco a ser codificado. Os dados residuais representam as diferenças de pixel entre o bloco original a ser codificado e o bloco de previsão. Um bloco intercodificado é codificado de acordo com um vetor de movimento que aponta para um bloco de amostras de referência que formam o bloco de previsão, e os dados

residuais indicando a diferença entre o bloco codificado e o bloco de previsão. Um bloco intracodificado é codificado de acordo com um modo de intracodificação e os dados residuais. Para a compressão adicional, os dados residuais podem ser transformados de domínio de pixel em domínio de transformação, resultando em coeficientes de transformação residuais, que podem, então, ser quantizados. Os coeficientes de transformação quantizados, dispostos inicialmente em um conjunto bidimensional, podem ser digitalizados a fim de produzirem um vetor unidimensional de coeficientes de transformação, e a codificação por entropia pode ser aplicada para se alcançar ainda mais compressão.

SUMÁRIO

[06] Em geral, essa descrição descreve as técnicas relacionadas com a derivação de vetor de movimento de lado de decodificador (DMVD). Por exemplo, essa descrição descreve as técnicas relacionadas com a aplicação de uma ou mais restrições à informação de movimento, tal como um vetor de movimento (MV) derivado pela DMVD, e/ou uma diferença MV entre um MV inicial e um MV derivado por DMVD. Essas técnicas podem ser aplicadas a qualquer um dos codecs de vídeo existentes, tal como HEVC (Codificação de Vídeo de Alta Eficiência) e/ou pode ser uma ferramenta de codificação eficiente em quaisquer padrões de codificação de vídeo no futuro. Quando a restrição é aplicada aos MVs e/ou diferenças de MV derivadas com DMVD, em determinados exemplos, apenas a informação de movimento derivado, que corresponde à restrição, é considerada informação de movimento válida. Dessa forma, a lista de MVs e o MV final

selecionado podem ser diferentes no caso de a restrição não ter sido utilizada, o que pode aumentar a eficiência da codificação e/ou reduzir a taxa de bit. Em alguns exemplos, a restrição inclui se os MVs derivados são simétricos, se as diferenças de MV entre os MVs iniciais e os MVs derivados pelos métodos DMVD são simétricos, se os MVs derivados são antissimétricos, se as diferenças de MV entre os MVs iniciais e os MVs derivados por métodos DMVD são antissimétricas. Em alguns exemplos, a ativação das restrições de MV/diferenças de MV simétricas/antissimétricas é sinalizada explicitamente. Em outros exemplos, as restrições podem ser determinadas implicitamente de acordo com alguma informação codificada. Em exemplos adicionais, as condições podem ser impostas às restrições, de modo que a restrição só se aplique em determinados exemplos.

[07] Em um exemplo, um método de decodificação de dados de vídeo a partir de uma sequência de bits de vídeo codificada, o método compreendendo gerar, utilizando a derivação de vetor de movimento de lado de decodificador (DMVD), uma pluralidade de vetores de movimento derivados, determinar um primeiro vetor de movimento derivado e um segundo vetor de movimento derivado a partir da pluralidade de vetores de movimento derivados, com base em uma métrica de custo, que compreende determinar que o primeiro vetor de movimento derivado e o segundo vetor de movimento derivado satisfazem pelo menos uma restrição em uma relação entre o primeiro vetor de movimento derivado e o segundo vetor de movimento derivado, a relação compreendendo: o primeiro vetor de movimento derivado e o segundo vetor de movimento

derivado possuindo uma diferença de vetor de movimento simétrica, o primeiro vetor de movimento derivado e o segundo vetor de movimento derivado possuindo uma diferença de vetor de movimento pseudo-simétrico, o primeiro vetor de movimento derivado e o segundo vetor de movimento derivado sendo antissimétricos, o primeiro vetor de movimento derivado e o segundo vetor de movimento derivado possuindo uma diferença de vetor de movimento antissimétrica, ou o primeiro vetor de movimento derivado e o segundo vetor de movimento derivado possuindo uma diferença de vetor de movimento antissimétrico, e decodificando um bloco atual utilizando o primeiro vetor de movimento derivado e o segundo vetor de movimento derivado determinados, utilizando a previsão dupla.

[08] Em outro exemplo, um método de codificação de dados de vídeo, o método compreendendo gerar, utilizando uma derivação de vetor de movimento de lado de decodificador (DMVD), uma pluralidade de vetores de movimento derivados, determinar um primeiro vetor de movimento derivado e um segundo vetor de movimento derivado a partir da pluralidade de vetores de movimento derivados com base em uma métrica de custo, compreendendo: determinar que o primeiro vetor de movimento derivado e o segundo vetor de movimento derivado satisfazem pelo menos uma restrição em uma relação entre o primeiro vetor de movimento derivado e o segundo vetor de movimento derivado, a relação compreendendo: o primeiro vetor de movimento derivado e o segundo vetor de movimento derivado possuindo uma diferença de vetor de movimento simétrica, o primeiro vetor de movimento derivado e o segundo vetor de movimento

derivado possuindo uma diferença de vetor de movimento pseudo-simétrica, o primeiro vetor de movimento derivado e o segundo vetor de movimento derivado sendo antissimétricos, o primeiro vetor de movimento derivado e o segundo vetor de movimento derivado possuindo uma diferença de vetor de movimento antissimétrica, ou o primeiro vetor de movimento derivado e o segundo vetor de movimento derivado possuindo uma diferença de vetor de movimento pseudo-antissimétrica, e codificando um bloco atual utilizando o primeiro vetor de movimento derivado e o segundo vetor de movimento derivado determinados, utilizando a previsão dupla.

[09] Em outro exemplo, um aparelho configurado para decodificar os dados de vídeo a partir de uma sequência de bits de vídeo codificada, o aparelho compreendendo uma memória configurada para armazenar os dados de vídeo, e um ou mais processadores em comunicação com a memória, os um ou mais processadores configurados para gerar, utilizando a derivação de vetor de movimento de lado de decodificador (DMVD), uma pluralidade de vetores de movimento derivados, determinar um primeiro vetor de movimento derivado e um segundo vetor de movimento derivado a partir da pluralidade de vetores de movimento derivados com base em uma métrica de custo, compreendendo determinar que o primeiro vetor de movimento derivado e o segundo vetor de movimento derivado satisfazem pelo menos uma restrição em uma relação entre o primeiro vetor de movimento derivado e o segundo vetor de movimento derivado, a relação compreendendo o primeiro vetor de movimento derivado e o segundo vetor de movimento derivado possuírem

uma diferença de vetor de movimento simétrica, o primeiro vetor de movimento derivado e o segundo vetor de movimento derivado possuírem uma diferença de vetor de movimento pseudo-simétrica, o primeiro vetor de movimento derivado e o segundo vetor de movimento derivados serem antissimétricos, o primeiro vetor de movimento derivado e o segundo vetor de movimento derivado possuírem uma diferença de vetor de movimento antissimétrica, ou o primeiro vetor de movimento derivado e o segundo vetor de movimento derivado possuírem uma diferença de vetor de movimento antissimétrica, e decodificar um bloco atual utilizando o primeiro vetor de movimento derivado e o segundo vetor de movimento derivado determinados, utilizando a previsão dupla.

[010] Em outro exemplo, um aparelho configurado para codificar os dados de vídeo, o aparelho compreendendo uma memória configurada para armazenar dados de vídeo, e um ou mais processadores em comunicação com a memória, os um ou mais processadores configurados para gerar, utilizando a derivação de vetor de movimento de lado de decodificador (DMVD), uma pluralidade de vetores de movimento derivados, determinar um primeiro vetor de movimento derivado e um segundo vetor de movimento derivado a partir da pluralidade de vetores de movimento derivado com base em uma métrica de custo, compreendendo uma determinação de que o primeiro vetor de movimento derivado e o segundo vetor de movimento derivado satisfazem a pelo menos uma restrição em uma relação entre o primeiro vetor de movimento derivado e o segundo vetor de movimento derivado, a relação compreendendo o primeiro vetor de

movimento derivado e o segundo vetor de movimento derivado possuírem uma diferença de vetor de movimento simétrica, o primeiro vetor de movimento derivado e o segundo vetor de movimento derivado possuírem uma diferença de vetor de movimento pseudo-simétrica, o primeiro vetor de movimento derivado e o segundo vetor de movimento derivado serem antissimétricos, o primeiro vetor de movimento derivado e o segundo vetor de movimento derivado possuírem uma diferença de vetor de movimento antissimétrico, ou o primeiro vetor de movimento derivado e o segundo vetor de movimento derivado possuírem uma diferença de vetor de movimento antissimétrico, e decodificar um bloco atual utilizando o primeiro vetor de movimento derivado e o segundo vetor de movimento derivado determinados utilizando a previsão dupla.

[011] Em outro exemplo, um aparelho configurado para decodificar os dados de vídeo a partir de uma sequência de bits de vídeo codificada, o aparelho compreendendo: meios para gerar, utilizando a derivação de vetor de movimento de lado de decodificador (DMVD), uma pluralidade de vetores de movimento derivados, meios para determinar um primeiro vetor de movimento derivado e um segundo vetor de movimento derivado a partir da pluralidade de vetores de movimento derivado com base em uma métrica de custo, compreendendo: meios para determinar que o primeiro vetor de movimento derivado e o segundo vetor de movimento derivado satisfazem pelo menos uma restrição em uma relação entre o primeiro vetor de movimento derivado e o segundo vetor de movimento derivado, a relação compreendendo: o primeiro vetor de movimento derivado e o segundo vetor de

movimento derivado possuírem uma diferença de vetor de movimento simétrica, o primeiro vetor de movimento derivado e o segundo vetor de movimento derivado possuírem uma diferença de vetor de movimento pseudo-simétrica, o primeiro vetor de movimento derivado e o segundo vetor de movimento derivado serem antissimétricos, o primeiro vetor de movimento derivado e o segundo vetor de movimento derivado possuírem uma diferença de vetor de movimento antissimétrica, ou o primeiro vetor de movimento derivado e o segundo vetor de movimento derivado possuírem uma diferença de vetor de movimento pseudo-antissimétrico, e meios para decodificar um bloco atual utilizando o primeiro vetor de movimento derivado e o segundo vetor de movimento derivado determinados, utilizando a previsão dupla.

[012] Em outro exemplo, um aparelho configurado para codificar os dados de vídeo, o aparelho compreendendo: meios para gerar, utilizando a derivação de vetor de movimento de lado de decodificador (DMVD), uma pluralidade de vetores de movimento derivados, meios para determinar um primeiro vetor de movimento derivado e um segundo vetor de movimento derivado a partir da pluralidade de vetores de movimento derivado com base em uma métrica de custo, compreendendo: meios para determinar que o primeiro vetor de movimento derivado e o segundo vetor de movimento derivado satisfazem pelo menos uma restrição em uma relação entre o primeiro vetor de movimento derivado e o segundo vetor de movimento derivado, a relação compreendendo o primeiro vetor de movimento derivado e o segundo vetor de movimento derivado possuírem uma diferença de vetor de movimento simétrica, o primeiro vetor de movimento derivado

e o segundo vetor de movimento derivado possuírem uma diferença de vetor de movimento pseudo-simétrico, o primeiro vetor de movimento derivado e o segundo vetor de movimento derivado serem antissimétricos, o primeiro vetor de movimento derivado e o segundo vetor de movimento derivado possuírem uma diferença de vetor de movimento antissimétrica, ou o primeiro vetor de movimento derivado e o segundo vetor de movimento derivado possuírem uma diferença de vetor de movimento antissimétrica, e meios para codificar um bloco atual utilizando o primeiro vetor de movimento derivado e o segundo vetor de movimento derivado determinados, utilizando uma previsão dupla.

[013] Em outro exemplo, essa descrição descreve um meio de armazenamento legível por computador, armazenando instruções que, quando executadas, fazem com que um ou mais processadores de um dispositivo configurado para gerar, utilizando a derivação de vetor de movimento de lado de decodificador (DMVD), uma pluralidade de vetores de movimento derivado, determinar um primeiro vetor de movimento derivado e um segundo vetor de movimento derivado a partir da pluralidade de vetores de movimento derivado com base em uma métrica de custo, compreendendo: determinar que o primeiro vetor de movimento derivado e o segundo vetor de movimento derivado satisfazem pelo menos uma restrição em uma relação entre o primeiro vetor de movimento derivado e o segundo vetor de movimento derivado, a relação compreendendo: o primeiro vetor de movimento derivado e o segundo vetor de movimento derivado possuírem uma diferença de vetor de movimento simétrica, o primeiro vetor de movimento derivado e o segundo vetor de movimento

derivado possuírem uma diferença de vetor de movimento pseudo-simétrica, o primeiro vetor de movimento derivado e o segundo vetor de movimento derivado serem antissimétricos, o primeiro vetor de movimento derivado e o segundo vetor de movimento derivado possuírem uma diferença de vetor de movimento antissimétrica, ou o primeiro vetor de movimento derivado e o segundo vetor de movimento derivado possuírem uma diferença de vetor de movimento pseudo-antissimétrica, e decodificar um bloco atual utilizando o primeiro vetor de movimento derivado e o segundo vetor de movimento derivado determinados utilizando uma previsão dupla.

[014] Em outro exemplo, essa descrição descreve um meio de armazenamento legível por computador armazenando instruções que, quando executadas, fazem com que um ou mais processadores de um dispositivo configurado para gerar, utilizando a derivação de vetor de movimento de lado de decodificador (DMVD), uma pluralidade de vetores de movimento derivados, determinar um primeiro vetor de movimento derivado e um segundo vetor de movimento derivado a partir da pluralidade de vetores de movimento derivado com base em uma métrica de custo, compreendendo: determinar que o primeiro vetor de movimento derivado e o segundo vetor de movimento derivado satisfazem pelo menos uma restrição em uma relação entre o primeiro vetor de movimento derivado e o segundo vetor de movimento derivado, a relação compreendendo: o primeiro vetor de movimento derivado e o segundo vetor de movimento derivado possuírem uma diferença de vetor de movimento simétrico, o primeiro vetor de movimento derivado e o segundo vetor de movimento

derivado possuírem uma diferença de vetor de movimento pseudo-simétrica, o primeiro vetor de movimento derivado e o segundo vetor de movimento derivado serem antissimétricos, o primeiro vetor de movimento derivado e o segundo vetor de movimento derivado possuírem uma diferença de vetor de movimento antissimétrica, ou o primeiro vetor de movimento derivado e o segundo vetor de movimento derivado possuírem uma diferença de vetor de movimento pseudo-antissimétrica, e decodificar um bloco atual utilizando o primeiro vetor de movimento derivado e o segundo vetor de movimento derivado determinados, utilizando-se a previsão dupla.

[015] Os detalhes de um ou mais exemplos são apresentados nos desenhos em anexo e na descrição abaixo. Outras características, objetivos e vantagens serão aparentes a partir da descrição e dos desenhos e das reivindicações.

BREVE DESCRIÇÃO DOS DESENHOS

[016] A figura 1 é um diagrama em bloco ilustrando um sistema de codificação e decodificação de vídeo ilustrativo que pode utilizar as técnicas para realizar a derivação de vetor de movimento de lado de decodificador (DMVD) dessa descrição;

[017] A figura 2 é um diagrama em bloco ilustrando um exemplo de um codificador de vídeo que pode implementar as técnicas para realizar DMVD dessa descrição;

[018] A figura 3 é um diagrama em bloco ilustrando um exemplo de um decodificador de vídeo que pode implementar as técnicas para realizar DMVD dessa descrição;

[019] As figuras 4A e 4B são um diagrama

conceitual ilustrando candidatos a vetor de movimento espacial derivados de blocos vizinhos;

[020] As figuras 5A e 5B são um diagrama conceitual ilustrando uma localização de bloco primário para um candidato a preditor de vetor de movimento temporal (TMVP);

[021] A figura 6 é um diagrama conceitual ilustrando os conceitos relacionados com a combinação bilateral para derivar a informação de movimento de um bloco atual;

[022] A figura 7 é um diagrama conceitual ilustrando os conceitos relacionados com a combinação de gabarito para derivar informação de movimento de um bloco atual;

[023] A figura 8 é um fluxograma ilustrando um exemplo de processo de combinação de gabarito de conversão ascendente de taxa de quadro (FRUC) ilustrativo;

[024] A figura 9 é um fluxograma ilustrando as mudanças propostas ilustrativas para o processo de combinação de gabarito FRUC da figura 8;

[025] A figura 10 é um diagrama conceitual ilustrando os conceitos relacionados com um exemplo do fluxo ótico bidirecional;

[026] A figura 11 é um diagrama conceitual ilustrando um exemplo do cálculo de gradiente para um bloco de 8 x 4;

[027] A figura 12 é um diagrama conceitual ilustrando os conceitos relacionados com a DMVD com base na combinação bilateral de gabarito;

[028] As figuras 13A e 13B são um diagrama

conceitual ilustrando os conceitos relacionados com a compensação de movimento de bloco sobreposto (OBMC);

[029] As figuras de 14A a 14D são diagramas conceituais ilustrando as ponderações OBMC;

[030] A figura 15 é um diagrama conceitual ilustrando os conceitos relacionados com a combinação bilateral para derivar a informação de movimento de um bloco atual;

[031] A figura 16 é um fluxograma ilustrando um método ilustrativo de decodificação de dados de vídeo de acordo com as técnicas descritas nessa descrição;

[032] A figura 17 é um fluxograma ilustrando um método ilustrativo de codificação de dados de vídeo de acordo com as técnicas descritas nessa descrição.

DESCRIÇÃO DETALHADA

[033] Essa descrição apresenta as técnicas relacionadas com a derivação de vetor de movimento de lado de decodificador (DMVD). As técnicas dessa descrição podem ser utilizadas como uma ferramenta eficiente em qualquer padrão de codificação de vídeo futuro.

[034] Em geral, essa descrição descreve as técnica para derivar informação de movimento (por exemplo, um ou mais vetores de movimento, precisão de um vetor de movimento e/ou diferença de vetor de movimento, e um ou mais índices de imagem de referência) utilizados para codificar blocos de dados de vídeo e para determinar, por um decodificador de vídeo, a mesma informação de movimento que foi determinada pelo codificador de vídeo. Utilizando-se as mesmas técnicas de derivação (ou alternativas) utilizadas pelo codificador de vídeo, um decodificador de

vídeo pode determinar, sem receber quaisquer elementos de sintaxe identificando especificamente a informação de movimento, qual informação de movimento foi utilizada para codificar os dados de vídeo. Em algumas situações de codificação, no entanto, o codificador de vídeo ainda pode sinalizar explicitamente a informação de movimento utilizada para codificar o bloco.

[035] Várias técnicas nessa descrição podem ser descritas com referência a um codificador de vídeo, que deve ser um termo genérico que pode se referir a um codificador de vídeo ou a um decodificador de vídeo. A menos que explicitamente mencionado o contrário, não se deve assumir que as técnicas descritas com relação a um codificador de vídeo ou a um decodificador de vídeo não possam ser realizadas pelo outro dentro de um codificador de vídeo ou um decodificador de vídeo. Por exemplo, em muitos casos, um decodificador de vídeo realiza a mesma técnica de codificação, ou algumas vezes uma alternativa, que um codificador de vídeo a fim de decodificar os dados de vídeo codificados. Em muitos casos, um codificador de vídeo também pode incluir um circuito de decodificação de vídeo, e, dessa forma, o codificador de vídeo realiza a decodificação do vídeo como parte da codificação de dados de vídeo. Dessa forma, a menos que mencionado o contrário, as técnicas descritas nessa descrição com relação a um decodificador de vídeo também podem ser realizadas por um codificador de vídeo, e vice-versa.

[036] Essa descrição também pode utilizar termos tal como uma camada atual, um bloco atual, uma imagem atual, uma fatia atual, etc. No contexto dessa

descrição, o termo atual deve identificar uma camada, bloco, imagem, fatia, etc. que esteja sendo codificada no momento, em oposição a, por exemplo, camadas, blocos, imagens e fatias codificados previamente ou blocos, imagens e fatias ainda a serem codificados.

[037] Um novo padrão de codificação de vídeo, chamado de Codificação de Vídeo com Alta Eficiência (HEVC) (também referido como ITU-T H.265), incluindo sua extensão de faixa, extensão de múltiplas visualizações (MV-HEVC) e extensão escalonável (SHVC), foi desenvolvido pela Equipe de Colaboração Conjunta em Codificação de Vídeo (JCT-VC), além de pela Equipe de Colaboração Conjunta em Desenvolvimento de Extensão de Codificação de Vídeo em 3D (HCT-3V) do Grupo de Especialistas em Codificação de Vídeo ITU-T (VCEG) e Grupo de Especialistas em Imagens em Movimento ISO/IEC (MPEG). A especificação HEVC, referida como HEVC WD doravante, está disponível em phenix.int-evry.fr/jct/doc_end_user/documents/14_Vienna/wg11/JCTVC-N1003-v1.zip.

[038] ITU-T VCEG (Q6/16) e ISO/IEC MPEG (JTC 1/SC 29/WG 11) estão estudando, no momento, a necessidade em potencial de se padronizar a tecnologia de codificação de vídeo futura com uma capacidade de compressão que excede de forma significativa a do padrão HEVC atual (incluindo suas extensões atuais e extensões near-term para codificação de conteúdo de tela e codificação de faixa dinâmica alta). Os grupos estão trabalhando em conjunto nessa atividade de exploração em um esforço colaborativo conjunto conhecido como Equipe de Exploração de Vídeo Conjunta (JVET) para avaliar os projetos de tecnologia de

compressão propostos por seus especialistas nessa área. O JVET se encontrou primeiramente entre 19 e 21 de outubro de 2015. Uma versão do software de referência, isso é, Modelo de Exploração Conjunta 5 (JEM 5), está disponível em jvet.hhi.fraunhofer.de/svn/svn_HMJEMSoftware/tags/HM-16.6-JEM-5.0. Uma descrição do algoritmo de JEM 5 está disponível em phenix.it-sudparis.eu/jvet/doc_end_user/current_document.php?id=2714.

[039] A figura 1 é um diagrama em bloco ilustrando um sistema de codificação e decodificação de vídeo ilustrativo 10 que pode utilizar as técnicas para realizar a DMVD dessa descrição. Como ilustrado na figura 1, o sistema 10 inclui um dispositivo fonte 12 que fornece dados de vídeo codificados a serem decodificados posteriormente por um dispositivo de destino 14. Em particular, o dispositivo fonte 12 fornece dados de vídeo para o dispositivo de destino 14 através de um meio legível por computador 16. O dispositivo fonte 12 e o dispositivo de destino 14 podem compreender qualquer um dentre uma ampla faixa de dispositivos, incluindo computadores desktop, computadores notebook (isso é, laptop), computadores tablet, caixas de decodificação, aparelhos telefônicos tal como os chamados "smartphones", os chamados "smart pads", televisores, câmeras, dispositivos de exibição, aparelhos de mídia digital, consoles de jogos de vídeo, dispositivo de sequenciamento de vídeo, ou similares. Em alguns casos, o dispositivo fonte 12 e o dispositivo de destino 14 podem ser equipados para comunicação sem fio.

[040] O dispositivo de destino 14 pode

receber os dados de vídeo codificados a serem decodificados através do meio legível por computador 16. O meio legível por computador 16 pode compreender qualquer tipo de meio ou dispositivo capaz de mover os dados de vídeo codificados do dispositivo fonte 12 para o dispositivo de destino 14. Em um exemplo, o meio legível por computador 16 pode compreender um meio de comunicação para permitir que o dispositivo fonte 12 transmita os dados de vídeo codificados diretamente para o dispositivo de destino 14 em tempo real. Os dados de vídeo codificados podem ser modulados de acordo com um padrão de comunicação, tal como um protocolo de comunicação sem fio, e transmitidos para o dispositivo de destino 14. O meio de comunicação pode compreender qualquer meio de comunicação com ou sem fio, tal como um espectro de frequência de rádio (RF) ou uma ou mais linhas de transmissão físicas. O meio de comunicação pode formar parte de uma rede com base em pacote, tal como uma rede de área local, uma rede de área ampla, ou uma rede global tal como a Internet. O meio de comunicação pode incluir roteadores, comutadores, estações base, ou qualquer outro equipamento que possa ser útil para facilitar a comunicação do dispositivo fonte 12 para o dispositivo de destino 14.

[041] Em alguns exemplos, os dados codificados podem ser enviados a partir da interface de saída 22 para um dispositivo de armazenamento. De forma similar, os dados codificados podem ser acessados a partir do dispositivo de armazenamento pela interface de entrada. O dispositivo de armazenamento pode incluir qualquer um dentre uma variedade de meios de armazenamento de dados

distribuídos ou acessados localmente, tal como um disco rígido, discos Blu-ray, DVDs, CD-ROMs, memória flash, memória volátil ou não volátil, ou qualquer outro meio de armazenamento digital adequado para armazenar os dados de vídeo decodificados. Em um exemplo adicional, o dispositivo de armazenamento pode corresponder a um servidor de arquivo ou outro dispositivo de armazenamento intermediário que possa armazenar o vídeo codificado gerado pelo dispositivo fonte 12. O dispositivo de destino 14 pode acessar os dados de vídeo armazenados a partir do dispositivo de armazenamento através de sequenciamento ou download. O servidor de arquivo pode ser qualquer tipo de servidor capaz de armazenar dados de vídeo codificados e transmitir esses dados de vídeo codificados para o dispositivo de destino 14. Servidores de arquivo ilustrativos incluem um servidor de rede (por exemplo, para um sítio de rede), um servidor FTP, dispositivos de armazenamento fixados à rede (NAS), ou um acionador de disco local. O dispositivo de destino 14 pode acessar os dados de vídeo codificados através de qualquer conexão de dados padrão, incluindo uma conexão com a Internet. Isso pode incluir um canal sem fio (por exemplo, uma conexão Wi-Fi), uma conexão com fio (por exemplo, DSL, modem a cabo, etc.), ou uma combinação de ambos que seja adequada para acessar os dados de vídeo codificados armazenados em um servidor de arquivo. A transmissão dos dados de vídeo codificados a partir do dispositivo de armazenamento pode ser uma transmissão de sequenciamento, uma transmissão em download, ou uma combinação das mesmas.

[042] As técnicas dessa descrição não são

limitadas necessariamente a aplicativos ou configurações sem fio. As técnicas podem ser aplicadas à codificação de vídeo para suportar qualquer um dentre uma variedade de aplicativos de multimídia, tal como difusões de televisão através do ar, transmissões de televisão a cabo, transmissões de televisão via satélite, transmissões de vídeo de sequenciamento na Internet, tal como sequenciamento adaptativo dinâmico através de HTTP (DASH), vídeo digital que é codificado em um meio de armazenamento de dados, decodificação de vídeo de dados armazenados em um meio de armazenamento de dados, ou outros aplicativos. Em alguns exemplos, o sistema 10 pode ser configurado para suportar a transmissão de vídeo de via única ou duas vias para suportar aplicativos, tal como sequenciamento de vídeo, reprodução de vídeo, difusão de vídeo e/ou vídeo telefonia.

[043] No exemplo da figura 1, o dispositivo fonte 12 inclui a fonte de vídeo 18, o codificador de vídeo 20 e a interface de saída 22. O dispositivo de destino 14 inclui a interface de entrada 28, o decodificador de vídeo 30, e o dispositivo de exibição 32. Em outros exemplos, um dispositivo fonte e um dispositivo de destino podem incluir outros componentes ou disposições. Por exemplo, o dispositivo fonte 12 pode receber os dados de vídeo de uma fonte de vídeo externa 18, tal como uma câmera externa. Da mesma forma, o dispositivo de destino 14 pode interfacear com um dispositivo de exibição externo, em vez de incluir um dispositivo de exibição integrado. De acordo com essa descrição, o codificador de vídeo 20 do dispositivo fonte 12 e/ou o decodificador de vídeo 30 do dispositivo de

destino 14 podem ser configurados para realizar uma ou mais técnicas DMVD descritas aqui. Por exemplo, o codificador de vídeo 20 e/ou o decodificador de vídeo 30 podem ser configurados para aplicar uma ou mais restrições a (1) um MV derivado pela DMVD, e/ou (2) uma diferença MV entre um MV inicial e um MV derivado pela DMVD. Uma restrição pode incluir uma limitação nos vetores de movimento derivados pelo processo DMVD. Restrições ilustrativas dessa descrição podem incluir restrições que impõem alguma relação predefinida entre o MV e MVDs em duas listas de imagem de referência (por exemplo, List0 e List1). Quando a restrição é aplicada a DMVD, em determinados exemplos, apenas a informação de movimento derivada, que corresponde à restrição, é considerada informação de movimento válida. Dessa forma, a lista de MVs e do MV final selecionado pode ser diferente se a restrição não tivesse sido utilizada. Em um primeiro exemplo, a restrição inclui se os MVs derivados são simétricos. Em um segundo exemplo, a restrição inclui se as diferenças de MV entre os MVs iniciais e os MVs derivados pelos métodos DMVD são simétricos. Em um terceiro exemplo a restrição inclui se os MVs derivados são antissimétricos. Em um quarto exemplo, a restrição inclui se as diferenças de MV entre os MVs iniciais e os MVs derivados pelos métodos DMVD são antissimétricos. Em alguns exemplos, a ativação das restrições de MV/diferenças entre MV simétricos/antissimétricos é sinalizada explicitamente. Em outros exemplos, as restrições podem ser determinadas implicitamente de acordo com alguma informação codificada.

[044] Em exemplos adicionais, as condições podem ser impostas às restrições de modo que a restrição se

aplique apenas em determinados exemplos. As condições podem incluir uma limitação em quando as restrições são ativadas. Por exemplo, (1) a restrição só pode ser ativada quando os MVs iniciais são simétricos; (2) a restrição só é ativada quando os MVs iniciais são pseudo-simétricos; (3) a restrição só é ativada quando os MVs iniciais são antissimétricos; (4) a restrição só é aplicada quando os MVs iniciais são pseudo-antissimétricos; (5) a restrição só é ativada quando os MVs iniciais não são simétricos; (6) a restrição só é ativada quando os MVs iniciais não são pseudo-simétricos; (7) a restrição só é ativada quando os MVs iniciais não são antissimétricos; (8) a restrição só é ativada quando os MVs iniciais não são pseudo-antissimétricos; (9) a restrição só é ativada quando as imagens de referência List0 e List1 dos MVs iniciais estão ambas antes ou ambas depois da imagem atual; (10) a restrição só é ativada quando as imagens de referência List0 e List1 dos MVs iniciais NÃO estão ambas antes ou ambas depois da imagem atual; (11) a restrição só é aplicada quando o índice de referência dos MVs iniciais são ambos zero; (12) a restrição só é ativada quando o índice de referência dos MVs iniciais NÃO são ambos iguais a zero; (13) a restrição só é ativada quando as distâncias POC entre a imagem de referência List0 e a imagem atual e as distâncias POC entre a imagem de referência de List1 e a imagem atual forem iguais; (14) a restrição é ativada apenas quando as distâncias POC entre a imagem de referência List1 e a imagem atual NÃO forem iguais; (15) a restrição da diferença de MV simétrica é aplicada quando os MVs iniciais não são simétricos; e/ou (16) a restrição da

diferença de MV antissimétrica é aplicada quando os MVs iniciais são simétricos.

[045] Em um exemplo adicional, a resolução de MV de restrição simétrica pode ser designada de forma determinística. Em um exemplo, para refinamento de pixel inteiro da Combinação Bilateral de Gabarito, nenhuma restrição simétrica/pseudo-simétrica/assimétrica deve ser imposta, e as restrições mencionadas acima são apenas um refinamento de movimento de precisão de meio pixel, de um quarto de pixel ou precisão mais alta. O nível de restrição (da resolução MV) pode ser sinalizado através de SPS/PPS/Cabeçalho de Fatia. A resolução também pode ser decidida juntamente com o fato de se outras ferramentas de refinamento de movimento, tal como BIO, conversão ascendente de taxa de quadro sub-pixel (FRUC) serem ativadas. Por exemplo, quando BIO é ativado, a restrição não deve ser imposta ao refinamento de movimento de precisão de um quarto de pixel ou superior.

[046] Em outro exemplo, o nível de restrição pode ser tornado adaptativo, de acordo com a diferença absoluta entre MVs da List0 (também referida como L0) e List1 (também referida como L1), a diferença absoluta escalonada (com base na distância POC relativa) entre MVs de List0 e List1, ou os valores SAD iniciais entre as amostras interpoladas de List0 e List1 (isso é, P0 e P1 nesse exemplo). Alternativamente, a razão dos valores SAD entre (gabarito atual P0) e gabarito atual, P1) pode ser utilizada para decidir em que resolução MV a restrição deve ser imposta. Nesse exemplo, se a razão dos valores SAD for inferior a um limite, então, nenhuma restrição simétrica

mencionada acima deve ser imposta.

[047] Adicionalmente, em alguns exemplos, uma única restrição (por exemplo, MVs simétricos, diferença de MV simétrica, MVS antissimétricos, ou diferença de MV antissimétrica) é aplicada aos MVs derivados com base em uma condição. Em outros exemplos, múltiplas restrições são aplicadas aos MVs derivados com base em múltiplas condições.

[048] O sistema ilustrado 10 da figura 1 é meramente um exemplo. As técnicas para realizar DMVD dessa descrição podem ser realizadas por qualquer dispositivo de codificação e/ou decodificação de vídeo digital. Por exemplo, as técnicas para restringir (1) um MV derivado por DMVD, e/ou (2) uma diferença de MV entre um MV inicial e um MV derivado de DMVD pode ser realizada por qualquer dispositivo de codificação e/ou decodificação de vídeo digital. Apesar de geralmente as técnicas dessa descrição serem realizadas por um dispositivo de codificação de vídeo, as técnicas também podem ser realizadas por um codificador/decodificador de vídeo, tipicamente referido como "CODEC". Ademais, as técnicas dessa descrição também podem ser realizadas por um pré-processador de vídeo. O dispositivo fonte 112 e o dispositivo de destino 14 são meramente exemplos de tais dispositivos de codificação nos quais o dispositivo fonte 12 gera os dados de vídeo codificados para transmissão para o dispositivo de destino 14. Em alguns exemplos, os dispositivos 12, 14 podem operar de uma forma substancialmente simétrica de modo que cada um dos dispositivos 12, 14 inclua componentes de codificação e decodificação de vídeo. Dessa forma, o sistema 10 pode

suportar a transmissão de via única ou duas vias entre os dispositivos de vídeo 12, 14, por exemplo, para sequenciamento de vídeo, reprodução de vídeo, difusão de vídeo ou vídeo telefonia.

[049] A fonte de vídeo 18 do dispositivo fonte 12 pode incluir um dispositivo de captura de vídeo, tal como uma câmera de vídeo, um arquivo de vídeo contendo vídeo previamente capturado e/ou uma interface de alimentação de vídeo para receber vídeo de um provedor de conteúdo de vídeo. Como uma alternativa adicional, a fonte de vídeo 18 pode gerar dados com base em gráficos de computador como vídeo fonte, ou uma combinação de vídeo ao vivo, vídeo arquivado, e vídeo gerado por computador. Em alguns casos, se a fonte de vídeo 18 for uma câmera de vídeo, o dispositivo fonte 12 e o dispositivo de destino 14 podem formar os chamados telefones com câmera ou vídeo telefones. Como mencionado acima, no entanto, as técnicas descritas nessa descrição podem ser aplicáveis à codificação de vídeo em geral, e podem ser aplicadas a aplicativos com e/ou sem fio. Em cada caso, o vídeo capturado, pré-capturado, ou gerado por computador pode ser codificado pelo codificador de vídeo 20. A informação do vídeo codificado pode, então, ser enviada pela interface de saída 22 para um meio legível por computador 16.

[050] O meio legível por computador 16 pode incluir meio transiente, tal como uma difusão sem fio ou transmissão de rede com fio, ou meio de armazenamento (isso é, meio de armazenamento não transitório), tal como disco rígido, flash drive, disco compacto, disco de vídeo digital, disco Blu-ray, ou outra mídia legível por

computador. Em alguns exemplos, um servidor de rede (não ilustrado) pode receber dados de vídeo codificados do dispositivo fonte 112 e fornecer dados de vídeo codificados para o dispositivo de destino 14, por exemplo, através da transmissão em rede. De forma similar, um dispositivo de computação de uma instalação de produção de meio, tal como uma instalação de estampagem de disco, pode receber dados de vídeo codificados a partir do dispositivo fonte 12 e produzir um disco contendo os dados de vídeo codificados. Portanto, o meio legível por computador 16 pode ser compreendido como incluindo um ou mais meios legíveis por computador de várias formas, em vários exemplos.

[051] A interface de entrada 28 do dispositivo de destino 14 recebe informação do meio legível por computador 16. A informação do meio legível por computador 16 pode incluir informação de sintaxe definida pelo codificador de vídeo 20, que também é utilizada pelo decodificador de vídeo 30, que inclui elementos de sintaxe que descrevem as características e/ou processamento dos blocos e outras unidades codificadas. O dispositivo de exibição 32 exibe os dados de vídeo decodificados para um usuário, e pode compreender qualquer um dentre uma variedade de dispositivos de exibição, tal como um tubo de raio catodo (CRT), um monitor de cristal líquido (LCD), um monitor de plasma, um monitor de diodo de emissão de luz orgânica (OLED), ou outro tipo de dispositivo de exibição.

[052] O codificador de vídeo 20 e o decodificador de vídeo 30 podem operar de acordo com um padrão de codificação de vídeo, tal como o padrão de Codificação de Vídeo de Alta Eficiência (HEVC), também

referido como ITU-T H.265. Alternativamente, o codificador de vídeo 20 e o decodificador de vídeo 30 podem operar de acordo com outros padrões da indústria ou proprietários, tal como o padrão ITU-T H.264, alternativamente referido como MPEG-4, Parte 10, Codificação de Vídeo Avançada (AVC), ou extensões de tais padrões. As técnicas dessa descrição, no entanto, não estão limitadas a qualquer padrão de codificação em particular. Outros exemplos de padrões de codificação de vídeo incluem MPEG-2 e ITU-T H.263. Apesar de não ilustrado na figura 1, em alguns aspectos, o codificador de vídeo 20 e o decodificador de vídeo 30 podem, cada um, ser integrados com um codificador e decodificador de áudio, e podem incluir unidades MUX-DEMUX adequadas, ou outro hardware e software, para manusear a codificação de áudio e vídeo em uma sequência de dados comum ou sequências de dados separadas. Se aplicável, as unidades MUX-DEMUX podem se conformar ao protocolo de multiplexador ITU H.223, ou outros protocolos, tal como o protocolo de datagrama de usuário (UDP).

[053] O codificador de vídeo 20 e o decodificador de vídeo 30, cada um, podem ser implementados como qualquer um dentre uma variedade de conjuntos de circuito de codificador e/ou decodificador adequado, tal como um ou mais microprocessadores, conjunto de circuitos de processamento (incluindo o conjunto de circuitos de função fixa e/ou conjunto de circuitos de processamento programável), processadores de sinal digital (DSPs), circuitos integrados específicos de aplicativo (ASICs), conjuntos de porta programável em campo (FPGAs), lógica discreta, software, hardware, firmware ou quaisquer

combinações dos mesmos. Quando as técnicas são implementadas parcialmente em software, um dispositivo pode armazenar instruções para software em um meio legível por computador não transitório adequado e executar as instruções em hardware utilizando um ou mais processadores para realizar as técnicas dessa descrição. Cada um dentre o codificador de vídeo 20 e o decodificador de vídeo 30 podem ser incluídos em um ou mais codificadores ou decodificadores, qualquer um dos quais pode ser integrado como parte de um codificador/decodificador (CODEC) combinado em um dispositivo respectivo. Um dispositivo incluindo o codificador de vídeo 20 e/ou o decodificador de vídeo 30 pode compreender um circuito integrado, um microprocessador e/ou um dispositivo de comunicação sem fio, tal como um telefone celular.

[054] Em geral, de acordo com ITU-T H.265, uma imagem de vídeo pode ser dividida em uma sequência de unidades de árvore de codificação (CTUs) (ou unidades de codificação maiores (LCUs)) que podem incluir ambas as amostras de luminescência e crominância. Alternativamente, as CTUs podem incluir dados monocromáticos (isso é, apenas amostras de luminescência). Os dados de sintaxe dentro de uma sequência de bits pode definir um tamanho para a CTU, que é uma unidade de codificação maior em termos do número de pixels. Uma fatia inclui um número de CTUs consecutivas na ordem de codificação. Uma imagem de vídeo pode ser dividida em uma ou mais fatias. Cada CTU pode ser dividida em unidades de codificação (CUs), de acordo com um quadtree. Em geral, uma estrutura de dados de quadtree inclui um nó por CU, com um nó raiz correspondendo à CTU.

Se uma CU for dividida em quatro sub-CUs, o nó correspondente à CU inclui quatro nós folha, cada um dos quais corresponde a uma das sub-CUs.

[055] Cada nó da estrutura de dados tipo quadtree pode fornecer dados de sintaxe para a CU correspondente. Por exemplo, um nó no quadtree pode incluir um indicador dividido, indicando se a CU que corresponde ao nó é dividida em sub-CUs. Os elementos de sintaxe para uma CU podem ser definidos de forma recursiva, e podem depender de se a CU foi dividida em sub-CUs. Se uma CU não for dividida adicionalmente, é referida como CU folha. Nessa descrição, quatro sub-CUs de uma CU folha serão referidas como CUs folha, mesmo se não houver divisão explícita da CU folha original. Por exemplo, se uma CU de tamanho de 16 x 16 não for dividida adicionalmente, as quatro sub-CUs de 8 x 8 também serão referidas como CUs folha apesar de a CU de 16 x 16 nunca ter sido dividida.

[056] Uma CU possui uma finalidade similar à de um macro bloco do padrão H.264, exceto que uma CU não possui uma distinção de tamanho. Por exemplo, uma CTU pode ser dividida em quatro nós criança (também referidos como sub-CUs), e cada nó criança pode, por sua vez, ser um nó parente e ser dividido em outros quatro nós criança. Um nó criança final, não dividido, referido como um nó folha do quadtree, compreende um nó de codificação, também referido como uma CU folha. Os dados de sintaxe associados com uma sequência de bits codificada podem definir um número máximo de vezes em que uma CTU pode ser dividida, referido como uma profundidade máxima de CU, e também pode definir um tamanho mínimo de nós de codificação. De acordo, uma

sequência de bits também pode definir uma menor unidade de codificação (SCU). Essa descrição utiliza o termo "bloco" para fazer referência a qualquer um dentre uma CU, uma unidade de previsão (PU) ou uma unidade de transformação (TU), no contexto de HEVC, ou estruturas de dados similares no contexto de outros padrões (por exemplo, macro blocos ou sub-blocos dos mesmos em H.264/AVC).

[057] Uma CU inclui um nó de codificação e unidades de previsão (PUs) e unidades de transformação (TUs) associadas com o nó de codificação. Um tamanho da CU corresponde a um tamanho do nó de codificação e é geralmente quadrado em formato. O tamanho da CU pode variar de 8 x 8 pixels até o tamanho da CTU com um tamanho máximo, por exemplo, de 64 x 64 pixels ou mais. Cada CU pode conter uma ou mais PUs e uma ou mais TUs. Os dados de sintaxe associados com uma CU podem descrever, por exemplo, a partição da CU em uma ou mais PUs. Os modos de partição podem diferir entre se a CU foi codificada por pulo ou modo direto, codificada por modo de intraprevisão, ou codificada por modo de interprevisão. As PUs podem ser divididas para que apresentem o formato não quadrado. Os dados de sintaxe associados com uma CU também podem descrever, por exemplo, a partição da CU em uma ou mais TUs de acordo com um quadtree. Uma TU pode ter formato quadrado ou não quadrado (por exemplo, retangular).

[058] O padrão HEVC permite as transformações de acordo com as TUs, que podem ser diferentes para diferentes CUs. As TUs são tipicamente dimensionadas com base no tamanho das PUs (ou partições de uma CU) dentro de uma CU determinada definida para uma CTU dividida, apesar

de esse nem sempre ser o caso. As TUs são tipicamente do mesmo tamanho ou menores que as PUs (ou partições de uma CU, por exemplo, no caso de intraprevisão). Em alguns exemplos, amostras residuais correspondentes a uma CU podem ser subdivididas em unidades menores utilizando uma estrutura de quadtree conhecida como "quad tree residual" (RQT). Os nós folha do RQT podem ser referidos como unidades de transformação (TUs). Os valores de diferença de pixel associados com as TUs podem ser transformados para produzir coeficientes de transformação que podem ser quantizados.

[059] Uma CU folha pode incluir uma ou mais unidades de previsão (PUs) quando previstas utilizando a interprevisão. Em geral, uma PU representa uma área espacial correspondente a toda ou a uma parte da CU correspondente, e pode incluir dados para recuperar e/ou gerar uma amostra de referência para a PU. Ademais, uma PU inclui dados relacionados com a previsão. Quando a CU é codificada por intermodo, uma ou mais PUs da CU podem incluir dados definindo a informação de movimento, tal como um ou mais vetores de movimento, ou as PUs podem ser codificadas por modo de pulo. Os dados que definem o vetor de movimento para uma PU podem descrever, por exemplo, um componente horizontal do vetor de movimento, um componente vertical do vetor de movimento, uma resolução para o vetor de movimento (por exemplo, precisão de um quarto de pixel ou precisão de um oitavo de pixel), uma imagem de referência para a qual o vetor de movimento aponta, e/ou uma lista de imagens de referência (por exemplo, List0 ou List1) para o vetor de movimento.

[060] CUs folha também podem ser previstas por intramodo. Em geral, a intraprevisão envolve prever uma CU folha (ou partições da mesma) utilizando um intramodo. Um codificador de vídeo pode selecionar um conjunto de pixels vizinhos, previamente codificados, para a CU folha utilizar para prever a CU folha (ou partições da mesma).

[061] Uma CU folha também pode incluir uma ou mais unidades de transformação (TUs). As unidades de transformação podem ser especificadas utilizando-se um RQT (também referida como uma estrutura de quadtree de TU), como discutido acima. Por exemplo, um indicador de divisão pode indicar se uma CU folha é dividida em quatro unidades de transformação.

[062] Ademais, as TUs das CUs folha também são associadas a estruturas de dados de quadtree respectivas, referidas como quadrees residuais (RQTs). Isso é, uma CU folha pode incluir um quadtree indicando como a CU folha é dividida em TUs. O nó raiz de um quadtree TU geralmente corresponde a uma CU folha, enquanto o nó raiz de um quadtree CU geralmente corresponde a uma CTU (ou LCU). As TUs do RQT que não são divididas são referidas como TUs folha. Em geral, essa descrição utiliza os termos CU e TU para fazer referência à CU folha e TU folha, respectivamente, a menos que notado o contrário.

[063] Uma sequência de vídeo inclui tipicamente uma série de quadros ou imagens de vídeo, começando com uma imagem de ponto de acesso randômico (RAP). Uma sequência de vídeo pode incluir dados de sintaxe em um conjunto de parâmetros de sequência (SPS) que caracteriza a sequência de vídeo. Cada fatia de uma imagem

pode incluir dados de sintaxe de fatia que descrevem um modo de codificação para a fatia respectiva. O codificador de vídeo 20 opera tipicamente em blocos de vídeo dentro das fatias de vídeo individuais a fim de codificar os dados de vídeo. Um bloco de vídeo pode corresponder a um nó de codificação dentro de uma CU. Os blocos de vídeo podem ter tamanhos fixos ou variáveis, e podem diferir em tamanho de acordo com um padrão de codificação especificado.

[064] Como um exemplo, a previsão pode ser realizada para PUs de vários tamanhos. Assumindo-se que o tamanho de uma CU em particular é de $2N \times 2N$, a intraprevisão pode ser realizada em tamanhos de PU de $2N \times 2N$ ou $N \times N$, e a interprevisão pode ser realizada em tamanhos simétricos de PU de $2N \times 2N$, $2N \times N$, $N \times 2N$ ou $N \times N$. A partição assimétrica para a interprevisão também pode ser realizada para tamanhos de PU de $2N \times nU$, $2N \times nD$, $nL \times 2N$ e $nR \times 2N$. Na partição assimétrica, uma direção de uma CU não é dividida enquanto que a outra direção é dividida em 25% e 75%. A parte da CU correspondente à 25% da partição é indicada por um "n" seguido por uma indicação de "para cima", "para baixo", "para a esquerda" e "para a direita". Dessa forma, por exemplo, " $2N \times nU$ " se refere a uma CU $2N \times 2N$ que é dividida horizontalmente com uma PU de $2N \times 0,5N$ em cima de uma PU $2N \times 1,5N$ no fundo.

[065] Nessa descrição, " $N \times N$ " e " N por N " podem ser utilizados de forma intercambiável para fazer referência às dimensões de pixel de um bloco de vídeo em termos de dimensões vertical e horizontal, por exemplo, 16×16 pixels ou 16 por 16 pixels. Em geral, um bloco de 16×16 terá 16 pixels em uma direção vertical ($y = 16$) e 16

pixels em uma direção horizontal ($x = 16$). Da mesma forma, um bloco $N \times N$ possui geralmente N pixels em uma direção vertical e N pixels em uma direção horizontal, onde N representa um valor inteiro não negativo. Os pixels em um bloco podem ser dispostos em fileiras e colunas. Ademais, os blocos não precisam necessariamente ter o mesmo número de pixels na direção horizontal como na direção vertical. Por exemplo, os blocos podem compreender $N \times M$ pixels, onde M não é necessariamente igual a N .

[066] Seguindo a codificação por intraprevisão ou interprevisão utilizando as PUs de uma CU, o codificador de vídeo 20 pode calcular dados residuais para as TUs da CU. As PUs podem compreender dados de sintaxe descrevendo um método ou modo de geração de dados e pixel de previsão no domínio espacial (também referido como domínio de pixel) e as TUs podem compreender coeficientes no domínio de transformação seguindo a aplicação de uma transformação, por exemplo, uma transformação de cosseno discreta (DCT), uma transformação de inteiro, uma transformação de wavelet, ou uma transformação conceitualmente similar a dados de vídeo residuais. Os dados residuais podem corresponder a diferenças de pixel entre os pixels da imagem não codificada e os valores de previsão correspondentes às PUs. O codificador de vídeo 20 pode formar as TUs para incluir coeficientes de transformação quantizados representativos dos dados residuais para a CU. Isso é, o codificador de vídeo 20 pode calcular os dados residuais (na forma de um bloco residual), transformar o bloco residual para produzir um bloco de coeficientes de transformação, e, então, quantizar

os coeficientes de transformação para formar os coeficientes de transformação quantizados. O codificador de vídeo 20 pode formar uma TU incluindo os coeficientes de transformação quantizados, além de outra informação de sintaxe (por exemplo, informação de divisão para a TU).

[067] Como notado, seguindo quaisquer transformações para produzir os coeficientes de transformação, o codificador de vídeo 20 pode realizar a quantização de coeficientes de transformação. A quantização geralmente se refere a um processo no qual os coeficientes de transformação são quantizados para, possivelmente, reduzir a quantidade de dados utilizada para representar os coeficientes, fornecendo compressão adicional. O processo de quantização pode reduzir a profundidade de bit associada com alguns ou todos os coeficientes. Por exemplo, um valor de bit igual a n pode ser arredondado para baixo para um valor de bit m durante a quantização, onde n é maior que m .

[068] Seguindo a quantização, o codificador de vídeo pode digitalizar os coeficientes de transformação, produzindo um vetor unidimensional a partir da matriz bidimensional incluindo os coeficientes de transformação quantizados. A digitalização pode ser projetada para colocar os coeficientes de mais energia (e, portanto, menos frequência) na frente do conjunto e colocar os coeficientes com menor energia (e, portanto, maior frequência) no fundo do conjunto. Em alguns exemplos, o codificador de vídeo 20 pode utilizar uma ordem de digitalização predefinida para digitalizar os coeficientes de transformação quantizados para produzir um vetor serializado que pode ser codificado por entropia. Em outros exemplos, o codificador de vídeo 20

pode realizar uma digitalização adaptativa. Depois da digitalização dos coeficientes de transformação quantizados para formar um vetor unidimensional, o codificador de vídeo 20 pode codificar por entropia o vetor unidimensional, por exemplo, de acordo com a codificação de comprimento variável adaptativa de contexto (CAVL), a codificação aritmética binária adaptativa de contexto (CABAC), a codificação aritmética binária adaptativa de contexto com base em sintaxe (SBAC), a codificação por Entropia por Partição de Intervalo de Probabilidade (PIPE), ou outra metodologia de codificação por entropia. O codificador de vídeo 20 também pode codificar por entropia os elementos de sintaxe associados com os dados de vídeo codificados para uso pelo decodificador de vídeo 30 na decodificação de dados de vídeo.

[069] Para realizar CABAC, o codificador de vídeo 20 pode designar um contexto dentro de um modelo de contexto para um símbolo a ser transmitido. O contexto pode se referir a, por exemplo, se os valores vizinhos do símbolos são diferentes de zero ou não. Para realizar CAVLC, o codificador de vídeo 20 pode selecionar um código de comprimento variável para um símbolo a ser transmitido. As palavras código em VLC podem ser construídas de modo que códigos relativamente mais curtos correspondam a símbolos mais prováveis, enquanto que códigos mais longos correspondam a símbolos menos prováveis. Dessa forma, o uso de VLC pode alcançar uma economia de bit maior, por exemplo, com relação à utilização de palavras código de mesmo comprimento para cada símbolo a ser transmitido. A determinação de probabilidade pode ser baseada em um

contexto designado ao símbolo.

[070] Em geral, o decodificador de vídeo 30 realiza um processo substancialmente similar, porém alternativo, que pode ser realizado pelo codificador de vídeo 20 para decodificar os dados codificados. Por exemplo, o decodificador de vídeo 30 quantiza de forma inversa e transforma de forma inversa os coeficientes de uma TU recebida para reproduzir um bloco residual. O decodificador de vídeo 30 utiliza um modo de previsão sinalizado (intra ou interprevisão) para formar um bloco previsto. Então, o decodificador de vídeo 30 combina o bloco previsto e o bloco residual (pixel por pixel) para reproduzir o bloco original. O processamento adicional pode ser realizado, tal como realizando um processo de desbloqueio para reduzir os artefatos visuais ao longo dos limites de bloco. Adicionalmente, o decodificador de vídeo 30 pode decodificar os elementos de sintaxe utilizando CABAC de uma forma substancialmente similar a, apesar de alternativa a, o processo de codificação CABAC do codificador de vídeo 20.

[071] O codificador de vídeo 20 pode enviar adicionalmente os dados de sintaxe, tal como os dados de sintaxe com base em bloco, dados de sintaxe com base em imagem, e dados de sintaxe com base em sequência, para o decodificador de vídeo 30, por exemplo, em um cabeçalho de imagem, um cabeçalho de bloco, um cabeçalho de fatia ou outros dados de sintaxe, tal como um conjunto de parâmetros de sequência (SPS), conjunto de parâmetros de imagem (PPS), ou conjunto de parâmetros de vídeo (VPS).

[072] A figura 2 é um diagrama em bloco

ilustrando um exemplo do codificador de vídeo 20 que pode ser configurado para realizar uma ou mais técnicas DMVD dessa descrição. Por exemplo, o codificador de vídeo 20 pode ser configurado para aplicar uma ou mais restrições a (1) um MV derivado pela DMVD e/ou (2) uma diferença de MV entre um MV inicial e um MV derivado de DMVD. Quando a restrição é aplicada a DMVD, em determinados exemplos, apenas a informação de movimento derivada, que corresponde à restrição, é considerada informação de movimento válida. Dessa forma, a lista de MVs e o MV final selecionado podem ser diferentes caso a restrição seja utilizada.

[073] Em um primeiro exemplo, a restrição inclui se os MVs derivados são simétricos. Em um segundo exemplo, a restrição inclui se as diferenças de MV entre os MVs iniciais e os MVs derivados pelos métodos DMVD são simétricas. Em um terceiro exemplo, a restrição inclui se os MVs derivados são antissimétricos. Em um quarto exemplo, a restrição inclui se as diferenças MV entre os MVs iniciais e os MVs derivados pelos métodos DMVD são antissimétricas. Em alguns exemplos, a ativação das restrições de diferenças MV/MV simétricos/antissimétricos é sinalizada explicitamente. Em outros exemplos, as restrições podem ser determinadas implicitamente de acordo com alguma informação codificada. Em exemplos adicionais, as condições podem ser localizadas nas restrições de modo que a restrição só se aplique a alguns exemplos.

[074] Em particular, o codificador de vídeo 20 pode realizar as técnicas DMVD dessa descrição durante um circuito de reconstrução, que inclui processos realizados pela unidade de quantização inversa 58, unidade

de transformação inversa 60, e somador 62. Adicionalmente, como discutido acima, o codificador de vídeo 20 pode sinalizar determinados valores que podem auxiliar um decodificador de vídeo, tal como o decodificador de vídeo 30, na realização da DMVD.

[075] O codificador de vídeo 20 pode realizar a intracodificação e intercodificação de blocos de vídeo dentro de fatias de vídeo. A intracodificação se baseia na previsão espacial para reduzir ou remover a redundância espacial no vídeo dentro de um quadro de vídeo ou imagem determinado. A intercodificação se baseia na previsão temporal para reduzir ou remover a redundância temporal no vídeo dentro dos quadros ou imagens adjacentes de uma sequência de vídeo. O intramodo (modo I) pode se referir a qualquer um dos vários modos de codificação com base em espaço. Os intermodos, tal como uma previsão unidirecional (modo P) ou revisão dupla (modo B), podem se referir a quaisquer dos vários modos de codificação com base em tempo.

[076] Como ilustrado na figura 2, o codificador de vídeo 20 recebe um bloco de vídeo atual dentro de um quadro de vídeo a ser codificado. No exemplo da figura 2, o codificador de vídeo 20 inclui a memória de dados de vídeo 66, a unidade de seleção de modo 40, a memória de imagem de referência 64 (que também pode ser referida como um armazenador de imagens decodificadas (DPB)), somador 50, unidade de processamento de transformação 52, unidade de quantização 54 e unidade de codificação por entropia 56. A unidade de seleção de modo 40, por sua vez, inclui a unidade de compensação de

movimento 44, a unidade de estimativa de movimento 42, a unidade de intraprevisão 46, e a unidade de partição 48. Para a reconstrução de bloco de vídeo, o codificador de vídeo 20 também inclui a unidade de quantização inversa 58, a unidade de transformação inversa 60 e o somador 62. Um filtro de desbloqueio (não ilustrado na figura 2) também pode ser incluído para filtrar os limites de bloco para remover os artefatos de bloqueio do vídeo reconstruído. Se desejável, o filtro de desbloqueio filtraria tipicamente a saída do somador 62. Filtros adicionais (em circuito ou pós circuito) também podem ser utilizados em adição ao filtro de desbloqueio. Tais filtros não são ilustrados por motivos de brevidade, mas, se desejável, podem filtrar a saída do somador 50 (como em um filtro de circuito de entrada).

[077] A memória de dados de vídeo 66 pode ser configurada para armazenar os dados de vídeo a serem codificados pelos componentes do codificador de vídeo 20. Os dados de vídeo na memória de dados de vídeo 101 podem ser obtidos, por exemplo, a partir a fonte de vídeo 18. O armazenador de imagem decodificada 116 pode ser uma memória de imagem de referência que armazena os dados de vídeo de referência para uso na codificação dos dados de vídeo pelo codificador de vídeo 20, por exemplo, nos modos de intra ou interprevisão. A memória de dados de vídeo 101 pode ser formada por qualquer um dentre uma variedade de dispositivos de memória, tal como uma memória de acesso randômico dinâmica (DRAM), incluindo DRAM sincronizada (SDRAM), RAM magneto resistiva (MRAM), RAM resistiva (RRAM), ou outros tipos de dispositivos de memória. A memória de dados de vídeo 101 pode ser fornecida pelo mesmo

dispositivo de memória ou dispositivos de memória separados. Em vários exemplos, a memória de dados de vídeo 101 pode estar em chip com outros componentes do codificador de vídeo 20, ou fora do chip com relação a esses componentes.

[078] Durante o processo de codificação, o codificador de vídeo 20 recebe um quadro ou fatia de vídeo a ser codificado. O quadro ou fatia pode ser dividido em múltiplos blocos de vídeo. A unidade de estimativa de movimento 42 e a unidade de compensação de movimento 44 realizam a codificação por interprevisão do bloco de vídeo recebido com relação a um ou mais blocos em um ou mais quadros de referência para fornecer previsão temporal. A unidade de intraprevisão 46 pode realizar, alternativamente, a codificação por intraprevisão do bloco de vídeo recebido com relação a um ou mais blocos vizinhos no mesmo quadro ou fatia que o bloco a ser codificado para fornecer previsão espacial. O codificador de vídeo 20 pode realizar múltiplas passagens de codificação, por exemplo, para selecionar um modo de codificação adequado para cada bloco de dados de vídeo.

[079] Ademais, a unidade de partição 48 pode dividir os blocos de dados de vídeo em sub-blocos, com base na avaliação dos esquemas de partição anteriores em passagens de codificação anteriores. Por exemplo, a unidade de partição 48 pode dividir inicialmente um quadro ou fatia em CTUs, e dividir cada uma das CTUs em sub-CUs com base na análise de distorção de taxa (por exemplo, otimização de distorção de taxa). A unidade de seleção de modo 40 pode produzir adicionalmente uma estrutura de dados de quadtree

indicativa da partição de uma CTU em sub-CUs. As CUs de nó folha do quadtree podem incluir uma ou mais PUs e uma ou mais TUs.

[080] A unidade de seleção de modo 40 pode selecionar um dos modos de previsão, intra ou inter, por exemplo, com base nos resultados de erro, e fornecer o bloco previsto resultante para o somador 50 para gerar dados residuais e para o somador 62 para reconstruir o bloco codificado para uso como um quadro de referência. A unidade de seleção de modo 40 também fornece elementos de sintaxe, tal como vetores de movimento, indicadores de intramodo, informação de partição, e outra informação de sintaxe similar, para a unidade de codificação por entropia 56.

[081] A unidade de estimativa de movimento 42 e a unidade de compensação de movimento 44 podem ser altamente integradas, mas são ilustradas separadamente para fins conceituais. A estimativa de movimento, realizada pela unidade de estimativa de movimento 42, é o processo para gerar os vetores de movimento, que estimam o movimento para os blocos de vídeo. Um vetor de movimento, por exemplo, pode indicar o deslocamento de uma PU de um bloco de vídeo dentro de um quadro de vídeo ou imagem atual com relação a um bloco de previsão dentro de um quadro de referência (ou outra unidade codificada) com relação ao bloco atual codificado dentro do quadro atual (ou outra unidade codificada). Um bloco de previsão é um bloco que é considerado muito próximo do bloco a ser codificado, em termos de diferença de pixel, que pode ser determinada pela soma da diferença absoluta (SAD), soma da diferença

quadrada (SSD) e outras métricas de diferença. Em alguns exemplos, o codificador de vídeo 20 pode calcular os valores para as posições de pixel subinteiro das imagens de referência armazenadas na memória de imagem de referência 64. Por exemplo, o codificador de vídeo 20 pode interpolar os valores das posições de um quarto de pixel, posições de um oitavo de pixel, ou outras posições de pixel fracionado da imagem de referência. Portanto, a unidade de estimativa de movimento 42 pode realizar uma busca por movimento com relação às posições de pixel inteiro e as posições de pixel fracionado e enviar um vetor de movimento com a precisão de pixel fracionado.

[082] A unidade de estimativa de movimento 42 calcula um vetor de movimento para uma PU de um bloco de vídeo em uma fatia intercodificada pela comparação da posição da PU com a posição de um bloco de previsão de uma imagem de referência. A imagem de referência pode ser selecionada a partir de uma primeira lista de imagens de referência (List0) ou uma segunda lista de imagens de referência (List1), cada uma das quais identifica uma ou mais imagens de referência armazenadas na memória de imagens de referência 64. A unidade de estimativa de movimento 42 envia o vetor de movimento calculado para a unidade de codificação por entropia 56 e a unidade de compensação de movimento 44.

[083] A compensação de movimento, realizada pela unidade de compensação de movimento 44, pode envolver coletar ou gerar o bloco de previsão com base no vetor de movimento determinado pela unidade de estimativa de movimento 42. Novamente, a unidade de estimativa de

movimento 42 e a unidade de compensação de movimento 44 podem ser integradas de forma funcional, em alguns exemplos. Depois do recebimento do vetor de movimento para a PU do bloco de vídeo atual, a unidade de compensação de movimento 44 pode localizar o bloco de previsão para o qual o vetor de movimento aponta em uma das listas de imagens de referência. O somador 50 forma um bloco de vídeo residual pela subtração de valores de pixel do bloco de previsão a partir dos valores de pixel do bloco de vídeo atual sendo codificado, formando os valores de diferença de pixel, como discutido abaixo. Em geral, a unidade de estimativa de movimento 42 realiza a estimativa de movimento com relação aos componentes de luminescência, e a unidade de compensação de movimento 44 utiliza os vetores de movimento calculados com base nos componentes de luminescência para ambos os componentes de crominância e luminescência. A unidade de seleção de modo 40 também pode gerar elementos de sintaxe associados com os blocos de vídeo e a fatia de vídeo para serem utilizados pelo decodificador de vídeo 30 na decodificação de blocos de vídeo da fatia de vídeo.

[084] A unidade de intraprevisão 46 pode intraprever um bloco atual, como uma alternativa à interprevisão realizada pela unidade de estimativa de movimento 42 e a unidade de compensação de movimento 44, como descrito acima. Em particular, a unidade de intraprevisão 46 pode determinar um modo de intraprevisão a ser utilizado para codificar um bloco atual. Em alguns exemplos, a unidade de intraprevisão 46 pode codificar um bloco atual utilizando vários modos de intraprevisão, por exemplo, durante passagens de codificação separadas, e a

unidade de intraprevisão 46 (ou unidade de seleção de modo 40, em alguns exemplos) pode selecionar um modo de intraprevisão adequado para ser utilizado para os modos testados.

[085] Por exemplo, a unidade de intraprevisão 46 pode calcular valores de distorção de taxa utilizando uma análise de distorção de taxa para os vários modos de intraprevisão testados, e selecionar o modo de intraprevisão possuindo as melhores características de distorção de taxa dentre os modos testados. A análise de distorção de taxa geralmente determina uma quantidade de distorção (ou erro) entre um bloco codificado e um bloco não codificado original que foi codificado para produzir o bloco codificado, além de uma taxa de bit (isso é, um número de bits) utilizada para produzir o bloco codificado. A unidade de intraprevisão 46 pode calcular as razões das distorções e taxas para os vários blocos codificados para determinar qual modo de intraprevisão exibe o melhor valor de distorção de taxa para o bloco.

[086] Depois de selecionar um modo de intraprevisão para um bloco, a unidade de intraprevisão 46 pode fornecer informação indicativa do modo de intraprevisão selecionado para o bloco para a unidade de codificação por entropia 56. A unidade de codificação por entropia 56 pode codificar a informação indicando o modo de intraprevisão selecionado. O codificador de vídeo 20 pode incluir na sequência de bits transmitida, dados de configuração que podem incluir uma pluralidade de tabelas de índice de modo de intraprevisão e uma pluralidade de tabelas de índice de modo de intraprevisão modificadas

(também referidas como tabelas de mapeamento de palavra código), definições de contextos de codificação para vários blocos, e indicações de um modo de intraprevisão mais provável, uma tabela de índice de modo de intraprevisão, e uma tabela de índice de modo de intraprevisão modificada para uso para cada um dos contextos.

[087] O codificador de vídeo 20 forma um bloco de vídeo residual pela subtração de dados de previsão da unidade de seleção de modo 40 a partir do bloco de vídeo original sendo codificado. O somador 50 representa o componente ou componentes que realizam essa operação de subtração. A unidade de processamento de transformação 52 aplica uma transformação, tal como uma transformação de cosseno discreto (DCT) ou uma transformação conceitualmente similar, ao bloco residual, produzindo um bloco de vídeo que compreende valores de coeficientes de transformação. As transformações de wavelet, transformações de inteiro, transformações de sub-banda, transformações de seno discreto (DSTs), ou outros tipos de transformações podem ser utilizadas em vez de uma DCT. Em qualquer caso, a unidade de processamento de transformação 52 aplica a transformação ao bloco residual, produzindo um bloco de coeficientes de transformação. A transformação pode converter a informação residual de um domínio de pixel em um domínio de transformação, tal como um domínio de frequência. A unidade de processamento de transformação 52 pode enviar os coeficientes de transformação resultantes para a unidade de quantização 54. A unidade de quantização 54 quantiza os coeficientes de transformação para reduzir ainda mais a taxa de bit. O processo de quantização 54

quantiza os coeficientes de transformação para reduzir ainda mais a taxa de bit. O processo de quantização pode reduzir a profundidade de bit associada com alguns ou todos os coeficientes. O grau de quantização pode ser modificado pelo ajuste de um parâmetro de quantização.

[088] Seguindo a quantização, a unidade de codificação por entropia 56 codifica por entropia os coeficientes de transformação quantizados. Por exemplo, a unidade de codificação por entropia 56 pode realizar a codificação de comprimento variável adaptativa de contexto (CAVLC), a codificação aritmética binária adaptativa de contexto (CABAC), codificação aritmética binária adaptativa de contexto com base em sintaxe (SBAC), codificação por entropia por divisão de intervalo de probabilidade (PIPE) ou outra técnica de codificação por entropia. No caso de codificação por entropia com base em contexto, o contexto pode ser baseado nos blocos vizinhos. Seguindo-se a codificação por entropia pela unidade de codificação por entropia 56, a sequência de bits codificada pode ser transmitida a outro dispositivo (por exemplo, decodificador de vídeo 30) ou arquivada para transmissão ou recuperação posterior.

[089] A unidade de quantização inversa 58 e a unidade de transformação inversa 60 aplicam quantização inversa e transformação inversa, respectivamente, para reconstruir o bloco residual no domínio de pixel. Em particular, o somador 62 adiciona o bloco residual reconstruído para o bloco de previsão de movimento compensado produzido mais cedo pela unidade de compensação de movimento 44 ou unidade de interprevisão 46 para

produzir um bloco de vídeo reconstruído para armazenamento na memória de imagens de referência 64. O bloco de vídeo reconstruído pode ser utilizado pela unidade de estimativa de movimento 42 e unidade de compensação de movimento 44 como um bloco de referência para intercodificar um bloco em um quadro de vídeo subsequente.

[090] A figura 3 é um diagrama em bloco ilustrando um exemplo do decodificador de vídeo 30 que pode ser configurado para realizar uma ou mais técnicas DMVD dessa descrição. Por exemplo, o decodificador de vídeo 30 pode ser configurado para aplicar uma ou mais restrições a (1) um MV derivado pela DMVD, e/ou (2) uma diferença de MV entre um MV inicial e um MV derivado pelo DMVD. Quando uma restrição é aplicada à DMVD, em determinados exemplos, apenas a informação de movimento derivada, que corresponde à restrição, é considerada informação de movimento válida. Dessa forma, a lista de MVs e o MV final selecionado podem ser diferentes caso a restrição não tenha sido utilizada.

[091] Em um primeiro exemplo, a restrição inclui se os MVs derivados são simétricos. Em um segundo exemplo, a restrição inclui se as diferenças MV entre os MVs iniciais e os MVs derivados pelos métodos DMVD são simétricas. Em um terceiro exemplo, a restrição inclui se os MVs derivados são antissimétricos. Em um quarto exemplo, a restrição inclui se as diferenças MV entre os MVs iniciais e os MVs derivados pelos métodos DMVD são antissimétricas. Em alguns exemplos, a ativação das restrições de MV/diferenças MV simétricas/antissimétricas é sinalizada explicitamente. Em outros exemplos, as restrições podem ser determinadas implicitamente de acordo

com alguma informação codificada. Em determinados exemplos, as condições podem ser impostas às restrições de modo que a restrição só se aplique em alguns exemplos.

[092] No exemplo da figura 3, o decodificador de vídeo 30 inclui a memória de dados de vídeo 68, uma unidade de decodificação por entropia 70, uma unidade de compensação de movimento 72, unidade de intraprevisão 74, unidade de quantização inversa 76, unidade de transformação inversa 78, memória de imagem de referência 82 e somador 80. O decodificador de vídeo 30 pode, em alguns exemplos, realizar uma passagem de decodificação geralmente alternada à passagem de codificação descrita com relação ao codificador de vídeo 20 (figura 2). A unidade de compensação de movimento 72 pode gerar dados de previsão com base nos vetores de movimento recebidos da unidade de decodificação por entropia 70, enquanto a unidade de intraprevisão 74 pode gerar dados de previsão com base nos indicadores de modo de intraprevisão recebidos da unidade de decodificação por entropia 70.

[093] A memória de dados de vídeo 68 pode armazenar os dados de vídeo codificados, tal como uma sequência de bits de vídeo codificada, a ser decodificada pelos componentes do decodificador de vídeo 30. Os dados de vídeo armazenados na memória de dados de vídeo 68 podem ser obtidos, por exemplo, a partir do meio legível por computador 16, por exemplo, a partir de uma fonte de vídeo local, tal como uma câmera, através de comunicação de rede com ou sem fio dos dados de vídeo, ou pelo acesso ao meio de armazenamento de dados físico. A memória de dados de vídeo 68 pode formar um armazenador de imagem codificada

(CPB) que armazena os dados de vídeo codificados a partir de uma sequência de bits de vídeo codificada. A memória de imagem de referência 82 pode ser uma memória que armazena os dados de vídeo de referência para uso na decodificação de dados de vídeo pelo decodificador de vídeo 30, por exemplo, nos modos de intra ou interprevisão, ou para envio. A memória de dados de vídeo 68 e a memória de imagens de referência 82 podem ser formadas por qualquer um dentre uma variedade de dispositivos de memória, tal como DRAM), incluindo SDRAM, MRAM, RRAM, ou outros tipos de dispositivos de memória. A memória de dados de vídeo 68 e a memória de imagens de referência 82 podem ser fornecidas pelo mesmo dispositivo de memória ou dispositivos de memória separados. Em vários exemplos, a memória de dados de vídeo 68 pode estar no chip com outros componentes do decodificador de vídeo 30 ou fora do chip com relação a esses componentes.

[094] A memória de dados de vídeo 68 recebe e armazena os dados de vídeo codificados (por exemplo, unidades NAL) de uma sequência de bits. A unidade de decodificação por entropia 70 pode receber os dados de vídeo codificados (por exemplo, unidades NAL) da memória de dados de vídeo 68 e pode analisar as unidades NAL para obter elementos de sintaxe. A unidade de decodificação por entropia 70 pode decodificar por entropia os elementos de sintaxe codificados por entropia nas unidades NAL. Durante o processo de decodificação, o decodificador de vídeo 30 recebe uma sequência de bits de vídeo codificada que representa os blocos de vídeo de uma fatia de vídeo codificada e elementos de sintaxe associados do codificador

de vídeo 20. A unidade de decodificação por entropia 70 do decodificador de vídeo 30 decodifica por entropia a sequência de bits para gerar os coeficientes quantizados, os vetores de movimento ou indicadores de modo de intraprevisão e outros elementos de sintaxe. A unidade de decodificação por entropia 70 envia os vetores de movimento para e outros elementos de sintaxe para a unidade de compensação de movimento 72. O decodificador de vídeo 30 pode receber os elementos de sintaxe no nível de fatia de vídeo e/ou nível de bloco de vídeo.

[095] Quando a fatia de vídeo é codificada como uma fatia intracodificada (I), a unidade de intraprevisão 74 pode gerar dados de previsão para um bloco de vídeo da fatia de vídeo atual com base em um modo de intraprevisão sinalizado e dados de blocos previamente decodificados do quadro ou imagem atual. Quando o quadro de vídeo é codificado como uma fatia intercodificada (isso é, B ou P), a unidade de compensação de movimento 72 produz blocos de previsão para um bloco de vídeo da fatia de vídeo atual com base nos vetores de movimento e outros elementos de sintaxe recebidos da unidade de decodificação por entropia 70. Os blocos de previsão podem ser produzidos a partir de uma das imagens de referência dentro de uma das listas de imagens de referência. O decodificador de vídeo 30 pode construir as listas de quadro de referência, List0 e List1, utilizando técnicas de construção padrão com base nas imagens de referência armazenadas na memória de imagens de referência 82. A unidade de compensação de movimento 72 determina a informação de previsão para um bloco de vídeo da fatia de vídeo atual pela análise dos vetores de

movimento e outros elementos de sintaxe, e utiliza a informação de previsão para produzir os blocos de previsão para o bloco de vídeo atual sendo decodificado. Por exemplo, a unidade de compensação de movimento 72 utiliza alguns dos elementos de sintaxe recebidos para determinar um modo de previsão (por exemplo, intraprevisão, interprevisão) utilizado para codificar os blocos de vídeo da fatia de vídeo, um tipo de fatia de interprevisão (por exemplo, fatia B ou fatia P), informação de construção para uma ou mais listas de imagens de referência para a fatia, vetores de movimento para cada bloco de vídeo intercodificado da fatia, situação de interprevisão para cada bloco de vídeo intercodificado da fatia, e outra informação para decodificar os blocos de vídeo na fatia de vídeo atual.

[096] A unidade de compensação de movimento 72 também pode realizar a interpolação com base nos filtros de interpolação. A unidade de compensação de movimento 72 pode utilizar filtros de interpolação como utilizado pelo codificador de vídeo 20 durante a codificação dos blocos de vídeo para calcular os valores interpolados para os pixels subinteiros dos blocos de referência. Nesse caso, a unidade de compensação de movimento 72 pode determinar os filtros de interpolação utilizados pelo codificador de vídeo 20 a partir dos elementos de sintaxe recebidos e utilizar os filtros de interpolação para produzir os blocos de previsão.

[097] A unidade de quantização inversa 76 quantiza de forma inversa, isso é, desquantiza, os coeficientes de transformação quantizados fornecidos na

sequência de bits e decodificados pela unidade de decodificação por entropia 70. O processo de quantização inversa pode incluir o uso de um parâmetro de quantização QP_y calculado pelo decodificador de vídeo 30 para cada bloco de vídeo na fatia de vídeo para determinar um grau de quantização e, da mesma forma, um grau de quantização inversa que deve ser aplicada.

[098] A unidade de transformação inversa 78 aplica uma transformação inversa, por exemplo, uma DCT inversa, uma transformação de inteiro inversa ou um processo de transformação inversa conceitualmente similar aos coeficientes de transformação a fim de produzir blocos residuais no domínio de pixel.

[099] Depois que a unidade de compensação de movimento 72 gera o bloco de previsão para o bloco de vídeo atual com base nos vetores de movimento e outros elementos de sintaxe, o decodificador de vídeo 30 forma um bloco de vídeo decodificado pela soma dos blocos residuais a partir da unidade de transformação inversa 78 com os blocos de previsão correspondentes gerados pela unidade de compensação de movimento 72. O somador 80 representa o componente ou componentes que realizam essa operação de soma. Se desejável, o filtro de desbloqueio também pode ser aplicado para filtrar os blocos decodificados a fim de remover os artefatos de bloqueio. Outros filtros de circuito (no circuito de codificação ou depois do circuito de codificação) também podem ser utilizados para suavizar as transições de pixel, ou de outra forma aperfeiçoar a qualidade de vídeo. Os blocos de vídeo decodificados em um quadro ou imagem determinado são, então, armazenados na

memória de imagens de referência 82, que armazena as imagens de referência utilizadas para a compensação de movimento subsequente. A memória de imagem de referência 82 também armazena o vídeo decodificado para apresentação posterior em um dispositivo de exibição, tal como o dispositivo de exibição 32 da figura 1.

[0100] De acordo com as técnicas dessa descrição, um codificador de vídeo, tal como o codificador de vídeo 20 e o decodificador de vídeo 30, pode realizar DMVD para derivar a informação de movimento para um bloco atual de dados de vídeo. Em particular, essas técnicas podem incluir qualquer uma ou todas as técnicas a seguir, sozinhas ou em qualquer combinação.

[0101] Um dos conceitos dessa descrição é aperfeiçoar a DMVD. As técnicas são elaboradas em vários aspectos diferentes destacados como discutido abaixo. As técnicas a seguir para se aperfeiçoar DMVD podem ser aplicadas individualmente. Alternativamente, qualquer combinação das técnicas descritas nessa descrição pode ser aplicada. As técnicas DMVD a seguir, aplicadas individualmente ou em qualquer combinação, podem aumentar a eficiência de codificação e/ou reduzir a taxa de bit. Por exemplo, restringir (1) um MV derivado pela DMVD, e/ou (2) uma diferença de MV entre um MV inicial e um MV derivado pela DMVD pode aumentar a eficiência de codificação e/ou reduzir a taxa de bit. Em cada um dos exemplos acima, as restrições podem ser aplicadas com base em uma condição.

[0102] Em HEVC, a unidade de codificação maior em uma fatia é chamada de bloco de árvore de codificação (CTB) ou unidade de árvore de codificação (CTU). Um CTB

contém um quadtree, os nós do qual são unidades de codificação. O tamanho de um CTB pode estar na faixa de 16 x 16 a 64 x 64 no perfil principal HEVC (apesar de tecnicamente os tamanhos de CTB de 8 x 8 poderem ser suportados). Uma unidade de codificação (CU), no entanto, pode ter o mesmo tamanho que um CTB e tão pequeno quanto 8 x 8. Cada unidade de codificação é codificada com um modo. Quando uma CU é intercodificada, pode ser adicionalmente dividida em 2 ou 4 unidades de previsão (PUs), ou se tornar apenas uma PU quando uma partição adicional não se aplica. Quando duas PUs estão presentes em uma CU, as mesmas podem ser retângulos de metade do tamanho ou dois retângulos com um tamanho de $\frac{1}{4}$ ou $\frac{3}{4}$ de CU. Quando a CU é intercodificada, um conjunto de informações de movimento está presente para cada PU. Adicionalmente, cada PU é codificada com um único modo de interprevisão para derivar o conjunto de informações de movimento.

[0103] No padrão HEVC, existem dois modos de interprevisão, chamados de modos de previsão de vetor de movimento misturado (pulo é considerado um caso especial de mistura) e avançado (AMVP) respectivamente para uma unidade de previsão (PU). Em qualquer modo AMVP ou de mistura, uma lista candidata a vetor de movimento (MV) é mantida para múltiplos previsores de vetor de movimento. Os vetores de movimento, além dos índices de referência no modo de mistura, da PU atual, são gerados escolhendo-se um candidato da lista de candidatos a MV.

[0104] A lista de candidatos a MV contém até 5 candidatos para o modo de mistura e apenas dois candidatos para o modo AMVP. Um candidato à mistura pode conter um

conjunto de informações de movimento, por exemplo, vetores de movimento correspondendo a ambas as listas de imagens de referência (List0 e List1) e os índices de referência. Se um candidato à mistura for identificado por um índice de mistura, as imagens de referência são utilizadas para a previsão dos blocos atuais, além dos vetores de movimento associados que são determinados. No entanto, sob o modo AMVP para cada direção de previsão em potencial da List0 ou List1, um índice de referência precisa ser sinalizado explicitamente, juntamente com um índice de previsão de MV (MVP) para a lista de candidatos a MV visto que o candidato a AMVP contém apenas um vetor de movimento. No modo AMVP, os vetores de movimento previstos podem ser refinados ainda mais. Como pode ser observado acima, um candidato à mistura corresponde a um conjunto inteiro de informação de movimento enquanto um candidato a AMVP contém apenas um vetor de movimento para uma direção de previsão específica e índice de referência. Os candidatos a ambos os modos são derivados de forma similar a partir dos mesmos blocos espacialmente ou temporalmente vizinhos.

[0105] O conceito das técnicas DVD serve para um codificador de vídeo (por exemplo, codificador de vídeo 20 ou decodificador de vídeo 30) para derivar a informação de movimento, tal como os vetores de movimento e direções de previsão, utilizando a informação previamente decodificada. Como utilizado aqui, o acrônimo "MV" representa vetor de movimento. List0 e List1 são duas listas que são construídas como listas de imagens no armazenador de imagens decodificadas (DPB) ou memória de imagens de referência 82 (da figura 3). Um índice chamado

de índice de imagens de referência é utilizado para identificar uma imagem em particular em uma dessas listas. Para a previsão única, uma imagem podem ser selecionada a partir de qualquer uma dessas listas. Para a previsão dupla, duas imagens são selecionadas, uma de cada lista (por exemplo, uma de List0 e outra de List1). Em abordagens atuais, os MVs de List0 e List1 derivados por DMVD são determinados de forma independente para algumas técnicas DMVD, tal como Combinação Bilateral de Gabarito e Combinação de Gabarito de conversão ascendente de taxa de quadro (FRUC). Nessa descrição, quando da derivação de MVs de previsão dupla (por exemplo, um par de MVs incluindo um MV List0 e um MV List1) com base nos MVs iniciais, uma restrição do MV simétrico/antissimétrico e/ou diferença de MV simétrica e/ou antissimétrica pode ser aplicada. Em alguns exemplos, mais de uma restrição pode ser aplicada, tal como aplicando uma restrição MV simétrica e uma restrição de diferença de MV simétrica. Uma restrição pode incluir uma ou mais condições. Em alguns exemplos, quando uma restrição é aplicada a DMVD (por exemplo, pela aplicação de uma restrição a um MV e/ou diferença MV derivada de DMVD), apenas a informação de movimento derivada, que corresponde (por exemplo, passa) à restrição pode ser considerada informação de movimento válida. Por exemplo, um MV derivado de DMVD que corresponde (por exemplo, passa) a uma restrição pode ser considerado um MV válido, e um MV derivado de DMVD que não corresponde à restrição pode ser considerado um MV inválido. Como outro exemplo, uma diferença de MV derivado de DMVD que corresponde a uma restrição pode ser considerada uma

diferença de MV válida, e uma diferença de MV derivada de DMVD que não corresponde à restrição pode ser considerada uma diferença MV inválida. Em alguns exemplos, um codificador de vídeo (por exemplo, codificador de vídeo 20 ou decodificador de vídeo 30) pode ser configurado para utilizar a informação de vetor de movimento válida para codificar os dados de vídeo (por exemplo, um bloco de dados de vídeo atual) e não utilizar informação de vetor de movimento inválida para codificar os dados de vídeo (por exemplo, um bloco de dados de vídeo atual). Em alguns exemplos, a informação de vetor de movimento válida pode, então, ser comparada com outra informação de vetor de movimento válida e a informação de vetor de movimento válida determinada como possuindo a métrica de custo mais baixa é selecionada.

[0106] Os candidatos a MV espaciais são derivados dos blocos vizinhos ilustrados na figura 4, para uma PU específica (PU0), apesar de os métodos para gerar os candidatos a partir dos blocos diferir para os modos de mistura e AMVP. No modo de mistura, até quatro candidatos a MV espaciais podem ser derivados com as ordens ilustradas na figura 4A com números, e a ordem ser a seguinte: esquerda (0, A1), acima (1, B1), direita e acima (2, B0), abaixo e à esquerda (3, A0), e acima e à esquerda (4, B2), como ilustrado na figura 4A.

[0107] Em um modo AVMP, os blocos vizinhos são divididos em dois grupos: grupo esquerdo consistindo do bloco 0 e do bloco 1, e o grupo acima consistindo de blocos 2, 3 e 4, como ilustrado na figura 4B. Para cada grupo, o candidato em potencial em um bloco vizinho referente à

mesma imagem de referência que a indicada pelo índice de referência sinalizado apresenta a maior prioridade para ser escolhido para formar um candidato final do grupo. É possível que todos os blocos vizinhos não contenham um vetor de movimento apontando para a mesma imagem de referência. Portanto, se tal candidato não puder ser encontrado, o primeiro candidato disponível será escalonado para formar o candidato final, dessa forma, as diferenças de distância temporal podem ser compensadas.

[0108] Os candidatos a preditor de vetor de movimento temporal (TMVP), se ativados e disponíveis, são adicionados à lista de candidatos a MV depois dos candidatos a vetor de movimento espaciais. O processo de derivação de vetor de movimento para o candidato TMVP é igual para ambos os modos de mistura e AMVP, no entanto, o índice de referência alvo para o candidato TMVP no modo de mistura pode ser sempre configurado para 0. A localização do bloco primário para a derivação do candidato TMVP é o bloco inferior direito fora da PU de mesma localização, como ilustrado na figura 5A como um bloco "T", para compensar a orientação para os blocos superior e esquerdo utilizados para gerar candidatos vizinhos espaciais. No entanto, se esse bloco estiver localizado fora da fileira CTB atual ou informação de movimento não estiver disponível, o bloco será substituído por um bloco central da PU.

[0109] Um vetor de movimento para o candidato TMVP é derivado da PU de mesma localização da imagem de mesma localização, indicada no nível de fatia. O vetor de movimento para a PU de mesma localização é chamado de MV de

mesma localização. Similar ao modo direto temporal em AVC, para derivar o vetor de movimento candidato a TMVP, o MV de mesma localização precisa ser escalonado para compensar as diferenças de distância temporal, como ilustrado na figura 5.

[0110] Vários aspectos do modo de mistura e do modo AMVP são discutidos abaixo.

[0111] Escalonamento de vetor de movimento: É considerado que o valor de vetores de movimento é proporcional à distância das imagens no tempo de apresentação. Um vetor de movimento associa duas imagens, a imagem de referência, e a imagem contendo o vetor de movimento (isto é, a imagem de contenção). Quando um vetor de movimento é utilizado para prever o outro vetor de movimento, a distância da imagem de contenção e a imagem de referência é calculada nos valores de Contagem de Ordem de Imagem (POC).

[0112] Para um vetor de movimento ser previsto, ambas suas imagens de contenção e imagens de referência associadas podem ser diferentes. Portanto, uma nova distância (com base em PIC) é calculada. E o vetor de movimento é escalonado com base nessas duas distâncias POC. Para um candidato vizinho espacial, as imagens de contenção para os dois vetores de movimento são iguais, enquanto as imagens de referência são diferentes. Em HEVC, o escalonamento do vetor de movimento se aplica a ambos TMVP e AMVP para candidatos espaciais e temporais vizinhos.

[0113] Geração de candidatos a vetor de movimento artificial: Se uma lista de candidatos a vetor de movimento não estiver completa (isto é, for inferior a um

número máximo predeterminado de candidatos), os candidatos a vetor de movimento artificial são gerados e inseridos no final da lista até que tenha todos os candidatos.

[0114] No modo de mistura, existem dois tipos de candidatos a MV artificiais: candidato combinado derivado apenas para fatias B e zero candidatos utilizados apenas para AMVP, se o primeiro tipo não fornecer candidatos artificiais suficientes.

[0115] Para cada par de candidatos que já está na lista de candidatos e possui informação de movimento necessária, os candidatos a vetor de movimento combinado bidirecional são derivados por uma combinação de vetor de movimento do primeiro candidato referente a uma imagem na List0 e o vetor de movimento de um segundo candidato com referência a uma imagem na List1.

[0116] Processo de preparação para inserção de candidato: Os candidatos de diferentes blocos podem surgir ao mesmo tempo, o que reduz a eficiência de uma lista de candidatos à mistura/AMVP. Um processo de treinamento é aplicado para solucionar esse problema. Compara um candidato com o outro na lista de candidatos atual para evitar inserir um candidato idêntico até determinado ponto. Para se reduzir a complexidade, apenas números limitados dos processos de preparação são aplicados, em vez de se comparar cada candidato em potência com todos os outros candidatos existentes.

[0117] Em um exemplo do software de referência JEM, existem várias ferramentas de intercodificação que derivam ou refinam o vetor de movimento (MV) para um bloco atual no lado do decodificador. Essas abordagens DMVD são

elaboradas como segue.

[0118] As figuras 6 e 7 são diagramas conceituais ilustrando os conceitos para derivação de vetor de movimento de padrão combinado (PMMVD). O modo PMMVD é um modo de mistura especial com base nas técnicas de Conversão Ascendente de Taxa de Quadro (FRUC). Com esse modo, a informação de movimento de um bloco não é sinalizada, mas derivada no lado do decodificador. Essa tecnologia foi incluída em um exemplo de JEM.

[0119] Um indicador FRUC é sinalizado para uma CU quando seu indicador de mistura é verdadeiro. Quando o indicador FRUC for falso, um índice de mistura é sinalizado e o modo de mistura regular é utilizado. Quando o indicador FRUC for verdadeiro, um indicador de modo FRUC adicional é sinalizado para indicar qual método (combinação bilateral ou combinação de gabarito) deve ser utilizada para derivar a informação de movimento para o bloco.

[0120] Durante o processo de derivação de movimento, um vetor de movimento inicial é primeiramente derivado para toda a CU com base na combinação bilateral ou combinação de gabarito. Primeiro, a lista de mistura da CU, ou as chamadas sementes PMMVD, é verificada e o candidato que resulta no menor custo de combinação (por exemplo, um custo de distorção de taxa (RD)) é selecionado como o ponto de partida. Então, uma busca local com base na combinação bilateral ou na combinação de gabarito em torno do ponto de partida é realizada e o MV que resulta no menor custo de combinação é considerado o MV para toda a CU. Subsequentemente, a informação de movimento é adicionalmente refinada no nível de sub-bloco com os

vetores de movimento CU derivados como pontos de partida.

[0121] De acordo com as presentes técnicas dessa descrição, as restrições podem ser aplicadas a vetores de movimento de CU derivados com base no fato de se, por exemplo, o MV derivado pela DMVD é simétrica ou antissimétrica, e/ou (2) uma diferença de MV entre um MV inicial e um MV derivado pela DMVD é simétrica ou antissimétrica. Em cada um dos exemplos acima, as restrições podem ser aplicadas com base em uma condição, por exemplo, em torno dos MVs iniciais ou MVs derivados.

[0122] Como ilustrado na figura 6, a combinação bilateral é utilizada para derivar a informação de movimento do bloco atual (Cur) encontrando-se a melhor combinação entre dois blocos de referência (R_0 e R_1) ao longo da trajetória de movimento do bloco atual em duas imagens de referência diferentes (Ref0 e Ref1). A trajetória de movimento pode incluir o percurso que um pixel em um bloco segue através do espaço e tempo quando considera uma sequência de imagens por exemplo, quadros de referência e o quadro atual) como um campo espaço-temporal contínuo tridimensional. Sob a consideração de uma trajetória de movimento contínua, os vetores de movimento MV0 e MV1 apontando para os dois blocos de referência (R_0 e R_1) são proporcionais às distâncias temporais entre a imagem atual (Cur) e as duas imagens de referência (Ref0 e Ref1). MVs derivados 600 e 602 são derivados utilizando-se combinação bilateral e apontam para os blocos de referência R'_0 e R'_1 , respectivamente. Como um caso especial, quando a imagem atual (Cur) está temporalmente entre as duas imagens de referência (Ref0 e Ref1) e a distância temporal da

imagem atual para as duas imagens de referência é igual, a combinação bilateral se torna um MV bidirecional espelhado. Consequentemente, os vetores de movimento derivados 600 e 602 são espalhados de forma similar.

[0123] Como ilustrado na figura 7, a combinação de gabarito é utilizada para derivar a informação de movimento do bloco atual (Cur) encontrando a melhor combinação entre um gabarito (blocos vizinhos superiores e/ou esquerdos do bloco atual) na imagem atual e um bloco (mesmo tamanho do gabarito) em uma imagem de referência (Ref0 e Ref1). Um gabarito pode incluir pixels vizinhos de um bloco que é utilizado para comparar um bloco de interesse (Cur) com as referências candidatas (R_0 com MV0 e R_1 com MV1) ou referências derivadas (R'_0 com MV 700 e R'_1 com MV 702) buscando blocos vizinhos de R_0 e R_1 . A referência mais similar é, então, utilizada como a previsão.

[0124] No codificador de vídeo 20, a decisão de se utiliza o modo de mistura FRUC para uma CU é baseada na seleção de custo RD como feito para um candidato à mistura normal. Isso é, os dois modos de combinação (combinação bilateral e combinação de gabarito) são ambos verificados para uma CU pela utilização de seleção de custo RD. O que leva ao menor custo é adicionalmente comparado com outros modos de CU. Se um modo de combinação FRUC for o mais eficiente, o indicador FRUC é configurado como verdadeiro para a CU e o modo de combinação relacionado é utilizado.

[0125] No 5o. encontro JVET, "Enhanced Template Matching in FRUC Mode," JVET-E0035, disponível em

<http://phenix.it-sudparis.eu/jvet/>, foi proposto para aperfeiçoar ainda mais a combinação de Gabarito FRUC. Um fluxograma de um modo de combinação de gabarito FRUC ilustrativo é ilustrado na figura 8. Na primeira etapa, um gabarito T_0 (e sua informação de movimento correspondente MV_0) combina com o gabarito atual T_c do bloco atual a partir das imagens de referência da $list0$. Na segunda etapa, o gabarito T_1 (e sua informação de movimento correspondente MV_1) é considerado a partir das imagens de referência de $list1$. A informação de movimento obtida MV_0 e MV_1 é utilizada para realizar a previsão dupla para gerar o previsor do bloco atual.

[0126] O modo de combinação de gabarito FRUC pode ser melhorado pela introdução da combinação de gabarito bidirecional e seleção adaptativa entre a previsão singular e a previsão dupla. As modificações ilustrativas relativas à figura 8 são sublinhadas na figura 9.

[0127] A combinação de gabarito bidirecional pode ser implementada com base na combinação de gabarito unidirecional. Como ilustrado na figura 8, um gabarito combinado T_0 é primeiramente encontrado na primeira etapa de combinação de gabarito a partir das imagens de referência $List0$ (800). Note-se que $List0$ aqui é considerado apenas um exemplo. De fato, se a $List0$ ou $List1$ utilizada na primeira etapa for adaptativa para o custo de distorção inicial entre o gabarito atual e o gabarito inicial na imagem de referência correspondente. O gabarito inicial pode ser determinado com a informação de movimento inicial do bloco atual que está disponível antes de realizar a primeira combinação de gabarito. A lista de

imagens de referência correspondendo ao custo de distorção de gabarito inicial mínimo será utilizada na primeira etapa de combinação de gabarito. Por exemplo, se o custo de distorção de gabarito inicial correspondendo à List0 não for maior do que o custo correspondente a List1, List0 é utilizada na primeira etapa de combinação de gabarito e List1 é utilizada na segunda etapa), então, o gabarito atual T_c do bloco atual é atualizado como segue:

$$T'_c = 2 * T_c - T_0$$

[0128] O gabarito atual atualizado T'_c , em vez do gabarito atual T_c , é utilizado para encontrar outro gabarito combinado T_1 a partir das imagens de referência List1 na segunda combinação de gabarito (802). Como resultado disso, o gabarito combinado T_1 é encontrado pela utilização conjunta das imagens de referência List0 e List1 (804). Esse processo de combinação é chamado de combinação de gabarito bidirecional.

[0129] A seleção entre a previsão singular e a previsão dupla para a previsão por compensação de movimento (MCP) pode ser baseada na distorção de combinação de gabarito. Como ilustrado na figura 9, durante a combinação de gabarito, distorção entre o gabarito T_- e T_c (o gabarito atual) pode ser calculada como $cost_0$ (900), o gabarito atual pode ser atualizado (902), e a distorção entre o gabarito T_1 e T'_c (o gabarito atual atualizado) pode ser calculado como $cost_1$ (904). Se $cost_0$ for inferior a $0,5 * cost_1$ (906), previsão singular com base em MV0 pode ser aplicada ao modo de combinação de gabarito FRUC (908); do contrário, a previsão dupla com base em MV0 e MV1 é aplicada (910). Note-se que $cost_0$ é comparado com $0,5 * cost_1$

visto que $cost1$ indica uma diferença entre o gabarito T_1 e T'_c (o gabarito atual atualizado), que é 2 vezes a diferença entre T_c (o gabarito atual) e sua previsão de $0,5 \cdot (T_0 + T_1)$. É notado que MCP pode ser aplicado ao refinamento de movimento de nível PU. O refinamento de movimento de nível sub-PU pode ser mantido inalterado.

[0130] De acordo com as técnicas dessa descrição, o codificador de vídeo 20 e o decodificador de vídeo 30 podem restringir os vetores de movimento derivados do modo de combinação de gabarito FRUC (ambos uni e bidirecional) no nível PU ou nível de sub-PU com base no fato de se (1) a restrição inclui se os MVs derivados são simétricos, (2) se as diferenças MV entre os MVs iniciais e os MVs derivados pelos métodos DMVD são simétricas, (3) se os MVs derivados são antissimétricos e/ou (4) se as diferenças MV entre os MVs iniciais e os MVs derivados pelos métodos DMVD são antissimétricas. Condições podem ser impostas às restrições de modo que a restrição se aplique apenas com base nos MVs iniciais ou gabarito FRUC.

[0131] A figura 10 é um diagrama conceitual ilustrando conceitos relacionados com o fluxo ótico bidirecional em um exemplo de JEM. O fluxo Ótico Bidirecional (BIO) é um refinamento de movimento no sentido de pixel que é realizado em cima da compensação de movimento no sentido de bloco em um caso de previsão dupla. Visto que BIO compensa o movimento fino dentro do bloco, a ativação de BIO resulta no aumento do tamanho de bloco para compensação de movimento. O refinamento de movimento de nível de amostra não exige uma busca exaustiva ou sinalização, visto que existe uma equação explícita que

fornece o vetor de movimento fino para cada amostra.

[0132] Deixe $I^{(k)}$ ser o valor de luminescência a partir de k de referência ($k = 0, 1$) depois do movimento do bloco de compensação, e $\partial I^{(k)} / \partial x$, $\partial I^{(k)} / \partial y$ são componentes horizontal e vertical do gradiente $I^{(k)}$, respectivamente. Assumindo-se que o fluxo ótico seja válido, o campo de vetor de movimento (v_x , v_y) é fornecido por uma equação

$$\partial I^{(k)} / \partial t + v_x \partial I^{(k)} / \partial x + v_y \partial I^{(k)} / \partial y = 0 \quad (1)$$

[0133] Combinar a equação de fluxo ótico com a interpolação Hermite para a trajetória de movimento de cada amostra permite se obter um polinômio singular de terceira ordem que combina ambas os valores de função $I^{(k)}$ e derivações $\partial I^{(k)} / \partial x$, $\partial I^{(k)} / \partial y$ nas extremidades. O valor desse polinômio com $t = 0$ é a previsão BIO:

$$\text{Pred}_{\text{BIO}} = 1/2 \cdot (I^{(0)} + I^{(1)} + v_x \cdot 2 \cdot (\tau_1 \partial I^{(1)} / \partial x - \tau_0 \partial I^{(0)} / \partial x) + v_y \cdot 2 \cdot (\tau_1 \partial I^{(1)} / \partial y - \tau_0 \partial I^{(0)} / \partial y)) \cdot \quad (2)$$

[0134] Aqui, τ_0 e τ_1 denotam a distância para os quadros de referência como ilustrado em uma figura 10. As distâncias τ_0 e τ_1 são calculadas com base no POC para Ref0 e Ref1:

$\tau_0 = \text{POC}(\text{atual}) - \text{POC}(\text{Ref0})$, $\tau_1 = \text{POC}(\text{Ref1}) - \text{POC}(\text{atual})$. Se ambas as previsões vierem da mesma direção de tempo (ambas do passado ou ambas do futuro) então, os sinais são diferentes $\tau_0 \cdot \tau_1 < 0$. Nesse caso, BIO é aplicado apenas se a previsão não vier do mesmo momento de tempo ($\tau_0 \neq \tau_1$), ambas as regiões referidas possuem o movimento diferente de

zero ($MVx_0, MVy_0, MVx_1, MVy_1 \neq 0$) e os vetores de movimento de bloco são proporcionais à distância de tempo ($MVx_0/MVx_1 = MVy_0/MVy_1 = -\tau_0/\tau_1$).

[0135] O campo de vetor de movimento (v_x, v_y) é determinado pela minimização da diferença Δ entre os valores nos pontos A e B (interseção da trajetória de movimento e planos de quadro de referência na figura 10). O modelo utiliza apenas o primeiro termo linear da expansão Taylor local para Δ :

$$\Delta = (I^{(0)} - I^{(1)}_0 + v_x(\tau_1 \partial I^{(1)} / \partial x + \tau_0 \partial I^{(0)} / \partial x) + v_y(\tau_1 \partial I^{(1)} / \partial y + \tau_0 \partial I^{(0)} / \partial y)) \quad (3)$$

[0136] Todos os valores em (1) dependem da localização da amostra (i', j'), que foi otimizada até agora. Assumindo-se que o movimento seja consistente no local circundante, minimiza-se Δ dentro da janela quadrada Ω $(2M + 1) \times (2M + 1)$ centralizada no ponto atualmente previsto (i, j):

$$(v_x, v_y) = \underset{v_x, v_y}{\operatorname{argmin}} \sum_{i', j' \in \Omega} \Delta^2[i', j'] \quad (4)$$

[0137] Para esse problema de otimização, uma solução simplificada é utilizada, criando a primeira minimização na direção vertical e, então, na direção horizontal. Resulta em

$$v_x = (s_1 + r) > m? \operatorname{clip3}(-thBIO, thBIO, -\frac{s_3}{(s_1 + r)}): 0 \quad (5)$$

$$v_y = (s_5 + r) > m? \operatorname{clip3}(-thBIO, thBIO, -\frac{s_6 - v_x s_2 / 2}{(s_5 + r)}): 0 \quad (6)$$

onde,

$$\begin{aligned}
s_1 &= \sum_{[i,j] \in \Omega} (\tau_1 \partial I^{(1)} / \partial x + \tau_0 \partial I^{(0)} / \partial x)^2; s_2 = \sum_{[i,j] \in \Omega} (I^{(1)} - I^{(0)}) (\tau_1 \partial I^{(1)} / \partial x + \tau_0 \partial I^{(0)} / \partial x) \\
s_3 &= \sum_{[i,j] \in \Omega} (\tau_1 \partial I^{(1)} / \partial x + \tau_0 \partial I^{(0)} / \partial x) (\tau_1 \partial I^{(1)} / \partial y + \tau_0 \partial I^{(0)} / \partial y) \\
s_4 &= \sum_{[i,j] \in \Omega} (\tau_1 \partial I^{(1)} / \partial y + \tau_0 \partial I^{(0)} / \partial y)^2; s_5 = \sum_{[i,j] \in \Omega} (I^{(1)} - I^{(0)}) (\tau_1 \partial I^{(1)} / \partial y + \tau_0 \partial I^{(0)} / \partial y)
\end{aligned} \tag{7}$$

[0138] A fim de se evitar a divisão por zero ou por um valor muito pequeno, os parâmetros de regularização r e m são introduzidos nas equações (2), (3).

$$r = 500 \cdot 4^{d-8} \tag{8}$$

$$m = 700 \cdot 4^{d-8} \tag{9}$$

[0139] Aqui d é a profundidade interna do bit do vídeo de entrada.

[0140] Em alguns exemplos, o refinamento MV de BIO pode ser pouco confiável devido a ruído ou movimento irregular. Portanto, em BIO, a magnitude do refinamento MV é restrita a um determinado limite (thBIO). O valor limite é determinado com base no fato de se todas as imagens de referência da imagem atual vêm todos de uma direção. Se todas as imagens de referência das imagens atuais da imagem atual forem de uma direção, o valor do limite é configurado para $12 \times 2^{14-d}$, do contrário, é configurado para $12 \times 2^{13-d}$.

[0141] Os gradientes para BIO são calculados ao mesmo tempo com a interpolação de compensação de movimento utilizando operações consistentes com o processo de compensação de movimento HEVC (Resposta a Impulso Finito separável em 2D (FIR)). A entrada para essa FIR separável em 2D é a mesma amostra de quadro de referência que o do processo de compensação de movimento e posição fracionada ($fracX$, $fracY$), de acordo com a parte fracionada do vetor de movimento de bloco. No caso do gradiente horizontal, o

sinal $\partial I / \partial x$ interpolado primeiro verticalmente utilizando *BIOfilterS* correspondendo à posição fracionada *fracY* com a mudança de de-scaling d-8, então, o filtro de gradiente *BIOfilterG* é aplicado na direção horizontal correspondendo à posição fracionada *fracX* com a mudança de de-scaling 18-d. No caso do gradiente vertical $\partial I / \partial y$, o primeiro filtro de gradiente é aplicado verticalmente utilizando *BIOfilterG* correspondente à posição fracionada *fracY* com a mudança de de-scaling d-8, então o deslocamento do sinal é realizado utilizando-se *BIOfilterS* na direção horizontal correspondente à posição fracionada *fracX* com mudança de de-scaling por 18-d. O comprimento do filtro de interpolação para cálculo de gradientes *BIOfilterG* e deslocamento de sinal *BIOfilterF* é mais curto (6 saídas) a fim de manter a complexidade razoável. A Tabela 1 ilustra os filtros utilizados para o cálculo de gradientes para diferentes posições fracionadas do vetor de movimento de bloco em BIO. A Tabela 2 ilustra os filtros de interpolação utilizados para a geração do sinal de previsão na BIO.

[0142] A figura 11 é um diagrama conceitual ilustrando um exemplo do cálculo de gradiente para um bloco de 8 x 4. Para um bloco de 8 x 4, um codificador de vídeo recolhe os previsores de movimento compensado e calcula os gradientes horizontal/vertical (HOR/VER) de todos os pixels dentro de um bloco atual, além de duas linhas externas dos pixels, visto que solucionar v_x e v_y para cada pixel precisa dos valores de gradiente HOR/VER e os previsores de movimento compensados dos pixels, dentro da janela Ω , centralizada em cada pixel como ilustrado na equação (4). E, em um exemplo de JEM, o tamanho dessa janela é

configurado para 5 x 5. Portanto, o codificador de vídeo (por exemplo, codificador de vídeo 20 e/ou decodificador de vídeo 30) recolhe os previsores de movimento compensado e calcula os gradientes para as duas linhas externas dos pixels em torno dos pontos A e B.

Tabela 1: Filtros para cálculo de gradientes em BIO

| posição de pixel fracionado | filtro de interpolação para o gradiente (BIOfilterG) |
|-----------------------------|--|
| 0 | { 8, -39, -3, 46, -17, 5} |
| 1/16 | { 8, -32, -13, 50, -18, 5} |
| 1/8 | { 7, -27, -20, 54, -19, 5} |
| 3/16 | { 6, -21, -29, 57, -18, 5} |
| $\frac{1}{4}$ | { 4, -17, -36, 60, -15, 4} |
| 5/16 | { 3, -9, -44, 61, -15, 4} |
| 3/8 | { 1, -4, -48, 61, -13, 3} |
| 7/16 | { 0, 1, -54, 60, -0, 2} |
| 1/2 | { 1, 4, -57, 57, -4, -1} |

Tabela 2: Filtros de interpolação para geração de sinal de previsão em BIO

| Posição de pixel fracionado | filtro de interpolação para o sinal de previsão (BIOfilterS) |
|-----------------------------|--|
| 0 | { 0, 0, 64, 0, 0, 0} |
| 1/16 | { 1, -3, 64, 4, -2, 0} |
| 1/8 | { 1, -6, 62, 9, -3, 1} |
| 3/16 | { 2, -8, 60, 14, -5, 1} |
| $\frac{1}{4}$ | { 2, -9, 57, 19, -7, 2} |
| 5/16 | { 3, -10, 53, 24, -8, 2} |
| 3/8 | { 3, -11, 50, 29, -9, 2} |
| 7/16 | { 3, -11, 44, 35, -10, 3} |
| 1/2 | { 1, -7, 38, 38, -7, 1} |

[0143] Nos exemplos de JEM, BIO é aplicado a todos os blocos previstos de forma bidirecional quando as duas previsões são de diferentes imagens de referência. Quando a compensação de iluminação local (LIC) é ativada

para uma CU, BIO é desativado.

[0144] De acordo com as técnicas dessa descrição, o codificador de vídeo 20 e o decodificador de vídeo 30 podem restringir os vetores de movimento derivados do BIO com base no fato de se (1) a restrição inclui se os MVs derivados são simétricos, (2) se as diferenças de MV entre os MVs iniciais e os MVs derivados pelos métodos DMVD são simétricas, (3) se os MVs derivados são antissimétricos e/ou (4) se as diferenças de MV entre os MVs iniciais e os MVs derivados pelos métodos DMVD são antissimétricos. As condições podem ser impostas às restrições de modo que a restrição só se aplique com base nos MVs iniciais.

[0145] A figura 12 é um diagrama conceitual ilustrando os conceitos relacionados com a derivação de vetor de movimento de lado de decodificador proposta (DMVD) com base na combinação de gabarito bilateral. Um codificador de vídeo pode gerar um gabarito bilateral como a combinação ponderada de dois blocos de previsão (R_0 e R_1), a partir de MV0 inicial de List0 e MV1 de List1, respectivamente, como ilustrado na figura 12. Em um exemplo, um gabarito bilateral pode ser gerado pela realização da média dos blocos de referência (R_0 e R_1) de List0 e List1 utilizando a fórmula: $(R_0 + R_1) \gg 1$ ou $(R_0 + R_1)/2$.

[0146] A operação de combinação de gabarito inclui calcular as medições de custo entre o gabarito gerado e a região de amostra (em torno do bloco de previsão inicial) na imagem de referência. Para cada uma de duas imagens de referência (R_0 e R_1), o MV que resulta em um custo mínimo de gabarito é considerado o MV atualizado

dessa lista para substituir o MV inicial (MV0 da List0 ou MV1 da List1). Finalmente, os dois novos MVs, isso é, MV0' e MV1', como ilustrado na figura 12, são utilizados para a previsão dupla regular. Como é comumente utilizado na estimativa de movimento de combinação de bloco, a soma das diferenças absolutas (SAD) é utilizada como a medida do custo.

[0147] A derivação de vetor de movimento de lado de decodificador (DMVD) pode ser aplicada ao modo de mistura da previsão dupla com uma dentre a imagem de referência no passado e outra da imagem de referência no futuro, sem a transmissão do elemento de sintaxe adicional. Em JEM 4.0, quando um candidato à mistura sub-CU, LIC, Affine, ou FRUC é selecionado para uma CU, a DMVD não é aplicada.

[0148] De acordo com as técnicas dessa descrição, o codificador de vídeo 20 e o decodificador de vídeo 30 podem restringir os vetores de movimento derivados da combinação bilateral de gabarito com base em DMVD, com base no fato de se (1) a restrição inclui se os MVs derivados são simétricos, (2) se as diferenças de MV entre os MVs iniciais e os MVs derivados pelos métodos DMVD são simétricas, (3) se os MVs derivados são antissimétricos, e/ou (4) se as diferenças de MV entre os MVs iniciais e os MVs derivados pelos métodos DMVD são antissimétricas. As condições podem ser impostas às restrições de modo que a restrição só se aplique com base nos MVs iniciais ou no gabarito bilateral.

[0149] As figuras 13A e 13B são diagramas conceituais ilustrando os conceitos relacionados com a

compensação de movimento em bloco sobreposto (OBMC) nos exemplos de JEM. OBMC tem sido utilizada para as gerações anteriores de padrões de vídeo, por exemplo, como em H.263. Nos exemplos de JEM, OBMC é realizada para os limites de bloco de Movimento Compensado (MC) exceto os limites direito e inferior de uma CU. Ademais, é aplicado a ambos os componentes de luminescência e crominância. Em um exemplo de JEM, um bloco MC corresponde a um bloco de codificação. Quando uma CU é codificada com o modo sub-CU (incluindo um modo de mistura sub-CU, Affine e FRUC), cada sub-bloco da CU é um bloco MC. Para processar os limites de CU de maneira uniforme, a OBMC é realizada no nível de sub-bloco para todos os limites de bloco MC, onde o tamanho de sub-bloco é configurado igual a 4×4 , como ilustrado nas figuras 13A e 13B.

[0150] O codificador de vídeo 20 e o decodificador de vídeo 30 podem aplicar OBMC ao sub-bloco atual, além de vetores de movimento atual, vetores de movimento de quatro sub-blocos vizinhos conectados, se disponíveis, e não são idênticos ao vetor de movimento atual, também são utilizados para derivar o bloco de previsão para o sub-bloco atual. Esses múltiplos blocos de previsão com base em múltiplos vetores de movimento são combinados para gerar o sinal de previsão final do sub-bloco atual.

[0151] As figuras 14A a 14D são diagramas conceituais ilustrando as ponderações OBMC. As figuras 14A a 14D ilustram um bloco de previsão com base em vetores de movimento de um sub-bloco vizinho denotado como P_N , com N indicando um índice para os sub-blocos vizinhos acima

(figura 14A), abaixo (figura 14B), à esquerda (figura 14C) e à direita (figura 14D) e o bloco de previsão com base nos vetores de movimento do sub-bloco atual é denotado como P_c . Quando P_N é baseado na informação de movimento de um sub-bloco vizinho que contém a mesma informação de movimento que o sub-bloco atual, o OBMC não é realizado a partir de P_N . Do contrário, cada pixel de P_N é adicionado ao mesmo pixel em P_c , isso é, quatro fileiras/colunas de P_N são adicionadas a P_c . Os fatores de ponderação $\{1/4, 1/8, 1/16, 1/32\}$ são utilizados para P_N e os fatores de ponderação $\{3/4, 7/8, 15/16, 31/32\}$ são utilizados para P_c . A exceção são blocos MC pequenos (isso é, quando altura ou largura do bloco de codificação é igual a 4 ou uma CU é codificada com o modo de sub-CU), para os quais apenas duas fileiras/colunas de P_N são adicionadas ao P_c . Nesse caso, os fatores de ponderação $\{1/4, 1/8\}$ são utilizados para P_N e os fatores de ponderação $\{3/4, 7/8\}$ são utilizados para P_c . Para P_N gerado com base em vetores de movimento de sub-blocos vizinhos verticalmente (horizontalmente), os pixels na mesma fileira (coluna) de P_N são adicionados a P_c com um mesmo fator de ponderação. É notado que BIO também é aplicado para derivar o bloco de previsão P_N .

[0152] Nos exemplos de JEM, para uma CU com um tamanho inferior a ou igual a 256 amostras de luminescência, um indicador de nível de CU é sinalizado para indicar se OBMC é aplicado ou não à CU atual. Para as CUs com um tamanho superior a 256 amostras de luminescência ou não codificadas com o modo AMVP, OBMC é aplicado por definição. No codificador, quando OBMC é aplicado a uma CU, seu impacto é levado em consideração durante o estágio de

estimativa de movimento. O sinal de previsão, pela utilização da informação de movimento do bloco vizinho superior e bloco vizinho esquerdo, é utilizado para compensar os limites superior e esquerdo do sinal original da CU atual, e, então, o processo de estimativa de movimento normal é aplicado.

[0153] De acordo com as técnicas dessa descrição, o codificador de vídeo 20 e o decodificador de vídeo 30 podem restringir os vetores de movimento derivados do OBMC com base no fato de se (1) as restrições incluem se os MVs derivados são simétricos, (2) se as diferenças de MV entre os MVs iniciais e os MVs derivados pelos métodos DMVD são simétricas, (3) se os MVs derivados são antissimétricos, e/ou (4) se as diferenças de MV entre os MVs iniciais e os MVs derivados pelos métodos DMVD são antissimétricas. As condições podem ser impostas às restrições de modo que a restrição se aplique apenas com base e MVs iniciais.

[0154] Métodos relacionados por DMVD (por exemplo, BIO, Combinação Bilateral FRUC, Combinação de Gabarito FRUC, combinação de Gabarito Bilateral e assim por diante) podem fornecer reduções de taxa de bit significativas. Essa descrição apresenta várias técnicas que podem aperfeiçoar adicionalmente DMVD, que pode aumentar a eficiência da codificação e/ou reduzir a taxa de bit. Por exemplo, restringir (1) um MV derivado pelo DMVD, e/ou (2) uma diferença de MV entre um MV inicial e um MV derivado pela DMVD pode aumentar a eficiência da codificação e/ou a reduzir a taxa de bit.

[0155] A tabela 3 abaixo descreve várias

terminologias utilizadas.

Tabela 3

| | |
|---|---|
| L0_POC_Init, L0_MV_X_Init, L0_MV_Y_Init, L0_POC_Init, L1_MV_X_Init, L1_MV_Y_Init | Os componentes X e Y dos MVs iniciais na List0 e List1 A contagem de ordem de imagem inicial para List0 e List1. A contagem de ordem de imagem pode ser POC como definido em HEVC. |
| L0_POC_Derived, L0_MV_X_Derived, L0_MV_Y_Derived, L1_POC_Derived, L1_MV_X_Derived | Os componentes X e Y dos MVs derivados pelos métodos DMVD para L0 e L1 |
| L1_MV_Y_Derived | A contagem da ordem de imagem derivada para List0 e List1. A contagem da ordem de imagem pode ser POC como definido em HEVC |
| L0_MV_X_Diff, L0_MV_Y_Diff, L1_MV_X_Diff, L1_MV_Y_Diff | Os componentes X e Y das diferenças de MV entre os MVs iniciais e os MVs derivados pelos métodos DMVD de List0 e List1. |
| CURR_POC | Contagem de Ordem de Imagem (por exemplo, POC como definido em HEVC) da fatia/imagem atual |

[0156] Vetores de Movimento Simétrico - Em alguns exemplos, MVs derivados por DMVD podem ser restringidos por um codificador de vídeo (por exemplo, codificador de vídeo 20 ou decodificador de vídeo 30), de modo que os MVs derivados sejam simétricos. Em um exemplo, para ser simétrico, um codificador de vídeo (por exemplo, o codificador de vídeo 20 ou decodificador de vídeo 30) pode ser configurado para restringir os MVs derivados, de modo

que os MVs derivados correspondam a uma ou mais (por exemplo, todas) das seguintes condições com relação um ao outro:

1. $L0_MV_X_Derived*(CURR_POC_L1_POC_Derived) = L1_MV_X_Derived*(CURR_POC_L0_POC_Derived)$, e/ou
2. $L0_MV_Y_Derived*(CURR_POC_L1_POC_Derived) = L1_MV_Y_Derived*(CURR_POC_L0_POC_Derived)$.

[0157] Em alguns exemplos, as condições (1) e (2) acima podem, cada uma, ser consideradas uma restrição MV simétrica separada, ou pode ser considerada uma restrição MV simétrica singular quando ambas forem aplicadas. A esse respeito, referência a uma restrição sendo ativada aqui pode fazer referência a uma condição sendo ativada, que opera como uma restrição. Por exemplo, referência a uma restrição de MV simétrica sendo ativada pode incluir (i) a condição (1) acima sendo ativada, (ii) a condição (2) acima sendo ativada, ou (iii) as condições (1) e (2) sendo ativadas.

[0158] Em alguns exemplos, para se alcançar simplificações adicionais e/ou aperfeiçoamentos de codificação, os MVs derivados pela DMVD podem ser restringidos por um codificador de vídeo (por exemplo, codificador de vídeo 20 ou decodificação de vídeo 30), de modo que os MVs derivados possam precisar corresponder a uma ou mais (por exemplo, todas, condições (a) e/ou (b) quando a condição (1) for verdadeira, e/ou condições (c) e/ou (d), quando a condição (2) for verdadeira) das condições abaixo (que podem ser referidas como condições pseudo-simétricas):

1. Quando as imagens de referência List0 e List1

estiverem ambas antes ou ambas depois da imagem atual (por exemplo,

$(\text{CURR_POC_L0_POC_Derived}) * (\text{CURR_POC_L1_POC_Derived}) > 0$),

a. $\text{L0_MV_X_Derived} = \text{L1_MV_X_Derived}$, e/ou

b. $\text{L0_MV_Y_Derived} = \text{L1_MV_Y_Derived}$;

e/ou

2. Quando as imagens de referência de List0 e List1 não estão, ambas, antes ou depois da imagem atual (por exemplo,

$(\text{CURR_POC_L0_POC_Derived}) * (\text{CURR_POC_L1_LOC_Derived}) < 0$);

c. $\text{L0_MV_X_Derived} = -1 * \text{L1_MV_X_Derived}$, e/ou

d. $\text{L0_MV_Y_Derived} = -1 * \text{L1_MV_Y_Derived}$.

[0159] Em alguns exemplos, um codificador de vídeo (por exemplo, o codificador de vídeo 20 ou o decodificador de vídeo 30) pode ser configurado para verificar ou determinar uma diferença POC sendo configurada para uso, por exemplo, a informação POC dos MVs iniciais e/ou informação POC dos MVs derivados. Por exemplo, um codificador de vídeo (por exemplo, codificador de vídeo 20 ou decodificador de vídeo 30) pode ser configurado para verificar ou determinar uma diferença POC com base, por exemplo, na informação POC dos MVs iniciais e/ou informação POC dos MVs derivados.

[0160] Diferenças de Vetor de Movimento Simétrico - Em alguns exemplos, as diferenças de MV entre os MVs iniciais e os MVs derivados por DMVD podem ser restringidas por um codificador de vídeo (por exemplo, codificador de vídeo 20 ou decodificador de vídeo 30), de modo que as diferenças de MV sejam simétricas. A diferença

de MV é a diferença entre os MVs iniciais e os MVs derivados por DMVD. A diferença de MV pode ser determinada com base em componente X e Y. Em um exemplo, para ser simétrico, um codificador de vídeo (por exemplo, codificador de vídeo 20 ou decodificador de vídeo 30) pode ser configurado para exigir que as diferenças de MV derivado correspondam a uma ou mais (por exemplo, todas) as condições a seguir:

1. $LO_MV_X_Diff * (CURR_POC - L1_POC_Derived) = L1_MV_X_Diff * (CURR_POC - L0_POC_Derived)$, e/ou
2. $L0_MV_Y_Diff * (CURR_POC - L1_POC_Derived) = L1_MV_Y_Diff * (CURR_POC - L0_POC_Derived)$.

[0161] Em alguns exemplos, para se alcançar simplificações e/ou aperfeiçoamentos de codificação adicionais, as diferenças MV derivadas pela DMVD podem ser restringidas por um codificador de vídeo (por exemplo, codificador de vídeo 20 ou decodificador de vídeo 30), de modo que as diferenças de MV derivadas possam precisar corresponder a uma ou mais (por exemplo, todas, condições (a) e/ou (b), quando a condição (1) for verdadeira, e/ou condições (c) e/ou (d), quando a condição (2) for verdadeira) das condições abaixo (que podem ser referidas como condições pseudo-simétricas).

1. Quando as imagens de referência List0 e List1 estão ambas antes ou ambas depois da imagem atual (por exemplo, $(CURR_POC_L0_POC_Init) * (CURR_POC - L1_POC_Init) > 0$),

- a. $L0_MV_X_Diff = L1_MV_X_Diff$; e/ou
- b. $L0_MV_Y_Diff = L1_MV_Y_Diff$;
- e/ou

2. Quando as imagens de referência List0 e List1 NÃO ESTÃO ambas antes ou ambas depois da imagem atual (por exemplo (por exemplo, $(CURR_POC - L0_POC_Init) * (CURR_POC - L1_POC_Init) < 0$),

c. $L0_MV_X_Diff = -1 * L1_MV_X_Diff$, e/ou

d. $L0_MV_Y_Diff = -1 * L1_MV_Y_Diff$.

[0162] Em alguns exemplos, um codificador de vídeo (por exemplo, o codificador de vídeo 20 ou decodificador de vídeo 30) pode ser configurado para verificar ou determinar uma diferença de POC sendo configurado para uso, por exemplo, a informação POC dos MVs iniciais e/ou a informação de POC dos MVs derivados. Por exemplo, um codificador de vídeo (por exemplo, codificador de vídeo 20 ou decodificador de vídeo 30) pode ser configurado para verificar ou determinar uma diferença de POC com base, por exemplo, na informação POC dos MVs iniciais e/ou informação POC dos MVs derivados.

[0163] Vetores de Movimento Antissimétrico - Em alguns exemplos, os MVs derivados pela DMVD podem ser restringidos por um codificador de vídeo (por exemplo, o codificador de vídeo 20 ou o decodificador de vídeo 30), de modo que os MVs derivados devam ser antissimétricos. Em um exemplo, para ser antissimétrico, um codificador de vídeo (por exemplo, o codificador de vídeo 20 ou o decodificador de vídeo 30) pode ser configurado para exigir que os MVs derivados correspondam a uma ou mais (por exemplo, todas) as condições a seguir:

1. $L0_MV_X_Derived * (CURR_POC - L1_POC_Derived)$
 $= -1 * L1_MV_X_Derived * (CURR_POC - L0_POC_Derived)$,
 e/ou

$$2. \text{L0_MV_Y_Derived} * (\text{CURR_POC} - \text{L1_POC_Derived}) \\ = -1 * \text{L1_MV_Y_Derived} * (\text{CURR_POC} - \text{L0_POC_Derived}).$$

[0164] Em alguns exemplos, para se alcançar as simplificações adicionais e/ou aperfeiçoamentos de codificação, as diferenças de MV derivadas por DMVD podem ser restringidas por um codificador de vídeo (por exemplo, o codificador de vídeo 20 ou o decodificador de vídeo 30), de modo que as diferenças de MV derivadas podem precisar corresponder a uma ou mais (por exemplo, todas, condições (a) e/ou (b), quando a condição (1) é verdadeira, e/ou condições (c) e/ou (d), quando a condição (2) é verdadeira) das condições abaixo (que podem ser referidas como condições pseudo-antissimétricas):

1. Quando as imagens de referência List0 e List1 estão ambas antes ou ambas depois da imagem atual (por exemplo, $(\text{CURR_POC} - \text{L0_POC_Derived}) * (\text{CURR_POC} - \text{L1_POC_Derived}) > 0$),

a. $\text{L0_MV_X_Derived} = -1 * \text{L1_MV_X_Derived}$, e/ou

b. $\text{L0_MV_Y_Derived} = -1 * \text{L1_MV_Y_Derived}$;

e/ou

2. Quando as imagens de referência de List0 e List1 NÃO ESTÃO ambas antes ou ambas depois da imagem atual (por exemplo, $(\text{CURR_POC} - \text{L0_POC_Derived}) * (\text{CURR_POC} - \text{L1_POC_Derived}) < 0$),

c. $\text{L0_MV_X_Derived} = \text{L1_MV_X_Derived}$, e/ou

d. $\text{L0_MV_Y_Derived} = \text{L1_MV_Y_Derived}$.

[0165] Em alguns exemplos, um codificador de vídeo (por exemplo, codificador de vídeo 20 ou decodificador de vídeo 30) pode ser configurado para verificar ou determinar uma diferença POC sendo configurada

para uso, por exemplo, a informação POC dos MVs iniciais e/ou informação POC dos MVs derivados. Por exemplo, um codificador de vídeo (por exemplo, codificador de vídeo 20 ou decodificador de vídeo 30) pode ser configurado para verificar ou determinar uma diferença POC com base, por exemplo, na informação POC dos MVs iniciais e/ou informação POC dos MVs derivados.

[0166] Diferenças de Vetor de Movimento Antissimétricas - Em alguns exemplos, as diferenças de MV entre os MVs iniciais e os MVs derivados por DMVD podem ser restringidas por um codificador de vídeo (por exemplo, codificador de vídeo 20 ou decodificador de vídeo 30), de modo que as diferenças de MV sejam antissimétricas. Em um exemplo, para ser antissimétrico, um codificador de vídeo (por exemplo, o codificador de vídeo 20 ou o decodificador de vídeo 30) pode ser configurado para exigir que as diferenças MV derivadas correspondam a uma ou mais (por exemplo, todas) as condições a seguir:

1. $L0_MV_X_Diff * (CURR_POC - L1_POC_Derived) = -1 * L1_MV_X_Diff * (CURR_POC - L0_POC_Derived)$, e/ou
2. $L0_MV_Y_Diff * (CURR_POC - L1_POC_Derived) = -1 * L1_MV_Y_Diff * (CURR_POC - L0_POC_Derived)$.

[0167] Em alguns exemplos, para se alcançar simplificações adicionais e/ou aperfeiçoamentos de codificação, as diferenças MV derivadas por DMVD podem ser restringidas por um codificador de vídeo (por exemplo, o codificador de vídeo 20 ou o decodificador de vídeo 30), de modo que as diferenças de MV derivadas possam precisar corresponder a uma ou mais (por exemplo, todas, condições (a) e/ou (b) quando a condição (1) for verdadeira e/ou

condições (c) e/ou (d) quando a condição (2) for verdadeira) das condições abaixo (que podem ser referidas como condições pseudo-antissimétricas):

1. Quando as imagens de referência de List0 e List1 estão ambas antes ou ambas depois da imagem atual (por exemplo, $(\text{CURR_POC} - \text{L0_POC_Init}) * (\text{CURR_POC} - \text{L1_POC_Init}) > 0$),

a. $\text{L0_MV_X_Diff} = -1 * \text{L1_MV_X_Diff}$, e/ou

b. $\text{L0_MV_Y_Diff} = -1 * \text{L1_MV_Y_Diff}$;

e/ou

2. Quando as imagens de referência List0 e List1 NÃO ESTÃO ambas antes ou ambas depois da imagem de referência (por exemplo, $(\text{CURR_POC} - \text{L0_POC_Init}) * (\text{CURR_POC} - \text{L1_POC_Init}) < 0$),

c. $\text{L0_MV_X_Diff} = \text{L1_MV_X_Diff}$, e/ou

d. $\text{L0_MV_Y_Diff} = \text{L1_MV_Y_Diff}$.

[0168] Em alguns exemplos, um codificador de vídeo (por exemplo, codificador de vídeo 20 ou decodificador de vídeo (30) podem ser configurados para verificar ou determinar uma diferença POC sendo configurada para uso, por exemplo, a informação POC dos MVs iniciais e/ou a informação POC dos MVs derivados. Por exemplo, um codificador de vídeo (por exemplo, codificador de vídeo 20 ou decodificador de vídeo 30) pode ser configurado para verificar ou determinar uma diferença POC com base, por exemplo, na informação POC dos MVs iniciais e/ou informação POC dos MVs derivados.

[0169] Condições para Aplicar as Restrições de MV/Diferenças de MV - A ativação das restrições (por exemplo, restrições de MV/diferença de MV

simétricas/antissimétricas) na informação de movimento DMVD pode, em alguns exemplos, ser sinalizada explicitamente ou pode ser determinada implicitamente de acordo com alguma informação codificada. Em alguns exemplos, a ativação da sinalização de uma restrição ou a determinação da ativação de uma restrição podem ser respectivamente descritas como sinalização que uma restrição recebeu ou está prestes a receber ou determinação de que a restrição foi ou está para ser aplicada.

[0170] Em alguns exemplos, um codificador de vídeo (por exemplo, o codificador de vídeo 20) pode ser configurado para ativar uma ou mais das restrições de MV e/ou diferença de MV descritas aqui. Por exemplo, o codificador de vídeo (por exemplo, o codificador de vídeo 20) pode ser configurado para sinalizar um ou mais valores indicativos de se uma ou mais restrições estão ativas.

[0171] Como um exemplo, um codificador de vídeo (por exemplo, codificador de vídeo 20) pode ser configurado para sinalizar um valor em uma sequência de bits correspondendo a um primeiro elemento de sintaxe, onde o valor correspondente ao primeiro elemento de sintaxe é indicativo de se uma ou mais restrições de MV simétricas foram ativadas. O valor pode ser um valor verdadeiro (por exemplo, um valor igual a 1) para indicar a ativação ou um valor falso (por exemplo, um valor igual a 0) para indicar a desativação, ou o valor pode ser um valor falso (por exemplo, um valor igual a 0) para indicar a ativação ou um valor falso (por exemplo, um valor igual a 1) para indicar a desativação. Em alguns exemplos, um codificador de vídeo (por exemplo, o codificador de vídeo 20) pode ser

configurado para sinalizar um valor correspondente ao primeiro elemento de sintaxe apenas quando uma ou mais restrições de MV simétricas são ativadas. Em outros exemplos, um codificador de vídeo (por exemplo, codificador de vídeo 20) pode ser configurado para sinalizar um valor correspondente ao primeiro elemento de sintaxe apenas quando uma ou mais restrições de MV simétricas forem ativadas.

[0172] Como outro exemplo, um codificador de vídeo (por exemplo, o codificador de vídeo 20) pode ser configurado para sinalizar um valor em uma sequência de bits correspondendo a um segundo elemento de sintaxe, onde o valor correspondente ao segundo elemento de sintaxe é indicativo de se uma ou mais restrições de diferença de MV simétricas foram ativadas. O valor pode ser um valor verdadeiro (por exemplo, um valor igual a 1) para indicar a ativação ou um valor falso (por exemplo, um valor igual a 0) para indicar a desativação, o valor pode ser um valor falso (por exemplo, um valor igual a 0) para indicar a ativação ou um valor falso (por exemplo, um valor igual a 1) para indicar a desativação. Em alguns exemplos, um codificador de vídeo (por exemplo, codificador de vídeo 20) pode ser configurado para sinalizar um valor correspondente ao segundo elemento de sintaxe apenas quando uma ou mais restrições de diferença de MV simétricas são ativadas. Em outros exemplos, um codificador de vídeo (por exemplo, o codificador de vídeo 20) pode ser configurado para sinalizar um valor correspondente ao segundo elemento de sintaxe apenas quando uma ou mais restrições de diferença de MV simétricas forem desativadas.

[0173] Como outro exemplo, um codificador de vídeo (por exemplo, o codificador de vídeo 20) pode ser configurado para sinalizar um valor em uma sequência de bits correspondendo a um terceiro elemento de sintaxe, onde o valor correspondente ao terceiro elemento de sintaxe é indicativo de se uma ou mais restrições de MV antissimétricas foram ativadas. O valor pode ser um valor verdadeiro (por exemplo, um valor igual a 1) para indicar a ativação ou um valor falso (por exemplo, um valor igual a 0) para indicar a desativação, ou o valor pode ser um valor falso (por exemplo, um valor igual a 0) para indicar a ativação ou um valor verdadeiro (por exemplo, um valor igual a 1) para indicar a desativação. Em alguns exemplos, um codificador de vídeo (por exemplo, o codificador de vídeo 20) pode ser configurado para sinalizar um valor correspondente ao terceiro elemento de sintaxe apenas quando uma ou mais restrições de MV antissimétricas são ativadas. Em outros exemplos, um codificador de vídeo (por exemplo, o codificador de vídeo 20) pode ser configurado para sinalizar um valor correspondente ao terceiro elemento de sintaxe apenas quando uma ou mais restrições de MV antissimétricas são desativadas.

[0174] Como outro exemplo, um codificador de vídeo (por exemplo, o codificador de vídeo 20) pode ser configurado para sinalizar um valor em uma sequência de bits correspondendo a um quarto elemento de sintaxe, onde o valor correspondente ao quarto elemento de sintaxe é indicativo de se uma ou mais restrições de diferença de MV antissimétricas foram ativadas. O valor pode ser um valor verdadeiro (por exemplo, um valor igual a 1) para indicar a

ativação ou um valor falso (por exemplo, um valor igual a 0) para indicar a desativação, ou o valor pode ser um valor falso (por exemplo, um valor igual a 0) para indicar a desativação ou um valor verdadeiro (por exemplo, um valor igual a 1) para indicar a desativação. Em alguns exemplos, um codificador de vídeo (por exemplo, codificador de vídeo 20) pode ser configurado para sinalizar um valor correspondente ao quarto elemento de sintaxe apenas quando uma ou mais restrições de diferença de MV antissimétricas são ativadas. Em outros exemplos, um codificador de vídeo (por exemplo, o codificador de vídeo 20) pode ser configurado para sinalizar um valor correspondente ao quarto elemento de sintaxe apenas quando uma ou mais restrições de diferença de MV antissimétricas forem ativadas.

[0175] Em alguns exemplos, um codificador de vídeo (por exemplo, decodificador de vídeo 30) pode ser configurado para receber uma sequência de bits, e determinar se uma ou mais das restrições de MV e/ou diferença de MV descritas aqui estão ativadas com base em um ou mais valores presentes na sequência de bits correspondendo a um ou mais elementos de sintaxe (por exemplo, um ou mais dentre o primeiro, segundo, terceiro e quarto elementos de sintaxe descritos aqui).

[0176] Em outros exemplos, um codificador de vídeo (por exemplo, o codificador de vídeo 20 e o decodificador de vídeo 30) pode ser configurado para determinar (por exemplo, determinar implicitamente) se uma ou mais das restrições de MV e/ou diferença MV descritas aqui são ativadas com base em outras informações

codificadas.

[0177] Várias condições ilustrativas para se ativar de forma adaptativa uma ou mais restrições descritas aqui (por exemplo, uma ou mais das restrições de MV/diferenças de MV simétricas/antissimétricas) são listadas como abaixo.

1. Em alguns exemplos, uma ou mais restrições descritas aqui (por exemplo, uma ou mais restrições de MV/diferença de MV simétricas/antissimétricas) só podem ser ativadas quando os MVs iniciais são simétricos.

2. Em alguns exemplos, uma ou mais restrições descritas aqui (por exemplo, uma ou mais restrições de MV e/ou diferença de MV simétricas/antissimétricas) só podem ser ativadas quando os MVs iniciais são pseudo-simétricos.

3. Em alguns exemplos, uma ou mais restrições descritas aqui (por exemplo, uma ou mais restrições de MV/diferença de MV simétricas/antissimétrica) só podem ser ativadas quando os MVs iniciais são antissimétricos.

4. Em alguns exemplos, uma ou mais restrições descritas aqui (por exemplo, uma ou mais restrições de MV/diferença de MV simétrica/antissimétrica) só podem ser ativadas quando os MVs iniciais são pseudo-antissimétricos.

5. Em alguns exemplos, uma ou mais restrições descritas aqui (por exemplo, uma ou mais restrições de MV/diferença de MV simétricas/antissimétricas) só podem ser ativadas quando os MVs iniciais não são simétricos.

6. Em alguns exemplos, uma ou mais restrições descritas aqui (por exemplo, uma ou mais restrições de MV/diferença de V simétricas/antissimétricas) só podem ser ativadas quando os MVs iniciais não são pseudo-simétricos.

7. Em alguns exemplos, uma ou mais restrições descritas aqui (por exemplo uma ou mais restrições MV/diferença de MV simétricas/antissimétricas) só podem ser ativadas quando os MVs iniciais não são antissimétricos.

8. Em alguns exemplos, uma ou mais restrições descritas aqui (por exemplo, uma ou mais restrições MV/diferença MV simétricas/antissimétricas) só podem ser ativadas quando os MVs iniciais não são pseudo-antissimétricos.

9. Em alguns exemplos, uma ou mais restrições descritas aqui (por exemplo, uma ou mais restrições MV/diferença de MV simétricas/antissimétricas) só podem ser ativadas quando as imagens de referência List0 e List1 dos MVs iniciais estiverem ambas antes ou ambas depois da imagem atual.

10. Em alguns exemplos, uma ou mais restrições descritas aqui (por exemplo, uma ou mais restrições MV/diferença MV simétricas e/ou antissimétricas) só podem ser ativadas quando as imagens de referência List0 e List1 dos MVs iniciais estão ambas antes ou ambas depois da imagem atual.

11. Em alguns exemplos, uma ou mais restrições descritas aqui (por exemplo, uma ou mais restrições MV/diferença de MV simétricas/antissimétricas) só podem ser ativadas quando o índice de referência dos MVs iniciais for igual a zero.

12. Em alguns exemplos, uma ou mais restrições descritas aqui (por exemplo, uma ou mais restrições MV/diferença de MV simétricas/antissimétricas) só podem ser ativadas quando o índice de referência dos MVs iniciais não

for igual a zero.

13. Em alguns exemplos, uma ou mais restrições descritas aqui (por exemplo, uma ou mais restrições de MV/diferença de MV simétricas/antissimétricas) só podem ser ativadas quando as distâncias POC entre a imagem de referência List1 e a imagem atual e as distâncias POC entre a imagem de referência List1 e a imagem atual são iguais.

14. Em alguns exemplos, uma ou mais restrições descritas aqui (por exemplo, uma ou mais restrições de MV/diferença de MV simétricas/antissimétricas) só podem ser ativadas quando as distâncias POC entre a imagem de referência List0 e a imagem atual e as distâncias POC entre a imagem de referência List1 e a imagem atual não são iguais.

15. Em alguns exemplos, uma ou mais restrições de diferença de MV simétricas podem ser aplicadas quando os MVs iniciais não são simétricos.

16. Em alguns exemplos, uma ou mais restrições de diferença de MV antissimétricas podem ser aplicadas quando os MVs iniciais são simétricos.

17. Em alguns exemplos, a resolução MV de uma ou mais das restrições simétricas pode ser designada de forma determinística. Em um exemplo, para o refinamento de pixel inteiro da Combinação de Gabarito Bilateral, nenhuma restrição simétrica/pseudo-simétrica/assimétrica pode ser imposta, e as restrições mencionadas acima são apenas no refinamento de movimento com precisão de meio pixel ou um quarto de pixel, ou superior. O nível de restrição (de resolução MV) pode ser sinalizado através de SPS/PPS/Cabeçalho de Fatia. A resolução também pode ser

decidida juntamente se outras ferramentas de refinamento de movimento, tal como BIO, FRUC de sub-pixel, forem ativadas. Por exemplo, quando BIO é ativado, a restrição pode não ser imposta ao refinamento de movimento com precisão de um quarto de pixel ou maior.

18. Em alguns exemplos, o nível de restrição (por exemplo, restrições que são ativadas) pode ser tornado adaptativo de acordo com a diferença absoluta entre os MVs de List0 e List1, a diferença absoluta escalonada (com base em sua distância POC relativa) entre os MVs de List0 e List1, ou os valores SAD iniciais entre as amostras interpoladas de List0 e List1 (isso é, P0 e P1, nesse exemplo). Em outros exemplos, a razão da soma dos valores de diferença absoluta (SAD) entre (gabarito atual, P0) e (gabarito atual, P1) pode ser utilizada para decidir a que resolução de MV a restrição deve ser imposta. Nesse exemplo, se a razão dos valores SAD for inferior a um limite, então, nenhuma restrição simétrica mencionada acima deve ser imposta.

[0178] A tabela 4 lista exemplos de condições e restrições aplicadas com base nas condições mencionadas acima sendo correspondidas.

[0179] Esse exemplo é ilustrado como Exemplo de Restrição 1 na Tabela 4, que também inclui outros exemplos de diferentes restrições.

Tabela 4: Exemplos de Condições e Restrições

| Exemplos de restrição | Condições | Restrições aplicadas |
|-----------------------|---------------------------------|-------------------------------|
| 1 | MVs iniciais não são simétricos | nenhuma restrição |
| | MVs iniciais não são simétricos | diferença MV pseudo-simétrica |

| | | |
|---|--|--------------------------------------|
| 2 | MVs iniciais são simétricos | nenhuma restrição |
| | MVs iniciais não são simétricos | diferença de MV simétrica |
| 3 | MVs iniciais são pseudo-simétricos | nenhuma restrição |
| | MVs iniciais não são pseudo-simétricos | diferença de MV pseudo-simétrica |
| 4 | MVs iniciais são pseudo-simétricos | nenhuma restrição |
| | MVs iniciais não são pseudo-simétricos | diferença de MV simétrica |
| 5 | MVs iniciais são simétricos | diferença de MV não pseudo-simétrica |
| | MVs iniciais não são simétricos | Diferença de MV pseudo-simétrica |
| 6 | MVs iniciais são simétricos | diferença de MV não simétrica |
| | MVs iniciais não são simétricos | diferença de MV simétrica |
| 7 | MVs iniciais são pseudo-simétricos | diferença de MV não pseudo-simétrica |
| | MVs iniciais não são pseudo-simétricos | diferença de MV pseudo-simétrica |
| 8 | MVs iniciais são pseudo-simétricos | diferença de MV não simétrica |
| | MVs iniciais não são pseudo-simétricos | diferença de MV simétrica |

[0180] A figura 15 ilustra um caso de uso ilustrativo da Combinação de Gabarito Bilateral, de acordo com a presente descrição. A figura 15 ilustra um exemplo onde uma ou mais restrições de diferença de MV pseudo-simétricas descritas aqui podem ser aplicadas à Combinação de Gabarito Bilateral apenas quando os MVs iniciais não forem simétricos. De acordo com tais exemplos, para a

Combinação de Gabarito Bilateral, quando os MVs iniciais não forem simétricos, os MVs finais podem ser derivados de modo que as diferenças de MV entre os MVs iniciais e os MVs finais sejam pseudo-simétricas. Por exemplo, como ilustrado na figura 15, o Gabarito Bilateral 1506 pode ser primeiramente derivado utilizando os MVs iniciais na List0 (L0_MV_Init, apontando para o bloco R_0 1502) e List1 (L1_MV_Init, apontando para o bloco R_1 1504) (por exemplo, os MVs de um candidato à mistura de previsão dupla), pela realização da média de R_0 e R_1 , e, então, o Gabarito Bilateral 1506 pode ser utilizado para buscar pelos blocos que melhor combinam na List0 e List1 para encontrar os melhores MVs de List0/List1. A Combinação de Gabarito Bilateral pode ser realizada dentro de uma faixa de busca predefinida (por exemplo, pixel inteiro de -1 a +1) centralizada nos MVs iniciais na List0/List1 buscando por blocos, por exemplo, R'_0 1508 e R'_1 1510. A diferença entre os blocos iniciais (bloco R_0 1502 e bloco R_1 1504) e os blocos buscados (R'_0 1508 e R'_1 1510) é ilustrada pela diferença de MV respectiva (denotada L0_MV_Diff e L1_MV_Diff). Em alguns exemplos, apenas os pares de MV que possuem uma diferença de MV pseudo-simétrica (comparando L0_MV_Diff e L1_MV_Diff) na List0 e List1 podem ser considerados MVs válidos. O par de MV válido com melhor custo de combinação pode, então, ser considerado os MVs derivados finais pela Combinação de Gabarito Bilateral. O exemplo ilustrado é denotado como o exemplo de restrição 1 na Tabela 4, ilustrada acima.

[0181] A figura 16 é um fluxograma ilustrando um método ilustrativo de acordo com as técnicas dessa

descrição. As técnicas da figura 16 podem ser implementadas por um ou mais dentro um dispositivo de destino 14 ou decodificador de vídeo 30 (ver figuras 1 e 3).

[0182] Em um exemplo da descrição, o decodificador de vídeo 30 pode ser configurado para gerar, utilizando a derivação de vetor de movimento de lado de decodificador (DMVD), uma pluralidade de vetores de movimento derivados (100). Em alguns exemplos, utilizar DMVD para gerar a pluralidade de vetores de movimento derivados compreende utilizar um dentro um processo BIO, um processo FRUC, um processo de Combinação Bilateral, um processo de combinação de gabarito FRUC, ou um processo de combinação de gabarito bilateral. Em alguns exemplos, o decodificador de vídeo 30 pode gerar a pluralidade de vetores de movimento derivados com base em uma determinação de se os primeiro e segundo vetores de movimento inicial correspondem a uma ou mais condições. O decodificador de vídeo 30 pode ser configurado para determinar um primeiro vetor de movimento derivado e um segundo vetor de movimento derivado a partir da pluralidade de vetores de movimento derivado com base em uma métrica de custo e determinar se uma restrição é correspondida (102). Em um exemplo, a métrica de custo inclui uma soma das diferenças absolutas. A restrição pode incluir se o primeiro vetor de movimento derivado e o segundo vetor de movimento derivado possuem uma diferença de vetor de movimento simétrica, possuem uma diferença de vetor de movimento pseudo-simétrica, são antissimétricos, possuem uma diferença de vetor de movimento antissimétrico, ou possuem uma diferença de vetor de movimento pseudo-antissimétrica. O decodificador de

vídeo 30 pode ser configurado para determinar se o primeiro vetor de movimento derivado e o segundo vetor de movimento derivado possuem uma diferença de vetor de movimento simétrica, possuem uma diferença de vetor de movimento pseudo-simétrica, são antissimétricos, possuem uma diferença de vetor de movimento antissimétrica, ou possuem uma diferença de vetor de movimento pseudo-antissimétrica. A restrição pode ser sinalizada através da sequência de bits, pode ser derivada, ou pré-programada.

[0183] Em um exemplo, o decodificador de vídeo 30 pode ser configurado para determinar que o primeiro vetor de movimento derivado e o segundo vetor de movimento derivado possuem a diferença de vetor de movimento simétrica compreende determinar que:

$$L0_MV_X_Diff * (CURR_POC - L1_POC_Derived) =$$

$$L1_MV_X_Diff * (CURR_POC - L0_POC_Derived) \quad \text{ou}$$

$$L0_MV_Y_Diff * (CURR_POC -$$

$$L1_POC_Derived) = L1_MV_Y_Diff * (CURR_POC - L0_POC_Derived),$$

onde: $L0_MV_X_Diff$ é um componente X de uma diferença entre um primeiro vetor de movimento inicial e o primeiro vetor de movimento derivado, $L0_MV_Y_Diff$ é um componente Y de uma diferença entre o primeiro vetor de movimento inicial e o primeiro vetor de movimento derivado, $L1_MV_X_Diff$ é um componente X de uma diferença entre um segundo vetor de movimento inicial e o segundo vetor de movimento derivado, $L1_MV_Y_Diff$ é um componente Y de uma diferença entre o segundo vetor de movimento inicial e o segundo vetor de movimento derivado, $CURR_POC$ é um POC atual de uma imagem atual, $L0_POC_Derived$ é uma POC derivada para uma primeira lista de vetores de movimento, e $L1_POC_Derived$ é uma POC

derivada para uma segunda lista de vetores de movimento.

[0184] Em outro exemplo, o decodificador de vídeo POC pode ser configurado para determinar que o primeiro vetor de movimento derivado e o segundo vetor de movimento derivado possuem a diferença de vetor de movimento antissimétrica o que compreende determinar que:

$$\begin{aligned} &L0_MV_X_Diff * (CURR_POC - L1_POC_Derived) = - \\ &1 * L1_MV_X_Diff * (CURR_POC - L0_POC_Derived), \quad \text{ou} \\ &L0_MV_Y_Diff * (CURR_POC - L1_POC_Derived) = - \\ &1 * L1_MV_Y_Diff * (CURR_POC - L0_POC_Derived), \quad \text{onde:} \end{aligned}$$

$L0_MV_X_Diff$ é um componente X de uma diferença entre um primeiro vetor de movimento inicial e o primeiro vetor de movimento derivado, $L0_MV_Y_Diff$ é um componente Y de uma diferença entre o primeiro vetor de movimento inicial e o primeiro vetor de movimento derivado, $L1_MV_X_Diff$ é um componente X de uma diferença entre um segundo vetor de movimento inicial e o segundo vetor de movimento derivado, $L1_MV_Y_Diff$ é um componente Y de uma diferença entre o segundo vetor de movimento inicial e o segundo vetor de movimento derivado, $CURR_POC$ é uma POC atual de uma imagem atual, $L0_POC_Derived$ é uma POC derivada para uma primeira lista de vetores de movimento, e $L1_POC_Derived$ é um POC derivada para uma segunda lista de vetores de movimento.

[0185] Em outro exemplo, o decodificador de vídeo 30 pode ser configurado para determinar que o primeiro vetor de movimento derivado e o segundo vetor de movimento derivado são antissimétricos e compreende determinar que:

$$\begin{aligned} &L0_MV_X_Derived * (CURR_POC - L1_POC_Derived) = - \\ &1 * L1_MV_X_Denied * (CURR_POC - L0_POC_Derived), \quad \text{ou} \\ &L0_MV_Y_Derived * (CURR_POC - L1_POC_Derived) = - \end{aligned}$$

$1 * L1_MV_Y_Derived * (CURR_POC - L0_POC_Derived)$, onde $L0_MV_X_Derived$ é um componente X do primeiro vetor de movimento derivado, $L0_MV_Y_Derived$ é um componente Y do primeiro vetor de movimento derivado, $L1_MV_X_Derived$ é um componente X do segundo vetor de movimento derivado, $L1_MV_Y_Derived$ é um componente Y do segundo vetor de movimento derivado, $CURR_POC$ é uma POC atual de uma imagem atual, $L0_POC_Derived$ é uma POC derivada para uma primeira lista de vetores de movimento e $L1_POC_Derived$ é uma POC derivada para uma segunda lista de vetores de movimento.

[0186] Em outro exemplo, o decodificador de vídeo 30 pode ser configurado para determinar que o primeiro vetor de movimento derivado e o segundo vetor de movimento derivado possuem a diferença de vetor de movimento pseudo-simétrica compreende determinar que $L0_MV_X_Diff = L1_MV_X_Diff$, $L0_MV_Y_Diff = L1_MV_Y_Diff$, $L0_MV_X_Diff = -1 * L1_MV_X_Diff$, ou $L0_MV_Y_Diff = -1 * L1_MV_Y_Diff$; onde: $L0_MV_X_Diff$ é um componente X de uma diferença entre um primeiro vetor de movimento inicial e o primeiro vetor de movimento derivado, $L0_MV_Y_Diff$ é um componente Y de uma diferença entre o primeiro vetor de movimento inicial e o primeiro vetor de movimento derivado, $L1_MV_X_Diff$ é um componente X de uma diferença entre um segundo vetor de movimento inicial e o segundo vetor de movimento derivado, e $L1_MV_Y_Diff$ é um componente Y de uma diferença entre o segundo vetor de movimento inicial e o segundo vetor de movimento derivado.

[0187] Em outro exemplo, o decodificador de vídeo 30 pode ser configurado para determinar que o primeiro vetor de movimento derivado e o segundo vetor de

movimento derivado possuem a diferença de vetor de movimento pseudo-antissimétrica e compreende uma determinação de que $L0_MV_X_Diff=L1_MV_X_Diff$, $L0_MV_Y_Diff=L1_MV_Y_Diff$, $L0_MV_X_Diff=-1*L1_MV_X_Diff$ ou $L0_MV_Y_Diff=-1*L1_MV_Y_Diff$; onde: $L0_MV_X_Diff$ é um componente X de uma diferença entre um primeiro vetor de movimento inicial e o primeiro vetor de movimento derivado, $L0_MV_Y_Diff$ é um componente Y de uma diferença entre o primeiro vetor e movimento inicial e o primeiro vetor de movimento derivado, $L1_MV_X_Diff$ é um componente X de uma diferença entre um segundo vetor de movimento inicial e o segundo vetor de movimento derivado, e $L1_MV_Y_Diff$ é um componente Y de uma diferença entre o segundo vetor de movimento inicial e o segundo vetor de movimento derivado.

[0188] Em outro exemplo, o decodificador de vídeo 30 pode ser configurado para determinar que um primeiro vetor de movimento inicial em uma lista de candidatos a vetor de movimento e um segundo vetor de movimento inicial na lista de candidatos a vetor de movimento satisfazem uma condição em uma relação entre o primeiro vetor de movimento inicial e o segundo vetor de movimento inicial, a relação compreendendo: o primeiro vetor de movimento inicial e o segundo vetor de movimento inicial serem simétricos, o primeiro vetor de movimento inicial e o segundo vetor de movimento inicial serem pseudo-simétricos, o primeiro vetor de movimento inicial e o segundo vetor de movimento inicial possuírem a diferença de vetor de movimento simétrica, o primeiro vetor de movimento inicial e o segundo vetor de movimento inicial serem antissimétricos, o primeiro vetor de movimento

inicial e o segundo vetor de movimento inicial serem pseudo-antissimétricos, ou o primeiro vetor de movimento inicial e o segundo vetor de movimento inicial possuírem a diferença de vetor de movimento antissimétrica, onde gerar a pluralidade de vetores de movimento derivado se baseia no primeiro vetor de movimento inicial e o segundo movimento inicial satisfazendo a pelo menos uma condição.

[0189] Em outro exemplo, o decodificador de vídeo 30 pode ser configurado para determinar se um primeiro vetor de movimento inicial em uma lista de candidatos a vetor de movimento e um segundo vetor de movimento inicial na lista de candidatos a vetor de movimento estão antes de uma imagem atual ou estão depois da imagem atual, onde determinar o primeiro vetor de movimento derivado e o segundo vetor de movimento derivado a partir da pluralidade de vetores de movimento derivado, com base na métrica de custo, é adicionalmente baseado na determinação.

[0190] Em outro exemplo, o decodificador de vídeo 30 pode ser configurado para determinar se uma primeira distância de contagem de ordem de imagem (POC), entre uma primeira imagem de referência, associada com uma primeira lista de candidatos a vetor de movimento e uma imagem atual é igual a zero, e uma segunda distância POC, entre uma segunda imagem de referência associada com uma segunda lista candidata a vetor de movimento e a imagem atual é igual a zero, onde determinar o primeiro vetor de movimento derivado e o segundo vetor de movimento derivado, a partir da pluralidade de vetores de movimento derivado com base na métrica de custo é a adicionalmente baseado na

determinação.

[0191] Em outro exemplo, o decodificador de vídeo 30 pode ser configurado para determinar se uma primeira distância de contagem de ordem de imagem (POC), entre uma primeira imagem de referência associada com uma primeira lista de candidatos a vetor de movimento e uma imagem atual, é igual a zero e uma segunda distância POC, entre uma segunda imagem de referência associada com uma segunda lista candidata a vetor de movimento e a imagem atual, é igual a zero, onde determinar o primeiro vetor de movimento derivado e o segundo vetor de movimento derivado, a partir da pluralidade de vetores de movimento derivado, com base na métrica de custo, gerando a pluralidade de vetores de movimento derivado, se baseia na determinação da primeira distância POC entre a primeira imagem de referência associada com a primeira lista de candidatos a vetor de movimento e a imagem atual, é igual a zero e a segunda distância POC entre a segunda imagem de referência associada com a segunda lista de candidatos a vetor de movimento e a imagem atual, é igual à determinação zero.

[0192] Em outro exemplo, o decodificador de vídeo 30 pode ser configurado para determinar a pelo menos uma restrição a partir da sequência de bits de vídeo codificada.

[0193] O decodificador de vídeo 30 pode ser configurado para decodificar um bloco atual utilizando o primeiro vetor de movimento derivado determinado e o segundo vetor de movimento derivado (104).

[0194] A figura 17 é um fluxograma ilustrando um método ilustrativo de acordo com as técnicas dessa

descrição. As técnicas da figura 17 podem ser implementadas por um ou mais dentre um dispositivo fonte 12 ou um codificador de vídeo 20 (ver figuras 1 e 2).

[0195] O codificador de vídeo 20 pode ser configurado para gerar uma pluralidade de vetores de movimento derivado utilizando DMVD (200). Em alguns exemplos, o codificador de vídeo 20 pode gerar a pluralidade de vetores de movimento derivado com base em uma determinação de se os primeiro e segundo vetores de movimento inicial correspondem a uma ou mais condições.

[0196] O codificador de vídeo 20 pode ser configurado para determinar um primeiro vetor de movimento derivado e um segundo vetor de movimento derivado a partir da pluralidade de vetores de movimento derivado com base em uma métrica de custo e determinando se uma restrição foi correspondida (202). A restrição pode incluir se o primeiro vetor de movimento derivado e o segundo vetor de movimento derivado possuem uma diferença de vetor de movimento simétrica, possuem uma diferença de vetor de movimento pseudo-simétrica, são antissimétricos, possuem uma diferença de vetor de movimento antissimétrica, ou possuem uma diferença de vetor de movimento pseudo-antissimétrica. O codificador de vídeo 20 pode ser configurado para codificar um bloco atual utilizando o primeiro vetor de movimento derivado e o segundo vetor de movimento derivado (204) determinados. O codificador de vídeo 20 pode ser configurado para codificar a restrição. O codificador de vídeo 20 pode ser adicionalmente configurado para enviar a restrição através de uma sequência de bits.

[0197] Deve-se reconhecer que dependendo do

exemplo, determinados atos ou eventos de qualquer uma das técnicas descritas aqui podem ser realizados em uma sequência diferente, podem ser adicionados, misturados ou deixados de fora completamente (por exemplo, nem todos os atos ou eventos descritos são necessários para a prática das técnicas). Ademais, em determinados exemplos, os atos ou eventos podem ser realizados simultaneamente, por exemplo, através de um processamento multissequencial, processamento interrompido, ou múltiplos processadores, em vez de sequencialmente.

[0198] Em um ou mais exemplos, as funções descritas podem ser implementadas em hardware, software, firmware, ou qualquer combinação dos mesmos. Se implementadas em software, as funções podem ser armazenadas em ou transmitidas como uma ou mais instruções ou código em um meio legível por computador e executadas por uma unidade de processamento com base em hardware. Meio legível por computador pode incluir o meio de armazenamento legível por computador, que corresponde a um meio tangível, tal como meio de armazenamento de dados, ou meio de comunicação incluindo qualquer meio que facilite a transferência de um programa de computador de um lugar para outro, por exemplo, de acordo com um protocolo de comunicação. Dessa forma, o meio legível por computador pode, geralmente, corresponder a (1) meio de armazenamento legível por computador tangível que é não transitório ou (2) um meio de comunicação, tal como um sinal ou onda portadora. O meio de armazenamento de dados pode ser qualquer meio disponível que possa ser acessado por um ou mais computadores ou um ou mais processadores para recuperar instruções, código e/ou

estruturas de dados para implementação das técnicas descritas nessa descrição. Um produto de programa de computador pode incluir um meio legível por computador.

[0199] Por meio de exemplo, e não de limitação, tal meio de armazenamento legível por computador pode compreender RAM, ROM, EEPROM, CD-ROM ou outro armazenamento em disco ótico, armazenamento em disco magnético, ou outros dispositivos de armazenamento magnéticos, memória flash ou qualquer outro meio que possa ser utilizado para armazenar o código de programa desejado na forma de instruções ou estruturas de dados e que possa ser acessado por um computador. Além disso, qualquer conexão é adequadamente chamada de meio legível por computador. Por exemplo, se as instruções forem transmitidas a partir de um sítio da rede, servidor ou outra fonte remota utilizando um cabo coaxial, o cabo de fibra ótica, o par torcido, linha de assinante digital (DSL), ou tecnologias sem fio, tal como infravermelho, rádio e micro-ondas, então, o cabo coaxial, o cabo de fibra ótica, o par torcido, DSL, ou tecnologias sem fio tal como infravermelho, rádio e micro-ondas são incluídos na definição de meio. Deve-se compreender, no entanto, que o meio de armazenamento legível por computador e o meio de armazenamento de dados não incluem conexões, ondas portadoras, sinais ou outro meio transitório, mas em vez disso, são direcionados para o meio de armazenamento tangível e não transitório. Disquete e disco, como utilizados aqui, incluem disco compacto (CD), disco a laser, disco ótico, disco versátil digital (DVD), disquete e disco Blu-ray, onde disquetes normalmente reproduzem

dados magneticamente, enquanto discos reproduzem os dados ópticamente com lasers. As combinações do acima devem ser incluídas também no escopo de meio legível por computador.

[0200] Instruções podem ser executadas por um ou mais processadores, tal como um ou mais processadores de sinal digital (DSPs), microprocessadores de finalidade geral, conjunto de circuitos de processamento (incluindo o conjunto de circuitos de função fixa e/ou conjunto de circuitos de processamento programável), circuitos integrados específicos de aplicativo (ASICs), conjuntos de porta programável em campo (FPGAs), ou outro conjunto de circuitos lógico discreto ou integrado equivalente. De acordo, o termo "processador", como utilizado aqui pode se referir a qualquer uma dentre a estrutura acima ou qualquer outra estrutura adequada para implementar as técnicas descritas aqui. Adicionalmente, em alguns aspectos, a funcionalidade descrita aqui pode ser fornecida dentro de módulos de hardware e/ou software dedicados configurados para codificar e decodificar, ou incorporados a um codec combinado. Além disso, as técnicas podem ser totalmente implementadas em um ou mais circuitos ou elementos lógicos.

[0201] As técnicas dessa descrição podem ser implementadas através de uma ampla variedade de dispositivos ou aparelhos, incluindo um aparelho sem fio, um circuito integrado (IC) ou um conjunto de ICs (por exemplo, um conjunto de chips). Vários componentes, módulos ou unidades são descritos nessa descrição para enfatizar os aspectos funcionais dos dispositivos configurados para realizar as técnicas descritas, mas não exigem, necessariamente, a realização por unidades de hardware

diferentes. Em vez disso, como descrito acima, várias unidades podem ser combinadas em uma unidade de hardware codec ou fornecidas por uma coleta de unidades de hardware interoperacional, incluindo um ou mais processadores como descrito acima, em conjunto com software e/ou firmware adequados.

[0202] Vários exemplos foram descritos. Esses e outros exemplos estão dentro do escopo das reivindicações a seguir.

REIVINDICAÇÕES

1. Método para decodificar dados de vídeo a partir de uma sequência de bits de vídeo codificada, o método compreendendo:

gerar, utilizando derivação de vetor de movimento de lado de decodificador (DMVD), uma pluralidade de vetores de movimento derivado;

determinar um primeiro vetor de movimento derivado e um segundo vetor de movimento derivado a partir da pluralidade de vetores de movimento derivado com base em uma métrica de custo, compreendendo:

determinar que o primeiro vetor de movimento derivado e o segundo vetor de movimento derivado satisfazem pelo menos uma restrição em uma relação entre o primeiro vetor de movimento derivado e o segundo vetor de movimento derivado, a relação compreendendo o primeiro vetor de movimento derivado e o segundo vetor de movimento derivado possuírem uma diferença de vetor de movimento simétrica, o primeiro vetor de movimento derivado e o segundo vetor de movimento derivado possuírem uma diferença de vetor de movimento pseudo-simétrico, o primeiro vetor de movimento derivado e o segundo vetor de movimento derivado serem antissimétricos, o primeiro vetor de movimento derivado e o segundo vetor de movimento derivado possuírem uma diferença de vetor de movimento antissimétrica, ou o primeiro vetor de movimento derivado e o segundo vetor de movimento derivado possuírem uma diferença de vetor de movimento antissimétrica; e

decodificar um bloco atual utilizando o primeiro vetor de movimento derivado e o segundo vetor de movimento

derivado determinados utilizando a previsão dupla.

2. Método, de acordo com a reivindicação 1, no qual utilizar DMVD para gerar a pluralidade de vetores de movimento derivado compreende utilizar um processo de fluxo Ótico Bidirecional (BIO), um processo de conversão ascendente de taxa de quadro (FRUC), um processo de Combinação Bilateral, um processo de combinação de gabarito FRUC, ou um processo de combinação de gabarito bilateral.

3. Método, de acordo com a reivindicação 1, no qual determinar que o primeiro vetor de movimento derivado e o segundo vetor de movimento derivado possuem a diferença de vetor de movimento simétrica compreende determinar que:

$$\begin{aligned} L0_MV_X_Diff * (CURR_POC - \\ L1_POC_Derived) = L1_MV_X_Diff * (CURR_POC - L0_POC_Derived) \text{ ou} \\ L0_MV_Y_Diff * (CURR_POC - \\ L1_POC_Derived) = L1_MV_Y_Diff * (CURR_POC - L0_POC_Derived), \end{aligned}$$

onde:

$L0_MV_X_Diff$ é um componente X de uma diferença entre um primeiro vetor de movimento inicial e o primeiro vetor de movimento derivado,

$L0_MV_Y_Diff$ é um componente Y de uma diferença entre o primeiro vetor de movimento inicial e o primeiro vetor de movimento derivado;

$L1_MV_X_Diff$ é um componente X de uma diferença entre um segundo vetor de movimento inicial e o segundo vetor de movimento derivado;

$L1_MV_Y_Diff$ é um componente Y de uma diferença entre o segundo vetor de movimento inicial e o segundo vetor de movimento derivado,

$CURR_POC$ é uma contagem de ordem de imagem atual

(POC) de uma imagem atual,

L0_POC_Derived é uma POC derivada para uma primeira lista de vetores de movimento, e

L1_POC_Derived é uma POC derivada para uma segunda lista de vetores de movimento.

4. Método, de acordo com a reivindicação 1, no qual determinar que o primeiro vetor de movimento derivado e o segundo vetor de movimento derivado possuem a diferença de vetor de movimento antissimétrica compreende determinar que:

$L0_MV_X_Diff * (CURR_POC - L1_POC_Derived) = -1 * L1_MV_X_Diff * (CURR_POC - L0_POC_Derived)$, ou

$L0_MV_Y_Diff * (CURR_POC - L1_POC_Derived) = -1 * L1_MV_Y_Diff * (CURR_POC - L0_POC_Derived)$

onde:

L0_MV_X_Diff é um componente X de uma diferença entre um primeiro vetor de movimento inicial e o primeiro vetor de movimento derivado;

L0_MV_Y_Diff é um componente Y de uma diferença entre o primeiro vetor de movimento inicial e o primeiro vetor de movimento derivado;

L1_MV_X_Diff é um componente X de uma diferença entre um segundo vetor de movimento inicial e o segundo vetor de movimento derivado;

L1_MV_Y_Diff é um componente Y de um diferença entre o segundo vetor de movimento inicial e o segundo vetor de movimento derivado.

CURR_POC é uma contagem de ordem de imagem atual (POC) de uma imagem atual;

L0_POC_Derived é uma POC derivada para uma

primeira lista de vetores de movimento, e

L1_POC_Derived é uma POC derivada para uma segunda lista de vetores de movimento.

5. Método, de acordo com a reivindicação 1, no qual determinar que o primeiro vetor de movimento derivado e o segundo vetor de movimento derivado são antissimétricos compreende determinar que:

$L0_MV_X_Derived * (CURR_POC - L1_POC_Derived) = -1 * L1_MV_X_Derived * (CURR_POC - L0_POC_Derived)$, ou

$L0_MV_Y_Derived * (CURR_POC - L1_POC_Derived) = -1 * L1_MV_Y_Derived * (CURR_POC - L0_POC_Derived)$,

onde:

L0_MV_X_Derived é um componente X do primeiro vetor de movimento derivado,

L0_MV_Y_Derived é um componente Y do primeiro vetor de movimento derivado,

L1_MV_X_Derived é um componente X do segundo vetor de movimento derivado;

L1_MV_Y_Derived é um componente Y do segundo valor de movimento derivado;

CURR_POC é uma contagem de ordem de imagem atual (POC) de uma imagem atual;

L0_POC_Derived é uma POC derivada para uma primeira lista de vetores de movimento, e

L1_POC_Derived é uma POC derivada para uma segunda lista de vetores de movimento.

6. Método, de acordo com a reivindicação 1, no qual determinar que o primeiro vetor de movimento derivado e o segundo vetor de movimento derivado possuem a diferença de vetor de movimento pseudo-simétrica compreende

determinar que:

$$L0_MV_X_Diff = L1_MV_Diff;$$

$$L0_MV_Y_Diff = L1_MV_Y_Diff;$$

$$L0_MV_X_Diff = -1*L1_MV_X_Diff; \text{ ou}$$

$$L0_MV_Y_Diff = -1*L1_MV_Y_Diff;$$

onde:

$L0_MV_X_Diff$ é um componente X de uma diferença entre um primeiro vetor de movimento inicial e o primeiro vetor de movimento derivado;

$L0_MV_Y_Diff$ é um componente Y de uma diferença entre o primeiro vetor de movimento inicial e o primeiro vetor de movimento derivado;

$L1_MV_X_Diff$ é um componente X de uma diferença entre um segundo vetor de movimento inicial e o segundo vetor de movimento derivado; e

$L1_MV_Y_Diff$ é um componente Y de uma diferença entre o segundo vetor de movimento inicial e o segundo vetor de movimento derivado.

7. Método, de acordo com a reivindicação 1, no qual determinar que o primeiro vetor de movimento derivado e o segundo vetor de movimento derivado possuem a diferença de vetor de movimento pseudo-antissimétrico compreende determinar que:

$$L0_MV_X_Diff = L1_MV_X_Diff;$$

$$L0_MV_Y_Diff = L1_MV_Y_Diff;$$

$$L0_MV_X_Diff = -1*L1_MV_X_Diff, \text{ ou}$$

$$L0_MV_Y_Diff = -1*L1_MV_Y_Diff;]$$

onde:

$L0_MV_X_Diff$ é um componente X de uma diferença entre um primeiro vetor de movimento inicial e o primeiro

vetor de movimento derivado;

L0_MV_Y_Diff é um componente Y de uma diferença entre o primeiro vetor de movimento inicial e o primeiro vetor de movimento derivado;

L1_MV_X_Diff é um componente X de uma diferença entre um segundo vetor de movimento inicial e o segundo vetor de movimento derivado; e

L1_MV_Y_Diff é um componente Y de uma diferença entre o segundo vetor de movimento inicial e o segundo vetor de movimento derivado.

8. Método, de acordo com a reivindicação 1, compreendendo adicionalmente:

determinar que um primeiro vetor de movimento inicial em uma lista de candidatos a vetor de movimento e um segundo vetor de movimento inicial na lista de candidatos a vetor de movimento satisfazem pelo menos uma condição em uma relação entre o primeiro vetor de movimento inicial e o segundo vetor de movimento inicial, a relação compreendendo: o primeiro vetor de movimento inicial e o segundo vetor de movimento inicial serem simétricos, o primeiro vetor de movimento inicial e o segundo vetor de movimento inicial serem pseudo-simétricos, o primeiro vetor de movimento inicial e o segundo vetor de movimento inicial possuírem a diferença de vetor de movimento simétrica, o primeiro vetor de movimento inicial e o segundo vetor de movimento inicial serem antissimétricos, o primeiro vetor de movimento inicial e o segundo vetor de movimento inicial serem pseudo-antissimétricos, ou o primeiro vetor de movimento inicial e o segundo vetor de movimento inicial possuírem a diferença de vetor de movimento antissimétrica;

onde a geração da pluralidade de vetores de movimento derivado se baseia no primeiro vetor de movimento inicial e segundo vetor de movimento inicial satisfazendo pelo menos uma condição.

9. Método, de acordo com a reivindicação 1, compreendendo adicionalmente:

determinar se um primeiro vetor de movimento inicial em uma lista de candidatos a vetor de movimento e um segundo vetor de movimento inicial na lista de candidatos a vetor de movimento estão antes de uma imagem atual ou estão depois da imagem atual;

onde determinar o primeiro vetor de movimento derivado e o segundo vetor de movimento derivado a partir da pluralidade de vetores de movimento derivado com base na métrica de custo se baseia adicionalmente na determinação.

10. Método, de acordo com a reivindicação 1, compreendendo adicionalmente:

determinar se uma primeira distância de contagem de ordem de imagem (POC), entre uma primeira imagem de referência associada com uma primeira lista de candidatos a vetor de movimento e uma imagem atual, é igual a zero e uma segunda distância POC, entre uma segunda imagem de referência associada com uma segunda lista de candidatos a vetor de movimento e a imagem atual, é igual a zero,

onde determinar o primeiro vetor de movimento derivado e o segundo vetor de movimento derivado da pluralidade de vetores de movimento derivado com base na métrica de custo se baseia adicionalmente na determinação.

11. Método, de acordo com a reivindicação 1, compreendendo adicionalmente:

determinar se uma primeira distância de contagem de ordem de imagem (POC), entre uma primeira imagem de referência associada com uma primeira lista de candidatos a vetor de movimento e uma imagem atual,, é igual a zero e uma segunda distância POC entre uma segunda imagem de referência associada com uma segunda lista de candidatos a vetor de movimento e a imagem atual, é igual a zero;

onde determinar o primeiro vetor de movimento derivado e o segundo vetor de movimento derivado a partir da pluralidade de vetores de movimento derivado com base na métrica de custo se baseia adicionalmente na determinação.

12. Método, de acordo com a reivindicação 1, compreendendo adicionalmente:

determinar a pelo menos uma restrição a partir da sequência de bits de vídeo codificada.

13. Método, de acordo com a reivindicação 1, no qual a métrica de custo inclui uma soma de diferenças absolutas.

14. Aparelho configurado para decodificar dados de vídeo a partir de uma sequência de bits de vídeo codificada, o aparelho compreendendo:

uma memória configurada para armazenar os dados de vídeo; e

um ou mais processadores em comunicação com a memória, os um ou mais processadores configurados para:

gerar, utilizando a derivação de vetor de movimento de lado de decodificador (DMVD), uma pluralidade de vetores de movimento derivado;

determinar um primeiro vetor de movimento derivado e um segundo vetor de movimento derivado a partir

da pluralidade de vetores de movimento derivado com base em uma métrica de custos, compreendendo:

uma determinação de que o primeiro vetor de movimento derivado e o segundo vetor de movimento derivado satisfazem pelo menos uma restrição em uma relação entre o primeiro vetor de movimento derivado e o segundo vetor de movimento derivado, a relação compreendendo: o primeiro vetor de movimento derivado e o segundo vetor de movimento derivado possuírem uma diferença de vetor de movimento simétrico, o primeiro vetor de movimento derivado e o segundo vetor de movimento derivado possuírem uma diferença de vetor de movimento simétrica, o primeiro vetor de movimento derivado e o segundo vetor de movimento derivado possuírem uma diferença de vetor de movimento pseudo-simétrica, o primeiro vetor de movimento derivado e o segundo vetor de movimento derivado serem antissimétricos, o primeiro vetor de movimento derivado e o segundo vetor de movimento derivado possuírem uma diferença de vetor de movimento antissimétrica, ou o primeiro vetor de movimento derivado e o segundo vetor de movimento derivado possuírem uma diferença de vetor de movimento pseudo-antissimétrica; e decodificar um bloco atual utilizando o primeiro vetor de movimento derivado e o segundo vetor de movimento derivado determinados utilizando a previsão dupla.

15. Aparelho, de acordo com a reivindicação 14, no qual utilizar DMVD para gerar a pluralidade de vetores de movimento derivado compreende utilizar um dentre um processo de fluxo Ótico Bidirecional, um processo de conversão ascendente de taxa de quadro (FRUC), um processo de Combinação Bilateral, um processo de combinação de

gabarito FRUC, ou um processo de combinação de gabarito bilateral.

16. Aparelho, de acordo com a reivindicação 14, no qual determinar que o primeiro vetor de movimento derivado e o segundo vetor de movimento derivado possuem a diferença de vetor de movimento simétrico compreende determinar que:

$$\begin{aligned} L0_MV_X_Diff * (CURR_POC - \\ L1_POC_Derived) = L1_MV_X_Diff * (CURR_POC - L0_POC_Derived) \text{ ou} \\ L0_MV_Y_Diff * (CURR_POC - \\ L1_POC_Derived) = L1_MV_Y_Diff * (CURR_POC - L0_POC_Derived), \end{aligned}$$

onde:

$L0_MV_X_Diff$ é um componente X de uma diferença entre um primeiro vetor de movimento inicial e o primeiro vetor de movimento derivado;

$L0_MV_Y_Diff$ é um componente Y de uma diferença entre o primeiro vetor de movimento inicial e o primeiro vetor de movimento derivado;

$L1_MV_X_Diff$ é um componente X de uma diferença entre um segundo vetor de movimento inicial e o segundo vetor de movimento derivado;

$L1_MV_Y_Diff$ é um componente Y de uma diferença entre o segundo vetor de movimento inicial e o segundo vetor de movimento derivado;

$CURR_POC$ é uma contagem de ordem de imagem atual (POC) de uma imagem atual,

$L0_POC_Derived$ é uma POC derivada para uma primeira lista de vetores de movimento, e

$L1_POC_Derived$ é uma POC derivada para uma segunda lista de vetores de movimento.

17. Aparelho, de acordo com a reivindicação 14, no qual determinar que o primeiro vetor de movimento derivado e o segundo vetor de movimento derivado possuem a diferença de vetor de movimento antissimétrica compreende determinar que:

$$\begin{aligned} &L0_MV_X_Diff * (CURR_POC - L1_POC_Derived) = - \\ &1 * L1_MV_X_Diff * (CURR_POC - L0_POC_Derived), \text{ ou} \\ &L0_MV_Y_Diff * (CURR_POC - L1_POC_Derived) = - \\ &1 * L1_MV_Y_Diff * (CURR_POC - L0_POC_Derived) \end{aligned}$$

onde:

$L0_MV_X_Diff$ é um componente X de uma diferença entre um primeiro vetor de movimento inicial e o primeiro vetor de movimento derivado;

$L0_MV_Y_Diff$ é um componente Y de uma diferença entre o primeiro vetor de movimento inicial e o primeiro vetor de movimento derivado;

$L1_MV_X_Diff$ é um componente X de uma diferença entre um segundo vetor de movimento inicial e o segundo vetor de movimento derivado;

$L1_MV_Y_Diff$ é um componente Y de uma diferença entre o segundo vetor e movimento inicial e o segundo vetor de movimento derivado;

$CURR_POC$ é uma contagem de ordem de imagem atual (POC) de uma imagem atual;

$L0_POC_Derived$ é uma POC derivada para uma primeira lista de vetores de movimento, e

$L1_POC_Derived$ é uma POC derivada para uma segunda lista de vetores de movimento.

18. Aparelho, de acordo com a reivindicação 14, no qual a determinação de que o primeiro vetor de movimento

derivado e o segundo vetor de movimento derivado são antissimétricos compreende determinar que:

$$\begin{aligned} &L0_MV_X_Derived*(CURR_POC-L1_POC_Derived)=- \\ &1*L1_MV_X_Derived*(CURR_POC-L0_POC_Derived), \text{ ou} \\ &L0_MV_Y_Derived*(CURR_POC-L1_POC_Derived)=- \\ &1*L1_MV_Y_Derived*(CURR_POC-L0_POC_Derived), \end{aligned}$$

onde:

$L0_MV_X_Derived$ é um componente X do primeiro vetor de movimento derivado,

$L0_MV_Y_Derived$ é um componente Y do primeiro vetor de movimento derivado;

$L1_MV_X_Derived$ é um componente X do segundo vetor de movimento derivado;

$L1_MV_Y_Derived$ é um componente Y do segundo vetor de movimento derivado;

$CURR_POC$ é uma contagem de ordem de imagem atual (POC) de uma imagem atual;

$L0_POC_Derived$ é uma POC derivada para uma primeira lista de vetores de movimento; e

$L1_POC_Derived$ é uma POC derivada para uma segunda lista de vetores de movimento.

19. Aparelho, de acordo com a reivindicação 14, no qual determinar que o primeiro vetor de movimento derivado e o segundo vetor de movimento derivado possuem a diferença de vetor de movimento pseudo-simétrico compreende determinar que:

$$\begin{aligned} &L0_MV_X_Diff=L1_MV_X_Diff; \\ &L0_MV_Y_Diff=L1_MV_Y_Diff; \\ &L0_MV_X_Diff=-1*L1_MV_X_Diff; \text{ ou} \\ &L0_MV_Y_Diff=-1*L1_MV_Y_Diff; \end{aligned}$$

onde:

$L0_MV_X_Diff$ é um componente X de uma diferença entre um primeiro vetor de movimento inicial e o primeiro vetor de movimento derivado;

$L0_MV_Y_Diff$ é um componente Y de uma diferença entre o primeiro vetor de movimento inicial e o primeiro vetor de movimento derivado;

$L1_MV_X_Diff$ é um componente X de uma diferença entre um segundo vetor de movimento inicial e o segundo vetor de movimento derivado; e

$L1_MV_Y_Diff$ é um componente Y de uma diferença entre um segundo vetor de movimento inicial e o segundo vetor de movimento derivado.

20. Aparelho, de acordo com a reivindicação 14, no qual determinar que o primeiro vetor de movimento derivado e o segundo vetor de movimento derivado possuem a diferença de vetor de movimento pseudo-antissimétrica compreende determinar que:

$$L0_MV_X_Diff = L1_MV_X_Diff;$$

$$L0_MV_Y_Diff = L1_MV_Y_Diff;$$

$$L0_MV_X_Diff = -1 * L1_MV_X_Diff, \text{ ou}$$

$$L0_MV_Y_Diff = -1 * L1_MV_Y_Diff,$$

onde:

$L0_MV_X_Diff$ é um componente X de uma diferença entre um primeiro vetor de movimento inicial e o primeiro vetor de movimento derivado;

$L0_MV_Y_Diff$ é um componente Y de uma diferença entre o primeiro vetor de movimento inicial e o primeiro vetor de movimento derivado;

$L1_MV_X_Diff$ é um componente X de uma diferença

entre um segundo vetor de movimento inicial e o segundo vetor de movimento derivado, e

L1_MV_Y_Diff é um componente Y de uma diferença entre o segundo vetor de movimento inicial e o segundo vetor de movimento derivado.

21. Aparelho, de acordo com a reivindicação 14, no qual um ou mais processadores são adicionalmente configurados para:

determinar que um primeiro vetor de movimento inicial em uma lista de candidatos a vetor de movimento e um segundo vetor de movimento inicial na lista de candidatos a vetor de movimento satisfazem pelo menos uma condição em uma relação entre o primeiro vetor de movimento inicial e o segundo vetor de movimento inicial, a relação compreendendo o primeiro vetor de movimento inicial e o segundo vetor de movimento inicial serem simétricos, o primeiro vetor de movimento inicial e o segundo vetor de movimento inicial serem pseudo-simétricos, o primeiro vetor de movimento inicial e o segundo vetor de movimento inicial possuírem a diferença de vetor de movimento simétrica, o primeiro vetor de movimento inicial e o segundo vetor de movimento inicial serem antissimétricos, o primeiro vetor de movimento inicial e o segundo vetor de movimento inicial serem pseudo-antissimétricos, ou o primeiro vetor de movimento inicial e o segundo vetor de movimento inicial possuírem a diferença de vetor de movimento antissimétrico,

onde gerar a pluralidade de vetores de movimento derivado se baseia no primeiro vetor de movimento inicial e segundo vetor de movimento inicial satisfazerem a pelo menos uma condição.

22. Aparelho, de acordo com a reivindicação 14, no qual um ou mais processadores são configurados adicionalmente para:

determinar se um primeiro vetor de movimento inicial em uma lista de candidatos a vetor de movimento e um segundo vetor de movimento inicial na lista de candidatos a vetor de movimento estão antes de uma imagem atual ou depois da imagem atual;

onde determinar o primeiro vetor de movimento derivado e o segundo vetor de movimento derivado a partir da pluralidade de vetores de movimento derivado com base na métrica de custo se baseia adicionalmente na determinação.

23. Aparelho, de acordo com a reivindicação 14, no qual um ou mais processadores são adicionalmente configurados para:

determinar se uma primeira distância de contagem de ordem de imagem (POC), entre uma primeira imagem de referência associada com uma primeira lista de candidatos a vetor de movimento e uma imagem atual, é igual a zero e uma segunda distância POC, entre uma segunda imagem de referência associada com uma lista de candidatos a vetor de movimento e a imagem atual, é igual a zero,

onde determinar o primeiro vetor de movimento derivado e o segundo vetor de movimento derivado a partir da pluralidade de vetores de movimento derivado com base na métrica de custo se baseia adicionalmente na determinação.

24. Aparelho, de acordo com a reivindicação 14, no qual um ou mais processadores são adicionalmente configurados para:

determinar se uma primeira distância de contagem

de ordem de imagem (POC), entre uma primeira imagem de referência associada com uma primeira lista de candidatos a vetor de movimento e uma imagem atual, é igual a zero e uma segunda distância POC, entre uma segunda imagem de referência associada com uma segunda lista de candidatos a vetor de movimento e a imagem atual, é igual a zero, onde determinar o primeiro vetor de movimento derivado e o segundo vetor de movimento derivado a partir da pluralidade de vetores de movimento derivado, com base na métrica de custo que gera a pluralidade de vetores de movimento derivado, se baseia adicionalmente na determinação da primeira distância POC, entre a primeira imagem de referência associada com a primeira lista de candidatos a vetor de movimento e a imagem atual, é igual a zero e a segunda distância POC, entre a segunda imagem de referência associada com a segunda lista de candidatos a vetor de movimento e a imagem atual, é igual à determinação zero.

25. Aparelho, de acordo com a reivindicação 14, no qual os um ou mais processadores são adicionalmente configurados para determinar a pelo menos uma restrição a partir da sequência de bits de vídeo codificado.

26. Aparelho, de acordo com a reivindicação 14, no qual a métrica de custo inclui uma soma de diferenças absolutas.

27. Aparelho configurado para decodificar os dados de vídeo a partir de uma sequência de bits de vídeo codificada, o aparelho compreendendo:

meios para gerar, utilizando a derivação de vetor de movimento de lado de decodificador (DMVD), uma pluralidade de vetores de movimento derivado;

meios para determinar um primeiro vetor de movimento derivado e um segundo vetor de movimento derivado a partir da pluralidade de vetores de movimento derivado com base em uma métrica de custo, compreendendo:

meios para determinar que o primeiro vetor de movimento derivado e o segundo vetor de movimento derivado satisfazem pelo menos uma restrição em uma relação entre o primeiro vetor de movimento derivado e o segundo vetor de movimento derivado, a relação compreendendo: o primeiro vetor de movimento derivado e o segundo vetor de movimento derivado possuindo uma diferença de vetor de movimento simétrica, o primeiro vetor de movimento derivado e o segundo vetor de movimento derivado possuindo uma diferença de vetor de movimento pseudo-simétrica, o primeiro vetor de movimento derivado e o segundo vetor de movimento derivado sendo antissimétricos, o primeiro vetor de movimento derivado e o segundo vetor de movimento derivado possuindo uma diferença de vetor de movimento antissimétrica, ou o primeiro vetor de movimento derivado e o segundo vetor de movimento derivado possuindo uma diferença de vetor de movimento pseudo-antissimétrica;

e

meios para decodificar um bloco atual utilizando o primeiro vetor de movimento derivado determinado e o segundo vetor de movimento derivado utilizando previsão dupla.

28. Aparelho, de acordo com a reivindicação 27, no qual utilizar DMVD para gerar a pluralidade de vetores de movimento derivado compreende utilizar um dentre um processo de fluxo Ótico Bidirecional (BIO), um processo de

conversão ascendente de taxa de quadro (FRUC), um processo de Combinação Bilateral, um processo de combinação de gabarito FRUC ou um processo de combinação de gabarito bilateral.

29. Aparelho configurado para codificar dados de vídeo, o aparelho compreendendo:

uma memória configurada para armazenar dados de vídeo; e

um ou mais processadores em comunicação com a memória, os um ou mais processadores configurados para:

gerar, utilizando a derivação de vetor de movimento de lado de decodificador (DMVD), uma pluralidade de vetores de movimento derivado;

determinar um primeiro vetor de movimento derivado e um segundo vetor de movimento derivado a partir da pluralidade de vetores de movimento derivado com base em uma métrica de custo, compreendendo:

determinar que o primeiro vetor de movimento derivado e o segundo vetor de movimento derivado satisfazem pelo menos uma restrição em uma relação entre um primeiro vetor de movimento derivado e o segundo vetor de movimento derivado, a relação compreendendo: o primeiro vetor de movimento derivado e o segundo vetor de movimento derivado possuindo uma diferença de vetor de movimento simétrica, o primeiro vetor de movimento derivado e o segundo vetor de movimento derivado possuindo uma diferença de vetor de movimento pseudo-simétrica, o primeiro vetor de movimento derivado e o segundo vetor de movimento derivado sendo antissimétricos, o primeiro vetor de movimento derivado e o segundo vetor de movimento derivado possuindo uma diferença

de vetor de movimento antissimétrica, ou o primeiro vetor de movimento derivado e o segundo vetor de movimento derivado possuindo uma diferença de vetor de movimento pseudo-antissimétrica; e

codificar um bloco atual utilizando o primeiro vetor de movimento derivado e o segundo vetor de movimento derivado utilizando a previsão dupla.

30. Aparelho, de acordo com a reivindicação 29, no qual utilizar DMVD para gerar a pluralidade de vetores de movimento derivado compreende utilizar um processo de fluxo Ótico Bidirecional (BIO), um processo de conversão ascendente de taxa de quadro (FRUC), um processo de Combinação Bilateral, um processo de combinação de gabarito FRUC, ou um processo de combinação de gabarito bilateral.

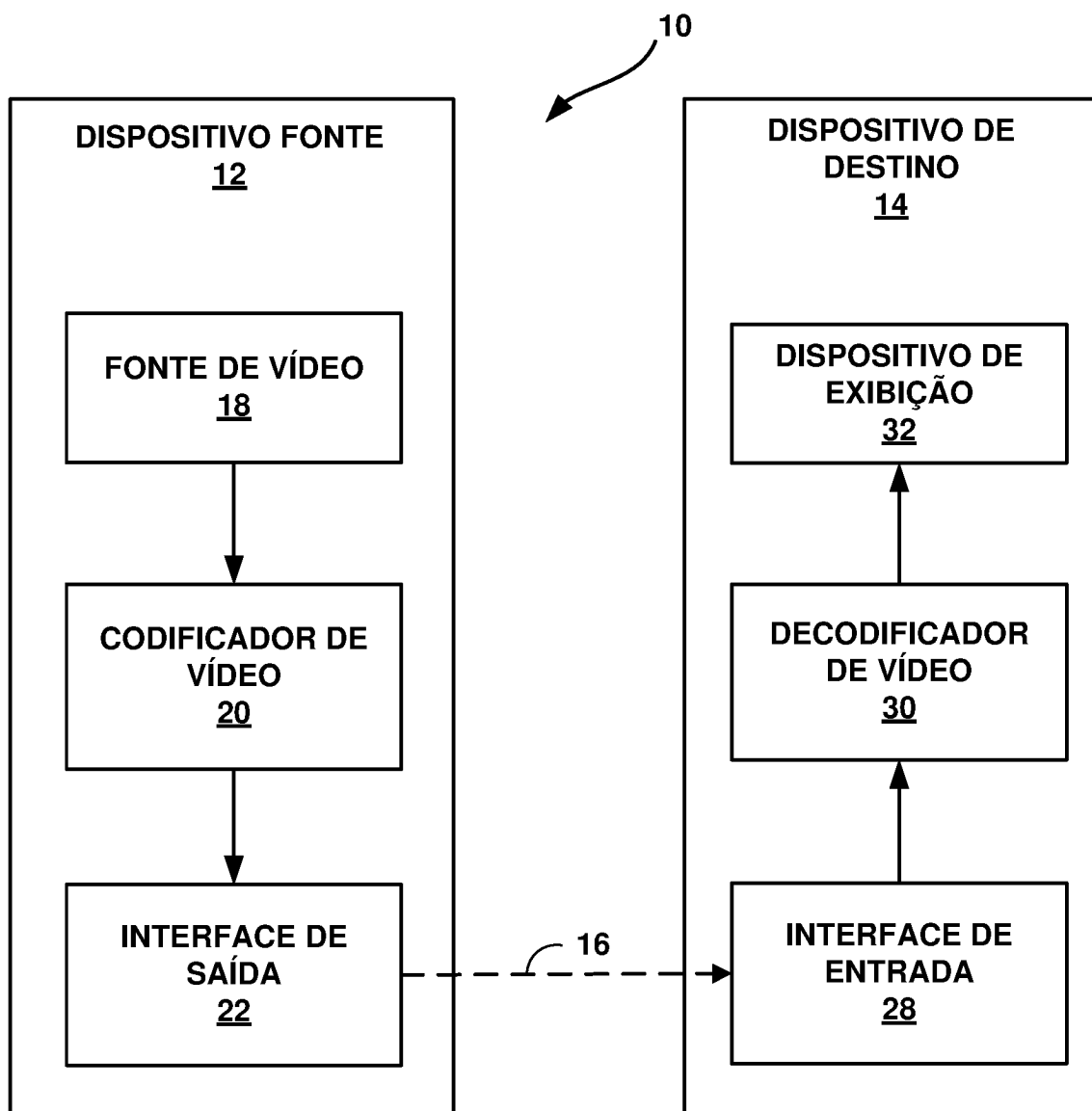


FIG. 1

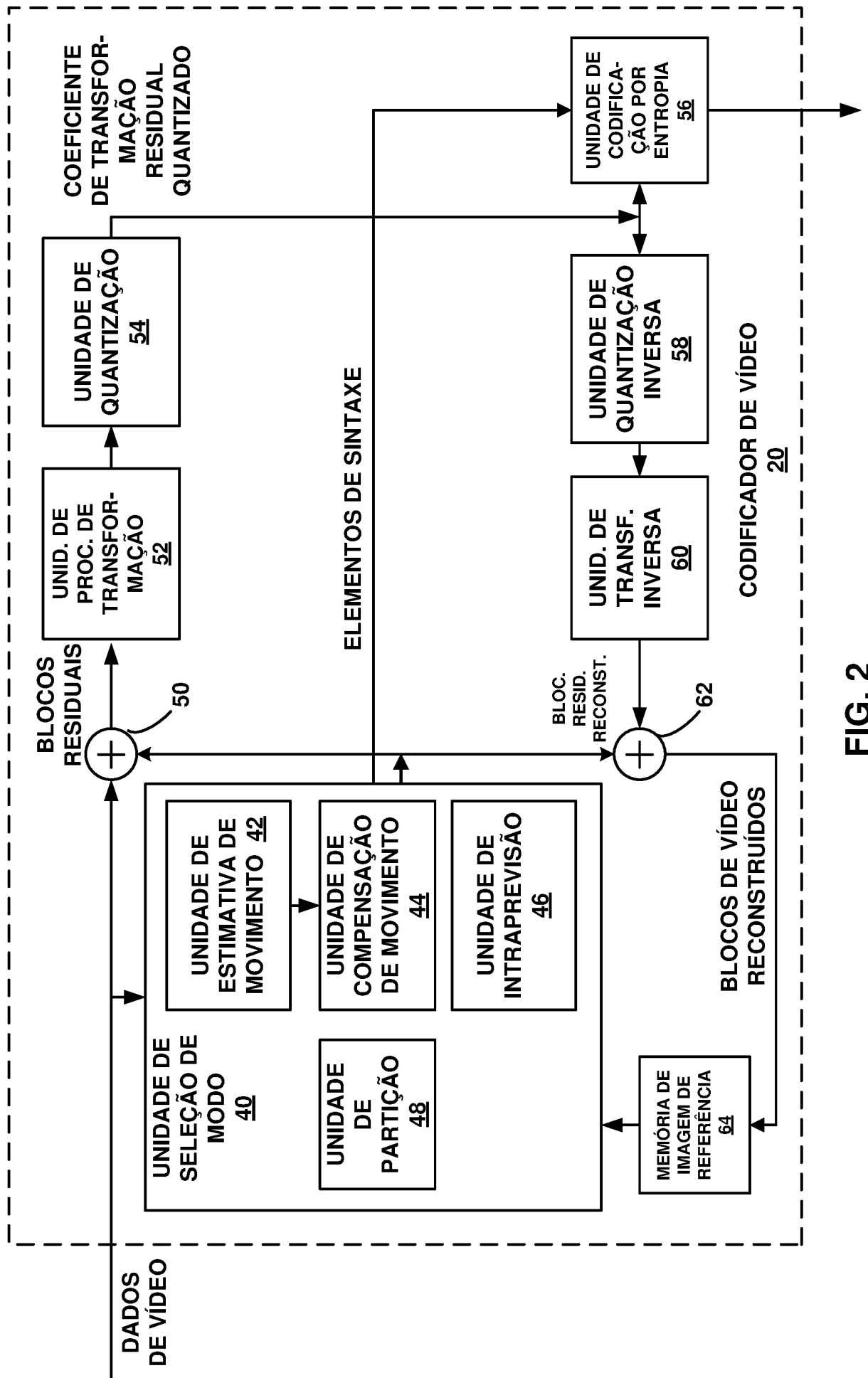


FIG. 2

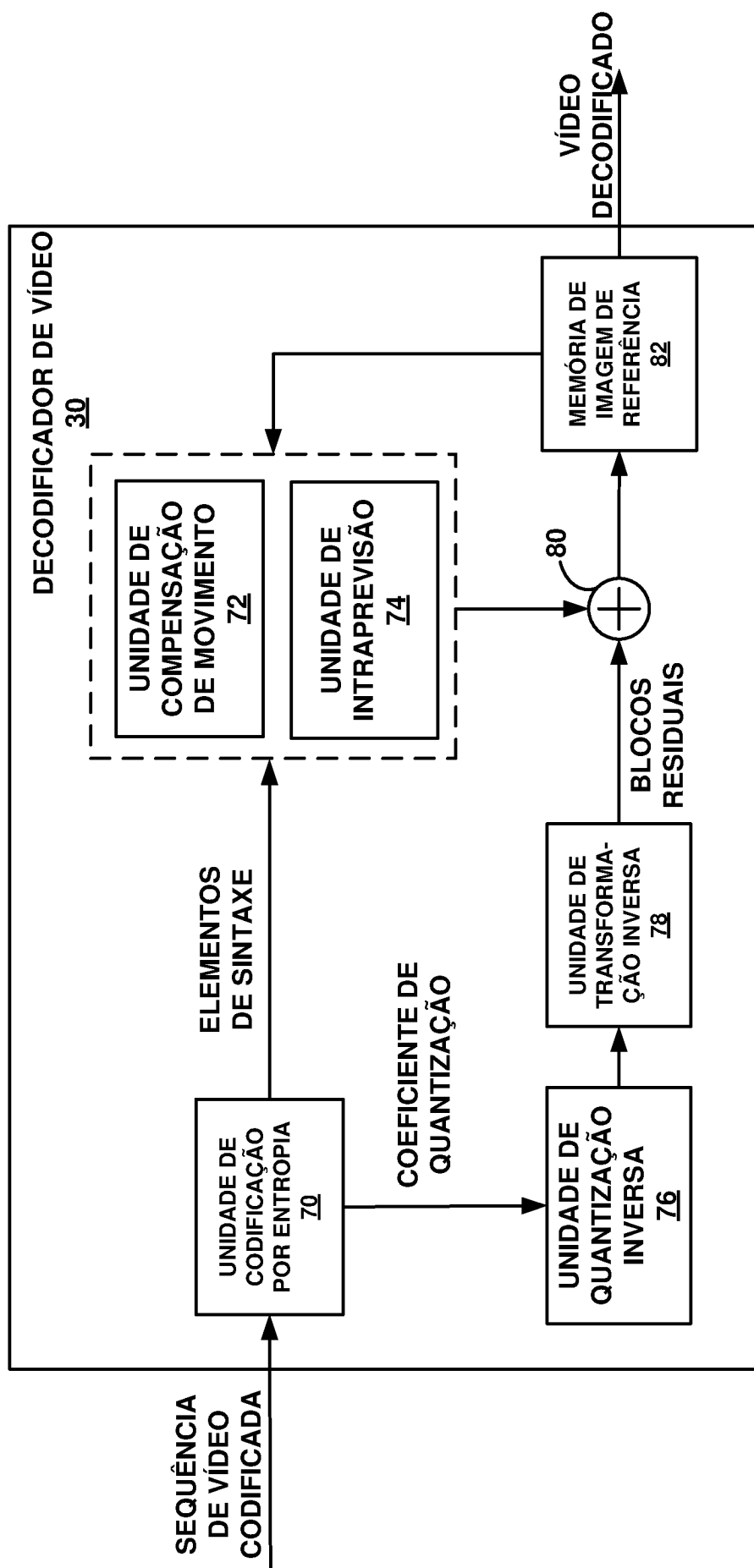


FIG. 3

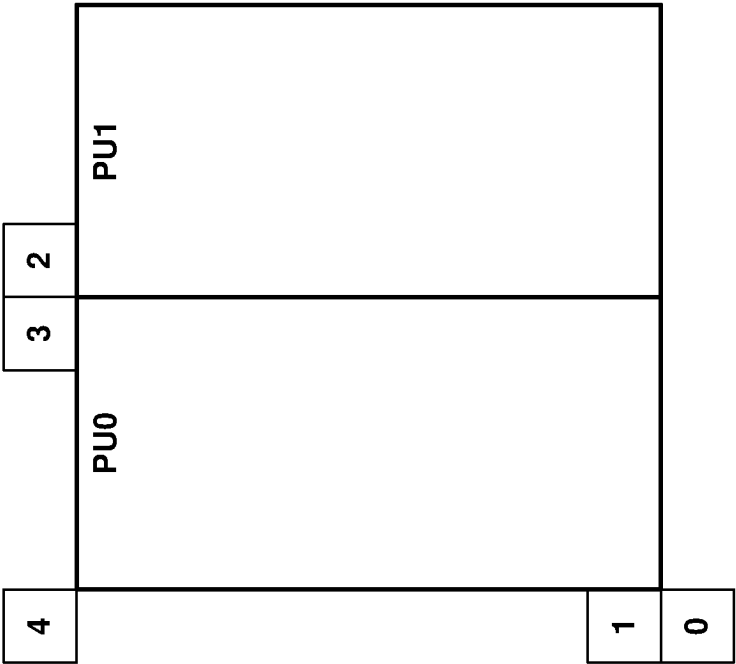


FIG. 4B

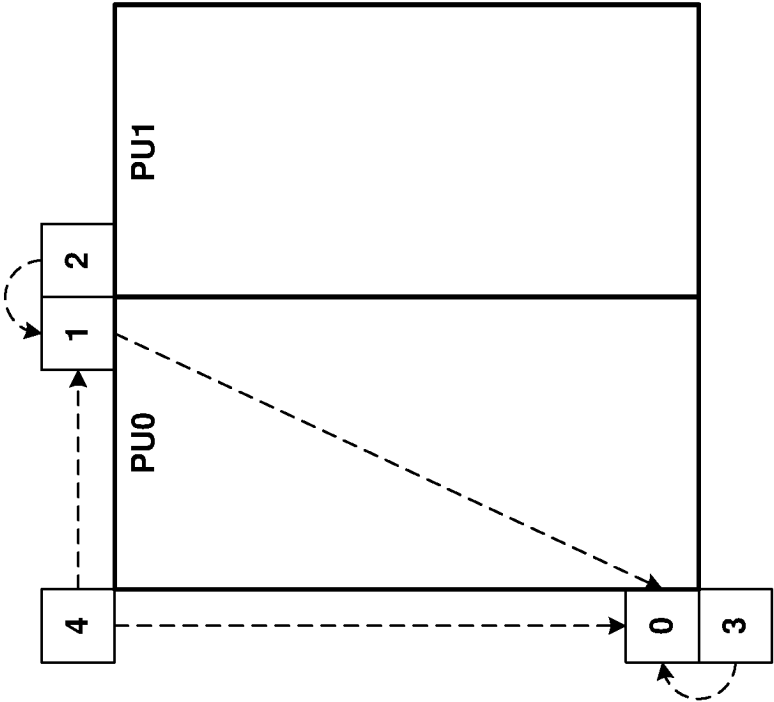


FIG. 4A

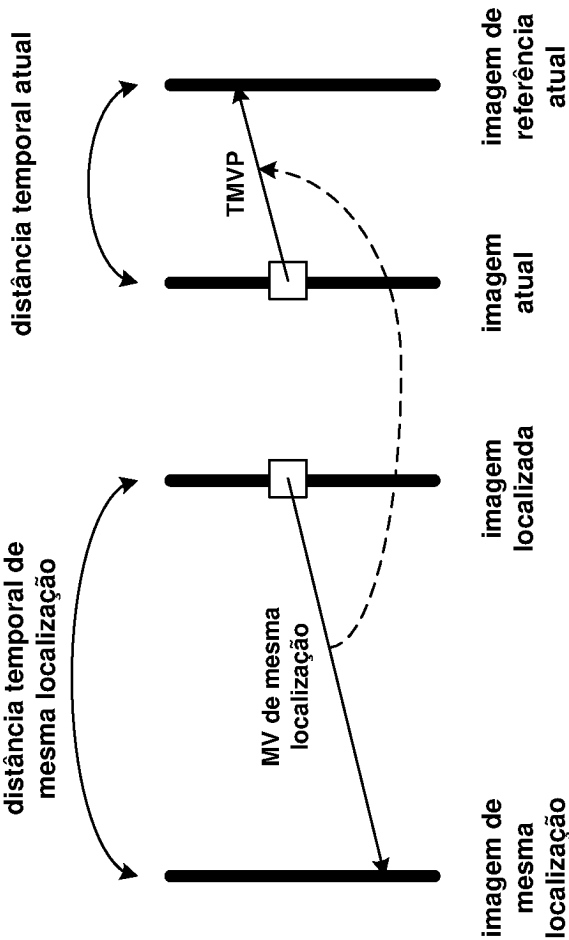


FIG. 5B

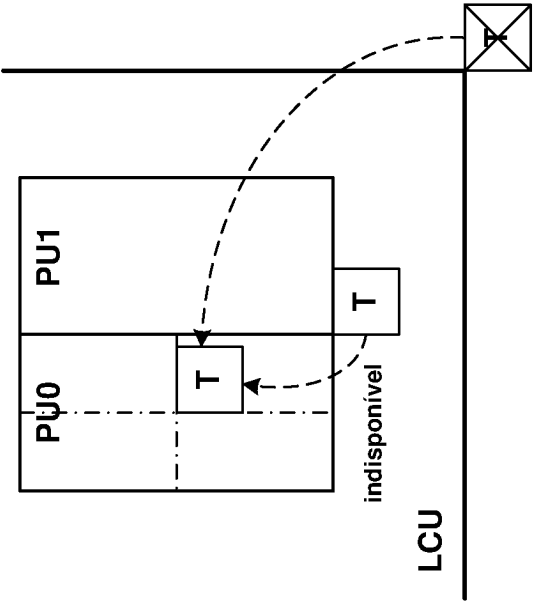


FIG. 5A

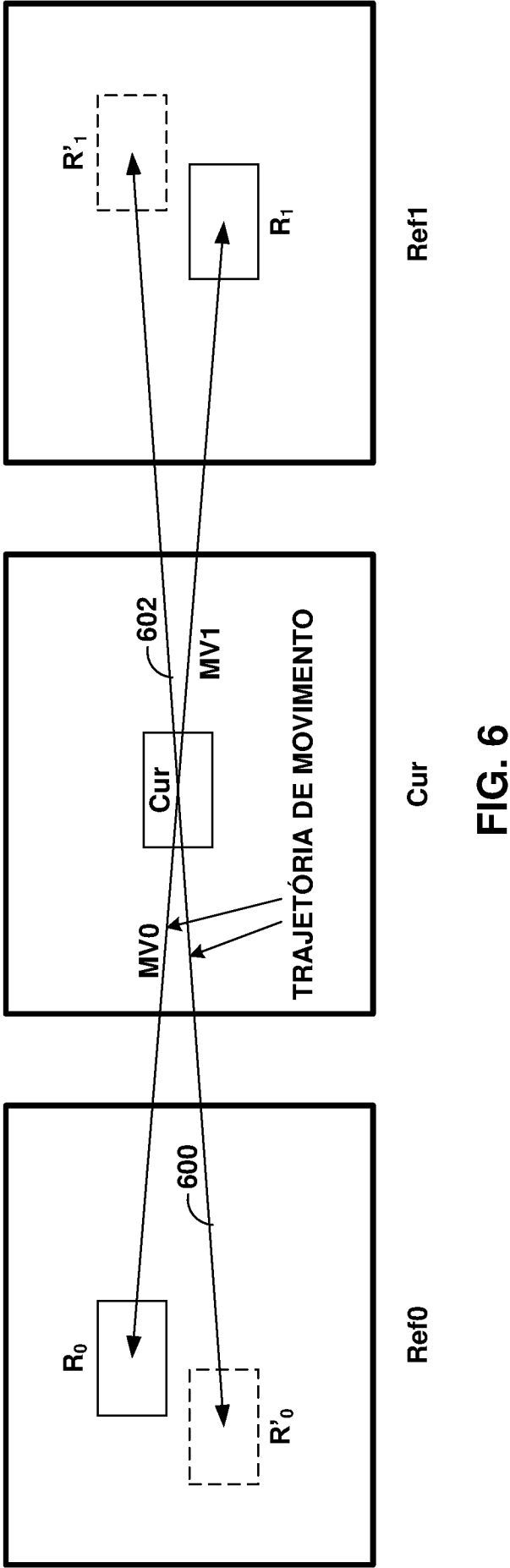


FIG. 6

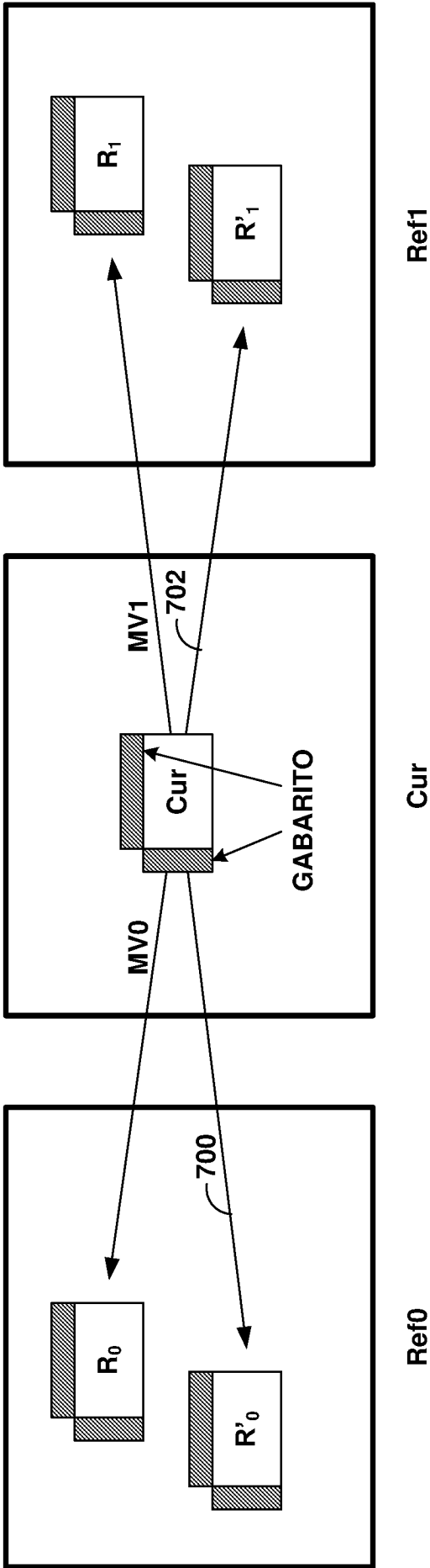
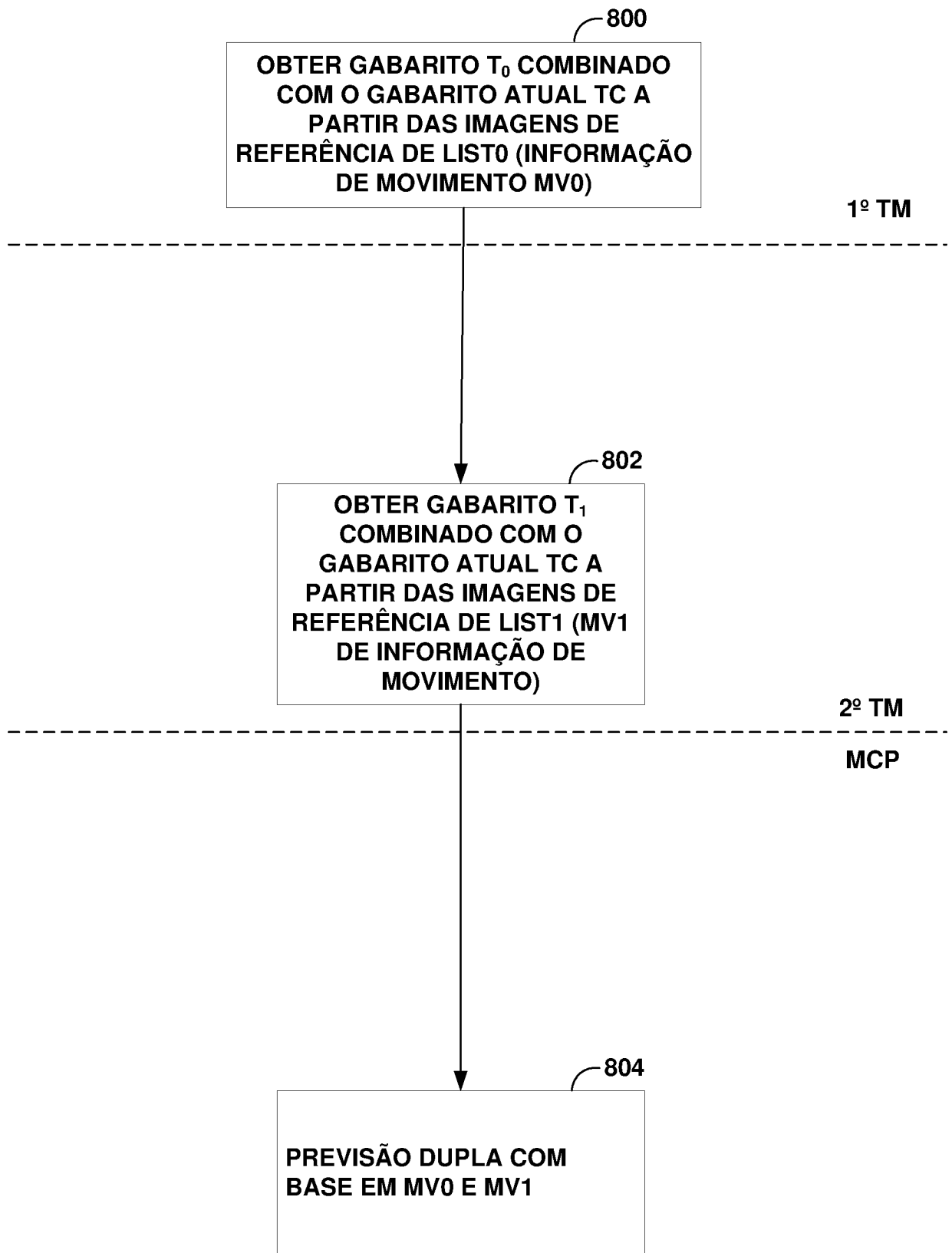


FIG. 7

**FIG. 8**

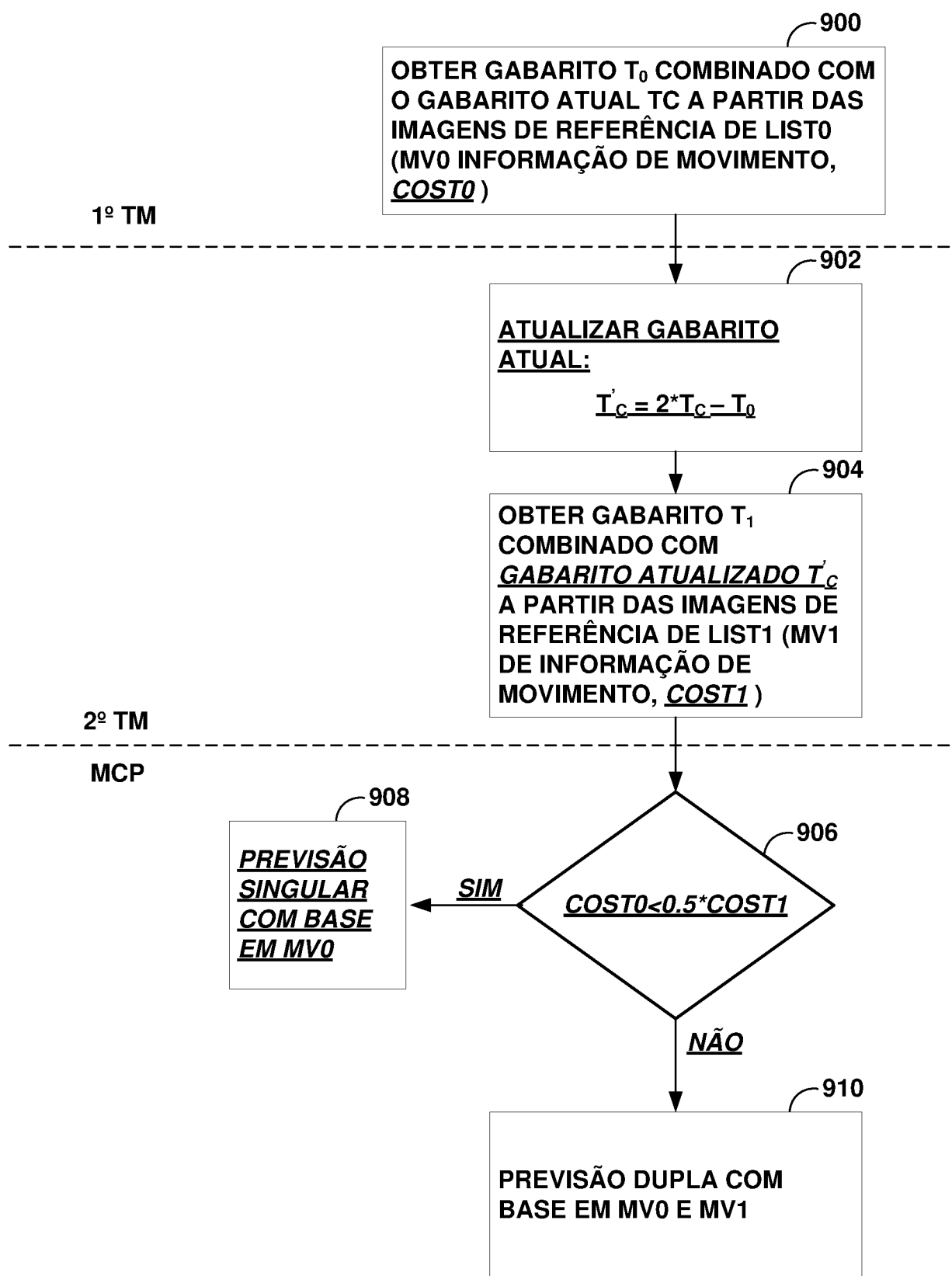


FIG. 9

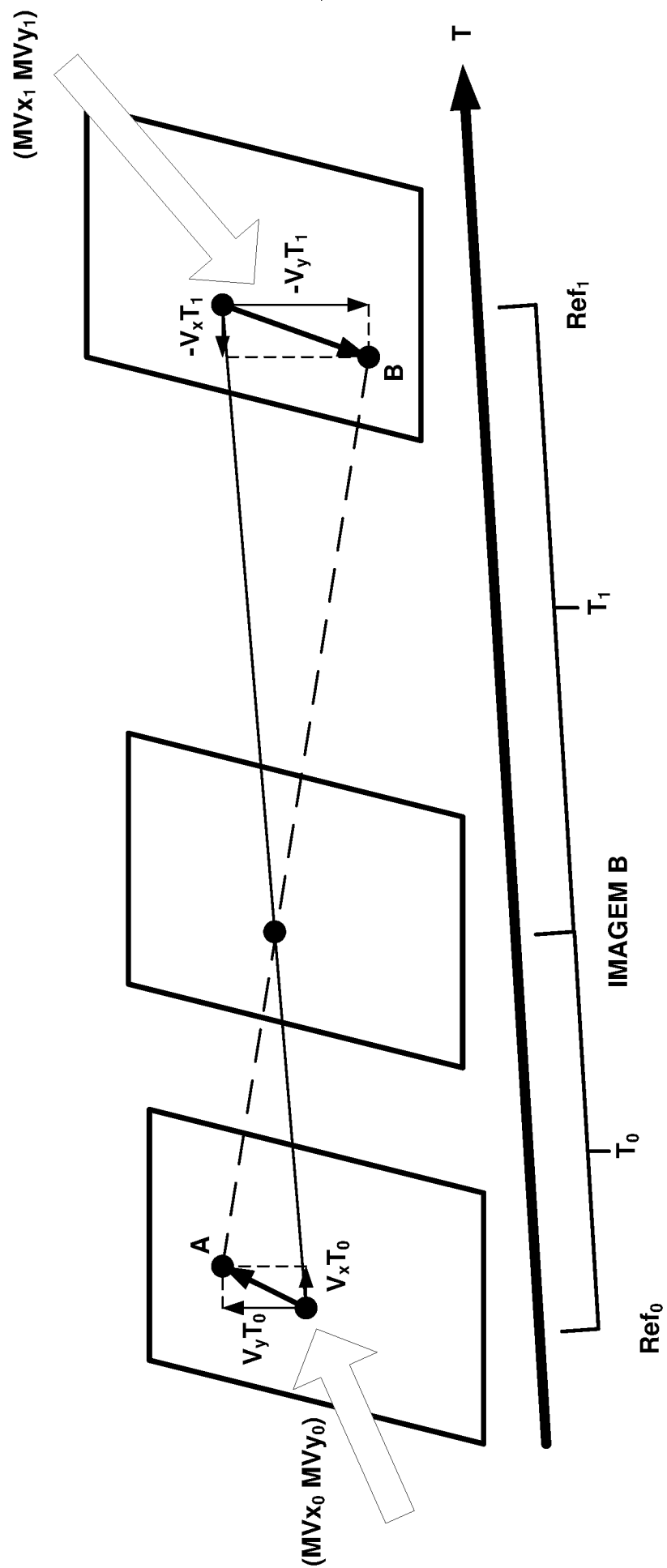


FIG. 10

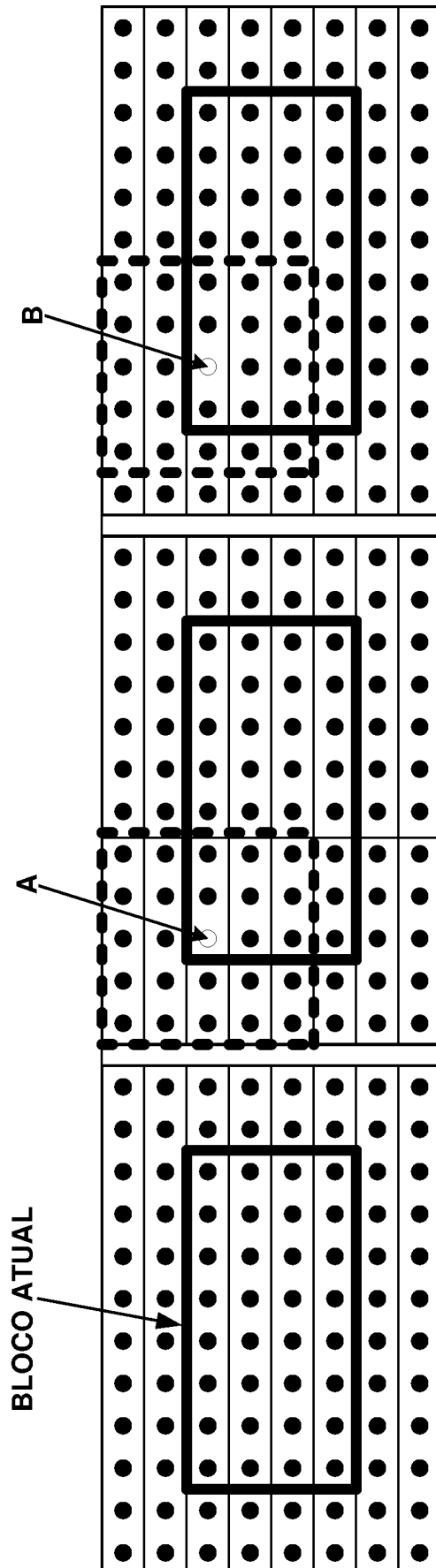


FIG. 11

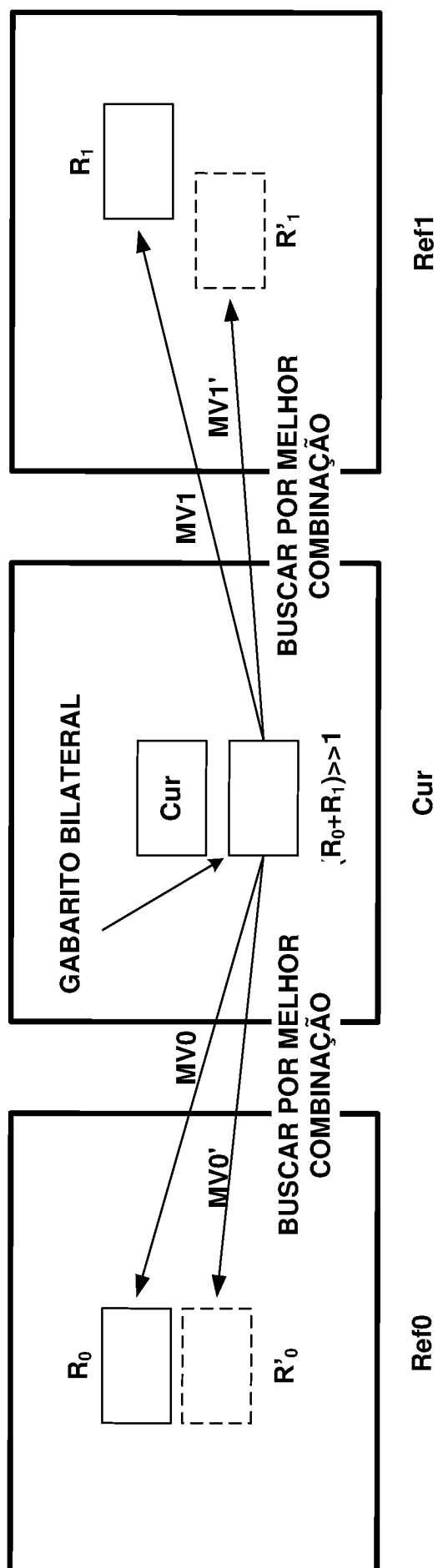


FIG. 12

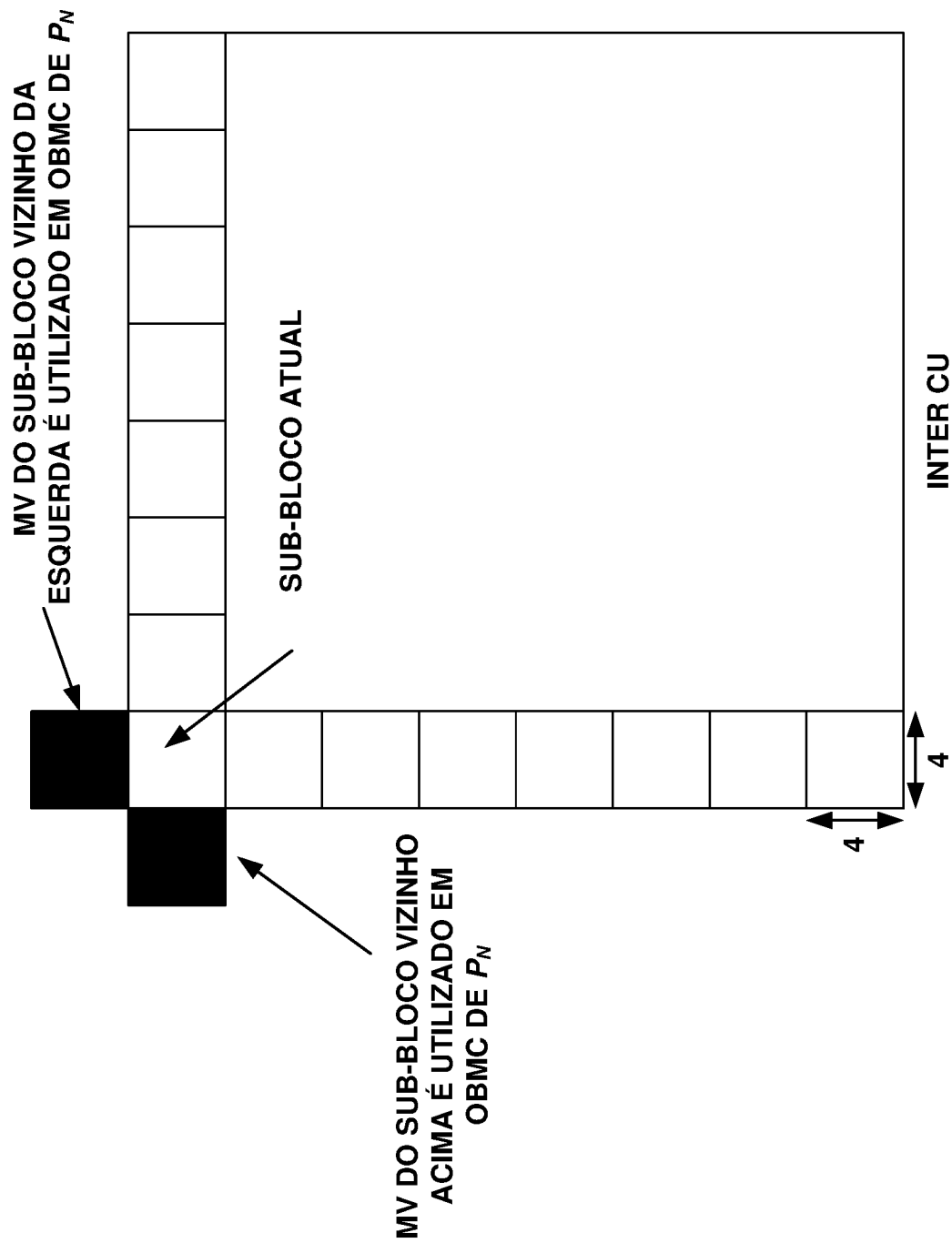


FIG. 13A

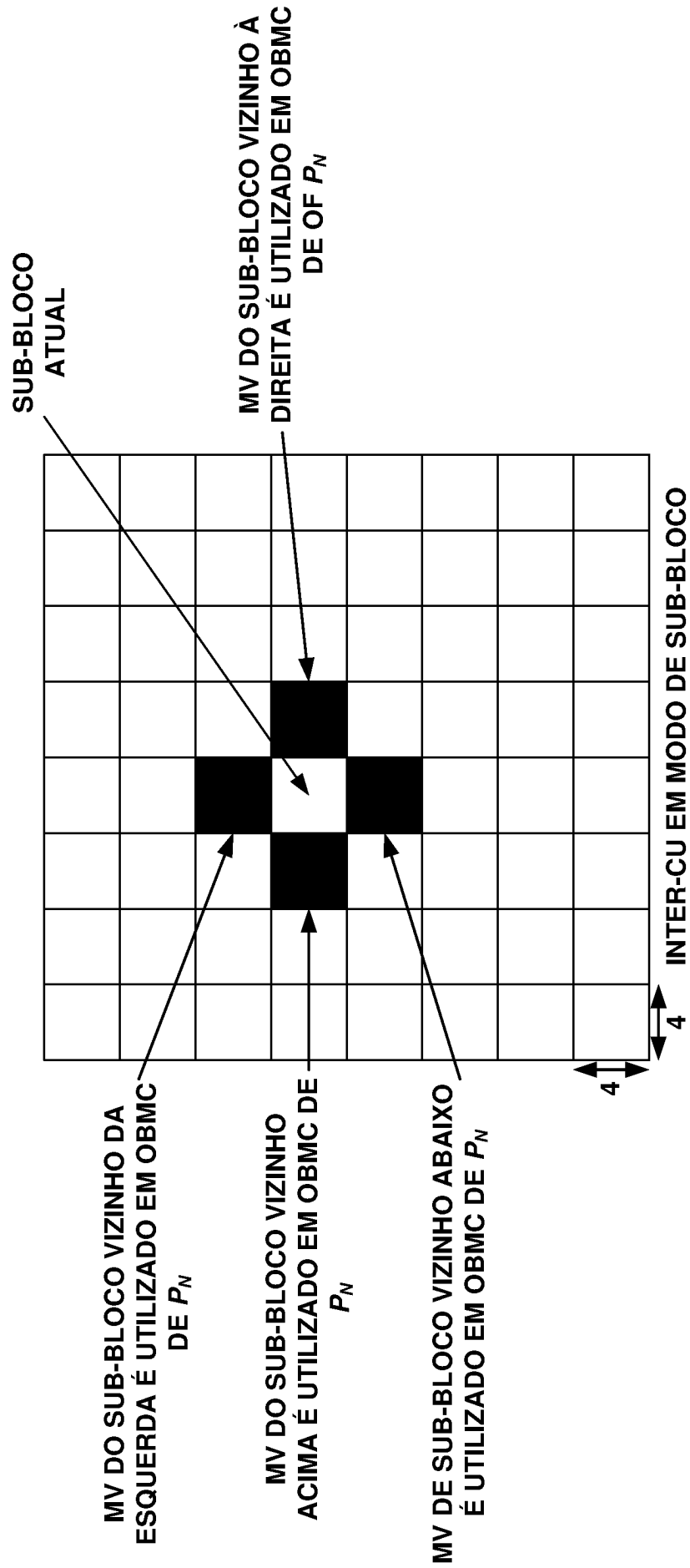


FIG. 13B

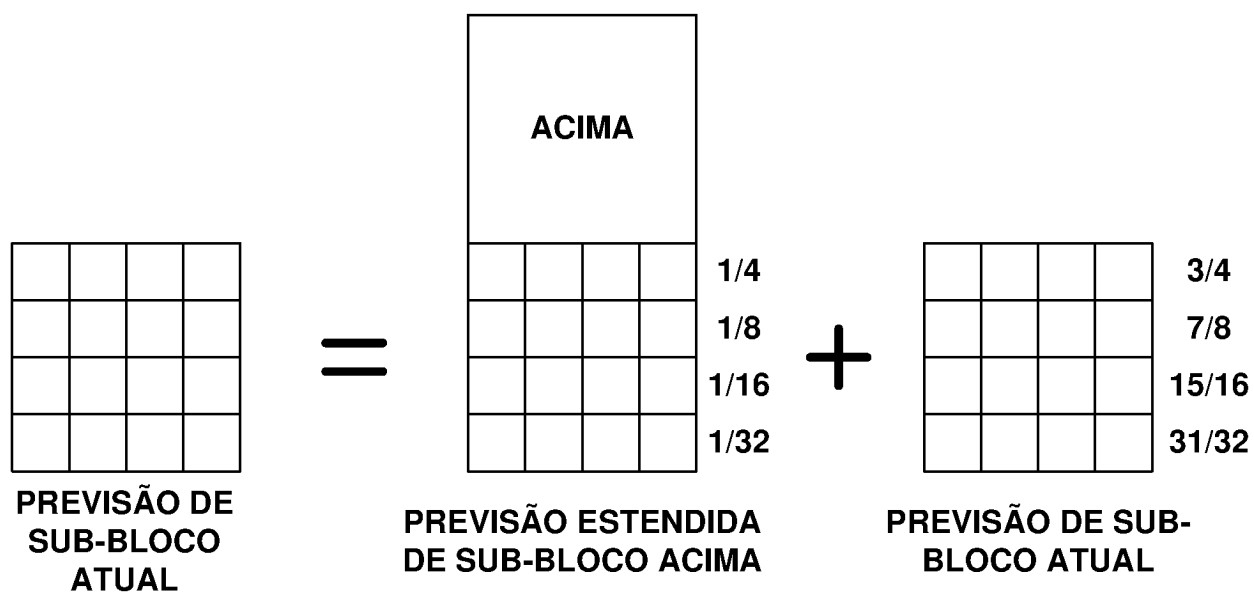


FIG. 14A

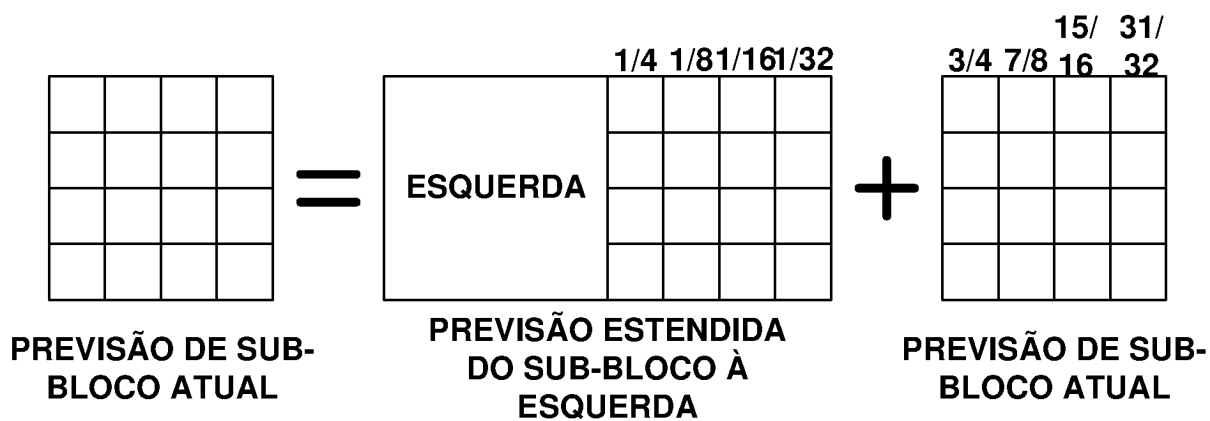


FIG. 14B

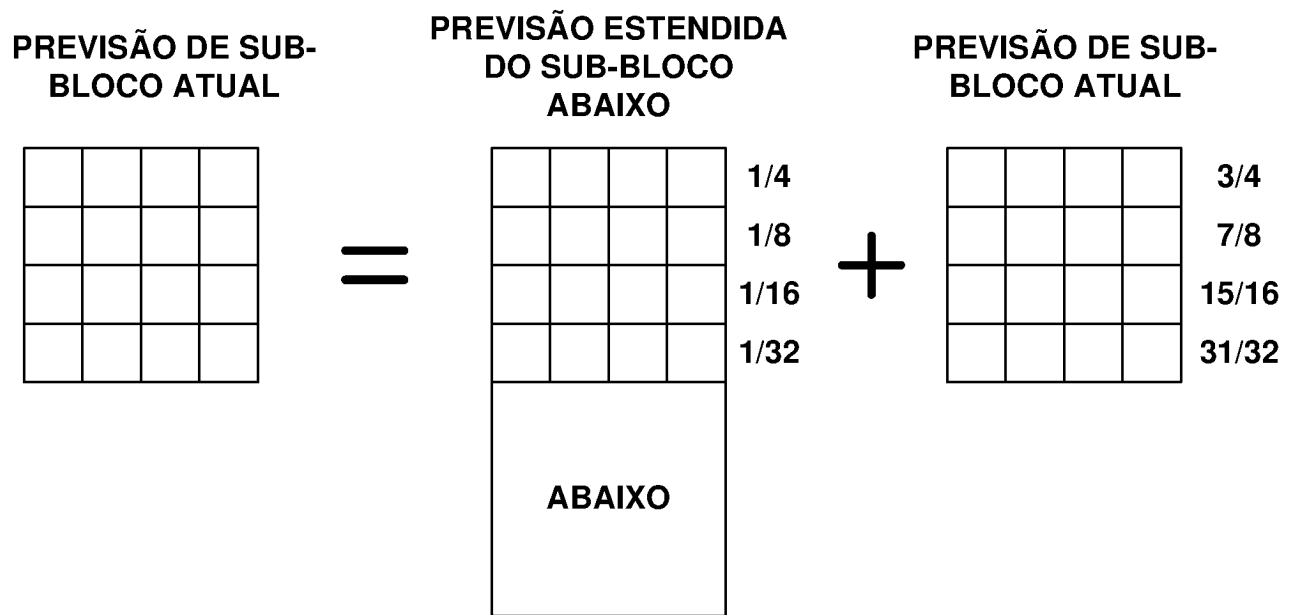


FIG. 14C



FIG. 14D

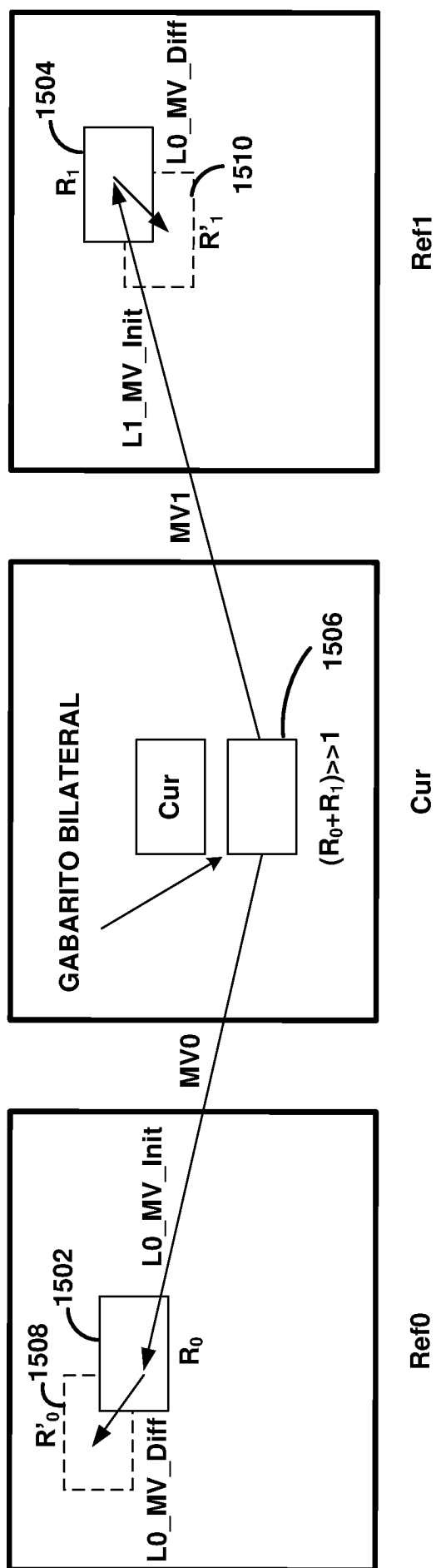


FIG. 15

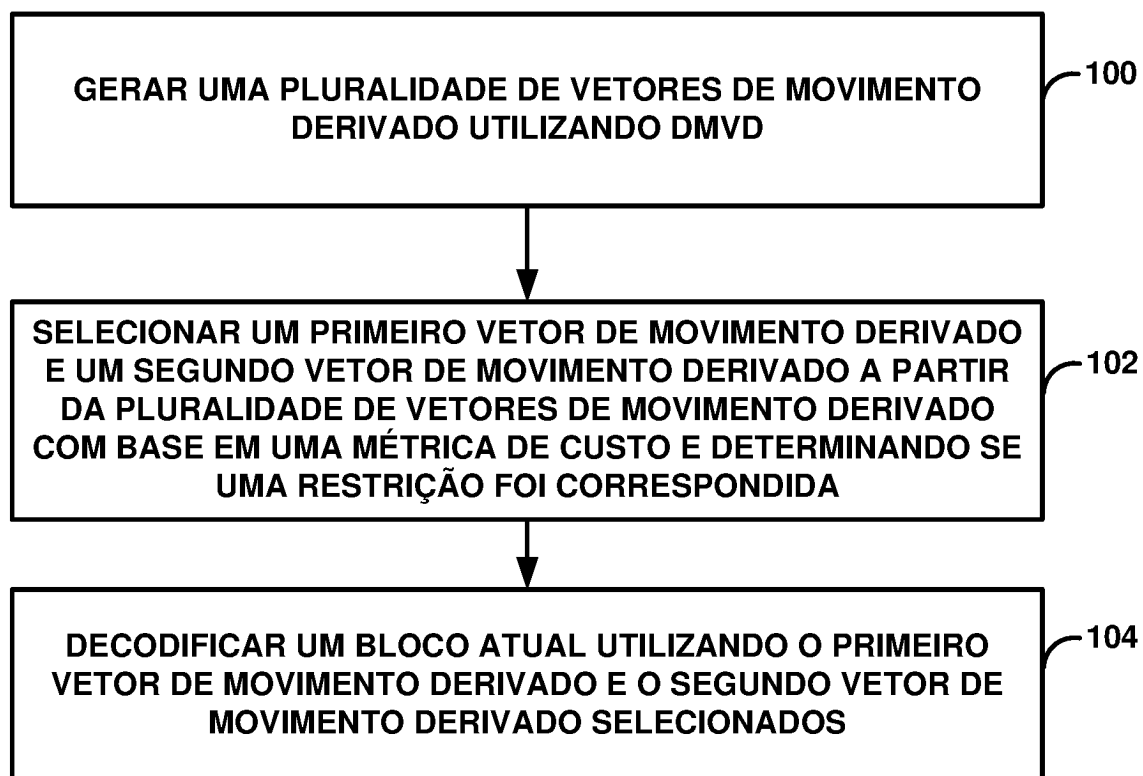


FIG. 16

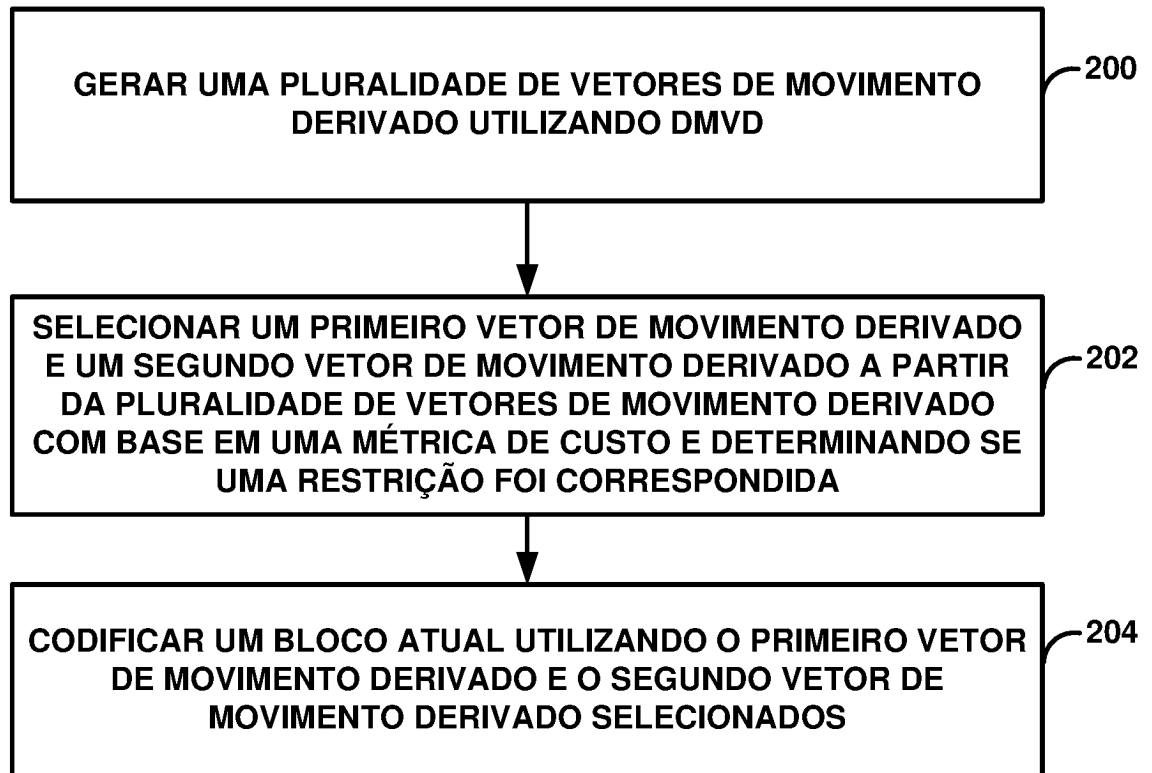


FIG. 17

RESUMO**"INFORMAÇÃO DE VETOR DE MOVIMENTO DE RESTRIÇÃO DERIVADA POR DERIVAÇÃO DE VETOR DE MOVIMENTO DE LADO DE DECODIFICADOR"**

As técnicas relacionadas com a derivação de vetor de movimento de lado de decodificador (DMVD) são descritas. Por exemplo, essa descrição descreve as técnicas relacionadas com aplicar uma ou mais restrições à informação de movimento, tal como um vetor de movimento (MV) derivado pela DMVD, e/ou uma diferença de MV entre um MV inicial e um MV derivado pela DMVD. Quando a restrição é aplicada à DMVD, em determinados exemplos, apenas a informação de movimento derivado, que corresponde à restrição, é considerada como informação de movimento válida. As condições podem ser impostas às restrições.