



**República Federativa do Brasil**

Ministério do Desenvolvimento, Indústria,  
Comércio e Serviços

Instituto Nacional da Propriedade Industrial



**(11) BR 112016007916-7 B1**

**(22) Data do Depósito:** 09/10/2014

**(45) Data de Concessão:** 02/05/2023

---

**(54) Título:** SINALIZAÇÃO PARA ARMAZENAMENTO TEMPORÁRIO DE FIGURAÇÃO SUBDECODIFICADA (SUB-DPB) COM BASE EM OPERAÇÕES DE DPB EM CODIFICAÇÃO DE VÍDEO

**(51) Int.Cl.:** H04N 19/70; H04N 19/463; H04N 19/31; H04N 19/423; H04N 19/597; (...).

**(30) Prioridade Unionista:** 08/10/2014 US 14/509,797; 10/10/2013 US 61/889,515; 14/10/2013 US 61/890,841.

**(73) Titular(es):** QUALCOMM INCORPORATED.

**(72) Inventor(es):** ADARSH KRISHNAN RAMASUBRAMONIAN; YE-KUI WANG.

**(86) Pedido PCT:** PCT US2014059925 de 09/10/2014

**(87) Publicação PCT:** WO 2015/054509 de 16/04/2015

**(85) Data do Início da Fase Nacional:** 08/04/2016

**(57) Resumo:** SINALIZAÇÃO PARA ARMAZENAMENTO TEMPORÁRIO DE FIGURAÇÃO SUBDECODIFICADA (SUB-DPB) COM BASE EM OPERAÇÕES DE DPB EM CODIFICAÇÃO DE VÍDEO. Trata-se de um método para decodificar dados de vídeo que inclui receber um fluxo de bits de vídeo criptados que inclui uma pluralidade de figurações e armazenar a pluralidade de figurações em um ou mais sub-DPBs. O método inclui adicionalmente receber um conjunto respectivo de parâmetros de sub-DPB para cada ponto de operação respectivo do fluxo de bits de vídeo criptados, aplicar o conjunto respectivo de parâmetros de sub-DPB a todas as camadas de um conjunto de camadas de saída para cada ponto de operação respectivo e realizar um processo de gerenciamento de sub-DPB nos um ou mais sub-DPBs de acordo com os conjuntos únicos respectivos recebidos de parâmetros de sub-DPB.

"SINALIZAÇÃO PARA ARMAZENAMENTO TEMPORÁRIO DE FIGURAÇÃO  
SUBDECODIFICADA (SUB-DPB) COM BASE EM OPERAÇÕES DE DPB EM  
CODIFICAÇÃO DE VÍDEO"

[0001] O presente pedido reivindica o benefício do Pedido Provisório nº U.S.61/889.515, depositado em 10 de outubro de 2013 e do Pedido Provisório nº U.S.61/890.841, depositado 14 de outubro de 2013, sendo que o conteúdo de cada um é incorporado a título de referência no presente documento.

CAMPO DA TÉCNICA

[0002] Esta revelação refere-se à codificação de vídeo.

ANTECEDENTES

[0003] As capacidades de vídeo digital podem ser incorporadas em uma ampla faixa de dispositivos, incluindo televisões digitais, sistemas de difusão direta digital, sistemas de difusão sem fio, assistentes digitais pessoais (PDAs), computadores do tipo laptop ou desktop, computadores do tipo tablet, leitores de e-book, câmeras digitais, dispositivo de gravação digitais, reprodutores de mídias digitais, dispositivos de jogo de video game, consoles de video game, telefones de rádio-satélite ou celular, os chamados de "telefones inteligentes", dispositivos de teleconferência por vídeo, dispositivos de transmissão contínua de vídeo e similares. Os dispositivos de vídeo digital implantam técnicas de codificação de vídeo, como as descritas nos padrões definidos por MPEG-2, MPEG-4, ITU-T H.263, ITU-T H.264/MPEG-4, Parte 10, Codificação de Vídeo Avançada (AVC), o padrão de Codificação de Vídeo de Alta Eficiência (HEVC) atualmente sob desenvolvimento e extensões de tais padrões. Os dispositivos de vídeo podem transmitir, receber, criptar, decodificar e/ou armazenar informações de vídeo digital de

maneira mais eficiente implantando-se tais técnicas de codificação de vídeo.

[0004] As técnicas de codificação de vídeo incluem previsão espacial (intrafiguração) e/ou previsão temporal (interfiguração) para reduzir ou remover a redundância inerente em sequências de vídeo. Para a codificação de vídeo com base em bloco, uma fatia de vídeo (por exemplo, um quadro de vídeo ou uma porção de um quadro de vídeo) pode ser particionada em blocos de vídeo, que também podem ser denominados de blocos em árvore, unidades de codificação (CUs) e/ou nós de codificação. Os blocos de vídeo em uma fatia intracodificada (I) de uma figuração são criptados com o uso de uma previsão espacial em relação às amostras de referência em blocos vizinhos na mesma figuração. Os blocos de vídeo em uma fatia intercodificada (P ou B) de uma figuração podem usar previsão espacial em relação às amostras de referência em blocos vizinhos na mesma previsão de figuração ou temporal em relação às amostras de referência em outras figurações de referência. As figurações podem ser denominadas como quadros, e as figurações de referência podem ser denominadas como quadros de referência.

[0005] A previsão temporal ou espacial resulta em um bloco preditivo para um bloco a ser codificado. Os dados residuais representam diferenças de pixel entre o bloco original a ser codificado e o bloco preditivo. Um bloco intercodificado é criptado de acordo com um vetor de movimento que aponta para um bloco de amostras de referência que forma o bloco preditivo, sendo que os dados residuais indicam a diferença entre o bloco codificado e o bloco preditivo. Um bloco intracodificado é criptado de acordo com um modo de intracodificação e os dados residuais. Para compressão adicional, os dados residuais

podem ser transformados do domínio de pixel para um domínio de transformada, resultando em coeficientes residuais que podem, em seguida, ser quantizados. O coeficiente de transformada quantizado, disposto inicialmente em uma matriz bidimensional e pode ter sua varredura realizada a fim de produzir um vetor monodimensional de coeficiente de transformada, e a codificação por entropia pode ser aplicada para conseguir ainda mais compressão.

#### SUMÁRIO

[0006] Em geral, essa revelação descreve técnicas de codificação de vídeo. Em particular, as técnicas se referem as técnicas de codificação de vídeo e sinalização que podem ser aplicadas aos armazenamentos temporários de figuração subdecodificada (sub-DPBs). Em um ou mais exemplos da revelação, técnicas para sinalizar parâmetros de DPB e de sub-DPB para uma ou mais camadas e/ou uma ou mais subcamadas são descritas. Utilizando-se as técnicas da revelação, a eficácia de sinalização pode ser aumentada para fluxos de bits de vídeo de múltiplas camadas que incluem uma ou mais subcamadas.

[0007] Em um exemplo da revelação, um método de decodificação de dados de vídeo compreende receber um fluxo de bits de vídeo criptados que inclui uma pluralidade de figurações, armazenar a pluralidade de figurações em um ou mais sub-DPBs, receber um conjunto respectivo de parâmetros de sub-DPB para cada ponto de operação respectivo do fluxo de bits de vídeo criptados, aplicar o conjunto respectivo de parâmetros de sub-DPB a todas as camadas de um conjunto de camadas de saída para cada ponto de operação respectivo, e realizar um processo de gerenciamento de sub-DPB nos um ou mais sub-DPBs de acordo com os conjuntos únicos respectivos recebidos de parâmetros de sub-DPB.

[0008] Em outro exemplo da revelação, um aparelho configurado para decodificar dados de vídeo compreende um ou mais sub-DPBs configurados para armazenar dados de vídeo e um decodificador de vídeo configurado para receber um fluxo de bits de vídeo criptados que inclui uma pluralidade de figurações, armazenar a pluralidade de figurações nos um ou mais sub-DPBs, receber um conjunto respectivo de parâmetros de sub-DPB para cada ponto de operação respectivo do fluxo de bits de vídeo criptados, aplicar o conjunto respectivo de parâmetros de sub-DPB a todas as camadas de um conjunto de camadas de saída para cada ponto de operação respectivo e realizar um processo de gerenciamento de sub-DPB nos um ou mais sub-DPBs de acordo com os conjuntos únicos respectivos recebidos de parâmetros de sub-DPB.

[0009] Em outro exemplo da revelação, um aparelho configurado para decodificar dados de vídeo compreende meios para receber um fluxo de bits de vídeo criptados que inclui uma pluralidade de figurações, meios para armazenar a pluralidade de figurações em um ou mais sub-DPBs, meios para receber um conjunto respectivo de parâmetros de sub-DPB para cada ponto de operação respectivo do fluxo de bits de vídeo criptados, meios para aplicar o conjunto respectivo de parâmetros de sub-DPB a todas as camadas de um conjunto de camadas de saída para cada ponto de operação respectivo, e meios para realizar um processo de gerenciamento de sub-DPB nos um ou mais sub-DPBs de acordo com os conjuntos únicos respectivos recebidos de parâmetros de sub-DPB.

[0010] Em outro exemplo essa revelação descreve instruções de armazenamento de mídia de armazenamento legível por computador que, quando executadas, levam um ou mais processadores de um

dispositivo configurado para decodificar dados de vídeo para receber um fluxo de bits de vídeo criptados que inclui uma pluralidade de figurações, armazenar a pluralidade de figurações nos um ou mais sub-DPBs, receber um conjunto respectivo de parâmetros de sub-DPB para cada ponto de operação respectivo do fluxo de bits de vídeo criptados, aplicar o conjunto respectivo de parâmetros de sub-DPB a todas as camadas de um conjunto de camadas de saída para cada ponto de operação respectivo, e realizar um processo de gerenciamento de sub-DPB nos um ou mais sub-DPBs de acordo com os conjuntos únicos respectivos recebidos de parâmetros de sub-DPB.

[0011] Em outro exemplo da revelação, um método de criptar dados de vídeo compreende reconstruir uma pluralidade de figurações de um fluxo de bits de vídeo criptados, armazenar a pluralidade de figurações em um ou mais sub-DPBs, gerar um conjunto respectivo de parâmetros de sub-DPB para cada ponto de operação respectivo do fluxo de bits de vídeo criptados, aplicar o conjunto respectivo de parâmetros de sub-DPB a todas as camadas de um conjunto de camadas de saída para cada ponto de operação respectivo, realizar um processo de gerenciamento de sub-DPB nos um ou mais sub-DPBs de acordo com os conjuntos únicos respectivos gerados de parâmetros de sub-DPB, e sinalizar o conjunto respectivo de parâmetros de sub-DPB no fluxo de bits de vídeo criptados.

[0012] Em outro exemplo da revelação, um aparelho configurado para criptar dados de vídeo compreende um ou mais sub-DPBs configurados para armazenar dados de vídeo, e um criptador de vídeo configurado para reconstruir uma pluralidade de figurações de um fluxo de bits de vídeo criptados, armazenar a pluralidade de figurações em um ou mais sub-DPBs, gerar um conjunto respectivo de parâmetros

de sub-DPB para cada ponto de operação respectivo do fluxo de bits de vídeo criptados, aplicar o conjunto respectivo de parâmetros de sub-DPB a todas as camadas de um conjunto de camadas de saída para cada ponto de operação respectivo, realizar um processo de gerenciamento de sub-DPB nos um ou mais sub-DPBs de acordo com os conjuntos únicos respectivos gerados de parâmetros de sub-DPB, e sinalizar o conjunto respectivo de parâmetros de sub-DPB no fluxo de bits de vídeo criptados.

[0013] Em outro exemplo da revelação, um aparelho configurado para criptar dados de vídeo compreende meios para reconstruir uma pluralidade de figurações de um fluxo de bits de vídeo criptados, meios para armazenar a pluralidade de figurações em um ou mais sub-DPBs, meios para gerar um conjunto respectivo de parâmetros de sub-DPB para cada ponto de operação respectivo do fluxo de bits de vídeo criptados, meios para aplicar o conjunto respectivo de parâmetros de sub-DPB a todas as camadas de um conjunto de camadas de saída para cada ponto de operação respectivo, meios para realizar um processo de gerenciamento de sub-DPB nos um ou mais sub-DPBs de acordo com os conjuntos únicos respectivos gerados de parâmetros de sub-DPB, e meios para sinalizar o conjunto respectivo de parâmetros de sub-DPB no fluxo de bits de vídeo criptados.

[0014] Em outro exemplo, essa revelação descreve instruções de armazenamento de mídia de armazenamento legível por computador que, quando executadas, levam um ou mais processadores de um dispositivo configurado para criptar dados de vídeo para reconstruir uma pluralidade de figurações de um fluxo de bits de vídeo criptados armazena a pluralidade de figurações em um ou mais sub-DPBs, gerar um conjunto respectivo de parâmetros de sub-DPB para cada ponto de

operação respectivo do fluxo de bits de vídeo criptados, aplicar o conjunto respectivo de parâmetros de sub-DPB a todas as camadas de um conjunto de camadas de saída para cada ponto de operação respectivo, realizar um processo de gerenciamento de sub-DPB nos um ou mais sub-DPBs de acordo com os conjuntos únicos respectivos gerados de parâmetros de sub-DPB, e sinalizar o conjunto respectivo de parâmetros de sub-DPB no fluxo de bits de vídeo criptados.

[0015] Os detalhes de um ou mais exemplos são apresentados nos desenhos anexos e na descrição abaixo. Outros atributos, objetivos e vantagens serão evidentes a partir da descrição e dos desenhos e a partir das reivindicações.

#### BREVE DESCRIÇÃO DOS DESENHOS

[0016] A Figura 1 é um diagrama de blocos que ilustra um sistema de criptação e decodificação de vídeo exemplificativo que pode utilizar as técnicas descritas nesta revelação.

[0017] A Figura 2 é um diagrama conceitual que ilustra uma ordem de decodificação de múltiplas vistas exemplificativa.

[0018] A Figura 3 é um diagrama conceitual que ilustra uma estrutura de previsão exemplificativa para codificação de múltiplas vistas.

[0019] A Figura 4 é um diagrama conceitual que ilustra camadas de codificação de vídeo escalonáveis exemplificativas.

[0020] A Figura 5 é um diagrama conceitual que mostra um armazenamento temporário de figuração decodificada (DPB) exemplificativo de acordo com os exemplos da revelação.

[0021] A Figura 6 é um diagrama de blocos que ilustra um criptador de vídeo exemplificativo que pode



implantar as técnicas descritas nesta revelação.

[0022] A Figura 7 é um diagrama de blocos que ilustra um decodificador de vídeo exemplificativo que pode implantar as técnicas descritas nesta revelação.

[0023] A Figura 8 é um fluxograma que mostra um método de criptação exemplificativo de acordo com um exemplo das técnicas desta revelação.

[0024] A Figura 9 é um fluxograma que mostra um método de decodificação exemplificativo de acordo com um exemplo das técnicas desta revelação.

#### DESCRIÇÃO DETALHADA

[0025] Em geral, essa revelação descreve técnicas para a sinalização de parâmetros relacionados a armazenamentos temporários de figuração decodificada (DPBs) em codificação de vídeo de múltiplas camadas. Em particular, essa revelação descreve diversas técnicas para sinalizar parâmetros relacionados ao gerenciamento de subunidades de um DPB (também chamado de sub-DPBs). Conforme será descrito abaixo em maiores detalhes, as técnicas desta revelação incluem técnicas para sinalizar parâmetros de sub-DPB para uma ou mais camadas e uma ou mais subcamadas de um fluxo de bits de vídeo de múltiplas camadas. Utilizando-se as técnicas da revelação, a eficácia de sinalização pode ser aumentada para fluxos de bits de vídeo de múltiplas camadas que incluem uma ou mais subcamadas.

[0026] No contexto desta revelação, uma camada pode ser uma camada-base ou uma ou mais camadas de aprimoramento em um processo de codificação de vídeo escalonável. Os processos de codificação de vídeo escalonáveis exemplificativos incluem H.264/SVC (codificação de vídeo escalonável) e extensões escalonáveis do padrão de codificação de vídeo de alta eficácia (HEVC),

tal como HEVC Escalonável (SHVC). Além disso, uma camada pode se referir a uma ou mais vistas de textura em codificação de vídeo de múltiplas vistas ou 3D. Além disso, uma vista de profundidade em codificação de vídeo 3D pode ser considerada como uma camada. Como outro exemplo, uma camada pode corresponder a uma vista única que inclui tanto componentes de vista de textura quanto componentes de vista de profundidade. Os processos de codificação de múltiplas vistas exemplificativas incluem H.264/MVC (codificação de múltiplas vistas) e extensões de múltiplas vistas do padrão HEVC, tal como HEVC de múltiplas vistas (MV-HEVC). De modo mais geral, uma camada pode se referir a um conjunto de unidades de camada de abstração de rede (NAL) que tem a mesma identificação de camada (por exemplo, `nuh_layer_id`).

[0027] Desse modo, o termo "camada" é usado de modo geral nessa revelação para se referir a camadas de base e/ou de aprimoramento no sentido de codificação de vídeo escalonável ou vistas (incluindo componentes de vista de textura e/ou componentes de vista de profundidade) no sentido de codificação de vídeo 3D e de múltiplas vistas. Consequentemente, o termo codificação de vídeo de múltiplas camadas se refere de modo geral a técnicas de codificação de vídeo escalonáveis, técnicas de codificação de vídeo de múltiplas vistas, e técnicas de codificação de vídeo 3D, que inclui múltiplas vistas mais codificação de profundidade. As técnicas desta revelação podem se aplicar a quaisquer situações de codificação de vídeo, que incluem extensões de múltiplas vistas, extensões de vídeo 3D e extensões escalonáveis de HEVC e/ou H.264/AVC.

[0028] No contexto desta revelação, o termo "subcamada" se refere a uma camada escalonável temporal de um fluxo de bits escalonável temporal. Isto é, um fluxo de bits no qual uma ou mais subcamadas diferentes podem ser

decodificadas para emitir o vídeo em razões de quadro diferentes. Uma camada pode incluir uma ou mais subcamadas.

[0029] As técnicas descritas abaixo podem ser aplicadas a técnicas de codificação de vídeo escalonável, de múltiplas vistas e 3D com base em codecs avançados, incluem a codificação de duas ou mais vistas de uma figuração com mapas de profundidade. Os padrões de codificação de vídeo incluem ITU-T H.261, ISO/IEC MPEG-1 Visual, ITU-T H.262 ou ISO/IEC MPEG-2 Visual, ITU-T H.263, ISO/IEC MPEG-4 Visual e ITU-T H.264 (também chamado de ISO/IEC MPEG-4 AVC), incluindo suas extensões de Codificação de Vídeo Escalonável (SVC) e de Codificação de Vídeo de Múltiplas Vistas (MVC). Além disso, um novo padrão de codificação de vídeo, a saber, Codificação de Vídeo de Alta Eficácia (HEVC), foi desenvolvida pela Equipe de Colaboração Conjunta em Codificação de Vídeo (JCT-VC) do Grupo de Especialistas em Codificação de Vídeo de ITU-T (VCEG) e do Grupo de Especialistas em Imagens com Movimento ISO/IEC (MPEG). O padrão HEVC é descrito em ITU-T H.265, Série H: AUDIOVISUAL AND MULTIMEDIA SYSTEMS, Infrastructure of Audiovisual Services— Coding of Moving Video, "High Efficiency Video Coding", abril de 2013 (doravante no presente documento, "HEVC").

[0030] Várias extensões para HEVC foram propostas. Uma das tais extensões é a Faixa de Extensão HEVC, descrita em "High Efficiency Video Coding (HEVC) Range Extensions text specification: Draft 4", JCTVC-N1005\_v1, agosto de 2013 (doravante no presente documento, "JCTVC-N1005"). Um Rascunho de Trabalho (WD) recente do padrão de HEVC escalonável (SHVC), intitulado "High efficiency video coding (HEVC) extensão escalonável draft 3", desenvolvida pela Equipe de Colaboração Conjunta em Codificação de Vídeo (JCT-VC) de ITU-T SG 16 WP 3 e ISO/IEC

JTC 1/SC 29/WG 11, 14º Reunião: Viena, AT, 25 de julho a 2 de agosto de 2013 e chamado de SHVC WD3, doravante no presente documento, é disponibilizado por [http://phenix.it-sudparis.eu/jct/doc\\_end\\_user/documents/14\\_Vienna/wg11/JCTVC-N1008-v3.zip](http://phenix.it-sudparis.eu/jct/doc_end_user/documents/14_Vienna/wg11/JCTVC-N1008-v3.zip). Uma extensão de múltiplas vistas para HEVC, a saber MV-HEVC, também é desenvolvida pela JCT-3 V. Um Rascunho de Trabalho de MV-HEVC (doravante no presente documento WD4), é disponibilizado por [http://phenix.it-sudparis.eu/jct2/doc\\_end\\_user/documents/4\\_Incheon/wg11/JCT3V-D1004-v3.zip](http://phenix.it-sudparis.eu/jct2/doc_end_user/documents/4_Incheon/wg11/JCT3V-D1004-v3.zip). Um rascunho de trabalho recente de MV-HEVC, intitulado "MV-HEVC Draft Text 8," Equipe de Colaboração Conjunta em Extensões de Codificação de Vídeo 3D de ITU-T SG 16 WP3 e ISO/IEC JTC 1/SC 29/WG11, 8º Reunião: Valência, ES, 29 de março - 4 de abril de 2014, está disponível em [http://phenix.it-sudparis.eu/jct3v/doc\\_end\\_user/documents/8\\_Valencia/wg11/JCT3V-H0002-V1.zip](http://phenix.it-sudparis.eu/jct3v/doc_end_user/documents/8_Valencia/wg11/JCT3V-H0002-V1.zip).

[0031] A Figura 1 é um diagrama de blocos que ilustra um sistema de criptação e decodificação de vídeo exemplificativo 10 que pode utilizar as técnicas para o gerenciamento e sinalização de armazenamento temporário de figuração decodificada (DPB) em um processo de codificação de vídeo de múltiplas camadas descrito nesta revelação. Conforme mostrado na Figura 1, o sistema 10 inclui um dispositivo de origem 12 que gera dados de vídeo criptados a serem decodificados posteriormente por um dispositivo de destino 14. O dispositivo de origem 12 e o dispositivo de destino 14 podem compreender qualquer um dentre uma ampla faixa de dispositivos, incluindo computadores de mesa, computadores do tipo notebook (isto é, do tipo laptop), computadores do tipo tablet, decodificadores de sinais, pares de telefone, tais como, os então chamados telefones "inteligentes", então chamados pads

"inteligentes", televisões, câmeras, dispositivos de exibição, reprodutores de mídias digital, consoles de jogos eletrônicos, dispositivos de transmissão contínua de vídeo ou semelhantes. Em alguns casos, o dispositivo de origem 12 e o dispositivo de destino 14 podem ser equipados para comunicação sem fio.

[0032] O dispositivo de destino 14 pode receber os dados de vídeo criptados a serem decodificados através de um enlace 16. O enlace 16 pode compreender qualquer tipo de mídia ou de dispositivo com capacidade para mover os dados de vídeo criptados do dispositivo de origem 12 ao dispositivo de destino 14. Em um exemplo, o enlace 16 pode compreender uma mídia de comunicação para possibilitar que um dispositivo de origem 12 transmite os dados de vídeo criptados diretamente a um dispositivo de destino 14 em tempo real. Os dados de vídeo criptados podem ser modulados de acordo com um padrão de comunicação, como um protocolo de comunicação sem fio e transmitidos para o dispositivo de destino 14. O meio de comunicação pode compreender qualquer mídia de comunicação sem fio ou com fio, como um espectro de radiofrequência (RF) ou uma ou mais linhas de transmissão físicas. A mídia de comunicação pode formar parte de uma rede com base em pacote, tal como, uma rede de área local, uma rede de longa distância ou uma rede global, tal como, a Internet. A mídia de comunicação pode incluir roteadores, comutadores, estações-base, ou qualquer outro equipamento que pode ser útil que para facilitar a comunicação do dispositivo de origem 12 com o dispositivo de destino 14.

[0033] Alternativamente, os dados criptados podem ser emitidos a partir da interface de saída 22 para um dispositivo de armazenamento 34. De modo semelhante, os dados criptados podem ser acessados a partir do dispositivo

de armazenamento 34 pela interface de entrada. O dispositivo de armazenamento 34 podem incluir qualquer um dentre uma variedade de mídias de armazenamento de dados distribuídos ou acessados localmente como um disco rígido, discos Blu-ray, DVDs, CD-ROMs, memória flash, memória volátil ou não volátil ou quaisquer outras mídias de armazenamento digital adequado para armazenar dados de vídeo criptados. Em um exemplo adicional, o dispositivo de armazenamento 34 pode corresponder a um servidor de arquivos ou outro dispositivo de armazenamento intermediário que pode manter o vídeo criptado gerado por dispositivo de origem 12. O dispositivo de destino 14 pode acessar dados de vídeo armazenados a partir do dispositivo de armazenamento 34 através de transmissão contínua ou transferência por download. O servidor de arquivos pode ser qualquer tipo de servidor com capacidade para armazenar dados de vídeo criptados e transmitir esses dados de vídeo criptados para o dispositivo de destino 14. Os servidores de arquivo exemplificativos incluem um servidor web (por exemplo, para um site da web), um servidor FTP, dispositivos de armazenamento anexado à rede (NAS) ou uma unidade de disco local. O dispositivo de destino 14 pode acessar os dados de vídeo criptados através de qualquer conexão de dados padrão, incluindo uma conexão de Internet. Isso pode incluir um canal sem fio (por exemplo, uma conexão Wi-Fi), uma conexão cabeada (por exemplo, DSL, modem a cabo, etc.), ou uma combinação dos dois que é adequada para cessar os dados de vídeo criptados armazenados em um servidor de arquivo. A transmissão de dados de vídeo criptados do dispositivo de armazenamento 34 pode ser uma transmissão contínua, uma transmissão por transferência por download ou uma combinação das mesmas.

[0034] As técnicas desta revelação para

gerenciamento e sinalização de DPB em um processo de codificação de vídeo de múltiplas camadas, tais como SHVC, MV-HEVC, ou 3D-HEVC como exemplos, não são limitadas necessariamente a aplicações ou configurações sem fio. As técnicas podem ser aplicadas a codificação de vídeo em suporte a qualquer uma dentre uma variedade de aplicativos multimídia, tais como difusões de televisão pelo ar, transmissões de televisão por cabo, transmissões de televisão por satélite, transmissões de vídeo contínuas, por exemplo, através da Internet, criptação de vídeo digital para o armazenamento em uma mídia de armazenamento de dados, decodificação de vídeo digital armazenado em uma mídia de armazenamento de dados ou outras aplicações. Em alguns exemplos, o sistema 10 pode ser configurado para suportar transmissão de vídeo unidirecional ou bidirecional para suportar aplicações como transmissão contínua de vídeo, reprodução de vídeo, difusão de vídeo e/ou telefonia por vídeo.

[0035] No exemplo da Figura 1, o dispositivo de origem 12 inclui uma fonte de vídeo 18, o criptador de vídeo 20 e uma interface de saída 22. Em alguns casos, a interface de saída 22 pode incluir um modulador/demodulador (modem) e/ou um transmissor. No dispositivo de origem 12, a fonte de vídeo 18 pode incluir uma fonte tal como um dispositivo de captura de vídeo, por exemplo, uma câmera de vídeo, um arquivo de vídeo que contém vídeo capturado anteriormente, uma interface de alimentação de vídeo para receber vídeo a partir de um provedor de conteúdo de vídeo, e/ou um sistema de gráficos de computador para gerar dados de gráficos de computador como vídeo de fonte, ou uma combinação de tais fontes. Como um exemplo, se a fonte de vídeo 18 for uma câmera de vídeo, o dispositivo de origem 12 e o dispositivo de destino 14 podem formar os então

chamados telefones de câmera ou videofones. No entanto, as técnicas descritas nesta revelação podem ser aplicáveis à codificação de vídeo em geral e podem ser aplicadas às aplicações com fio e/ou sem fio.

[0036] O vídeo capturado, pré-capturado ou gerado por computador pode ser criptado pelo criptador de vídeo 20. Os dados de vídeo criptados podem ser transmitidos diretamente ao dispositivo de destino 14 através da interface de saída 22 do dispositivo fonte 12. Os dados de vídeo criptados também podem (ou alternativamente) ser armazenados no dispositivo de armazenamento 34 para o acesso posterior pelo dispositivo de destino 14 ou outros dispositivos, para decodificação e/ou reprodução.

[0037] O dispositivo de destino 14 inclui uma interface de entrada 28, um decodificador de vídeo 30 e um dispositivo de exibição 32. Em alguns casos, a interface de entrada 28 pode incluir um receptor e/ou um modem. A interface de entrada 28 do dispositivo de destino 14 recebe os dados de vídeo criptados através do enlace 16. Os dados de vídeo criptados comunicados através do enlace 16, ou fornecidos no dispositivo de armazenamento 34, podem incluir uma variedade de elementos de sintaxe gerados pelo criptador de vídeo 20 para o uso por um decodificador de vídeo, tal como decodificador de vídeo 30, na decodificação dos dados de vídeo. Tais elementos de sintaxe podem ser incluídos com dados de vídeo criptados transmitidos em uma mídia de comunicação, armazenados em uma mídia de armazenamento, ou armazenados em um servidor de arquivo.

[0038] O dispositivo de exibição 32 pode ser integrado com, ou externo ao dispositivo de destino 14. Em alguns exemplos, o dispositivo de destino 14 pode incluir um dispositivo de exibição integrado e também pode ser



configurado para realizar interface com um dispositivo de exibição externo. Em outros exemplos, o dispositivo de destino 14 pode ser um dispositivo de exibição. Em geral, 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 visores como um visor de cristal líquido (LCD), um visor de plasma, um visor de diodo emissor de luz orgânico (OLED) ou outro tipo de dispositivo de exibição.

[0039] O criptador de vídeo 20 e o decodificador de vídeo 30 podem operar de acordo com um padrão de codificação de vídeo, como a padrão de Codificação de Vídeo de Alta Eficiência (HEVC), e pode se conformar ao Modelo de Teste HEVC (HM). Em particular, em alguns exemplos, o criptador de vídeo 20 e o decodificador de vídeo podem operar de acordo com as extensões de HEVC que suportam codificação de vídeo escalonável, de múltiplas vistas e/ou de múltiplas vistas mais profundidade. Em outros exemplos, o criptador de vídeo 20 e o decodificador de vídeo 30 podem operar de acordo com outros padrões de proprietário ou de indústria, como o padrão ITU-T H.264, referido de modo alternativo como MPEG-4, Parte 10, Codificação de Vídeo Avançada (AVC), ou extensões de tais padrões, incluem H.264/SVC. As técnicas desta revelação, entretanto, não se limitam a qualquer padrão de codificação particular. Outros exemplos padrões de compressão de vídeo incluem MPEG-2 e ITU-T H.263. Em particular, de acordo com técnicas desta revelação, o criptador de vídeo 20 e o decodificador de vídeo 30 podem operar de acordo com um padrão de codificação de vídeo com a capacidade para criptar de modo escalonável (por exemplo, SHVC) e/ou de múltiplas vistas (por exemplo, MV-HEVC ou 3D-HEVC).

[0040] Embora não seja mostrado na Figura 1,

em alguns aspectos, o criptador de vídeo 20 e o decodificador de vídeo 30 podem ser, cada um, integrados a um criptador e decodificador de áudio, e podem incluir unidades MUX-DEMUX apropriadas, ou outro hardware e software, para manipular a criptação tanto de áudio quanto de vídeo em um fluxo de dados comum ou fluxos de dados separadas. Caso aplicável, em alguns exemplos, as unidades MUX-DEMUX podem se conformar ao protocolo multiplexador ITU H.223, ou outros protocolos como protocolo de datagrama de usuário (UDP).

[0041] O criptador de vídeo 20 e o decodificador de vídeo 30, cada um, podem ser implantados como qualquer um dentre uma variedade de conjunto de circuitos de criptador adequado, como um ou mais microprocessadores, processadores de sinal digital (DSPs), circuitos integrados de aplicação específica (ASICs), matrizes de porta programáveis por campo (FPGAs), lógica discreta, software, hardware, firmware ou quaisquer combinações dos mesmos. Quando as técnicas são implantadas parcialmente em software, um dispositivo pode armazenar instruções para o software em uma mídia legível por computador não transitório adequada e executar as instruções em hardware com o uso de um ou mais processadores para realizar as técnicas desta revelação. Cada um dentre o criptador de vídeo 20 e o decodificador de vídeo 30 podem estar incluídos em um ou mais criptadores ou decodificadores, um dos quais pode ser integrado como parte de um criptador/decodificador (CODEC) combinado em um dispositivo respectivo.

[0042] A seguinte seção da revelação fornecerá um conhecimento do padrão HEVC. As tentativas de padronização HEVC se baseiam em um modelo de evolução de um dispositivo de codificação de vídeo denominado de Modelo de

Teste HEVC (HM). O HM considera diversas capacidades adicionais para dispositivos de codificação de vídeo em relação a dispositivos existentes de acordo com, por exemplo, ITU-T H.264/AVC. Por exemplo, enquanto o H.264 fornece nove modos de criptação de intraprevisão, o HM pode fornecer até trinta e três modos de criptação de intraprevisão.

[0043] Em geral, o modelo em funcionamento do HM descreve que um quadro de vídeo ou figuração pode ser dividido em uma sequência de blocos em árvore (também chamados de blocos de árvore de codificação (CTBs)) ou as maiores unidades de codificação (LCUs) que incluem tanto amostras luma quanto croma. Um bloco de árvore tem um propósito similar ao de um macro bloco do padrão H.264. Uma fatia inclui vários blocos em árvore consecutivos em ordem de codificação. Um quadro de vídeo ou figuração pode ser particionado em uma ou mais fatias. Cada bloco de árvore pode ser partido em unidades de codificação (CUs) de acordo com uma árvore quadrática. Por exemplo, um bloco de árvore, como um nó raiz da árvore quadrática, pode ser dividida em quatro nós filhos e cada nó filho pode, por sua vez, ser um nó pai e ser dividido em outros quatro nós filhos. Um nó filho não dividido final, como um nó de folha da árvore quadrática, compreende um nó de codificação, isto é, um bloco de vídeo codificado. Os dados de sintaxe associados a um fluxo de dados codificados podem definir um número máximo de vezes que um bloco de árvore pode ser dividido e também podem definir um tamanho mínimo dos nós de codificação.

[0044] Uma CU inclui um nó de codificação e unidades de previsão (PUs) E unidades de transformada (TUs) associadas ao nó de codificação. Um tamanho da CU corresponde de modo geral a um tamanho do nó de codificação

e deve ter, tipicamente, formato quadrado. O tamanho da CU pode variar de 8x8 pixels até o tamanho do bloco de árvore com um máximo de 64x64 pixels ou maior. Cada CU pode conter uma ou mais PUs e uma ou mais TUs. Os dados de sintaxe associados a 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 é criptada em salto ou de modo direto, criptada de modo de intraprevisão ou criptada de modo de interprevisão. As PUs podem ser particionadas para serem não quadradas em formato. Os dados de sintaxe associados a uma CU também podem descrever, por exemplo, a partição da CU em uma ou mais TUs de acordo com uma árvore quadrática. Uma TU pode ter formato quadrado ou não quadrado.

[0045] O padrão HEVC prevê transformações de acordo com TUs, que podem ser diferentes para diferentes CUs. As TUs são dimensionadas tipicamente com base no tamanho de PUs dentro de uma determinada CU definida por uma LCU particionada, embora esse possa nem sempre ser o caso. As TUs têm tipicamente o mesmo tamanho ou são menores que as PUs. Em alguns exemplos, as amostras residuais correspondentes a uma CU podem ser subdivididas em unidades menores com o uso de uma estrutura de árvore quadrática conhecida como "árvore quadrática residual" (RQT). Os nós de folha da RQT podem ser chamados como unidades de transformada (TUs). Os valores de diferença de pixel associados às TUs podem ser transformados para produzir o coeficiente de transformada, que pode ser quantizado.

[0046] Em geral, uma PU inclui dados relacionados ao processo de previsão. Por exemplo, quando a PU é criptada em intermodos, a PU pode incluir dados que descrevem um modo de intraprevisão para a PU. Como outro exemplo, quando a PU for criptada em intermodo, a PU pode

incluir dados que definem um vetor de movimento para a PU. 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 figuração de referência para a qual o vetor de movimento aponta e/ou uma lista de figurações de referência (por exemplo, Lista 0, Lista 1 ou Lista C) para o vetor de movimento, que pode ser indicado por uma direção de previsão.

[0047] Em geral, uma TU é usada para os processos de transformada e de quantização. Uma determinada CU que tem uma ou mais PUs também podem incluir uma ou mais unidades de transformada (TUs). Após a previsão, o criptador de vídeo 20 pode calcular os valores residuais a partir do bloco de vídeo identificado pelo nó de codificação de acordo com a PU. O nó de codificação é, então, atualizado referente aos valores residuais em vez do bloco de vídeo original. Os valores residuais compreendem valores de diferença de pixel que podem ser transformados em coeficientes de transformada, quantizados e digitalizados com o uso das informações de transformadas e de outras transformadas especificados nas TUs para produzir coeficientes de transformada em série para codificação de entropia. O nó de codificação pode ser novamente substituído para se referir a esses coeficientes de transformada em série. Esta revelação usa tipicamente o termo "bloco de vídeo" para se referir a um nó de codificação de uma CU. Em alguns casos específicos, esta revelação também pode usar o termo "bloco de vídeo" para se referir a um bloco de árvore, isto é, CTB, LCU ou uma CU, que inclui um nó de codificação e PUs e TUs.

[0048] Uma sequência de vídeo inclui tipicamente uma série de quadros ou figurações de vídeo. Um grupo de figurações (GOP) compreende geralmente uma série de uma ou mais das figurações de vídeo. Um GOP pode incluir dados de sintaxe em um cabeçalho do GOP, um cabeçalho de uma ou mais das figurações ou, em outro lugar, que descreve um certo número de figurações incluídas no GOP. Cada fatia de uma figuração pode incluir uma fatia de dados de sintaxe que descrevem um modo de criptação para a respectiva fatia. O criptador de vídeo 20 opera tipicamente em blocos de vídeo dentro de fatias de vídeo individuais a fim de criptar 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 variados e podem diferir em tamanho de acordo com um padrão de codificação especificado.

[0049] Como um exemplo, o HM sustenta uma previsão em vários tamanhos de PU. Presumindo-se que o tamanho de uma CU particular seja  $2N \times 2N$ , o HM sustenta intraprevisão em tamanhos de PU de  $2N \times 2N$  ou  $xN$  e interprevisão em tamanhos simétricos de PU de  $2N \times 2N$ ,  $2N \times N$ ,  $N \times 2N$  ou  $N \times N$ . O HM também sustenta a partição assimétrico para interprevisão em tamanhos de PU de  $2N \times nU$ ,  $2N \times nD$ ,  $nL \times 2N$  e  $nR \times 2N$ . Na partição assimétrico, uma direção de uma CU não é particionada, enquanto a outra direção é particionada em 25% e 75%. a porção da CU correspondente aos 25% de partição é indicada por um "n" seguida por uma indicação de "Cima", "Baixo", "Esquerda" ou "Direita". Dessa forma, por exemplo, " $2N \times nU$ " se refere a uma CU de  $2N \times 2N$  que é particionado horizontalmente com uma PU de  $2N \times 0.5N$  no topo e uma PU de  $2N \times 1.5N$  no fundo.

[0050] Nessa revelação, " $N \times N$ " e " $N$  por  $N$ " podem ser usados de modo intercambiável para se referir às

dimensões de pixel de um bloco de vídeo em termos de dimensões horizontal e vertical, por exemplo, 16x16 pixels ou 16 por 16 pixels. Em geral, um bloco de 16x16 terá 16 pixels em uma direção vertical ( $y = 16$ ) e 16 pixels em uma direção horizontal ( $x = 16$ ). De modo semelhante, um bloco NxN geralmente tem N pixels em uma direção vertical e N pixels em uma direção horizontal, em que N representa um valor de número inteiro não negativo. Os pixels em um bloco podem ser representados em fileiras e colunas. Além disso, os blocos não necessariamente têm a mesma quantidade de pixels tanto na direção horizontal quanto na direção vertical. Por exemplo, os blocos podem compreender Pixels de NxM, em que M não é necessariamente igual a N.

[0051] Após codificação de intrapredição e interpredição com o uso das PUs de uma CU, o criptador de vídeo 20 pode calcular dados residuais os quais as transformadas especificadas pelas TUs da CU são aplicadas. Os dados residuais podem corresponder a diferenças de pixel entre pixels da figuração não criptada e valores de previsão correspondentes às CUs. O criptador de vídeo 20 pode formar as TUs que incluem os dados residuais para a CU, e, então, transformar os dados residuais para produzir o coeficiente de transformada.

[0052] Após quaisquer transformações para produzir os coeficientes de transformada, o criptador de vídeo 20 pode realizar a quantização do coeficiente de transformada. A quantização, em geral, se refere a um processo no qual os coeficientes de transformada são quantizados para reduzir possivelmente a quantidade de dados usados para representar os coeficientes, fornecendo compressão adicional. O processo de quantização pode reduzir a profundidade de bit associada a alguns ou todos os coeficientes. Por exemplo, um valor n-bit pode ser

arredondado para baixo para um valor  $m$ -bit durante a quantização, em que  $n$  é maior que  $m$ .

[0053] Em alguns exemplos, o criptador de vídeo 20 pode utilizar uma ordem de varredura predefinida para varrer os coeficientes de transformada quantizados para produzir um vetor em série que pode ser criptado por entropia. Em outros exemplos, o criptador de vídeo 20 pode realizar uma varredura adaptativa. Após digitalizar os coeficientes de transformada quantizados para formar um vetor unidimensional, criptador de vídeo 20 pode criptar por entropia o vetor unidimensional, por exemplo, de acordo com a codificação aritmética binária adaptável por contexto (CABAC) ou outra metodologia de criptação por entropia. Para realizar a CABAC, o criptador de vídeo 20 pode atribuir um contexto dentro de um modelo de contexto a um símbolo a ser transmitido. O contexto pode se relacionar a, por exemplo, se valores vizinhos do símbolo são diferentes de zero ou não. O criptador de vídeo 20 também pode criptar por entropia elementos de sintaxe associados aos dados de vídeo criptados para uso pelo decodificador de vídeo 30 na decodificação dos dados de vídeo.

[0054] As extensões de HEVC também estão em desenvolvimento atualmente em JCT-3 V e JCT-VC. Em JCT-3V, duas extensões de HEVC relacionadas a múltiplas vistas, a extensão de múltiplas vistas (MV-HEVC) e extensão de vídeo 3D (3D-HEVC) são desenvolvidas. Além disso, duas extensões de AVC, a MVC+D e a 3D-AVC são desenvolvidas.

[0055] Versões exemplificativas dos padrões vigentes são listados conforme o seguinte:

[0056] - T. Suzuki, M. M. Hannuksela, Y. Chen, S. Hattori, G. Sullivan, "MVC

[0057] Extension for Inclusion of Depth Maps Draft Text 6," JCT3V-C1001, Equipe de Colaboração Conjunta



em Desenvolvimento de Extensão de Codificação de Vídeo 3D de ITU-T SG 16 WP 3 e ISO/IEC JTC 1/SC 29/WG 11, 4º reunião, disponível em [http://phenix.int-evry.fr/jct2/doc\\_end\\_user/documents/3\\_Geneva/wg11/JCT3V-C1001-v3.zip](http://phenix.int-evry.fr/jct2/doc_end_user/documents/3_Geneva/wg11/JCT3V-C1001-v3.zip).

[0058] - M. M. Hannuksela, Y. Chen, T. Suzuki, J.-R. Ohm, G. Sullivan, "3D-AVC Draft Text 8," JCT3V-F1002, Equipe de Colaboração Conjunta em Desenvolvimento de

[0059] Extensão de Codificação de Vídeo 3D de ITU-T SG 16 WP 3 e ISO/IEC JTC 1/SC 29/WG 11, 6º Reunião, disponível em [http://phenix.int-evry.fr/jct2/doc\\_end\\_user/documents/6\\_Geneva/wg11/JCT3 V-F 1002-v3.zip](http://phenix.int-evry.fr/jct2/doc_end_user/documents/6_Geneva/wg11/JCT3 V-F 1002-v3.zip).

[0060] - JCT3 V-F 1004, "MV-HEVC Draft Text 6," G. Tech, K. Wegner, Y. Chen, M.

[0061] Hannuksela, Equipe de Colaboração Conjunta em Desenvolvimento de Extensão de Codificação de Vídeo 3D

[0062] de U-T SG 16 WP 3 e ISO/IEC JTC 1/SC 29/WG 11, 6º Reunião, disponível em [http://phenix.int-evry.fr/jct2/doc\\_end\\_user/documents/6\\_Geneva/wg11/JCT3V-F1004-v6.zip](http://phenix.int-evry.fr/jct2/doc_end_user/documents/6_Geneva/wg11/JCT3V-F1004-v6.zip). Gerhard Tech, Krzysztof Wegner, Ying Chen, Sehoon Yea, "3D-HEVC Draft Text 2," JCT3V-F1001, Equipe de Colaboração Conjunta em Desenvolvimento de

[0063] Extensão de Codificação de Vídeo 3D de ITU-T SG 16 WP 3 e ISO/IEC JTC 1/SC 29/WG 11, 6º Reunião, disponível em [http://phenix.int-evry.fr/jct2/doc\\_end\\_user/documents/6\\_Geneva/wg11/JCT3 V-F 1001 -v2](http://phenix.int-evry.fr/jct2/doc_end_user/documents/6_Geneva/wg11/JCT3 V-F 1001 -v2).

[0064] As técnicas de codificação de vídeo de múltiplas vistas de extensão do padrão de H.264/ Codificação de Vídeo Avançada (AVC) serão agora discutidas. No entanto, as técnicas desta revelação podem ser

aplicáveis com qualquer padrão de codificação de vídeo que suporta a codificação de múltiplas vistas e/ou propostas de múltiplas vistas de codificação 3D para o padrão HEVC emergente (por exemplo, MV-HEVC e 3D-HEVC).

[0065] A codificação de vídeo de múltiplas vistas (MVC) é uma extensão de H.264/AVC. Uma ordem de decodificação de MVC típica (isto é, uma ordem de fluxo de bits) é mostrada na Figura 2. A disposição de ordem de decodificação é chamada de codificação de primeiro no tempo. Observa-se que a ordem de decodificação de unidades de acesso pode não ser idêntica à ordem de saída ou de exibição das unidades de acesso. Na Figura 2, S0 a S7 se referem, cada uma, a diferentes vistas do vídeo de múltiplas vistas. T1 a T8 representam, cada um, um instante do tempo de saída. Uma unidade de acesso pode incluir as figurações codificadas de todas as vistas para um instante do tempo de saída. Por exemplo, uma primeira unidade de acesso pode incluir todas as vistas S0 a S7 para o instante do tempo T1, uma segunda unidade de acesso pode incluir todas as vistas S0 a S7 para o instante do tempo T2, e assim por diante.

[0066] Para propósitos de brevidade, a revelação pode usar as seguintes definições:

[0067] componente de vista: Uma *representação codificada* de uma *vista* em uma *única unidade de acesso*.

[0068] Quando uma vista inclui tanto uma representação de textura codificada quanto uma representação de profundidade, um componente de vista pode incluir um componente de vista de textura e um componente de vista de profundidade.

[0069] componente de vista de textura: Uma *representação codificada* da textura de uma vista em uma *única unidade de acesso*.

[0070] componente de vista de profundidade: Uma *representação codificada* da profundidade de uma vista em uma *única unidade de acesso*.

[0071] Conforme discutido acima, no contexto desta revelação, cada um dentre um componente de vista, componente de vista de textura e um componente de vista de profundidade pode ser chamado de modo geral de uma camada. Na Figura 2, cada uma dentre as vistas inclui conjuntos de figurações. Por exemplo, a vista S0 inclui um conjunto de figurações 0, 8, 16, 24, 32, 40, 48, 56 e 64, a vista S1 inclui um conjunto de figurações 1, 9, 17, 25, 33, 41, 49, 57 e 65, e assim por diante. Cada conjunto inclui dois tipos de figurações: uma figuração é chamada de componente de vista de textura, e a outra figuração é chamada de componente de vista de profundidade. O componente de vista de textura e o componente de vista de profundidade dentro de um conjunto de figurações de uma vista pode ser considerado como correspondente um ou ao outro. Por exemplo, o componente de vista de textura dentro de um conjunto de figurações de uma vista é considerado como correspondente ao componente de vista de profundidade dentro do conjunto das figurações da vista, e vice-versa (isto é, o componente de vista de profundidade correspondente ao seu componente de vista de textura no conjunto, e vice-versa). Conforme usado nesta revelação, um componente de vista de textura que corresponde a um componente de vista de profundidade pode ser considerado como parte de uma mesma vista que o componente de profundidade em uma unidade de acesso única.

[0072] O componente de vista de textura inclui o conteúdo de imagem real que é exibido. Por exemplo, o componente de vista de textura pode incluir componentes de luma (Y) e de croma (Cb e Cr). O componente de vista de

profundidade pode indicar profundidades relativas dos pixels no componente de vista de textura correspondente do componente de vista de profundidade. Como um exemplo, o componente de vista de profundidade é uma imagem em escala de cinza que inclui apenas valores de luma. Em outras palavras, o componente de vista de profundidade pode não transportar qualquer conteúdo de imagem, porém, em vez disso, pode fornecer uma medida das profundidades relativas dos pixels no componente de vista de textura correspondente.

[0073] Por exemplo, um pixel inteiramente branco no componente de vista de profundidade indica que o pixel ou pixels correspondentes do mesmo no componente de vista de textura correspondentes está mais próxima da perspectiva do espectador, e um pixel inteiramente preto no componente de vista de profundidade indica que o pixel ou pixels correspondentes do mesmo no componente de vista de textura correspondentes está mais distante da perspectiva do espectador. As diversas tonalidades de cinza entre preto e branco indicam níveis de profundidade diferentes, de modo que um aumento na escuridão da tonalidade de cinza de um pixel na vista de profundidade é indicativo de um aumento no nível de profundidade associada ao pixel correspondente na vista de textura. Por exemplo, um pixel muito cinza no componente de vista de profundidade indica que pixel correspondentes do mesmo no componente de vista de textura está mais distante que um pixel levemente cinza no componente de vista de profundidade. Devido ao fato de que apenas a escala de cinza é necessária para identificar a profundidade de pixels, o componente de vista de profundidade não precisa incluir componentes de croma, uma vez que os valores de cor para o componente de vista de profundidade podem não ter propósito algum.

[0074] O componente de vista de profundidade que usa apenas valores de luma (por exemplo, valores de intensidade) para identificar a profundidade é fornecido para propósitos de ilustração e não deve ser considerado como limitativo. Em outros exemplos, qualquer técnica pode ser utilizada para indicar profundidades relativas dos pixels no componente de vista de textura.

[0075] Uma estrutura de previsão de MVC típica (incluindo tanto a previsão de interfiguração dentro de cada vista quanto a previsão entre vistas) para a codificação de vídeo de múltiplas vistas é mostrada na Figura 3. A estrutura de previsão da Figura 3 também pode ser usada com MV-HEVC. Direções de previsão indicadas por setas, em que o objeto apontado usa o objeto de apontamento como a referência de previsão. Em MVC, previsão entre vistas é suportada pela compensação de movimento de disparidade, com o uso da sintaxe da compensação de movimento H.264/AVC, porém, permite que uma figuração em uma vista diferente seja usada como uma figuração de referência.

[0076] No exemplo da Figura 3, as oito vistas (que têm IDs de vista "S0" a "S7") são ilustradas, e as doze localizações temporais ("T0" a "T11") são ilustradas para cada vista. Ou seja, cada fileira na Figura 3 corresponde a uma vista, ao passo que cada coluna indica uma localização temporal.

[0077] Embora a MVC tenha uma chamada vista de base, que é decodificável por decodificadores H.264/AVC, e pares de vistas estéreos possam ser suportados também pela MVC, a vantagem da MVC é que a mesma pode suportar um exemplo que usa mais que duas vistas como uma entrada de vídeo 3D e decodifica esse vídeo 3D representado pelas múltiplas vistas. Um renderizador de um cliente que tem um

decodificador de MVC pode esperar o conteúdo de vídeo 3D com múltiplas vistas.

[0078] As figurações na Figura 3 são indicadas na interseção de cada fileira e cada coluna. O padrão H.264/AVC pode usar o termo quadro para representar uma porção do vídeo. A presente revelação pode usar o tempo figuração e quadro de maneira intercambiável.

[0079] As figurações na Figura 3 são ilustradas com o uso de um bloco incluindo uma letra, sendo que a letra indica se a figuração correspondente é intracodificada (ou seja, uma I-figuração), ou uma intercodificada em uma direção (ou seja, como uma P-figuração) ou em múltiplas direções (ou seja, como uma B-figuração). Em geral, as previsões são indicadas por setas, em que as figurações apontadas usam as figurações apontadas para referência de previsão. Por exemplo, a P-figuração da vista S2 em uma localização temporal T0 é prevista a partir da I-figuração da vista S0 em uma localização temporal T0.

[0080] Conforme a criptação de vídeo de única vista, as figurações de uma sequência de vídeos de codificação de vídeo de múltiplas vistas pode ser criptada de maneira prevista em relação às figurações em diferentes localizações temporais. Por exemplo, a b-figuração da vista S0 na localização temporal T1 tem uma seta apontada para a mesma a partir da I-figuração da vista S0 na localização temporal T0, o que indica que a b-figuração é prevista a partir da I-figuração. No entanto, adicionalmente, no contexto de criptação de vídeo de múltiplas vistas, as figurações podem ser previstas entre vistas. Ou seja, um componente de vista pode usar os componentes de vista em outras vistas para referência. Em MVC, por exemplo, a previsão entre vistas é notada como se o componente de vista em outra vista fosse uma referência de interprevisão.

As referências de entre vistas potenciais são sinalizadas na extensão de MVC de Conjunto de Parâmetros de Sequência (SPS) e podem ser modificadas pelo processo de construção de lista de figurações de referência, o que possibilita o ordenamento flexível das referências de interprevisão ou de previsão entre vistas. A previsão entre vistas é também um atributo de uma extensão com múltiplas vistas proposta de HEVC, incluindo 3D-HEVC (múltiplas vistas mais profundidade).

[0081] A Figura 3 fornece vários exemplos de previsão entre vistas. As figurações da vista S1, no exemplo da Figura 3, são ilustradas como previstas a partir das figurações em diferentes localizações temporais da vista S1, assim como, previstas entre vistas a partir das figurações das vistas S0 e S2 nas mesmas localizações temporais. Por exemplo, a b-figuração da vista S1 na localização temporal T1 é prevista a partir de cada uma dentre as figurações b da vista S1 nas localizações temporais T0 e T2, assim como, as figurações b das vistas S0 e S2 na localização temporal T1.

[0082] Em alguns exemplos, a Figura 3 pode ser vista como ilustrando os componentes de vista de textura. Por exemplo, as figurações I, P, B e b ilustradas na Figura 3 podem ser consideradas como componentes de vista de textura para cada uma das vistas. Em conformidade com as técnicas descritas na presente revelação, para cada um dos componentes de vista de textura ilustrados na Figura 3 há um componente de vista de profundidade correspondente. Em alguns exemplos, os componentes de vista de profundidade podem ser previstos de maneira semelhante àquela ilustradas na Figura 3 para os componentes de vista de textura correspondentes.

[0083] A codificação de duas vistas também

pode ser suportada pela MVC. Uma das vantagens de MVC é que um criptador de MVC pode obter mais que duas vistas como uma entrada de vídeo 3D e um decodificador de MVC pode decodificar tal representação de múltiplas vistas. Desse modo, qualquer renderizador com um decodificador de MVC pode decodificar conteúdo de vídeo 3D com mais que duas vistas.

[0084] Conforme discutido acima, em MVC (ou MV-HEVC), previsão entre vistas é permitida dentre as figurações na mesma unidade de acesso (que significa, em alguns exemplos, com o mesmo instante do tempo). Durante a codificação de uma figuração em uma dentre as vistas de não base, uma figuração pode ser adicionada a uma lista de figurações de referência caso esteja em uma vista diferente, porém, dentro de um mesmo instante do tempo. Uma figuração de referência entre vistas de previsão pode ser colocada em qualquer posição de uma lista de figurações de referência, exatamente como qualquer figuração de referência de interprevisão. Conforme mostrado na Figura 3, um componente de vista pode usar os componentes de vista em outras vistas para referência. Em MVC, a previsão entre vistas é notada como se o componente de vista em outra vista fosse uma referência de interprevisão.

[0085] Conforme mostrado na Figura 3, um componente de vista pode usar os componentes de vista em outras vistas para referência. Isso é chamado de previsão entre vistas. Em MVC, a previsão entre vistas é notada como se o componente de vista em outra vista fosse uma referência de interprevisão. As referências de entre vistas potenciais, no entanto, são sinalizadas na extensão de MVC de Conjunto de Parâmetros de Sequência (SPS) (conforme mostrado na Tabela 1 abaixo) e podem ser modificadas pelo processo de construção de lista de figurações de



referência, o que possibilita o ordenamento flexível das referências de interprevisão ou de previsão entre vistas.

TABELA 1 - SPS MVC TABELA DE SINTAXE DE EXTENSÃO  
DE MVC DE SPS

seq_parameter_set_mvc_extension() {		<b>Descriptor</b>
<b>num_views_minus1</b>		ue(v)
for( i = 0; i <= num_views_minus1; i++ )		
<b>view_id[ i ]</b>		ue(v)
for( i = 1; i <= num_views_minus1; i++ ) {		
<b>num_anchor_refs_l0[ i ]</b>		ue(v)
for( j = 0; j < num_anchor_refs_l0[ i ]; j++ )		
<b>anchor_ref_l0[ i ][ j ]</b>		ue(v)
<b>num_anchor_refs_l1[ i ]</b>		ue(v)
for( j = 0; j < num_anchor_refs_l1[ i ]; j++ )		
<b>anchor_ref_l1[ i ][ j ]</b>		ue(v)
}		
for( i = 1; i <= num_views_minus1; i++ ) {		
<b>num_non_anchor_refs_l0[ i ]</b>		ue(v)
for( j = 0; j < num_non_anchor_refs_l0[ i ]; j++ )		
<b>non_anchor_ref_l0[ i ][ j ]</b>		ue(v)
<b>num_non_anchor_refs_l1[ i ]</b>		ue(v)
for( j = 0; j < num_non_anchor_refs_l1[ i ]; j++ )		
<b>non_anchor_ref_l1[ i ][ j ]</b>		ue(v)
}		
<b>num_level_values_signalled_minus1</b>		ue(v)
for( i = 0; i <= num_level_values_signalled_minus1; i++ )		
{		
<b>level_idc[ i ]</b>		u(8)
<b>num_applicable_ops_minus1[ i ]</b>		ue(v)
for( j = 0; j <= num_applicable_ops_minus1[ i ]; j++ ) {		
<b>applicable_op_temporal_id[ i ][ j ]</b>		u(3)
<b>applicable_op_num_target_views_minus1[ i ][ j ]</b>		ue(v)
for( k = 0; k <=		
applicable_op_num_target_views_minus1[ i ][ j ]; k++ )		
<b>applicable_op_target_view_id[ i ][ j ][ k ]</b>		ue(v)
<b>applicable_op_num_views_minus1[ i ][ j ]</b>		ue(v)
}		
}		
}		

[0086] Na extensão de MVC de SPS, para cada vista, o número de vistas que pode ser usado para formar

lista de figuração de referência 0 e lista de figuração de referência 1 são sinalizadas. As relações de previsão para uma figuração de âncora, como sinalizadas na extensão de MVC de SPS, podem ser diferentes da relação de previsão para uma figuração de não âncora (sinalizada na extensão de MVC de SPS) da mesma vista.

[0087] A próxima seção discutirá codificação de vídeo múltiplas vistas e 3D em relação a HEVC. Em particular, as técnicas exemplificativas desta revelação são aplicáveis ao codificar duas ou mais vistas. A pluralidade de figurações de vídeo para cada vista pode ser chamada de componentes de vista de textura. Para 3D-HEVC, cada componente de vista de textura pode ter um componente de vista de profundidade correspondente. MV-HEVC não usa componentes de vista de profundidade. Os componentes de vista de textura incluem conteúdo de vídeo (por exemplo, componentes de luma e de croma de valores de pixel), e os componentes de vista de profundidade (por exemplo, para 3D-HEVC) podem indicar as profundidades relativas dos pixels dentro dos componentes de vista de textura.

[0088] As técnicas exemplificativas desta revelação se referem à codificação de dados de vídeo 3D pela codificação de dados de textura ou dados de textura e dados de profundidade. Em geral, o termo "textura" é usado para descrever valores de luminância (isto é, brilho ou "luma") de valores de imagem e crominância (isto é, cor ou "croma") da imagem. Em alguns exemplos, uma imagem de textura pode incluir um conjunto de dados de luminância e dois conjuntos de dados de crominância, para matizes de azul (Cb) e matizes de vermelho (Cr). Em determinados formatos de croma, tais como 4:2:2 ou 4:2:0, os dados de croma têm resolução reduzida em relação aos dados de luma. Isto é, a resolução espacial de componentes de crominância

é menor que a resolução espacial de componentes de luminância correspondentes, por exemplo, uma metade ou um quarto da resolução de luminância.

[0089] Os dados de profundidade descrevem de modo geral os valores de profundidade para dados de textura correspondentes. Por exemplo, uma imagem de profundidade pode incluir um conjunto de pixels de profundidade que descrevem, cada um, a profundidade para dados de textura correspondentes. Os dados de profundidade podem ser usados para determinar a disparidade horizontal para os dados de textura correspondentes. Desse modo, um dispositivo que recebe os dados de textura e de profundidade pode exibir uma primeira imagem de textura para uma vista (por exemplo, uma vista de olho esquerdo) e usar os dados de profundidade para modificar a primeira imagem de textura para gerar uma segunda imagem de textura para a outra vista (por exemplo, uma vista de olho direito) deslocando-se os valores de pixel da primeira imagem pelos valores de disparidade horizontal determinados com base nos valores de profundidade. Em geral, a disparidade horizontal (ou simplesmente "disparidade") descrevem o deslocamento espacial horizontal de um pixel em uma primeira vista a um pixel correspondente em uma segunda vista, em que os dois pixels correspondem à mesma porção do mesmo objeto que o representado nas duas vistas.

[0090] Em ainda outros exemplos, dados de profundidade podem ser definidos por pixels em uma dimensão z perpendicular ao plano de imagem, de modo que uma profundidade associada a um determinado pixel seja definida em relação a um plano de disparidade zero definido para a imagem. Tal profundidade pode ser usada para criar disparidade horizontal para exibir o pixel, de modo que o pixel seja exibido de modo diferente para olhos esquerdo e

direito, dependendo do valor de profundidade de dimensão  $z$  do pixel em relação ao plano de disparidade zero. O plano de disparidade zero pode se alterar para porções diferentes de uma sequência de vídeo, e a quantidade de profundidade em relação ao plano de disparidade zero também pode se alterar. Pixels localizados no plano de disparidade zero podem ser definidos de modo semelhante para os olhos esquerdo e direito. Pixels localizados na parte frontal do plano de disparidade zero podem ser exibidos em localizações diferentes para o olho esquerdo e olho direito (por exemplo, com disparidade horizontal) de modo a criar uma percepção de que o pixel aparenta sair da imagem na direção  $z$  perpendicular ao plano de imagem. Pixels localizados na parte de trás do plano de disparidade zero podem ser exibidos com um ligeiro embaçamento, para apresentar uma ligeira percepção da profundidade, ou podem ser exibidos em localizações diferentes para o olho esquerdo e olho direito (por exemplo, com disparidade horizontal que é oposta àquela de pixels localizados na parte frontal do plano de disparidade zero). Diversas outras técnicas também podem ser usadas para transmitir ou definir dados de profundidade para uma imagem.

[0091] Para cada pixel no componente de vista de profundidade, pode haver um ou mais pixels correspondentes no componente de vista de textura. Por exemplo, se as resoluções espaciais do componente de vista de profundidade e do componente de vista de textura forem as mesmas, cada pixel no componente de vista de profundidade corresponde a um pixel no componente de vista de textura. Se a resolução espacial do componente de vista de profundidade for menor que aquela do componente de vista de textura, então, cada pixel no componente de vista de profundidade corresponde a múltiplos pixels no componente

de vista de textura. O valor do pixel no componente de vista de profundidade pode indicar a profundidade relativa dos um ou mais pixels correspondentes na vista de textura.

[0092] Em alguns exemplos, um criptador de vídeo sinaliza dados de vídeo para os componentes de vista de textura e os componentes de vista de profundidade correspondentes para cada uma dentre as vistas. Um decodificador de vídeo utiliza tanto os dados de vídeo dos componentes de vista de textura quanto os componentes de vista de profundidade para decodificar o conteúdo de vídeo das vistas para a exibição. Uma exibição, então, exibe o vídeo de múltiplas vistas para produzir vídeo 3D.

[0093] As extensões escalonáveis de HEVC também são desenvolvidas por JCT-VC. A Figura 4 é um diagrama conceitual que ilustra um exemplo de codificação de vídeo escalonável. Embora a Figura 4 é descrita em relação a H.264/AVC e SVC, deve ser entendido que camadas similares podem ser codificadas com o uso de outros esquemas de codificação de vídeo de múltiplas camadas, que incluem extensão escalonável de HEVC. O exemplo da Figura 4 mostra camadas codificadas com o uso do mesmo codec. Em outros exemplos, as camadas podem ser codificadas com o uso de um codec de múltiplos padrões. Por exemplo, uma camada-base pode ser codificada com o uso de H.264/AVC, enquanto uma camada de aprimoramento pode ser codificada com o uso de uma extensão escalonável para HEVC. Desse modo, as referências a SVC abaixo podem se aplicar a codificação de vídeo escalonável de modo geral, e não se limitam à extensão escalonável de H.264/AVC.

[0094] Em SVC, as escalabilidades podem ser permitidas em três dimensões que incluem, for exemplo, dimensões espaciais, temporais e de qualidade (representadas como uma taxa de bit ou sinal para taxa de

ruído (SNR)). Em geral, uma representação melhor pode ser alcançada normalmente pela adição a uma representação em qualquer dimensão. Por exemplo, no exemplo da Figura 4, a camada 0 é codificada em Um Quarto de Formato Intermediário Comum (QCIF) que tem uma taxa de quadro de 7,5 Hz e uma taxa de bit de 64 kilobytes por segundo (KBPS). Além disso, a camada 1 é codificada em QCIF que tem uma taxa de quadro de 15 Hz e uma taxa de bit de 64 KBPS, a camada 2 é codificada em CIF que tem uma taxa de quadro de 15 Hz e uma taxa de bit de 256 KBPS, a camada 3 é codificada em QCIF que tem uma taxa de quadro de 7,5 Hz e uma taxa de bit de 512 KBPS, e a camada 4 é codificada em 4CIF que tem uma taxa de quadro de 30 Hz e uma taxa de bit de Megabytes por segundo (MBPS). Deve ser entendido que o número, os conteúdos e disposições particulares das camadas mostradas na Figura 4 são aprimorados para fins exemplificativos apenas.

[0095] De qualquer maneira, uma vez que um criptador de vídeo (tal como criptador de vídeo 20) tenha criptado o conteúdo em tal uma forma escalonável, um decodificador de vídeo (tal como decodificador de vídeo 30) pode usar uma ferramenta extratora para adaptar o conteúdo entregue real de acordo com as exigências de aplicação, que podem depender, por exemplo, do cliente ou do canal de transmissão.

[0096] Em SVC, as figurações que têm a camada espacial e de qualidade mais baixa são compatíveis tipicamente com H.264/AVC. No exemplo da Figura 4, as figurações com a camada espacial e de qualidade mais baixa (por exemplo, as figurações na camada 0 e na camada 1, com resolução de QCIF) podem ser compatíveis com H.264/AVC. Entre essas, aquelas figurações do nível temporal mais baixo forma a camada-base temporal (por exemplo, camada 0).

Essa camada-base temporal (por exemplo, camada 0) pode ser aprimorada com figurações de níveis temporais mais altos (por exemplo, camada 1).

[0097] Além da camada compatível de H.264/AVC, diversas camadas de aprimoramento espaciais e/ou de qualidade podem ser adicionadas para fornecer escalabilidades espaciais e/ou de qualidade. Cada camada de aprimoramento espacial ou de qualidade pode ser escalonável temporalmente, com a mesma estrutura de escalabilidade temporal que a camada compatível de H.264/AVC.

[0098] Conforme descrito acima, o termo "camada" é usado de modo geral nessa revelação para se referir a camadas-base individuais ou camadas de aprimoramento individuais no sentido de codificação de vídeo escalonável, ou vista de textura e/ou vistas de profundidade no sentido de codificação de vídeo 3D e de múltiplas vistas. Uma vez que as figurações para cada um dentre as camadas (por exemplo, camadas de MV-HEVC ou camadas de SHVC), sejam decodificadas, por exemplo, pelo decodificador de vídeo 30 ou pelo circuito de reconstrução de criptador de vídeo 20, as figurações para a camada decodificada são armazenadas em um DPB. Um DPB é um armazenamento temporário ou memória que é usado para armazenar figurações, e nesta revelação, as figurações de camadas decodificadas ao usar técnicas de codificação de vídeo de múltiplas camadas. As figurações das camadas decodificadas armazenadas no DPB podem ser usadas como referências para interpretação (incluindo a compensação de movimento, previsão entre vistas e intercamada), para reordenação de saída e atraso de saída.

[0099] De acordo com diversos exemplos desta revelação, um DPB pode ser particionado em subunidades (por exemplo, chamado de sub-DPBs). Cada sub-DPB pode, então,



ser configurada para armazenar uma figuração ou figurações, ou de modo mais geral, informações de figuração, para um tipo de camada particular. Por exemplo, um DPB pode ser particionado de modo que um sub-DPB seja configurado para armazenar figurações de uma camada-base para um processo de codificação de vídeo escalonável. Outro sub-DPB pode ser configurado para armazenar figurações de uma primeira camada de aprimoramento de um processo de codificação de vídeo escalonável. Em ainda outro sub-DPB pode ser configurado para armazenar figurações de uma segunda camada de aprimoramento de um processo de codificação de vídeo escalonável, e assim por diante.

[00100] De acordo com diversos exemplos desta revelação, conforme será descrito em maiores detalhes abaixo, o criptador de vídeo 20 também pode ser configurado para gerenciar cada sub-DPB de modo independente um do outro, tal como nos termos de marcar figurações para remoção do DPB, remoção da figuração decodificada do DPB, bem como a alocação de tamanho para cada sub-DPB. No entanto, o criptador de vídeo 20 pode realizar enlace da saída de figurações decodificadas a partir de sub-DPBs diferentes, por exemplo, com base nos valores de tempos de saída respectivos e/ou de contagem de ordem de figuração (POC) associados a cada figuração decodificada.

[00101] Um dentre o criptador de vídeo 20 e o decodificador de vídeo 30 ou ambos (e/ou um ou mais componentes dos mesmos) podem implantar múltiplos modos diferentes de operações com base em sub-DPB. Por exemplo, dois modos de operação com base em sub-DPB podem diferir em termos da maneira na qual as figurações são associadas a cada sub-DPB. De acordo com o primeiro modo (chamado no presente documento de "modo de camada específica"), cada camada no ponto de operação pode ser associada a um sub-DPB

separado, e o número de sub-DPBs pode ser igual ao número de camadas no conjunto de camadas correspondente ao ponto de operação. De acordo com um segundo modo (chamado no presente documento de "modo de resolução específica"), todas as figurações que compartilham a mesma resolução, formato de croma e profundidade de bit compartilham o mesmo sub-DPB. O criptador de vídeo 20 pode especificar o modo de camada específica de modo relativamente fácil. No entanto, o modo de camada específica pode ser menos eficaz em termos de uso de memória de DPB, quando comparado ao modo de resolução específica.

[00102] A Figura 5 é um diagrama conceitual que mostra o DPB 900 que é configurada para armazenar diferentes componentes de camada decodificados (por exemplo, figurações para tipos de camada diferentes) em subunidades diferentes. O DPB 900 pode formar parte de um decodificador de vídeo e/ou criptador de vídeo (por exemplo, criptador de vídeo 20 e decodificador de vídeo 30). Cada um dos sub-DPBs 910A a representa localizações de armazenamento para figuração ou figurações de tipos diferentes de camadas decodificadas. Por exemplo, o sub-DPB 910A pode ser configurado para armazenar uma ou mais figurações de textura de uma vista-base para um fluxo de bits de vídeo compatível de MV-HEVC, enquanto o sub-DPB 910B pode ser usado para armazenar figurações de textura de uma vista dependente para o fluxo de bits de vídeo compatível de MV-HEVC. De modo semelhante, o sub-DPB 910C pode ser usado para armazenar figurações de uma camada-base em um processo de codificação de vídeo escalonável (por exemplo, SHVC), enquanto o sub-DPB 910D pode ser usado para armazenar um nível de figurações para uma camada de aprimoramento no processo de codificação de vídeo escalonável. Os sub-DPBs adicionais podem ser configurados

para armazenar figurações de níveis adicionais de camada de aprimoramento ou figurações de vistas dependentes de textura adicionais. Cada um dos sub-DPBs 910A a D contém unidades separadas que são configuradas para armazenar uma ou mais figurações de uma camada decodificada para um tipo de camada particular associado ao sub-DPB. Deve ser observado que os dados de vídeo codificados com MV-HEVC e dados de vídeo codificados com SHVC não são decodificados necessariamente ao mesmo tempo. Isto é, os dados de vídeo codificados de SHVC e MV-HEVC não estão tipicamente no mesmo fluxo de bits de vídeo criptados, porém, em vez disso, são tipicamente codificados separadamente.

[00103] O decodificador de vídeo 30 pode ser configurado adicionalmente para remover as figurações de camadas decodificadas a partir da pluralidade de sub-DPBs 910A a D. Em alguns exemplos, a remoção de figurações a partir de cada sub-DPB determinado (por exemplo, sub-DPB 910A) pode ser gerenciado separadamente (isto é, as figurações podem ser removidas a partir de um sub-DPB de modo independente a partir da remoção de outras figurações a partir de outros sub-DPBs). Em outros exemplos, as figurações podem ser removidas a partir de um ou mais sub-DPBs com base em um ponto de operação. Um ponto de operação particular é associado a um número particular de camadas que são decodificadas para uma figuração particular a ser emitida. Por exemplo, para SHVC, um ponto de operação pode apenas exigir uma figuração de uma camada-base a ser decodificada. Em outro ponto de operação para SHVC, uma figuração de uma camada-base e figurações de uma ou mais camadas de aprimoramento podem ser decodificadas para produzir uma figuração de saída. No ponto de operação que exige tanto uma figuração de uma camada-base quanto figurações de uma ou mais camadas de aprimoramento, a

remoção de figurações pode ser realizada para cada sub-DPB usado para tal ponto de operação (isto é, cada sub-DPB que armazenada figurações para uma camada que é usada para a figuração a ser emitida). Isto é, uma figuração de uma camada-base e as figurações correspondentes das uma ou mais camadas de aprimoramento associadas a um determinado ponto de operação podem ser removidas a partir de seus sub-DPBs respectivos ao mesmo tempo.

[00104] O criptador de vídeo 20 e o decodificador de vídeo 30 podem ser configurados adicionalmente para marcar os componentes de camada decodificada (por exemplo, figurações de camadas) como não usadas para referência ou usadas para referência na pluralidade de sub-DPBs 910A-D. Novamente, o processo para marcar figurações como não usadas para referência pode ser realizado separadamente para cada sub-DPB. Isto é, o criptador de vídeo 20 e o decodificador de vídeo 30 podem ser configurados para realizar um processo de marcação nas figurações em cada um dentre a pluralidade dos sub-DPBs independentemente, sendo que o processo de marcação marca as figurações como usadas para referência ou marca as figurações como não usadas para referência. Em outro exemplo, o processo de marcação pode ser realizado para todos os sub-DPBs que armazenam as figurações para um número particular de camadas relacionadas a um ponto de operação particular.

[00105] Em HEVC e outros padrões de codificação de vídeo, a operação de DPB 900 é frequentemente especificada em relação a um decodificador hipotético de referência (HRD). O criptador de vídeo 20 e o decodificador de vídeo 30 podem ser configurados para gerenciar o DPB 900 para realizar diversas ações, que incluem marcar figurações decodificadas armazenadas no DPB 900 como "não usada para

referência" (isto é, incapaz de ser usada como uma figuração de referência para um processo de interprevisão), marcar uma figuração decodificada para saída (por exemplo, para o dispositivo de exibição 32) e marcar uma figuração decodificada para remoção a partir de DPB 900 (também chamado de "aumento"). Em HEVC, o processo de aumento inclui tanto marcar uma figuração para saída quanto marcar uma figuração para remoção. Uma figuração é removida tipicamente (isto é, aumentada) do DPB 900 quando a figuração não é mais necessária como uma figuração de referência para interprevisão e não é mais necessária para a saída.

[00106] A conformidade de fluxos de bits é especificada definindo-se o comportamento do HRD. O DPB 900 pode ser considerado como um componente do HRD. Conforme descrito acima, DPB 900 pode ser implantado como uma ou mais unidades de memória configurados para armazenar figurações reconstruídas em um criptador de vídeo (por exemplo, criptador de vídeo 20) ou figurações decodificadas em um decodificador de vídeo (por exemplo, decodificador de vídeo 30). Em algumas propostas para SHVC e MV-HEVC, as operações de DPB não são especificadas claramente, porém, uma suposição simples é que todas as figurações com a mesma chegada de tempo de saída no DPB 900 (isto é, são armazenadas no DPB 900) no mesmo instante de tempo. A saída de figurações a partir do DPB 900 é determinada com base em como os parâmetros de HRD são definidos.

[00107] Por exemplo, os parâmetros de HRD podem ser definidos para conformidade de ordem de saída. Como exemplo, a ordem de saída de figurações decodificadas de camada pode ser especificada por um ou mais elementos de sintaxe cujo valor especifica o número de figurações presentes para uma camada particular. Como outro exemplo,

para conformidade de tempo de saída, o tempo de saída pode ser especificado por um ou mais elementos de sintaxe cujos valores indicam os tempos de saída de figuração. O decodificador de vídeo 30 pode ser configurado para derivar configurações de HRD a partir dos valores de parâmetros sinalizados em um fluxo de bits de vídeo criptados. Os parâmetros de HRD podem ser sinalizados em uma estrutura de formato de dados (por exemplo, chamada `hrd_parameters`). Os parâmetros exemplificativos podem incluir uma mensagem de informações de aprimoramento suplementar de período de armazenamento temporário (SEI) e uma mensagem de SEI de temporização de figuração. Em alguns exemplos, uma mensagem de SEI de informações de unidade de decodificação (DU) também pode ser definida. Em alguns exemplos, a saída das figurações em DPB 900 ocorre de modo que todas as figurações associadas ao mesmo tempo de saída sejam emitidas no mesmo instante. Nos exemplos desta revelação, as figurações em sub-DPBs diferentes podem ser emitidas ao mesmo tempo no caso de cada uma dessas figurações (isto é, as figurações nos sub-DPBs diferentes) serem associadas ao mesmo tempo de saída.

[00108] Em algumas propostas exemplificativas para HEVC, os parâmetros que são relacionados às operações de DPB são sinalizadas em um ou mais dentre um conjunto de parâmetros de sequência (SPS) e/ou um conjunto de parâmetros de vídeo (VPS). Um SPS é uma estrutura de sintaxe que contém elementos de sintaxe que se aplicam a zero ou sequências de vídeo codificadas mais inteiras (CVSs) conforme indicado pelos valores de elementos de sintaxe encontrados em um conjunto de parâmetros de figuração (PPS) e/ou um cabeçalho de fatia. Um VPS é uma estrutura de sintaxe que contém elemento de sintaxe que e aplica a zero ou CVSs mais inteiras conforme determinado

pelo valor de elemento de sintaxe encontrado em um SPS, PPS, e/ou cabeçalho de fatia. Em geral, elementos de sintaxe sinalizados em um VPS se aplicarão a um número maior de figurações que o número de figurações ao qual o elemento de sintaxe sinalizado em um SPS se aplica. Conforme definido em HEVC, uma CVS é uma sequência de unidades de acesso que consiste de, em ordem de decodificação, uma unidade de acesso (AU) que contém uma figuração de acesso intra aleatório (IRAP) associado a uma `nuh_layer_id` igual a um valor de 0 e com um `NoRaslOutputFlag` igual a um valor de 1 seguido por AU zero ou outras AUs até a próxima AU (isto é, até, porém, sem incluir a próxima AU), em ordem de decodificação, que contém uma figuração de IRAP com `nuh_layer_id` igual a um valor de 0 e `NoRaslOutputFlag` igual a um valor de 1. Uma unidade de acesso é um conjunto de unidades de camada de abstração de rede (NAL) que são consecutivas em ordem de decodificação e contêm uma ou mais configurações codificadas que têm, cada uma, o mesmo tempo de saída.

[00109] Elementos de sintaxe exemplificativos que são relacionados às operações de DPB incluem, porém, sem limitação, o elemento de sintaxe `max_dec_pic_buffering_minus1`, o elemento de sintaxe `max_num_reorder_pics`, e o elemento de sintaxe `max_latency_increase_plus1`, que podem, cada um, ser sinalizados no SPS e/ou no VPS. O elemento de sintaxe `max_dec_pic_buffering_minus1`, o elemento de sintaxe `max_num_reorder_pics`, e o elemento de sintaxe `max_latency_increase_plus1` pode ser chamado coletivamente de parâmetros de DPB e/ou de parâmetros de sub-DPB.

[00110] O valor do elemento de sintaxe `max_dec_pic_buffering_minus1` especifica o tamanho máximo permitido de um DPB ou sub-DPB para uma CVS nas unidades de

armazenamentos temporários de figuração. Isto é, um DPB ou um sub-DPB podem ser particionados em um ou mais armazenamentos temporários de figuração, em que cada armazenamento temporário de figuração armazena uma figuração (por exemplo, uma figuração associada a uma camada particular). Em alguns exemplos, o valor de `max_dec_pic_buffering_minus1` é limitado a uma faixa de a partir de 0 a algum tamanho de DPB e/ou sub-DPB máximo predeterminado.

[00111] O valor do elemento de sintaxe `max_num_reorder_pics` especifica o número máximo permitido de figurações que pode ser armazenado em um DPB e/ou sub-DPB e que pode preceder qualquer figuração na CVS na ordem de decodificação e segue a mesma figuração na ordem de saída. O valor do elemento de sintaxe `max_latency_increase_plus1`, quando não for igual a zero (isto é, com base no valor do elemento de sintaxe `max_latency_increase_plus1` que é um valor diferente de zero), é uma indicação de tempo de latência que é usado para determinar o número máximo de figurações que pode ser armazenado em um DPB e/ou sub-DPB que pode preceder qualquer figuração na CVS na ordem de saída e segue a mesma figuração na ordem de decodificação. Se o número de figurações em um DPB e/ou sub-DPB exceder o número máximo de figurações indicado pelo valor do elemento de sintaxe `max_num_reorder_pics` ou o valor determinado a partir do valor do elemento de sintaxe `max_latency_increase_plus1`, uma ou mais figurações devem ser removidas do DPB e/ou do sub-DPB.

[00112] Em um exemplo, o criptador de vídeo 20 pode ser configurado para gerar e sinalizar os parâmetros de DPB e/ou parâmetros de sub-DPB mencionados anteriormente para cada subcamada, por exemplo, se (isto é, com base em)



o valor de um elemento de sintaxe de comutação (por exemplo, a `sub_layer_ordering_info_present_flag`) é igual a 1. Conforme descrito acima, uma subcamada é uma camada escalonável temporal de um fluxo de bits escalonável temporal. De acordo com esse exemplo, quando a sinalização de subcamada não é permitida, apenas um valor é sinalizado para cada um dentre os elementos de sintaxe, e os valores sinalizados de cada um desses elementos de sintaxe de parâmetro de sub-DPB (isto é, o elemento de sintaxe `max_dec_pic_buffering_minus1`, o elemento de sintaxe `max_num_reorder_pics`, e o elemento de sintaxe `max_latency_increase_plus1`) são inferidos (isto é, determinados sem o recebimento de sinalização explícita) para ser o mesmo para todas as subcamadas temporais. De outro modo, quando a sinalização de subcamada é permitida (por exemplo, o valor de elemento de sintaxe `sub_layer_ordering_info_present_flag` é igual a 0), os valores dos elementos de sintaxe de parâmetro de sub-DPB são sinalizados explicitamente para cada subcamada. A sinalização dos parâmetros de sub-DPB no SPS é destacada na Tabela 2 abaixo em *itálico*. Sinalização semelhante pode ser usada no VPS.

TABELA 2 - PARÂMETROS DE DPB NO SPS

seq_parameter_set_rbsp( ) {	<b>Descriptor</b>
<b>sps_video_parameter_set_id</b>	u(4)
if( nuh_layer_id == 0 ) {	
<b>sps_max_sub_layers_minus1</b>	u(3)
<b>sps_temporal_id_nesting_flag</b>	u(1)
profile_tier_level( 1, sps_max_sub_layers_minus1 )	
}	
<b>sps_seq_parameter_set_id</b>	ue(v)
<b>chroma_format_idc</b>	ue(v)
if( chroma_format_idc == 3 )	
<b>separate_colour_plane_flag</b>	u(1)
<b>pic_width_in_luma_samples</b>	ue(v)
<b>pic_height_in_luma_samples</b>	ue(v)
<b>conformance_window_flag</b>	u(1)
if( conformance_window_flag ) {	
<b>conf_win_left_offset</b>	ue(v)
<b>conf_win_right_offset</b>	ue(v)
<b>conf_win_top_offset</b>	ue(v)
<b>conf_win_bottom_offset</b>	ue(v)
}	
<b>bit_depth_luma_minus8</b>	ue(v)
<b>bit_depth_chroma_minus8</b>	ue(v)
<b>log2_max_pic_order_cnt_lsb_minus4</b>	ue(v)
<b>sps_sub_layer_ordering_info_present_flag</b>	u(1)
for( i = ( sps_sub_layer_ordering_info_present_flag ? 0 : sps_max_sub_layers_minus1 ); i <= sps_max_sub_layers_minus1; i++ ) {	
<b>sps_max_dec_pic_buffering_minus1[ i ]</b>	ue(v)
<b>sps_max_num_reorder_pics[ i ]</b>	ue(v)
<b>sps_max_latency_increase_plus1[ i ]</b>	ue(v)
}	
...	

[00113] Na versão-base de HEVC (algumas vezes

chamada de HEVC 1), apenas uma única camada está presente (por exemplo, conforme indicado pelo elemento de sintaxe `nuh_layer_id` que tem um valor igual a 0), e, portanto, o conjunto de camadas decodificado e a saída de conjunto de camadas são a mesma. No contexto de codificação de vídeo de múltiplas camadas (por exemplo, SHVC ou MV-HEVC), o número de conjuntos de camadas e conjuntos de camadas de saída não são necessariamente o mesmo, visto que o número total de conjuntos de camadas pode não ser totalmente saída para cada ponto de operação. Isto é, pontos de operação de SHVC incluem frequentemente a decodificação de uma camada-base e uma ou mais camadas de aprimoramento. De modo semelhante, os pontos de operação de MV-HEVC frequentemente incluem a decodificação de duas ou mais vistas. No entanto, nem todas as camadas ou vistas podem ser usadas para emitir uma determinada figuração.

[00114] As técnicas para a remoção das figurações a partir do DPB serão discutidas agora. As técnicas exemplificativas para HEVC envolvem duas formas de especificação da remoção de figurações do DPB. As mesmas incluem um processo de remoção de DPB com base em figuração e um processo de remoção de DPB com base em AU. Em um processo de remoção de DPB com base em figuração, as figurações em uma AU podem ser removidas individualmente em tempos diferentes. Em operações de remoção de DPB com base em AU, todas as figurações na AU são removidas ao mesmo tempo. As propostas atuais para operações de remoção de DPB para SHVC e MV-HEVC não especificam claramente como as figurações devem ser removidas.

[00115] Propostas anteriores para parâmetros de DPB de sinalização, incluem parâmetros para sub operações de DPB (isto é, parâmetros de sub-DPB), tais como aqueles propostos em Ramasubramonian, et al, "MV-HEVC/SHVC HLS: On

DPB Operations", Equipe Colaborativa Conjunta em Extensões de Codificação de Vídeo em 3D de ITU-T SG 16 WP 3 e ISO/IEC JTC 1/SC 29/WG 11, 5º Encontro: Viena, AT, 27 de julho a 2 de agosto de 2013 (doravante no presente documento, "JCTVC-N0093\_v4") pode ser sub ideal. Como exemplo, de acordo com o esquema de sinalização para operações com base em sub-DPB em JCTVC-N0093\_v4, os três parâmetros relacionados a DPBs e sub-DPBs (isto é, o elemento de sintaxe `max_dec_pic_buffering_minus1`, o elemento de sintaxe `max_num_reorder_pics`, e o elemento de sintaxe `max_latency_increase_plus1` descritos acima) são sinalizados para cada conjunto de camadas de saída, cada camada no conjunto de camadas de saída, e para cada subcamada temporal dentro de cada camada. Tal esquema de sinalização pode ser sub ideal.

[00116] Em particular, os elementos de sintaxe de sinalização que indicam o número máximo de figurações de reordenação e da latência máxima para cada camada em cada conjunto de camadas de saída podem ser redundantes, causando potencialmente o desperdício de recursos de computação (por exemplo, recursos de processador, capacidade de armazenamento, etc.) e/ou largura de banda de rede. Por exemplo, a saída de uma ou mais figurações de um conjunto de camadas de saída em uma unidade de acesso (AU) pode ser simultânea. Adicionalmente, emitir figurações a partir de uma segunda AU antes que todas as figurações em uma primeira AU sejam emitidas pode não ser permitido. Portanto, os parâmetros de sub-DPB de sinalização para cada camada em um conjunto de camadas de saída podem ser redundantes, visto que valores diferentes para parâmetros de sub-DPB podem não ser possíveis. Desse modo, pode ser redundante enviar informações de reordenação e/ou de latência para cada camada em um conjunto de camadas de

saída.

[00117] Para mitigar ou eliminar ineficácias e outros problemas potenciais causados por tal redundância de dados sinalizados, o criptador de vídeo 20 e/ou o decodificador de vídeo 30 podem implantar uma ou mais dentre as técnicas descritas no presente documento. Em um ou mais exemplos da revelação, o criptador de vídeo 20 pode ser configurado para sinalizar um conjunto de parâmetros de sub-DPB para cada conjunto de camadas de saída correspondente a um ponto de operação. O único conjunto de parâmetros de sub-DPB sinalizado para cada conjunto de camadas de saída é, então, usado pelo decodificador de vídeo 30 para cada camada respectiva em cada conjunto de camadas de saída. Dessa forma, a eficácia de sinalização é aumentada, visto que a sinalização redundante de parâmetros de sub-DPB é evitada. Além disso, os valores dos parâmetros de sub-DPB podem ser harmonizados através de todas as camadas de um conjunto de camadas de saída.

[00118] Em um exemplo da revelação, para o conjunto de camadas de saída de cada ponto de operação, o criptador de vídeo 20 pode ser configurado para gerar e sinalizar, no VPS, apenas um valor (isto é, um único valor) de um elemento de sintaxe indicativo do número máximo de figurações de reordenação e apenas um valor (isto é, um único valor) um elemento de sintaxe indicativo da latência máxima. Isto é, para cada ponto de operação relacionada a um VPS particular, o criptador de vídeo 20 gera um número máximo de figurações de reordenação elemento de sintaxe e um elemento de sintaxe de latência máxima. Consequentemente, o número total de elementos de sintaxe de figuração de reordenação e elementos de sintaxe de latência máxima em cada VPS é dependente do número de pontos de operação definidos para o fluxo de bits de vídeo. Desse

modo, menos exemplos de elementos de sintaxe de figuração de reordenação e elementos de sintaxe de latência máxima são sinalizados em comparação às propostas anteriores para HEVC em que os elementos de sintaxe de figuração de reordenação e elementos de sintaxe de latência máxima são sinalizados para cada conjunto de camadas de saída, cada camada no conjunto de camadas de saída, e para cada subcamada temporal dentro de cada camada.

[00119] Conforme será descrito abaixo em maiores detalhes com referência à primeira implantação exemplificativa, o elemento de sintaxe indicativo do número máximo de figurações de reordenação e o elemento de sintaxe indicativo da latência máxima podem ser sinalizados em uma tabela de tamanho de DPB e podem ser transmitidos em um VPS. Em outros exemplos da revelação, o elemento de sintaxe indicativo do número máximo de figurações de reordenação e o elemento de sintaxe indicativo da latência máxima podem ser transmitidos em outras estruturas de dados que inclui um PPS.

[00120] Mediante o recebimento do elemento de sintaxe indicativo do número máximo de figurações de reordenação e do elemento de sintaxe indicativo da latência máxima, o decodificador de vídeo 30 pode aplicar valores respectivos dos elementos de sintaxe indicativos do número máximo de figurações de reordenação e indicativos da latência máxima para o conjunto de camadas de saída para um ponto de operação particular.

[00121] Em um exemplo, os elementos de sintaxe indicativos do número máximo de figurações de reordenação e indicativos da latência máxima podem corresponde a apenas uma camada de uma camada de saída. Por exemplo, os elementos de sintaxe podem corresponder à camada que tem uma id de camada mais alta (por exemplo, conforme indicado

pelo elemento de sintaxe `nuh_layer_id`). Nesse exemplo, o criptador de vídeo 20 pode ser configurado para sinalizar um sinalizador para indicar para o decodificador de vídeo 30 a possibilidade de um valor do número máximo de figurações de reordenação e de um valor da latência máxima estarem associados à camada com o maior valor de `nuh_layer_id` no ponto de operação. O decodificador de vídeo 30 seria configurado para inferir o valor dos elementos de sintaxe para o conjunto de camadas de saída como o mesmo que o valor recebido para a camada que tem a id de camada mais alta.

[00122] Em outro exemplo da revelação, o criptador de vídeo 20 pode sinalizar um valor de id (por exemplo, com o uso do elemento de sintaxe `nuh_layer_id`) para indicar para o decodificador de vídeo 30 a camada com a qual o um valor do número máximo de figurações de reordenação e o um valor da latência máxima estão associados. Novamente, o decodificador de vídeo 30 seria configurado para inferir o valor dos elementos de sintaxe para o conjunto de camadas de saída como o mesmo que o valor recebido para a camada que tem a id de camada mais alta.

[00123] Em outro exemplo da revelação, para cada ponto de operação, o criptador de vídeo 20 pode ser configurado para sinalizar um sinalizador para indicar o decodificador de vídeo 30 a possibilidade dos parâmetros de sub-DPB (por exemplo, um elemento de sintaxe que indica o número máximo de figurações de reordenação e/ou um elemento de sintaxe que indica a latência máxima) serem sinalizados apenas para a camada com o maior valor de `nuh_layer_id` no conjunto de camadas de saída (isto é, a camada mais alta), ou que os parâmetros de sub-DPB são sinalizados para cada camada no conjunto de camadas de saída. Se o valor do

sinalizador indicar (isto é, com base na indicação de sinalizador) que os parâmetros de sub-DPB são sinalizados apenas para a camada mais alta, o decodificador de vídeo 30 pode ser configurado para inferir o valor dos parâmetros de sub-DPB para o conjunto de camadas de saída como o mesmo valor que os parâmetros de sub-DPB para a camada mais alta.

[00124] Em outro exemplo da revelação, para cada conjunto de camadas de saída, o criptador de vídeo 20 pode sinalizar um sinalizador para especificar para o decodificador de vídeo 30 a possibilidade dos parâmetros de sub-DPB (por exemplo, os elementos de sintaxe indicativos de tamanho de sub-DPB, o número máximo de figurações de reordenação, e/ou da latência máxima) serem sinalizados para apenas uma subcamada (por exemplo, a subcamada com ID temporal igual a 0) ou para mais que uma subcamada. Por exemplo, o criptador de vídeo 20 pode gerar o sinalizador para um primeiro valor para indicar que as informações de parâmetros de sub-DPB são sinalizadas para apenas uma subcamada. Nesse caso, o decodificador de vídeo 30, então, inferiria o valor dos parâmetros de sub-DPB para todas as subcamadas outras como o mesmo valor que os parâmetros de sub-DPB sinalizados atualmente para a uma subcamada (por exemplo, a subcamada com ID temporal igual a 0).

[00125] Por outro lado, o criptador de vídeo 20 pode definir o sinalizador para um segundo valor para indicar que os parâmetros de sub-DPB podem ser sinalizados para múltiplas subcamadas. Em um exemplo, quando o valor do sinalizador indica (isto é, com base na indicação de sinalizador) que os parâmetros de sub-DPB podem ser sinalizados para múltiplas subcamadas, o criptador de vídeo 20 pode ser configurado para gerar elementos de sintaxe para os parâmetros de sub-DPB para todas as subcamadas de um conjunto de camadas de saída. Em outro exemplo, conforme



será explicado em maiores detalhes abaixo com referência às Tabelas 3 e 4, o criptador de vídeo 20 pode ser configurado para gerar um segundo sinalizador que indica, para cada subcamada do conjunto de camadas de saída, a possibilidade ou não de parâmetros de sub-DPB adicionais serem sinalizados para uma subcamada particular. Quando o segundo sinalizador indica que os parâmetros de sub-DPB adicionais devem ser sinalizados para uma subcamada particular, o criptador de vídeo 20 gera e sinaliza os parâmetros de sub-DPB adicionais para a subcamada particular. Se o segundo sinalizador indicar que nenhum parâmetro de sub-DPB adicional será sinalizado para uma subcamada particular, o decodificador de vídeo 30 pode ser configurado para inferir os valores dos parâmetros de sub-DPB para a subcamada particular a partir de parâmetros de sub-DPB sinalizados anteriormente para o conjunto de camadas de saída (por exemplo, a partir dos parâmetros de sub-DPB para a subcamada que tem uma ID temporal igual a 0, ou a partir dos parâmetros de sub-DPB para a subcamada anterior). Em alguns exemplos, quando o segundo sinalizador indicar que os parâmetros de sub-DPB não são sinalizados para uma subcamada particular, os parâmetros de sub-DPB das subcamadas subsequentes àquela subcamada particular também não são sinalizados, e o decodificador de vídeo 30 pode ser configurado para inferir o valor dos parâmetros de sub-DPB para essas subcamadas subsequentes a partir dos parâmetros de sub-DPB sinalizados anteriormente para o conjunto de camadas de saída.

[00126] O decodificador de vídeo 30 pode ser configurado adicionalmente para realizar um processo de gerenciamento de sub-DPB (por exemplo, marcar figurações para a saída, marcar figurações como não usadas para referência, emitir figurações, e/ou remover figurações) de

acordo com técnicas de sinalização de parâmetro de sub-DPB desta revelação. Por exemplo, o decodificador de vídeo 30 pode ser configurado para realizar o processo de aumento em um nível de unidade de acesso, em vez de em um nível de figuração. Isso é devido ao fato de que, de acordo com as técnicas desta revelação, os parâmetros de sub-DPB se aplicam a um conjunto de camadas de saída completo e não apenas a um subconjunto das camadas. Também nesse sentido, a necessidade de verificar os valores dos parâmetros de sub-DPB para IDs de camada particular é removida, visto que todas as camadas de um conjunto de camadas de saída particular compartilham os mesmos parâmetros de sub-DPB. Maiores detalhes de processos de gerenciamento de DPB, de acordo com as técnicas desta revelação, são descritos abaixo com referência à primeira implantação exemplificativa.

[00127] As técnicas a seguir são exemplos adicionais de técnicas de sinalização para sinalizar os parâmetros de sub-DPB. Em um exemplo, para cada ponto de operação, o criptador de vídeo 20 pode sinalizar um sinalizador para especificar para o decodificador de vídeo 30 a possibilidade dos parâmetros de sub-DPB serem sinalizados. Alternativamente, o decodificador de vídeo 30 pode inferir um valor do sinalizador, em exemplos em que o criptador de vídeo 20 não sinaliza o sinalizador. Em outro exemplo, para cada ponto de operação, o criptador de vídeo 20 pode sinalizar um sinalizador para indicar para o decodificador de vídeo 30 a possibilidade de: a) os tamanhos de sub-DPB serem sinalizados apenas para o sub-DPB do índice mais alto e para os outros sub-DPBs inferidos como iguais ao tamanho do sub-DPB com o índice mais alto, ou b) os tamanhos de sub-DPB serem sinalizados para cada sub-DPB no ponto de operação.

[00128] As seções a seguir descrevem implantações exemplificativas das técnicas desta revelação em relação às técnicas de codificação de HEVC. As implantações exemplificativas serão descritas com o uso da terminologia e definições das especificações HEVC, SHVC, MV-HEVC e/ou 3D-HEVC. As definições de quaisquer termos não definidos explicitamente abaixo podem ser encontradas nas especificações HEVC, SHVC, MV-HEVC e/ou 3D-HEVC, como referência acima.

[00129] Em uma primeira implantação exemplificativa das técnicas descritas no presente documento, são descritas técnicas para sinalizar parâmetros de DPB e de sub-DPB. Elementos de sintaxe exemplificativos associados à primeira implantação exemplificativa são mostrados na Tabela 3 e Tabela 4 abaixo. Os elementos de sintaxe relacionados às técnicas desta revelação são distinguidos na Tabela 3 e na Tabela 4 por estarem em negrito e sublinhados. As exclusões das contribuições de HEVC anteriores são mostradas com um ~~tachado~~. As porções sublinhadas da Tabela 1 e da Tabela 2 são destinadas às alterações introduzidas pela Implantação Exemplificativa nº 1 para sintaxe existente.

TABELA 3 - SINTAXE DE EXTENSÃO DE CONJUNTO DE PARÂMETRO DE VÍDEO

<code>vps_extension() {</code>	<b>Descritor</b>
<code>avc_base_layer_flag</code>	<code>u(1)</code>
<code>...</code>	
<code>cross_layer_irap_aligned_flag</code>	<code>u(1)</code>
<b><u><code>dpb_size_table()</code></u></b>	
<code>direct_dep_type_len_minus2</code>	<code>ue(v)</code>
<code>....</code>	
<code>}</code>	

[00130] Conforme mostrado na Tabela 3, o criptador de vídeo 20 pode ser configurado para sinalizar uma tabela de tamanho de DPB (`dpb_size_table`) em uma sintaxe de extensão de VPS. A tabela de tamanho de DPB pode incluir parâmetros relacionados de DPB e de sub-DPB, que inclui elementos de sintaxe que têm valores indicativos de tamanho de sub-DPB, o número máximo de figurações de reordenação, e/ou a latência máxima (os parâmetros de sub-DPB).

TABELA 4 - SINTAXE DE TABELA DE TAMANHO DE DPB

dpb_size_table( ) {	
for( i = 1; i < NumOutputLayerSets; i++ ) {	
<b>sub_layer_flag_info_present_flag[ i ]</b>	u(1)
for( j = 0; j <= vps_max_sub_layers_minus1; j++ ) {	
if( j > 0 && sub_layer_flag_info_present_flag[ i ] && sub_layer_dpb_info_present_flag[ i ][ j - 1 ] )	
<b><u>sub_layer_dpb_info_present_flag[ i ][ j ]</u></b>	u(1)
if( sub_layer_dpb_info_present_flag[ i ][ j ] ) {	
for( k = 0; k < NumSubDpbs[ i ]; k++ )	
<b>max_vps_dec_pic_buffering_minus1[ i ][ k ][ j ]</b>	ue(v)
<b>max_vps_num_reorder_pics[ i ][ j ]</b>	ue(v)
<b>max_vps_latency_increase_plus1[ i ][ j ]</b>	ue(v)
}	
}	
}	
}	

[00131] Conforme mostrado na Tabela 4, o criptador de vídeo 20 é configurado para sinalizar uma tabela de tamanho de DPB que inclui elementos de sintaxe cujos valores especificam um tamanho de sub-DPB (`max_vps_dec_pic_buffering_minus1 [i][k][j]`), o número máximo de figurações de reordenação (`max_vps_num_reorder_pics[i][j]`), e a latência máxima de sub-DPB (`max_vps_latency_increase_plus1 [i][j]`). O For Loop "for(i = 1; i < NumOutputLayerSets; i++)" especifica que o tamanho de sub-DPB, o número máximo de figurações de reordenação, e a latência máxima de sub-DPB são sinalizados para cada conjunto de camadas de saída i (isto é, para cada ponto de operação) até o número máximo de conjuntos de

camadas de saída (NumOutputLayerSets). Consequentemente, o decodificador de vídeo 30 é configurado para aplicar o tamanho máximo de sub-DPB recebido, o número máximo de figurações de reordenação, e a latência máxima de sub-DPB para cada camada em um conjunto de camadas de saída de um ponto de operação.

[00132] Adicionalmente, conforme mostrado na Tabela 4, o criptador de vídeo 20 também pode ser configurado para gerar e sinalizar um sinalizador (sub\_layer\_flag\_info\_present\_flag[i]) que indica a possibilidade dos valores sinalizados indicativos de tamanho máximo de sub-DPB, de número máximo de figurações de reordenação, e de latência máxima de sub-DPB serem sinalizados para apenas uma subcamada de cada camada, ou para mais que uma subcamada de uma camada. Por exemplo, se o valor de sub\_layer\_flag\_info\_present\_flag[i] for 1, os valores dos elementos de sintaxe indicam o tamanho máximo de sub-DPB, o número máximo de figurações de reordenação e a latência máxima de sub-DPB serão usados para todas as subcamadas. No entanto, se o valor de sub\_layer\_flag\_info\_present\_flag[i] for 0, elementos de sintaxe separados cujos valores indicam o tamanho máximo de sub-DPB, o número máximo de figurações de reordenação, e a latência máxima de sub-DPB podem ser sinalizados para mais que uma subcamada. Nesse exemplo, decodificador de vídeo 30, com base no valor dos elementos de sintaxe respectivos, pode ser configurado para usar valores diferentes para o tamanho máximo de sub-DPB, o número máximo de figurações de reordenação e a latência máxima de sub-DPB para duas ou mais subcamadas diferentes.

[00133] No caso de o valor de sub\_layer\_flag\_info\_present\_flag[i] ser 1 (indicando que os parâmetros de sub-DPB serão sinalizados para mais que uma

subcamada), o criptador de vídeo 20 pode ser configurado adicionalmente para sinalizar um elemento de sintaxe adicional (`sub_layer_dpb_info_present_flag[i][j]`) para cada subcamada em cada sub-DPB que indica a possibilidade ou não dos parâmetros de sub-DPB serem sinalizados para uma subcamada particular.

[00134] No exemplo da Tabela 4, a declaração se (`"if(j > 0 && sub_layer_flag_info_present_flag[i]&& sub_layer_dpb_info_present_flag[i][j - 1])"`) exige que criptador de vídeo 20 sinaliza `sub_layer_dpb_info_present_flag[i][j]` para uma subcamada particular apenas se o valor de `sub_layer_flag_info_present_flag[i]` for 1 e o valor `sub_layer_dpb_info_present_flag` da subcamada anterior (`sub_layer_dpb_info_present_flag[i][j - 1]`) tiver um valor de 1. Isto é, o criptador de vídeo 20 não sinaliza a `sub_layer_dpb_info_present_flag`, ou quaisquer parâmetros de sub-DPB, se os parâmetros de sub-DPB não forem sinalizados para uma subcamada anterior.

[00135] Em um exemplo da revelação, a exigência de verificar o valor de `sub_layer_dpb_info_present_flag` para uma subcamada anterior é removida. Em vez disso, a `sub_layer_dpb_info_present_flag` pode ser sinalizada para todas as subcamadas, independentemente de quaisquer valores anteriores da `sub_layer_dpb_info_present_flag`.

[00136] As semânticas associadas aos elementos de sintaxe da Tabela 4 acima são descritas conforme o seguinte. Conforme usado no presente documento, `NumSubDpbs[i]` pode denotar um número de sub-DPBs exigido para um *i*-ésimo conjunto de camadas de saída.

[00137] O valor de elemento de sintaxe `sub_layer_flag_info_present_flag[i]` igual a 1 indica que `sub_layer_dpb_info_present_flag[i][j]` está presente para

todas as subcamadas no  $i$ -ésimo conjunto de camadas de saída. O número máximo de subcamadas é indicado pelo valor de elemento de sintaxe `vps_max_sub_layers_minus1`. Em outras palavras, quando o criptador de vídeo 20 gera uma `sub_layer_flag_info_present_flag[i]` com um valor de 1, o decodificador de vídeo 30 determina que um sinalizador adicional (isto é, `sub_layer_dpb_info_present_flag[i][j]`) será sinalizado para todas as subcamadas do  $i$ -ésimo conjunto de camadas de saída.

[00138] Por outro lado, se o criptador de vídeo 20 gerar elemento de sintaxe `sub_layer_flag_info_present_flag[i]` com um valor igual a 0, o decodificador de vídeo 30 pode determinar que o elemento de sintaxe `sub_layer_flag_info_present_flag[i]` especifica que, para cada valor de  $j$  maior que 0, `sub_layer_dpb_info_present_flag[i][j]` não está presente, e o decodificador de vídeo 30 pode inferir o valor como igual a 0. Desse modo, o criptador de vídeo 20 usaria os valores sinalizados inicialmente dos parâmetros de sub-DPB (isto é, `max_vps_dec_pic_buffering_minus1[i][k][j]`, `max_vps_num_reorder_pics[i][j]`, e `max_vps_latency_increase_plus1[i][j]`) para todas as subcamadas.

[00139] Conforme descrito acima, se o criptador de vídeo 20 gerar a `sub_layer_flag_info_present_flag[i]` com um valor de 1, o criptador de vídeo 20 também gera uma `sub_layer_dpb_info_present_flag[i][j]` para cada subcamada de um conjunto de camadas de saída. Se o valor do elemento de sintaxe `sub_layer_dpb_info_present_flag[i][j]` é igual a 1, então, decodificador de vídeo 30 pode receber adicionalmente o tamanho de sub-DPB (`max_vps_dec_pic_buffering_minus1[i][k][j]`), para subcamada  $j$  para cada sub-DPB  $k$  na faixa de 0 ao número



máximo de sub-DPBs ( $\text{NumSubDpbs}[i]-1$ ), inclusive. O decodificador de vídeo 30 também é configurado para receber elementos de sintaxe que indicam a latência máxima ( $\text{max\_vps\_latency\_increase\_plus1}[i][j]$ ), e o número máximo de figurações de reordenação ( $\text{max\_vps\_num\_reorder\_pics}[i][j]$ ) para a subcamada  $j$ .

[00140] Por outro lado, se decodificador de vídeo 30 determinar que o valor do elemento de sintaxe  $\text{sub\_layer\_dpb\_info\_present\_flag}[i][j]$  é igual a 0, então o decodificador de vídeo 30 pode determinar que o elemento de sintaxe  $\text{sub\_layer\_dpb\_info\_present\_flag}[i][j]$  especifica que os valores de  $\text{max\_vps\_dec\_pic\_buffering\_minus1}[i][k][j]$  são iguais a  $\text{max\_vps\_dec\_pic\_buffering\_minus1}[i][k][j-1]$  (isto é, igual ao valor do elemento de sintaxe de tamanho de DPB para a subcamada anterior no conjunto de camadas de saída) para  $k$  na faixa de 0 a ( $\text{NumSubDpbs}[i]-1$ ), inclusive. Ademais, o decodificador de vídeo 30 pode determinar adicionalmente que os valores do número máximo de elemento de sintaxe de figurações de reordenação ( $\text{max\_vps\_num\_reorder\_pics}[i][j]$ ) e elemento de sintaxe de latência máxima ( $\text{max\_vps\_latency\_increase\_plus1}[i][j]$ ) para a subcamada  $j$  são definidos iguais para serem iguais ao valor daqueles elementos de sintaxe para a subcamada anterior no conjunto de camadas de saída (isto é,  $\text{max\_vps\_num\_reorder\_pics}[i][j-1]$  e  $\text{max\_vps\_latency\_increase\_plus1}[i][j-1]$ , respectivamente). Adicionalmente, o decodificador de vídeo 30 pode inferir o valor do elemento de sintaxe  $\text{sub\_layer\_dpb\_info\_present\_flag}[i][0]$  para qualquer valor possível  $i$  a ser igual a 1. Isto é, o decodificador de vídeo 30 pode ser configurado para inferir um valor de 1 para a  $\text{sub\_layer\_dpb\_info\_present\_flag}$  para 0° (isto é, primeiro) subcamada de um conjunto de camadas de saída.

Adicionalmente, o decodificador de vídeo 30 pode inferir o valor de elemento de sintaxe `sub_layer_dpb_info_present_flag[i][k]` é igual a 0 para  $k$  na faixa de  $j + 1$  para `vps_max_sub_layers_minus1`, inclusive.

[00141] O elemento de sintaxe `max_vps_dec_pic_buffering_minus1 [i][k][j]` mais 1 especifica o tamanho máximo exigido do  $k$ -ésimo sub-DPB para as CVS no  $i$ -ésimo conjunto de camadas de saída nas unidades de armazenamentos temporários de armazenamento de figuração quando `HighestTid` é igual a  $j$ . Quando  $j$  for maior que 0, o decodificador de vídeo 30 pode determinar que `max_vps_dec_pic_buffering_minus1 [i][k][j]` é maior que ou igual ao valor de `max_vps_dec_pic_buffering_minus1 [i][k][j-1]`. Quando `max_vps_dec_pic_buffering_minus1 [i][k][j]` não está presente para  $j$  na faixa de 1 a  $(vps\_max\_sub\_layers\_minus1-1)$ , inclusive, o decodificador de vídeo 30 pode inferir o valor de `max_vps_dec_pic_buffering_minus1 [i][k][j]` para ser igual ao valor de `max_vps_dec_pic_buffering_minus1 [i][k][j-1]`.

[00142] O elemento de sintaxe `max_vps_num_reorder_pics[i][j]` indica o número máximo de figurações de reordenação. Isto é, o número máximo permitido de unidades de acesso que contêm uma figuração com `PicOutputFlag` igual a 1 que pode preceder qualquer unidade de acesso auA que contém uma figuração com `PicOutputFlag` igual a 1 no  $i$ -ésimo conjunto de camadas de saída na CVS na ordem de decodificação e segue a unidade de acesso auA que contém uma figuração com `PicOutputFlag` igual a 1 em ordem de saída, quando `HighestTid` é igual a  $j$ . Quando `max_vps_num_reorder_pics[i][j]` não está presente para  $j$  na faixa de 1 a  $(vps\_max\_sub\_layers\_minus1-1)$ , inclusive, devido a `sub_layer_dpb_info_present_flag[i][j]` ser igual a 0, o decodificador de vídeo 30 pode inferir o

valor do elemento de sintaxe `max_vps_num_reorder_pics[i][j]` como igual ao valor de `max_vps_num_reorder_pics[i][j-1]`.

[00143] O valor de elemento de sintaxe `max_vps_latency_increase_plus1 [i][j]` indica a latência máxima. O elemento de sintaxe `max_vps_latency_increase_plus1 [i][j]` diferente de 0 é usado para computar o valor do elemento de sintaxe `VpsMaxLatencyPictures[i][k][j]`, o valor do que especifica o número máximo de unidades de acesso que contêm uma figuração com PicOutputFlag igual a 1 no *i*-ésimo conjunto de camadas de saída que pode preceder qualquer unidade de acesso auA que contém uma figuração com PicOutputFlag igual a 1 na CVS na ordem de saída e segue a unidade de acesso auA que contém uma figuração com PicOutputFlag igual a 1 em ordem de decodificação quando `HighestTid` é igual a *j*. Em exemplos em que `max_vps_latency_increase_plus1 [i][j]` não está presente para *j* na faixa de 1 para  $(vps\_max\_sub\_layers\_minus1-1)$ , inclusive, devido a `sub_layer_dpb_info_present_flag[i][j]` ser igual a 0, o decodificador de vídeo 30 pode inferir o valor de `max_vps_latency_increase_plus1 [i][j]` como igual a `max_vps_latency_increase_plus1 [i][j-1]`.

[00144] Quando `max_vps_latency_increase_plus1 [i][j]` não é igual a 0, o criptador de vídeo 20 pode especificar o valor de `VpsMaxLatencyPictures[i][k][j]` conforme o seguinte:

$$\begin{aligned} &VpsMaxLatencyPictures[i][j]= \\ &\quad max\_vps\_num\_reorder\_pics[i][j]+ \\ &\quad max\_vps\_latency\_increase\_plus1 [i][j] - 1 \end{aligned}$$

[00145] Isto é, o número máximo de figuração de latência é igual ao número máximo de figurações de reordenação mais o valor do elemento de sintaxe de latência máxima.

[00146] Quando `max_vps_latency_increase_plus1[i][k]` é igual a 0, nenhum limite correspondente é expresso. O valor de `max_vps_latency_increase_plus1[i][j]` deve estar na faixa de 0 a  $(2^{32})-2$ , inclusive.

[00147] O seguinte mostra as alterações à saída e remoção do processo de figurações relativo à subcláusula HEVC F.13.5.2.a fim de implantar as técnicas exemplificativas desta revelação. Novamente, as adições são mostradas em negrito e sublinhado, e as exclusões são mostradas com um ~~tachado~~.

[00148] A saída e remoção de figurações do DPB antes da decodificação da figuração atual pelo decodificador de vídeo 30 (porém, após analisar o cabeçalho de fatia da primeira fatia da figuração atual) ocorre instantaneamente quando a primeira unidade de decodificação da figuração atual é removida do CPB e prossegue conforme o seguinte: O processo de decodificação para RPS, conforme especificado na subcláusula F.8.3.2, é invocado.

[00149] - Se a figuração atual for uma figuração de IRAP com `NoRaslOutputFlag` igual a 1 e com `nuh_layer_id` igual a 0 que não é a figuração 0, o decodificador de vídeo 30 pode aplicar as seguintes etapas ordenadas aplicas:

[00150] 1. O decodificador de vídeo 30 pode derivar a variável `NoOutputOfPriorPicsFlag` para o decodificador em teste conforme o seguinte:

[00151] - Se a figuração atual é uma figuração de CRA, o decodificador de vídeo 30 pode definir a `NoOutputOfPriorPicsFlag` como igual a 1 (independentemente do valor de `no_output_of_prior_pics_flag`).

[00152] - De outro modo, se o valor de `pic_width_in_luma_samples`, de `pic_height_in_luma_samples`, ou de `sps_max_dec_pic_buffering_minus1[HighestTid]` derivado

a partir do SPS ativo for diferente do valor de `pic_width_in_luma_samples`, de `pic_height_in_luma_samples`, ou de `sps_max_dec_pic_buffering_minus1[HighestTid]`, respectivamente, derivado a partir do SPS ativo para a figuração precedente na ordem de decodificação com o mesmo valor de `nuh_layer_id` que a figuração atual, o decodificador de vídeo 30 pode definir `NoOutputOfPriorPicsFlag` para 1 em teste, independentemente do valor de `no_output_of_prior_pics_flag`. No entanto, em exemplos preferidos, o decodificador de vídeo 30 pode não definir `NoOutputOfPriorPicsFlag` para 1 em teste sob as condições especificadas nesse marcador. OBSERVAÇÃO - Embora a configuração de `NoOutputOfPriorPicsFlag` como igual a `no_output_of_prior_pics_flag` possa ser preferida sob essas condições, o decodificador de vídeo 30, em teste, é permitido para definir `NoOutputOfPriorPicsFlag` para 1 sob essas condições.

[00153] - De outro modo, o decodificador de vídeo 30 pode definir `NoOutputOfPriorPicsFlag` como igual a `no_output_of_prior_pics_flag`.

[00154] 2. O decodificador de vídeo 30 pode aplicar o valor de `NoOutputOfPriorPicsFlag` derivado em teste aplicado para o HRD conforme o seguinte:

[00155] - Se `NoOutputOfPriorPicsFlag` for igual a 1, o decodificador de vídeo 30 pode esvaziar todos os armazenamentos temporários de armazenamento de figuração em todos os sub-DPBs no DPB sem a saída das figurações contidas nos armazenamentos temporários de armazenamento de figuração esvaziados, e o decodificador de vídeo 30 pode definir a completude de sub-DPB como igual a 0 para todos os sub-DPBs.

[00156] - De outro modo (isto é, `NoOutputOfPriorPicsFlag` é igual a 0), o decodificador de

vídeo 30 pode esvaziar todos os armazenamentos temporários de armazenamento de figuração que contém uma figuração que é marcada como "não necessário para saída" e "não usado para referência" sem a saída, e o decodificador de vídeo 30 pode esvaziar todos os armazenamentos temporários de armazenamento de figuração não vazios no DPB invocando-se repetidamente o processo de "aumento" especificado na subcláusula F.13.5.2.4, e o decodificador de vídeo 30 pode definir a completude de sub-DPB como igual a 0.

[00157] - De outro modo (por exemplo, se a figuração atual não for uma figuração de IRAP com NoRaslOutputFlag igual a 1 ou com nuh\_layer\_id diferente de 0), o decodificador de vídeo 30 pode esvaziar todos os armazenamentos temporários de armazenamento de figuração que contém uma figuração que é marcada como "não necessário para saída" e "não usado para referência" sem saída. Para cada armazenamento temporário de armazenamento de figuração que é esvaziado, o decodificador de vídeo 30 pode diminuir a completude de sub-DPB associada a tal armazenamento temporário de armazenamento de figuração em um. A variável currLayerId é definida como igual a nuh\_layer\_id da figuração atual decodificada, a variável currSubDpbId é definida como igual ao índice do sub-DPB associado à figuração atual decodificada, e quando um ou mais dentre as seguintes condições forem verdadeiras, o processo de "aumento" especificado na subcláusula F.13.5.2.4 é invocado repetidamente enquanto diminui adicionalmente a completude de DPB em um para cada armazenamento temporário de armazenamento de figuração adicional que é esvaziado, até que nenhuma das seguintes condições seja verdadeira:

[00158] - O número de unidades de acesso que contêm pelo menos uma figuração ~~figurações com nuh\_layer\_id igual a currLayerId no sub-DPB que é associado à figuração~~

~~atual~~, que ~~são~~ é marcado como "necessário para saída" é maior que `sps_max_num_reorder_pics[HighestTid]` do SPS ativo (quando uma sequência de vídeo codificada em conformidade com um ou mais dentre os perfis especificados no Anexo A é decodificada aplicando-se o processo de decodificação especificado nas cláusulas 2 a 10) ou `max_vps_num_reorder_pics[TargetOutputLayerId][LayerIdxInVps[currLayerId]][HighestTid]` do VPS de camada ativa para o valor de `currLayerId` (quando uma sequência de vídeo codificada em conformidade com um ou mais dentre os perfis especificados no Anexo G ou H é decodificada aplicando-se o processo de decodificação especificado nas cláusulas 2 a 10, Anexo F, e Anexo G ou H).

[00159] - `sps_max_latency_increase_plus1[HighestTid]` do SPS ativo (quando uma sequência de vídeo codificada em conformidade com um ou mais dentre os perfis especificados no Anexo A é decodificada aplicando-se o processo de decodificação especificado nas cláusulas 2 a 10) ou `VpsMaxLatencyPictures[TargetOutputLayerId][LayerIdxInVps[currLayerId]][HighestTid]` do VPS ativo (quando uma sequência de vídeo codificada em conformidade com um ou mais dentre os perfis especificados no Anexo G ou H é decodificada aplicando-se o processo de decodificação especificado nas cláusulas 2 a 10, Anexo F, e Anexo G ou H) é diferente de 0 e há pelo menos uma unidade de acesso que contém uma figuração que é marcada como "necessária para saída" figuração com `nuh_layer_id` igual a `currLayerId` no DPB que é marcado como "necessário para saída" para o qual a variável associada `PicLatencyCount[currLayerId]` é maior que ou igual a `SpsMaxLatencyPictures[HighestTid]` derivada a partir do SPS ativo (quando uma sequência de vídeo codificada em conformidade com um ou mais dentre os perfis especificados

no Anexo A é decodificada aplicando-se o processo de decodificação especificado nas cláusulas 2 a 10) ou  $VpsMaxLatencyPictures[TargetOutputLayerId][LayerId \times InvVps[currLayerId]][HighestTid]$  a partir do VPS ativo para o valor de currLayerId (quando uma sequência de vídeo codificada em conformidade com um ou mais dentre os perfis especificados no Anexo G ou H é decodificada aplicando-se o processo de decodificação especificado nas cláusulas 2 a 10, Anexo F, e Anexo G ou H).

[00160] - O número de figurações com nuh\_layer\_id igual a currLayerId no sub-DPB associado é maior que ou igual a  $sps\_max\_dec\_pic\_buffering\_minus1[HighestTid] + 1$  do SPS ativo (quando uma sequência de vídeo codificada em conformidade com um ou mais dentre os perfis especificados no Anexo A é decodificada aplicando-se o processo de decodificação especificado nas cláusulas 2 a 10) ou  $max\_vps\_dec\_pic\_buffering\_minus1[TargetOutputLayerId][currSubDpbld][HighestTid] + 1$  (quando uma sequência de vídeo codificada em conformidade com um ou mais dentre os perfis especificados no Anexo G ou H é decodificada aplicando-se o processo de decodificação especificado nas cláusulas 2 a 10, Anexo F, e Anexo G ou H).

[00161] Conforme mostrado acima nas adições e exclusões marcadas, o processo de saída e de remoção de figuração (também chamado "aumento") é atualizado para refletir as técnicas de sinalização de parâmetro de sub-DPB desta revelação. Uma alteração é que as condições para continuar o processo de aumento têm como base as unidades de acesso, em vez de as figurações. Isso é devido ao fato de que, de acordo com as técnicas desta revelação, os parâmetros de sub-DPB se aplicam a uma camada de saída. Também nesse sentido, a necessidade de verificar os valores



dos parâmetros de sub-DPB para IDs de camada particular (por exemplo, `LayerIdxInVps` e `currLayerId`) é removida, visto que todas as camadas de um conjunto de camadas de saída particular compartilham os mesmos parâmetros de sub-DPB.

[00162] O seguinte mostra alterações nos processos de decodificação de figuração, de marcação, de aumento adicional e de armazenamento em relação à subcláusula C.1.1.1.1 de HEVC a fim de implantar as técnicas exemplificativas desta revelação. Novamente, as adições são mostradas em negrito e sublinhado, e as exclusões são mostradas com um ~~tachado~~.

[00163] Os processos especificados nessa subcláusula ocorrem instantaneamente quando o criptador de vídeo 20 e/ou decodificador de vídeo 30 remove a última unidade de decodificação de unidade de acesso  $n$  que contém a figuração atual a partir do CPB. O criptador de vídeo 20 e/ou o decodificador de vídeo 30 podem definir a variável `currLayerId` como igual a `nuh_layer_id` da figuração decodificada atual, e pode definir `currSubDpbId` como igual ao índice do sub-DPB associado à camada com a qual figuração atual está associada. As alterações introduzidas pelas técnicas desta revelação para a processos de decodificação de figuração especificado, de marcação, de aumento e de armazenamento adicionais são denotadas abaixo com o uso de texto sublinhado e tachado.

[00164] Para a unidade de acesso que contém uma figuração no sub-DPB associado que é marcado como "necessário para saída", o criptador de vídeo 20 e/ou o decodificador de vídeo 30 pode definir a variável associada `PicLatencyCount` como igual a `PicLatencyCount + 1`.

[00165] O decodificador de vídeo 30 pode considerar a figuração atual como decodificada após a

última unidade de decodificação da figuração ser decodificada. A figuração atual decodificada é armazenada em um armazenamento temporário de armazenamento de figuração vazio no DPB, e o seguinte se aplica:

[00166] - Se a figuração atual decodificada tem PicOutputFlag como igual a 1, o decodificador de vídeo 30 pode marcar a figuração atual decodificada como "necessário para saída" e a variável PicLatencyCount associada à unidade de acesso atual é definida como igual a 0.

[00167] - De outro modo (por exemplo, a figuração atual decodificada tem PicOutputFlag como igual a 0), o decodificador de vídeo 30 pode marcar a figuração atual decodificada como "não necessário para saída".

[00168] Adicionalmente, o decodificador de vídeo 30 pode marcar a figuração atual decodificada como "usada para referência em curto prazo".

[00169] Quando um ou mais dentre as seguintes condições são verdadeiras, o decodificador de vídeo 30 pode invocar repetida/iterativamente o processo de "aumento" especificado na subcláusula F.13.5.2.4, até que nenhuma das seguintes condições seja verdadeira:

[00170] - O número de unidades de acesso que contêm pelo menos uma figuração que é marcada como "necessário para saída" é maior que sps\_max\_num\_reorder\_pics[HighestTid] do SPS ativo quando uma sequência de vídeo codificada em conformidade com um ou mais dentre os perfis especificados no Anexo A é decodificada aplicando-se o processo de decodificação especificado nas cláusulas 2 a 10) ou max\_vps\_num\_reorder\_pics[TargetOutputLayerId][HighestTid] do VPS ativo para o valor de currLayerId (quando uma sequência de vídeo codificada em conformidade com um ou mais dentre os perfis especificados no Anexo G ou H é

decodificada aplicando-se o processo de decodificação especificado nas cláusulas 2 a 10, Anexo F, e Anexo G ou H).

[00171] - sps\_max\_latency\_increase\_plus1 [HighestTid] é diferente de 0 (quando uma sequência de vídeo codificada em conformidade com um ou mais dentre os perfis especificados no Anexo A é decodificada aplicando-se o processo de decodificação especificado nas cláusulas 2 a 10) ou VpsMaxLatencyPictures[TargetOutputLayerId][HighestTid] o VPS ativo (quando uma sequência de vídeo codificada em conformidade com um ou mais dentre os perfis especificados no Anexo G ou H é decodificada aplicando-se o processo de decodificação especificado nas cláusulas 2 a 10, Anexo F, e Anexo G ou H) é diferente de 0 e há pelo menos uma unidade de acesso que contém uma figuração marcada como "necessária para saída" para a qual a variável associada PicLatencyCount que é maior que ou igual a SpsMaxLatencyPictures[HighestTid] derivado a partir do SPS ativo (quando uma sequência de vídeo codificada em conformidade com um ou mais dentre os perfis especificados no Anexo A é decodificada aplicando-se o processo de decodificação especificado nas cláusulas 2 a 10) ou VpsMaxLatencyPictures[TargetOutputLayerId] [HighestTid] a partir do VPS ativo para o valor de currLayerId (quando uma sequência de vídeo codificada em conformidade com um ou mais dentre os perfis especificados no Anexo G ou H é decodificada aplicando-se o processo de decodificação especificado nas cláusulas 2 a 10, Anexo F, e Anexo G ou H).

[00172] Conforme mostrado acima nas adições e exclusões marcadas, o processo de saída e de remoção de figuração (também chamado "aumento") é atualizado para

refletir as técnicas de sinalização de parâmetro de sub-DPB desta revelação. Uma alteração é que as condições para continuar o processo de aumento têm como base as unidades de acesso, em vez de as figurações. Isso é devido ao fato de que, de acordo com as técnicas desta revelação, os parâmetros de sub-DPB se aplicam a um conjunto de camadas de saída completo e não apenas a um subconjunto das camadas.

[00173] O seguinte é uma descrição de uma segunda implantação exemplificativa em relação à primeira implantação exemplificativa descrita acima. Em contraste à primeira implantação exemplificativa, de acordo com a segunda implantação exemplificativa, o criptador de vídeo 20 pode sinalizar uma ID de camada que especifica a camada com a qual os valores de `max_vps_num_reorder_pics[i][j]` e `max_vps_latency_increase_plus1 [i][j]` estão associados. As alterações à sintaxe e às semânticas introduzidas pela segunda implantação exemplificativa, são distinguidas abaixo com o uso de formatação em negrito e sublinhado e ~~tachado~~, conforme foi mostrado acima. A sintaxe e as semânticas dos outros elementos de sintaxe (isto é, denotados pelo texto que não é sublinhado nem tachado) permanecem as mesmas, de acordo com a segunda implantação exemplificativa

TABELA 5 - SINTAXE DE TABELA DE TAMANHO DE DPB

dpb_size_table() {	
for( i = 1; i < NumOutputLayerSets; i++ ) {	
<b><u>sub_layer_flag_info_present_flag[i]</u></b>	u(1)
for( j = 0; j <= vps_max_sub_layers_minus1; j++ ) {	
if( j > 0 && sub_layer_flag_info_present_flag[i] &&	
sub_layer_dpb_info_present_flag[i][j-1] )	
<b><u>sub_layer_dpb_info_present_flag[i][j]</u></b>	u(1)
if( sub_layer_dpb_info_present_flag[i][j] ) {	
for( k = 0; k < NumSubDpbs[i]; k++ )	
max_vps_dec_pic_buffering_minus1[i][k][j]	ue(v)
<b><u>reorder_info_layer_id[i][j]</u></b>	
max_vps_num_reorder_pics[i][j]	ue(v)
max_vps_latency_increase_plus1[i][j]	ue(v)
}	
}	
}	
}	

[00174] A Tabela 5 acima descreve alterações de sintaxe introduzidas pela segunda implantação exemplificativa em relação à primeira implantação exemplificativa. As alterações nas semânticas, conforme introduzidas pela segunda implantação exemplificativa, são descritas abaixo.

[00175] O decodificador de vídeo 30 pode determinar que o elemento de sintaxe `reorder_info_layer_id[i][j]` especifica o valor de `nuh_layer_id` da camada que pertence ao *i*-ésimo conjunto de camadas de saída ao qual os valores de `max_vps_num_reorder_pics[i][j]` e `max_vps_latency_increase_plus1[i][j]` se aplicam. Em exemplos em que o elemento de sintaxe `reorder_info_layer_id[i][j]` não está presente, o

decodificador de vídeo 30 pode inferir o valor de `reorder_info_layer_id[i][j]` como igual a 0. Alternativamente, em alguns exemplos, quando `reorder_info_layer_id[i][j]` não está presente, o decodificador de vídeo 30 pode inferir o valor do `reorder_info_layer_id[i][j]` como igual ao `nuh_layer_id` da camada mais alta no *i*-ésimo conjunto de camadas de saída. Alternativamente, em alguns exemplos, o criptador de vídeo 20 pode sinalizar um sinalizador que especifica para o decodificador de vídeo 30 a possibilidade do elemento de sintaxe `reorder_info_layer_id[i][j]` ser sinalizado.

[00176] O decodificador de vídeo 30 pode determinar que o elemento de sintaxe `max_vps_num_reorder_pics[i][j]` indica o número máximo permitido de figurações com `nuh_layer_id` igual a `reorder_info_layer_id[i][j]` que pode preceder qualquer figuração com `nuh_layer_id` igual a `reorder_info_layer_id[i][j]` no *i*-ésimo conjunto de camadas de saída na CVS na ordem de decodificação, e segue tal figuração com `nuh_layer_id` igual a `reorder_info_layer_id[i][j]` em ordem de saída, e quando `HighestTid` é igual a *j*. Quando `max_vps_num_reorder_pics[i][j]` não está presente para *j* na faixa de 1 a (`vps_max_sub_layers_minus1`-1), inclusive, devido a `sub_layer_dpb_info_present_flag[i][j]` ser igual a 0, o decodificador de vídeo 30 pode inferir o valor do elemento de sintaxe `max_vps_num_reorder_pics[i][j]` como igual a `max_vps_num_reorder_pics[i][j-1]`.

[00177] Em exemplos em que o valor de `max_vps_latency_increase_plus1[i][j]` não é igual a 0, decodificador de vídeo 30 pode usar o valor do elemento de sintaxe `max_vps_latency_increase_plus1 [i][j]` para computar o valor de `VpsMaxLatencyPictures[i][k][j]`. O valor do

elemento de sintaxe `VpsMaxLatencyPictures[i][k][j]`, por sua vez, pode especificar o número máximo de figurações com `nuh_layer_id` igual a `reorder_info_layer_id[i][j]` no *i*-ésimo conjunto de camadas de saída que pode preceder qualquer figuração com `nuh_layer_id` igual a `reorder_info_layer_id[i][j]` na CVS na ordem de saída e segue tal figuração com `nuh_layer_id` igual a `reorder_info_layer_id[i][j]` na ordem de decodificação quando `HighestTid` é igual a *j*. Em exemplos em que `max_vps_latency_increase_plus1 [i][j]` não está presente para *j* na faixa de 1 para (`vps_max_sub_layers_minus1-1`), inclusive, devido ao valor de `sub_layer_dpb_info_present_flag[i][j]` ser igual a 0, o decodificador de vídeo 30 pode inferir o valor de `max_vps_latency_increase_plus1 [i][j]` como igual a `max_vps_latency_increase_plus1 [i][j-1]`.

[00178] Em exemplos em que `max_vps_latency_increase_plus1 [i][j]` não é igual a 0, o valor de `VpsMaxLatencyPictures[i][k][j]` é especificado por uma fórmula conforme o seguinte:

$$\begin{aligned} \text{VpsMaxLatencyPictures}[i][j] = \\ \text{max\_vps\_num\_reorder\_pics}[i][j] + \\ \text{max\_vps\_latency\_increase\_plus1}[i][j] - 1 \end{aligned}$$

[00179] Por outro lado, em exemplos em que `max_vps_latency_increase_plus1 [i][j]` é igual a 0, nenhum limite correspondente é expresso. Nesses exemplos, o criptador de vídeo 20 e/ou decodificador de vídeo 30 pode determinar que o valor de `max_vps_latency_increase_plus1 [i][j]` está na faixa de 0 a (`2A32-2`), inclusive.

[00180] O seguinte descreve uma terceira implantação exemplificativa em relação à primeira implantação exemplificativa. No entanto, em contraste com a primeira implantação exemplificativa, de acordo com a

terceira implantação exemplificativa, o criptador de vídeo 20 pode sinalizar um sinalizador para especificar a possibilidade do número máximo de figurações de reordenação e da latência máxima serem sinalizados para cada camada no conjunto de camadas de saída. Por exemplo, de acordo com a terceira implantação exemplificativa, o criptador de vídeo 20 pode definir o sinalizador a um primeiro valor para indicar que o número máximo de figurações de reordenação e da latência máxima são sinalizados para cada camada no conjunto de camadas de saída, e para um segundo valor para indicar que o número máximo de figurações de reordenação e da latência máxima não são sinalizados para cada camada no conjunto de camadas de saída. As alterações introduzidas pela terceira implantação exemplificativa, em comparação à sintaxe fornecida pela primeira implantação exemplificativa, são distinguidos pelo sublinhado e negrito na Tabela 6 abaixo.

TABELA 6 - SINTAXE DE TABELA DE TAMANHO DE DPB



dpb_size_table() {	
for( i = 1; i < NumOutputLayerSets; i++ ) {	
<b><u>sub_layer_flag_info_present_flag[ i ]</u></b>	u(1)
for( j = 0; j <= vps_max_sub_layers_minus1; j++ ) {	
if( j > 0 && sub_layer_flag_info_present_flag[ i ] && sub_layer_dpb_info_present_flag[ i ][ j - 1 ] )	
<b><u>sub_layer_dpb_info_present_flag[ i ][ j ]</u></b>	u(1)
if( sub_layer_dpb_info_present_flag[ i ][ j ] ) {	
for( k = 0; k < NumSubDpbs[ i ]; k++ )	
<b><u>max_vps_dec_pic_buffering_minus1[ i ][ k ][ j ]</u></b>	ue(v)
<b><u>reorder_info_only_highest_layer_flag[ i ][ j ]</u></b>	u(1)
for( k = reorder_info_only_highest_layer_flag[ i ][ j ] ? NumLayersInIdList[ output_layer_set_idx_minus1[ i ] + 1 ] - 1 : 0; k <= NumLayersInIdList[ output_layer_set_idx_minus1[ i ] + 1 ] - 1; k++ ) {	
<b><u>max_vps_num_reorder_pics[ i ][ k ][ j ]</u></b>	ue(v)
<b><u>max_vps_latency_increase_plus1[ i ][ k ][ j ]</u></b>	ue(v)
}	
}	
}	
}	
}	

[00181] Novas semânticas introduzidas pela terceira implantação exemplificativa em relação à tabela de tamanho de DPB da Tabela 6, em comparação à primeira implantação exemplificativa, são descritas abaixo. Quaisquer semânticas que não são descritas abaixo em relação à terceira implantação exemplificativa permanecem a mesma que na primeira implantação exemplificativa.

[00182] Em exemplos em que criptador de vídeo 20 define que sub\_layer\_flag\_info\_present\_flag[i] igual a

1, o `sub_layer_flag_info_present_flag[i]` especifica que `sub_layer_dpb_info_present_flag[i][j]` está presente para subcamadas `vps_max_sub_layers_minus1`). Por outro lado, em exemplos em que criptador de vídeo 20 define `sub_layer_flag_info_present_flag[i]` igual a 0, o `sub_layer_flag_info_present_flag[i]` especifica que, para cada valor de `j` maior que 0, `sub_layer_dpb_info_present_flag[i][j]` não está presente. Em alguns exemplos em que o criptador de vídeo 20 define o `sub_layer_flag_info_present_flag[i]` para 0, decodificador de vídeo 30 pode inferir o valor do `sub_layer_dpb_info_present_flag[i][j]` como igual a 0.

[00183] Em exemplos em que o criptador de vídeo define `sub_layer_dpb_info_present_flag[i][j]` igual a 1, o `sub_layer_dpb_info_present_flag[i][j]` especifica o decodificador de vídeo 30 que `max_vps_dec_pic_buffering_minus1 [i][k][j]`, para `k` na faixa de 0 a `(NumSubDpbs[i]-1)`, inclusive, e `reorder_info_only_highest_layer_flag[i][j]` estão presentes para a `j`-ésima subcamada. Por outro lado, em exemplos em que o criptador de vídeo 20 define `sub_layer_dpb_info_present_flag[i][j]` igual a 0, `sub_layer_dpb_info_present_flag[i][j]` especifica ao decodificador de vídeo 30 que os valores de `max_vps_dec_pic_buffering_minus1 [i][k][j]` são iguais a `max_vps_dec_pic_buffering_minus1 [i][k][j - 1]` para `k` na faixa de 0 a `(NumSubDpbs[i]- 1)`, inclusive, e que os valores `max_vps_num_reorder_pics[i][k][j]` e `max_vps_latency_increase_plus1 [i][k][j]` são definidos iguais a `max_vps_num_reorder_pics[i][k][j - 1]` e `max_vps_latency_increase_plus1 [i][k][j - 1]`, respectivamente, para `k` na faixa de 0 a `(NumLayersInIdList[output_layer_set_idx_minus1 [i]+ 1]- 1)`,

inclusive. O decodificador de vídeo 30 pode inferir o valor de `sub_layer_dpb_info_present_flag[i][0]` para qualquer valor possível de `i` como igual a 1.

[00184] O valor de `max_vps_dec_pic_buffering_minus1[i][k][j]` mais 1 especifica o tamanho máximo exigido do `k`-ésimo sub-DPB para as CVS no `i`-ésimo conjunto de camadas de saída nas unidades de armazenamentos temporários de armazenamento de figuração quando `HighestTid` é igual a `j`. Quando `j` for maior que 0, `max_vps_dec_pic_buffering_minus1[i][k][j]` deve ser maior que ou igual a `max_vps_dec_pic_buffering_minus1[i][k][j-1]`. Quando `max_vps_dec_pic_buffering_minus1[i][k][j]` não está presente para `j` na faixa de 1 a  $(\text{vps\_max\_sub\_layers\_minus1} - 1)$ , inclusive, o decodificador de vídeo 30 pode inferir `max_vps_dec_pic_buffering_minus1[i][k][j]` como igual a `max_vps_dec_pic_buffering_minus1[i][k][j-1]`.

[00185] Em exemplos em que `reorder_info_only_highest_layer_flag[i][j]` é igual a 1, `reorder_info_only_highest_layer_flag[i][j]` indica ao decodificador de vídeo 30 que os elementos de sintaxe `max_vps_num_reorder_pics[i][NumLayersInIdList[output_layer_set_idx_minus1[i]+1]-1][j]` e `max_vps_latency_increase_plus1[i][NumLayersInIdList[output_layer_set_idx_minus1[i]+1]-1][j]` estão presentes e `max_vps_num_reorder_pics[i][k][j]` e `max_vps_latency_increase_plus1[i][k][j]`, para `k` na faixa de 0 a  $(\text{NumLayersInIdList[output\_layer\_set\_idx\_minus1}[i]+1]-2)$ , inclusive, são inferidas (por exemplo, pelo decodificador de vídeo 30) como iguais a `max_vps_num_reorder_pics[i][NumLayersInIdList[output_layer_set_idx_minus1[i]+1]-1][j]` e `max_vps_latency_increase_plus1`

[i][NumLayersInIdList[output\_layer\_set\_idx\_minus 1 [i]+ 1]- 1][j], respectivamente. Por outro lado, em exemplos em que reorder\_info\_only\_highest\_layer\_flag[i][j] é igual a 0, o reorder\_info\_only\_highest\_layer\_flag[i][j] indica ao decodificador de vídeo 30 que os elementos de sintaxe max\_vps\_num\_reorder\_pics[i][k][j] e max\_vps\_latency\_increase\_plus1 [i][k][j] são sinalizados (por exemplo, pelo criptador de vídeo 20) para k na faixa de 0 a (NumLayersInIdList[output\_layer\_set\_idx\_minus1 [i]+ 1]- 1), inclusive.

[00186] O elemento de sintaxe max\_vps\_num\_reorder\_pics[i][k][j] indica (por exemplo, ao decodificador de vídeo 30) o número máximo permitido de figurações com nuh\_layer\_id igual a layer\_id\_in\_nuh[k] que pode preceder qualquer figuração com nuh\_layer\_id igual a layer\_id\_in\_nuh[k] no i-ésimo conjunto de camadas de saída na CVS na ordem de decodificação e segue tal figuração com nuh\_layer\_id igual a layer\_id\_in\_nuh[k] na ordem de saída, e quando HighestTid é igual a j. Quando max\_vps\_num\_reorder\_pics[i][k][j] não está presente para j na faixa de 1 para (vps\_max\_sub\_layers\_minus1 - 1), inclusive, devido ao sub\_layer\_dpb\_info\_present\_flag[i][j] ser igual a 0, infere-se (por exemplo, pelo decodificador de vídeo 30) como igual a max\_vps\_num\_reorder\_pics[i][k][j - 1]. Por outro lado, em exemplos em que max\_vps\_num\_reorder\_pics[i][k][j] não está presente para k na faixa de 0 a (NumLayersInIdList[output\_layer\_set\_idx\_minus1 [i]+ 1]- 2), inclusive, devido a sub\_layer\_dpb\_info\_present\_flag[i][j] ser igual a 1 e reorder\_info\_only\_highest\_layer\_flag[i][j] igual a 0, decodificador de vídeo 30 pode inferir max\_vps\_num\_reorder\_pics[i][k][j] como igual a max\_vps\_num\_reorder\_pics[i][NumLayersInIdList[output\_layer\_

set\_idx\_minus1 [i] + i] - i][j - i].

[00187] Em exemplos em que max\_vps\_latency\_increase\_plus1[i][k][j] não é igual a 0, o decodificador de vídeo 30 pode usar max\_vps\_latency\_increase\_plus1 [i][k][j] para computar o valor de VpsMaxLatencyPictures[i][k][j], que especifica o número máximo de figurações com nuh\_layer\_id igual a layer\_id\_in\_nuh[k] no i-ésimo conjunto de camadas de saída que pode preceder qualquer figuração com nuh\_layer\_id igual a layer\_id\_in\_nuh[k] na CVS na ordem de saída e segue tal figuração com nuh\_layer\_id igual a layer\_id\_in\_nuh[k] na ordem de decodificação quando HighestTid é igual a j. Em exemplos em que max\_vps\_latency\_increase\_plus1 [i][k][j] não estiver presente para j na faixa de 1 a (vps\_max\_sub\_layers\_minus1 - 1), inclusive, devido a sub\_layer\_dpb\_info\_present\_flag[i][j] como igual a 0, decodificador de vídeo 30 pode inferir max\_vps\_latency\_increase\_plus1 [i][k][j] como igual a max\_vps\_latency\_increase\_plus1 [i][k][j-1]. Em exemplos em que max\_vps\_latency\_increase\_plus1 [i][k][j] não está presente para k na faixa de 0 a (NumLayersInIdList[output\_layer\_set\_idx\_minus1 [i] + 1] - 2), inclusive, devido a sub\_layer\_dpb\_info\_present\_flag[i][j] ser igual a 1 e reorder\_info\_only\_highest\_layer\_flag[i][j] igual a 0, decodificador de vídeo 30 pode inferir max\_vps\_latency\_increase\_plus1 [i][k][j] como igual a max\_vps\_latency\_increase\_plus1 [i][NumLayersInIdList[output\_layer\_set\_idx\_minus1 [i] + 1] - 1][j].

[00188] Em exemplos em que max\_vps\_latency\_increase\_plus1 [i][k][j] não é igual a 0, o valor de VpsMaxLatencyPictures[i][k][j] é especificado por uma fórmula conforme o seguinte:

$$\begin{aligned} & \text{VpsMaxLatencyPictures}[i][k][j] = \\ & \text{max\_vps\_num\_reorder\_pics}[i][k][j] + \\ & \text{max\_vps\_latency\_increase\_plus1}[i][k][j] - 1 \end{aligned}$$

[00189] Em exemplos em que  $\text{max\_vps\_latency\_increase\_plus1}[i][k][j]$  é igual a 0, nenhum limite correspondente é expresso. O valor de  $\text{max\_vps\_latency\_increase\_plus1}[i][k][j]$ , em tais exemplos, deve estar na faixa de 0 a  $(2^{\wedge}32 - 2)$ , inclusive.

[00190] A Figura 6 é um diagrama de blocos que ilustra um criptador de vídeo exemplificativo 20 que pode implantar as técnicas para sinalização de parâmetro de sub-DPB descritas nesta revelação. O criptador de vídeo 20 pode ser configurado para criptar dados de vídeo de acordo com quaisquer técnicas de criptação de vídeo, que inclui HEVC e H.264/AVC, bem como extensão escalonável, de múltiplas vistas e 3D de tais padrões. O exemplo da Figura 6 será explicada com referência a HEVC. Nesse sentido, o circuito de criptação de vídeo mostrado na Figura 6 pode ser aplicado a cada camada de um processo de criptação de vídeo (isto é, camadas-base e camadas de aprimoramento), para cada vista de um processo de codificação de vídeo de múltiplas vistas, ou tanto para vistas de textura quanto de profundidade de um processo de codificação de vídeo 3D.

[00191] O criptador de vídeo 20 pode realizar intra e intercodificação de blocos de vídeo dentro de fatias de vídeo. A intracodificação tem base na previsão espacial para reduzir ou remover a redundância espacial no vídeo dentro de um quadro de vídeo ou figuração dados. A intercodificação tem base na previsão temporal para reduzir ou remover a redundância temporal no vídeo dentro de quadros ou figurações adjacentes de uma sequência de vídeo. O intramodo (modo I) pode se referir a qualquer um dentre vários modos de compressão com base em espaço. Os

intermodos, como previsão unidirecional (modo P) ou biprevisão (modo B), podem se referir a qualquer um dentre vários modos de compressão com base em tempo. Além disso, o criptador de vídeo 20 pode realizar a previsão entre vistas e/ou previsão intercamada entre vistas ou camadas diferentes, conforme descrito acima.

[00192] No exemplo da Figura 6, criptador de vídeo 20 inclui memória de dados de vídeo 40, unidade de processamento de previsão 41, DPB 64, somador 50, unidade de processamento de transformada 52, unidade de quantização 54 e unidade de criptação por entropia 56. A unidade de processamento de previsão 41 inclui unidade de estimativa de movimento e disparidade 42 uma unidade de compensação de movimento e de disparidade 44 e uma unidade de processamento de intraprevisão 46. Para o bloco de vídeo reconstrução, o criptador de vídeo 20 também inclui a unidade de quantização inversa 58, a unidade de processamento de transformada inversa 60 e o somador 62. Um filtro de desbloqueio (não mostrado na Figura 6) também pode ser incluído para filtrar os limites do bloco para remover artefatos de característica de bloqueio provenientes de vídeo reconstruído. Caso desejado, o filtro de desbloqueio pode filtrar tipicamente o resultado do somador 62. Filtros de circuito adicionais (em laço ou pós-laço) também podem ser usados adicionalmente ao filtro de desbloqueio.

[00193] A memória de dados de vídeo 40 pode armazenar dados de vídeo a serem criptados pelos componentes de criptador de vídeo 20. Os dados de vídeo armazenados na memória de dados de vídeo 40 podem ser obtidos, por exemplo, a partir da fonte de vídeo 18. DPB 64 é um exemplo de um armazenamento temporário de figuração de decodificação (DPB) que armazena dados de vídeo de

referência para o uso em dados de vídeo de criptação pelo criptador de vídeo 20 (por exemplo, em modos de intra e intercodificação, também chamados de modos de codificação de intra e interprevisão). A memória de dados de vídeo 40 e DPB 64 podem ser formados por qualquer um dentre uma variedade de dispositivos de memória, tal como memória dinâmica de acesso aleatório (DRAM), que inclui DRAM síncrona (SDRAM), RAM magnetorresistiva (MRAM), RAM resistiva (RRAM) ou outros tipos de dispositivos de memória. A memória de dados de vídeo 40 e DPB 64 podem ser fornecidos pelo menos dispositivo de memória ou dispositivos de memória separados. Em vários exemplos, a memória de dados de vídeo 40 pode ser no chip com outros componentes de criptador de vídeo 20, ou fora do chip em relação àqueles componentes.

[00194] Conforme mostrado na Figura 6, o criptador de vídeo 20 recebe dados de vídeo e pode ser configurado para particionar os dados em blocos de vídeo. Essa partição também pode incluir a partição em fatias, em cascata, ou outras unidades maiores, bem como partição de bloco de vídeo, por exemplo, de acordo com uma estrutura de árvore do tipo quadtree de LCUs e CUs. O criptador de vídeo 20 ilustra, de modo geral, os componentes que criptam blocos de vídeo dentro de uma fatia de vídeo a ser criptada. A fatia pode ser dividida em múltiplos blocos de vídeo (e possivelmente em conjuntos de blocos de vídeo também chamados de cascata). A unidade de processamento de previsão 41 também pode selecionar um dentre uma pluralidade de modos de codificação possíveis, tal como um dentre uma pluralidade de modos de intracodificação ou um dentre uma pluralidade de modos de intercodificação ou modos de codificação entre vistas, para o bloco de vídeo atual com base nos resultados de erro (por exemplo, taxa de



codificação e o nível de distorção). A unidade de processamento de previsão 41 pode fornecer o bloco intra ou intercodificado resultante para o somador 50 para gerar os dados de bloco residual e para o somador 62 para reconstruir o bloco criptado para o uso como uma figuração de referência.

[00195] A unidade de processamento de intraprevisão 46 dentro da unidade de processamento de previsão 41 pode realizar a codificação intrapreditiva do bloco de vídeo atual em relação a um ou mais blocos próximos no mesmo quadro ou fatia como o bloco atual a ser codificado para fornecer a compressão espacial. A unidade de estimativa de movimento e de disparidade 42 e unidade de compensação de movimento e de disparidade 44 dentro da unidade de processamento de previsão 41 realiza a codificação interpreditiva e/ou codificação entre vistas do bloco de vídeo atual em relação a um ou mais blocos preditivos em uma ou mais figurações de referência, camadas de figuração de referência e/ou vistas de referência para fornecer previsão temporal e entre vistas.

[00196] A unidade de estimativa de movimento e de disparidade 42 pode ser configurada para determinar o modo de interprevisão e/ou modo de previsão entre vistas para uma fatia de vídeo de acordo com uma padronização predeterminada para uma sequência de vídeo. A padronização predeterminada pode designar fatias de vídeo na sequência como fatias P ou fatias B. A unidade de estimativa de movimento e de disparidade 42 e a unidade de compensação de movimento e de disparidade 44 podem ser altamente integradas, mas são ilustradas separadamente para propósitos de conceito. A estimativa de movimento, realizada pela unidade de estimativa de movimento e de disparidade 42, é o processo de geração de vetores de

movimento, que estima o movimento para blocos de vídeo. Um vetor de movimento, por exemplo, pode indicar que o deslocamento de uma PU de um bloco de vídeo dentro de um quadro ou figuração de vídeo atual em relação a um bloco preditivo dentro de uma figuração de referência. A estimativa de disparidade, realizada pela unidade de estimativa de movimento e de disparidade 42, é o processo de gerar vetores de disparidade, que pode ser usado para prever um bloco codificado atualmente a partir de um bloco em uma vista diferente.

[00197] Um bloco preditivo é um bloco que se encontra em correspondência muito próxima à PU do bloco de vídeo a ser codificado, em termos de diferença de pixels, que pode ser determinado pelo somatório da diferença absoluta (SAD), somatório da diferença quadrada (SSD) ou outras medidas de diferença. Em alguns exemplos, o criptador de vídeo pode calcular valores para posições de pixel subinteiro de figurações de referência armazenadas no DPB 64. Por exemplo, o criptador de vídeo 20 pode interpolar valores de posições de um quarto de pixel, posições de um oitavo de pixel ou outras posições fracionadas de pixel da figuração de referência. Portanto, a unidade de estimativa de movimento 42 pode realizar uma busca de movimento em relação a todas as posições de pixel e posições fracionadas de pixel e emitir um vetor de movimento com precisão fracionada de pixel.

[00198] A unidade de estimativa de movimento e disparidade 42 calcula um vetor de movimento (para previsão de movimento compensado) e/ou um vetor de disparidade (para previsão de disparidade compensada) para uma PU de um bloco de vídeo em uma fatia intercodificada ou entre vistas prevista pela comparação da posição da PU em relação à posição de um bloco preditivo de uma figuração de

referência. A figuração de referência pode ser selecionada a partir de uma primeira lista de figurações de referência (Lista 0) ou uma segunda lista de figurações de referência (Lista 1), cada uma das quais identifica uma ou mais figurações de referência armazenadas no DPB 64. Para previsão entre vistas, a figuração de referência está em uma vista diferente. A unidade de estimativa de movimento e de disparidade 42 envia o vetor de movimento e/ou valor de disparidade calculado para a unidade de criptação por entropia 56 e a unidade de compensação de movimento e de disparidade 44.

[00199] A compensação de movimento e/ou compensação de disparidade, realizada pela unidade de compensação de movimento e de disparidade 44, pode envolver obter ou gerar o bloco preditivo com base no vetor de movimento determinado pela estimativa de movimento e/ou estimativa de disparidade, realizando-se, possivelmente, interpolações para precisão de subpixel. Mediante o recebimento do vetor de movimento e/ou de disparidade para a PU do bloco de vídeo atual, a unidade de compensação de movimento e disparidade 44 pode localizar o bloco preditivo para o qual o vetor de movimento e/ou vetor de disparidade apontam em uma das listas de figurações de referência. O criptador de vídeo 20 forma um bloco de vídeo residual subtraindo-se valores de pixel do bloco preditivo dos valores de pixel do bloco de vídeo atual que é codificado, o que forma a diferença de valores de pixels. Os valores de diferença de pixel formam dados residuais para o bloco, e podem incluir tanto componentes de diferença de luma quanto de croma. O somador 50 representa o componente ou os componentes que realizam essa operação de subtração. A unidade de seleção de movimento e de disparidade 44 também pode gerar elementos de sintaxe associados aos blocos de

vídeo e a fatia de vídeo para uso pelo decodificador de vídeo 30 na decodificação dos blocos de vídeo da fatia de vídeo.

[00200] A unidade de intraprevisão 46 pode intraprever ou calcular um bloco atual, como uma alternativa à interprevisão realizada pela unidade de estimativa de movimento e de disparidade 42 e a unidade de compensação de movimento e de disparidade 44, conforme descrito acima. Em particular, a unidade de processamento de intraprevisão 46 pode determinar um modo de intraprevisão para usar para criptar um bloco atual. Em alguns exemplos, a unidade de processamento de intraprevisão 46 pode criptar um bloco atual com o uso de vários modos intraprevisões, por exemplo, durante passos de criptação separados, e a unidade de processamento de intraprevisão 46 (ou uma unidade de seleção de modo 41, em alguns exemplos) pode selecionar um modo de intraprevisão apropriado para usar a partir dos modos testados. Por exemplo, a unidade de processamento de intraprevisão 46 pode calcular valores de distorção de taxa com o uso de uma análise de distorção de taxa para os vários modos de intraprevisão testados, e selecionar o modo de intraprevisão que tem as melhores características de distorção de taxa entre os modos testados. A análise de distorção de taxa geralmente determina uma quantidade de distorção (ou erro) entre um bloco criptado e um bloco original não criptado que foi criptado para produzir o bloco criptado, assim como uma taxa de bits (ou seja, vários bits) usados para produzir o bloco criptado. A unidade de processamento de intraprevisão 46 pode calcular razões a partir de distorções e taxas para os vários blocos criptados para determinar qual modo de intraprevisão exibe o melhor valor de distorção de taxa para o bloco.

[00201] De qualquer maneira, após selecionar um modo de intraprevisão para um bloco, a unidade de processamento de intraprevisão 46 pode fornecer informações que indicam o modo de intraprevisão selecionado para o bloco em relação à unidade de criptação por entropia 56. A unidade de criptação por entropia 56 pode criptar a indicação de informações do modo de intraprevisão selecionada de acordo com as técnicas desta revelação. O criptador de vídeo 20 pode incluir nos dados de configuração de fluxo de bits transmitidos, que podem incluir uma pluralidade de tabelas de índices de modo de intraprevisão e uma pluralidade de tabelas de índices de modo de intraprevisão modificados (também referidas como tabelas de mapeamento de palavras-código), as definições de criptação de contextos para vários blocos, e indicações de um modo de intraprevisão mais provável, uma tabela de índices de modo de intraprevisão e uma tabela de índices de modo de intraprevisão modificados para usar para cada um dos contextos.

[00202] Após a unidade de processamento de previsão 41 gerar o bloco preditivo para o bloco de vídeo atual através da interprevisão ou da intraprevisão, o criptador de vídeo 20 forma um bloco de vídeo residual subtraindo-se o bloco preditivo a partir do bloco de vídeo atual. Os dados de vídeo residuais no bloco residual podem ser incluídos em uma ou mais TUs e aplicados à unidade de processamento de transformada 52. A unidade de processamento de transformada 52 transforma os dados de vídeo residuais em coeficientes de transformada residuais com o uso de uma transformada, tal como uma transformada de cosseno distinta (DCT) ou uma transformada conceitualmente semelhante. A unidade de processamento de transformada 52 pode converter os dados de vídeo residuais a partir de um

domínio de pixel a um domínio de transformada, tal como um domínio de frequência.

[00203] A unidade de processamento de transformada 52 pode enviar o coeficiente de transformada resultante para a unidade de quantização 54. A unidade de quantização 54 quantiza o coeficiente de transformada para reduzir ainda mais a taxa de bits. O processo de quantização pode reduzir a profundidade de bits associada a alguns ou todos os coeficientes. O grau de quantização pode ser modificado ajustando-se um parâmetro de quantização. Em alguns exemplos, a unidade de quantização 54, então, pode realizar uma varredura da matriz, incluindo o coeficiente de transformada quantizado. Alternativamente, a unidade de criptação por entropia 56 pode realizar a varredura.

[00204] Após a quantização, a unidade de criptação por entropia 56 cripta por entropia o coeficiente de transformada quantizado. Por exemplo, uma unidade de criptação por entropia 56 pode realizar codificação de comprimento variável adaptativa a contexto (CAVLC), codificação aritmética binária adaptativa a contexto (CABAC), codificação aritmética binária adaptativa a contexto com base em sintaxe (SBAC), codificação por entropia de partição de Intervalo de Probabilidade (PIPE) ou outra metodologia ou técnica de criptação por entropia. Após a criptação por entropia pela unidade de criptação por entropia 56, o fluxo de bits criptado pode ser transmitido para o decodificador de vídeo 30 ou arquivado para transmissão ou recuperação posterior pelo decodificador de vídeo. A unidade de criptação por entropia 56 também pode criptar por entropia os vetores de movimento e os outros elementos de sintaxe para a fatia de vídeo atual é codificada.

[00205] A unidade de quantização regredir 58 e

a unidade de processamento de transformada regressir 60 aplicam quantização regressir e transformada regressir, respectivamente, para reconstruir o bloco residual no domínio de pixels, para uso posterior como um bloco de referência de uma figuração de referência. Essa parte do criptador de vídeo 20 algumas vezes chamado do circuito de reconstrução, e decodifica eficazmente um bloco de vídeo criptado para o uso como uma figuração de referência em interpretação. As figurações reconstruídas e/ou figurações de camada reconstruídas são armazenadas no DPB 64.

[00206] A unidade de compensação de movimento e de disparidade 44 pode calcular um bloco de referência adicionando-se o bloco residual a um bloco preditivo de uma dentre as figurações de referência em uma dentre as listas de figuração de referência 64. A unidade de compensação de movimento e de disparidade 44 também pode aplicar um ou mais filtros de interpolação ao bloco residual reconstruído para calcular valores de pixel subinteiros para uso na estimativa de movimento. O somador 62 adiciona o bloco residual reconstruído ao bloco de previsão compensado de movimento produzido pela unidade de compensação de movimento 44 para produzir um bloco de referência para o armazenamento no DPB 64. O bloco de referência pode ser usado pela unidade de estimativa de movimento e de disparidade 42 e pela unidade de compensação de movimento e de e de disparidade 44 como um bloco de referência para interpretar um bloco em um quadro de vídeo subsequente ou figuração.

[00207] Conforme descrito acima em referência à Figura 5, o DPB 64 pode ser particionado em uma pluralidade de sub-DPBs, em que cada sub-DPB é configurado para armazenar figurações para um tipo diferente de camada em um processo de codificação de vídeo de múltiplas camadas.

Conforme discutido acima, e conforme será discutido em maiores detalhes abaixo em relação às Figuras 8, o criptador de vídeo 20 pode ser configurado para realizar as técnicas de sinalização de parâmetro de sub-DPB desta revelação.

[00208] A Figura 7 é um diagrama de blocos que ilustra um decodificador de vídeo exemplificativo que pode implantar as técnicas de sinalização de parâmetro de sub-DPB descritas nesta revelação. No exemplo da Figura 7, decodificador de vídeo 30 inclui memória de dados de vídeo 69, unidade de decodificação por entropia 80, unidade de processamento de previsão 81, unidade de quantização inversa 86, unidade de processamento de transformação inversa 88, somador 90 e DPB 92. A unidade de processamento de previsão 81 inclui uma unidade de compensação de movimento e de disparidade 82 e uma unidade de processamento de intraprevisão 84. O decodificador de vídeo 30 pode, em alguns exemplos, realizar um passo de decodificação geralmente recíproco ao passo de criptação descrito em relação ao criptador de vídeo 20 a partir da Figura 6.

[00209] A memória de dados de vídeo 69 pode armazenar dados de vídeo, tal como um fluxo de bits de vídeo criptado, a ser decodificado pelos componentes de decodificador de vídeo 30. Os dados de vídeo armazenados na memória de dados de vídeo 69 podem ser obtidos, por exemplo, a partir do dispositivo 34, a partir de uma fonte de vídeo local, tal como uma câmera, através de comunicação de rede sem fio ou com fio de dados de vídeo ou através do acesso de meio de armazenamento de dados físicos. A memória de dados de vídeo 69 pode formar um armazenamento temporário de figuração codificada (CPB) que armazena dados de vídeo criptados a partir de um fluxo de bits de vídeo



criptado.

[00210] Durante o processo de decodificação, o decodificador de vídeo 30 recebe um fluxo de bits de vídeo criptado que representa blocos de vídeo de uma fatia de vídeo criptada e elementos de sintaxe associados provenientes do criptador de vídeo 20. A unidade de decodificação por entropia 80 do decodificador de vídeo 30 decodifica por entropia o fluxo de bits para gerar coeficientes quantizados, vetores de movimento, vetores de disparidade e outros elementos de sintaxe. A unidade de decodificação por entropia 80 encaminha os vetores de movimento, vetores de disparidade e outros elementos de sintaxe para a unidade de processamento de previsão 81. O decodificador de vídeo 30 pode receber os elementos de sintaxe no nível de fatia de vídeo e/ou no nível de bloco de vídeo.

[00211] Quando a fatia de vídeo é codificada como uma fatia intracodificada (I), unidade de processamento de intraprevisão 84 da unidade de processo de previsão 81 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 provenientes de blocos decodificados anteriormente do quadro ou figuração atual. Quando o quadro de vídeo é codificado como uma fatia intercodificada (por exemplo, B ou P) ou fatia prevista de entre vistas, a unidade de compensação de movimento e de disparidade 82 de unidade de processamento de previsão 81 produz blocos preditivos para um bloco de vídeo da fatia de vídeo atual com base nos vetores de movimento, vetores de disparidade e outros elementos de sintaxe recebidos a partir da unidade de decodificação por entropia 80. Os blocos preditivos podem ser produzidos a partir de uma das figurações de referência dentro de uma das listas de

figurações de referência. O decodificador de vídeo 30 pode construir as listas de quadro de referência, a Lista 0 e a Lista 1, com o uso de técnicas de construção padrão com base em figuras de referência armazenadas no DPB 92.

[00212] A unidade de compensação de movimento e de disparidade 82 determina informações de previsão para um bloco de vídeo da fatia de vídeo atual analisando-se os vetores de movimento e outros elementos de sintaxe, e usa as informações de previsão para produzir os blocos preditivos para o bloco de vídeo atual em decodificação. Por exemplo, a unidade de compensação de movimento e disparidade 82 usa uma parte dos elementos de sintaxe recebidos para determinar um modo de previsão (por exemplo, intra ou interprevisão) usado para codificar os blocos de vídeo da fatia de vídeo, um tipo de fatia de interprevisão ou entrevistas (por exemplo, fatia B ou fatia P), informações de construção para uma ou mais dentre as listas de figuras de referência para a fatia, vetores de movimento e/ou disparidade para cada bloco de vídeo intercriptado da fatia, situações de interprevisão para cada bloco de vídeo intercodificado da fatia e outras informações para decodificar os blocos de vídeo na fatia de vídeo atual.

[00213] A unidade de compensação de movimento e de disparidade 82 também pode realizar a interpolação com base em filtros de interpolação. A unidade de compensação de movimento e de disparidade 82 por usar filtros de interpolação conforme usado pelo criptador de vídeo 20 durante a criptação dos blocos de vídeo para calcular valores interpolados para pixels subinteiros de blocos de referência. Nesse caso, a unidade de compensação de movimento e de disparidade 82 pode determinar os filtros de interpolação usados pelo criptador de vídeo a partir dos

elementos de sintaxe recebidos e usar os filtros de interpolação para produzir blocos preditivos.

[00214] A unidade de quantização inversa 86 quantiza inversamente, por exemplo, de-quantiza, o coeficiente de transformada quantizado fornecido no fluxo de bits e decodificado pela unidade de decodificação por entropia 80. O processo de quantização inversa pode incluir o uso de um parâmetro de quantização calculado pelo criptador de vídeo 20 para cada bloco de vídeo na fatia de vídeo para determinar um grau de quantização e, de modo similar, um grau de quantização inversa que deve ser aplicado. A unidade processamento de transformada inversa 88 aplica uma transformada inversa, por exemplo, uma DCT inversa, uma transformada de número inteiro inversa ou um processo de transformada inversa conceitualmente similar, ao coeficiente de transformada a fim de produzir blocos residuais no domínio de pixels.

[00215] Após a unidade de compensação de movimento e de disparidade 82 gerar o bloco preditivo para o bloco de vídeo atual com base nos vetores de movimento e de disparidade e outros elementos de sintaxe, o decodificador de vídeo 30 forma um bloco de vídeo decodificado somando-se os blocos residuais provenientes da unidade de processamento de transformada inversa 88 com os blocos preditivos correspondentes gerados pela unidade de compensação de movimento 82. O somador 90 representa o componente ou os componentes que realizam essa operação de soma. Caso desejado, um filtro de desbloqueio também pode ser aplicado para filtrar os blocos decodificados a fim de remover os artefatos de característica de bloqueio. Outros filtros em circuito (tanto no circuito de codificação quanto após o circuito de codificação) também podem ser usados para suavizar as transições de pixel, ou, de outro

modo, melhorar a qualidade de vídeo.

[00216] O vídeo decodificado em um dito quadro ou figuração são, então, armazenados no DPB 92, que armazenam blocos de figuração de referência (ou figurações de camada em um processo de codificação de vídeo de múltiplas camadas) usado para compensação de movimento subsequente. DPB 92 também armazena o vídeo decodificado para apresentação em um dispositivo de exibição, como o dispositivo de exibição 32 da Figura 1. Conforme descrito acima em referência à Figura 5, o DPB 92 pode ser particionado em uma pluralidade de sub-DPBs, em que cada sub-DPB é configurado para armazenar figurações para um tipo diferente de camada em um processo de codificação de vídeo de múltiplas camadas. Conforme discutido acima, e conforme será discutido em maiores detalhes abaixo em relação às Figuras 9, decodificador de vídeo 300 pode ser configurado para realizar as técnicas de sinalização de parâmetro de sub-DPB desta revelação.

[00217] A Figura 8 é um fluxograma que mostra um método de criptação exemplificativo de acordo com as técnicas desta revelação. As técnicas da Figura 8 podem ser implantadas por uma ou mais unidades de hardware de criptador de vídeo 20, incluindo o DPB 64.

[00218] Em um exemplo da revelação, o criptador de vídeo 20 pode ser configurado para reconstruir uma pluralidade de figurações de um fluxo de bits de vídeo criptados (800), e armazenar a pluralidade de figurações em um ou mais sub-DPBs (802). Isto é, o DPB 64 pode ser dividido uma ou mais sub-DPBs. O criptador de vídeo 20 pode ser configurado adicionalmente para gerar um conjunto respectivo de parâmetros de sub-DPB para cada ponto de operação respectivo do fluxo de bits de vídeo criptados(804), aplicar o conjunto respectivo de parâmetros

de sub-DPB a todas as camadas de um conjunto de camadas de saída para cada ponto de operação respectivo (806), e realizar um processo de gerenciamento de sub-DPB nos um ou mais sub-DPBs de acordo com os conjuntos únicos respectivos gerados de parâmetros de sub-DPB (808). Em um exemplo da revelação, parâmetros de sub-DPB incluem pelo menos um dentre um elemento de sintaxe indicativo de um número máximo de figurações de reordenação permitido para os um ou mais sub-DPBs e um elemento de sintaxe indicativo de uma latência máxima permitida para os um ou mais sub-DPBs. O criptador de vídeo 20 pode ser configurado adicionalmente para sinalizar o conjunto respectivo de parâmetros de sub-DPB no fluxo de bits de vídeo criptados (810).

[00219] Em outro exemplo da revelação, o criptador de vídeo 20 pode ser configurado para gerar o conjunto respectivo de parâmetros de sub-DPB em um conjunto de parâmetros de vídeo (VPS). Em outro exemplo da revelação, o criptador de vídeo 20 pode ser configurado para gerar um primeiro elemento de sintaxe de subcamada respectivo que indica se os parâmetros de sub-DPB são sinalizados ou não para cada subcamada de cada conjunto de camadas de saída respectivo. Em outro exemplo da revelação, o criptador de vídeo 20 pode ser configurado para gerar, no caso em que o primeiro elemento de sintaxe de subcamada que indica que os parâmetros de sub-DPB podem ser sinalizados para cada subcamada de um conjunto de camadas de saída particular, receber um segundo elemento de sintaxe de subcamada respectivo que indica se os parâmetros de sub-DPB são sinalizados ou não para subcamadas do conjunto de camadas de saída particulares respectivas. Em outro exemplo da revelação, o criptador de vídeo 20 pode ser configurado para gerar, no caso em que o segundo elemento de sintaxe de subcamada respectivo indica que os parâmetros de sub-DPB

são sinalizados para a subcamada particular de um conjunto adicional de parâmetros de sub-DPB para a subcamada particular.

[00220] Em outro exemplo da revelação o processo de gerenciamento de sub-DPB inclui pelo menos uma dentre marcar figurações como não usadas para referência, marcar as figurações para saída, e remover figurações do sub-DPB, e em que a realização do processo de gerenciamento de sub-DPB compreende realizar o processo de gerenciamento de sub-DPB em um nível de unidade de acesso com o uso dos conjuntos respectivos recebidos de parâmetros de sub-DPB. O criptador de vídeo 20 pode ser configurado para realizar o processo de gerenciamento de sub-DPB no nível de unidade de acesso emitindo-se figurações com base no número de unidades de acesso em um DPB que contém pelo menos uma figuração marcada como necessária para a saída, marcando as figurações como não usadas para referência em um nível por camada (por exemplo, para cada camada, de modo independente), e removendo as figurações para o sub-DPB em um nível por camada (por exemplo, para cada camada, de modo independente).

[00221] A Figura 9 é um fluxograma que mostra um método de decodificação exemplificativo de acordo com as técnicas desta revelação. As técnicas da Figura 9 podem ser implantadas por uma ou mais unidades de hardware de decodificador de vídeo 30, incluindo o DPB 92.

[00222] Em um exemplo da revelação, decodificador de vídeo 30 pode ser configurado para receber um fluxo de bits de vídeo criptados que incluem uma pluralidade de figurações (900), e armazenar a pluralidade de figurações em um ou mais sub-DPBs (902). Isto é, o DPB 92 pode ser dividido uma ou mais sub-DBPs. O decodificador de vídeo 30 pode ser configurado adicionalmente para

receber um conjunto respectivo de parâmetros de sub-DPB para cada ponto de operação respectivo do fluxo de bits de vídeo criptados(904), aplicar o conjunto respectivo de parâmetros de sub-DPB a todas as camadas de um conjunto de camadas de saída para cada ponto de operação respectivo (906), e realizar um processo de gerenciamento de sub-DPB nos um ou mais sub-DPBs de acordo com os conjuntos únicos respectivos recebidos de parâmetros de sub-DPB (908). Em um exemplo da revelação, parâmetros de sub-DPB incluem pelo menos um dentre um elemento de sintaxe indicativo de um número máximo de figurações de reordenação permitido para os um ou mais sub-DPBs e um elemento de sintaxe indicativo de uma latência máxima permitida para os um ou mais sub-DPBs.

[00223] Em outro exemplo da revelação, o decodificador de vídeo 30 pode ser configurado para receber o conjunto respectivo de parâmetros de sub-DPB em um conjunto de parâmetros de vídeo (VPS). Em outro exemplo da revelação, o decodificador de vídeo 30 pode ser configurado para receber um primeiro elemento de sintaxe de subcamada respectivo que indica se os parâmetros de sub-DPB são sinalizados ou não para cada subcamada de cada conjunto de camadas de saída respectivo.

[00224] Em outro exemplo da revelação, o decodificador de vídeo 30 pode ser configurado para aplicar, no caso em que o primeiro elemento de sintaxe de subcamada que indica que os parâmetros de sub-DPB não são sinalizados para cada subcamada de um conjunto de camadas de saída particular, o conjunto respectivo de parâmetros de sub-DPB para todas as subcamadas do conjunto de camadas de saída particulares respectivas. Em outro exemplo da revelação, o decodificador de vídeo 30 pode ser configurado para receber, no caso em que o primeiro elemento de sintaxe

de subcamada que indica que os parâmetros de sub-DPB podem ser sinalizados para cada subcamada de um conjunto de camadas de saída particular, receber um segundo elemento de sintaxe de subcamada respectivo que indica se os parâmetros de sub-DPB são sinalizados ou não para subcamadas do conjunto de camadas de saída particulares respectivas.

[00225] Em outro exemplo da revelação, o decodificador de vídeo 30 pode ser configurado para aplicar, no caso em que o segundo elemento de sintaxe de subcamada respectivo indica que os parâmetros de sub-DPB não são sinalizados para uma subcamada particular o conjunto respectivo recebido de parâmetros de sub-DPB para a subcamada particular. Adicionalmente, no caso do segundo elemento de sintaxe de subcamada respectivo que indica que os parâmetros de sub-DPB são sinalizados para a subcamada particular, decodificador de vídeo 30 podem ser configurados para receber um conjunto adicional de parâmetros de sub-DPB para a subcamada particular.

[00226] Em outro exemplo da revelação, o decodificador de vídeo 30 pode ser configurado para aplicar, no caso em que o segundo elemento de sintaxe de subcamada respectivo indica que os parâmetros de sub-DPB não são sinalizados para uma subcamada particular com uma ID de subcamada X, um último conjunto respectivo recebido de parâmetros de sub-DPB para subcamadas que têm a ID de subcamada maior que X

[00227] Em outro exemplo da revelação o processo de gerenciamento de sub-DPB inclui pelo menos uma dentre marcar figurações como não usadas para referência, marcar as figurações para saída, e remover figurações do sub-DPB, e em que a realização do processo de gerenciamento de sub-DPB compreende realizar o processo de gerenciamento de sub-DPB em um nível de unidade de acesso com o uso dos



conjuntos respectivos recebidos de parâmetros de sub-DPB. O decodificador de vídeo 30 pode ser configurado para realizar o processo de gerenciamento de sub-DPB no nível de unidade de acesso emitindo-se figurações com base no número de unidades de acesso em um DPB que contém pelo menos uma figuração marcada como necessária para a saída, marcando as figurações como não usadas para referência em um nível por camada (por exemplo, para cada camada, de modo independente), e removendo as figurações para o sub-DPB em um nível por camada (por exemplo, para cada camada, de modo independente).

[00228] Deve ser reconhecido que, dependendo do exemplo, certos atos ou eventos de quaisquer uma dentre as técnicas descritas no presente documento podem ser realizados em uma sequência diferente, podem ser adicionados, fundidos ou deixados de fora todos juntos (por exemplo, nem todos os atos e eventos descritos são necessários para a prática das técnicas). Ademais, em certos exemplos, os atos e eventos podem ser realizados de modo concorrente, por exemplo, através de processamento multifiletado, processamento interrompido ou em múltiplos processadores, em vez de sequencialmente.

[00229] Em um ou mais exemplos, as funções descritas podem ser implantadas em hardware, software, firmware ou qualquer combinação dos mesmos. Caso implantado em software, as funções podem ser armazenadas em, ou transmitidas sobre, como uma ou mais instruções ou código em uma mídia legível por computador e executadas por uma unidade de processamento com base em hardware. As mídias legíveis por computador podem incluir mídias de armazenamento legíveis por computador, que correspondem a uma mídia tangível como mídias de armazenamento de dados ou mídias de comunicação que incluem qualquer mídia que

facilite a transferência de um programa de computador proveniente de um lugar para outro, por exemplo, de acordo com um protocolo de comunicação. Desse modo, as mídias legíveis por computador geralmente podem corresponder a (1) mídias tangíveis de armazenamento legíveis por computador que sejam não transitórias ou (2) uma mídia de comunicação tal como um sinal ou onda portadora. As mídias de armazenamento de dados podem ser quaisquer mídias disponíveis que possam ser acessadas por um ou mais computadores ou um ou mais processadores para recuperar instruções, estruturas de código e/ou dados para a implantação das técnicas descritas nesta revelação. Um produto de programa de computador pode incluir uma mídia legível por computador. Conforme usado no presente documento, o termo 'sinalização' pode incluir armazenar ou de outra forma incluir dados com um fluxo de bits criptado. Em outras palavras, em diversos exemplos de acordo com essa revelação, o termo 'sinalização' pode ser associado à comunicação em tempo real de dados, ou alternativamente, à comunicação que não é realizada em tempo real.

[00230] A título de exemplo, e não de limitação, tais mídias de armazenamento legíveis por computador podem compreender RAM, ROM, EEPROM, CD-ROM ou outro armazenamento de disco óptico, armazenamento de disco magnético ou outros dispositivos de armazenamento magnético, ou qualquer outra mídia que possa ser usada para armazenar o código de programa desejado na forma de instruções ou estruturas de dados e que possa ser acessado por um computador. Também, qualquer conexão pode ser propriamente denominada uma mídia legível por computador. Por exemplo, se as instruções forem transmitidas a partir de um sítio da web, servidor ou outra fonte remota com o uso de um cabo coaxial, cabo de fibra óptica, par trançado,

linha de inscrição digital (DSL) ou tecnologias sem fio como infravermelho, rádio e micro-onda, então, o cabo coaxial, o cabo de fibra óptica, o par trançado, a DSL ou as tecnologias sem fio como infravermelho, rádio e micro-onda estão incluídos na definição de mídia. Deve ser entendido, entretanto, que as mídias de armazenamento legíveis por computador e as mídias de armazenamento de dados não incluem conexões, ondas transportadoras, sinais ou outras mídias transitórias, mas são, em vez disso, direcionadas para mídias não transientes e tangíveis. Disco magnético e disco óptico, conforme usado no presente documento, incluem disco compacto (CD), disco laser, disco ótico, disco versátil digital (DVD), disquete e disco blu-ray, em que os discos magnéticos normalmente reproduzem os dados de modo magnético, enquanto os discos ópticos reproduzem os dados de modo óptico com lasers. As combinações dos supracitados também devem ser abrangidas pelo escopo de meios legíveis por computador.

[00231] As instruções podem ser executadas por um ou mais processadores, como um ou mais processadores de sinal digital (DSPs), microprocessadores para propósitos gerais, circuitos integrados de aplicação específica (ASICs), matrizes lógicas programáveis por campo (FPGAs) ou outro conjunto de circuitos lógicos equivalentes integrados ou discretos. Portanto, o termo "processador", conforme usado no presente documento pode se referir a qualquer uma das estruturas supracitadas ou qualquer outra estrutura adequada para a implantação das técnicas descritas no presente documento. Adicionalmente, em alguns aspectos, a funcionalidade descrita no presente documento pode ser fornecida dentro de módulos dedicados de hardware e/ou software configurados para criptar e decodificar ou incorporados em um codec combinado. Também, as técnicas

podem ser totalmente implantadas em um ou mais circuitos ou elementos lógicos.

[00232] As técnicas dessa revelação podem ser implantadas em uma ampla variedade de dispositivos ou aparelhos, incluindo um monofone, um circuito integrado (IC) ou um conjunto de ICs (por exemplo, um conjunto de chips). Vários componentes, módulos ou unidades são descritos nessa revelação para enfatizar os aspectos funcionais dos dispositivos configurados para realizar as técnicas reveladas, mas não exigem necessariamente a realização por diferentes unidades de hardware. Em vez disso, conforme descrito acima, várias unidades podem ser combinadas em uma unidade de hardware de codec ou fornecidas por uma coleção de unidades de hardware interoperativos, incluindo um ou mais processadores conforme descrito acima, em conjunto com software e/ou firmware adequados.

[00233] Vários exemplos foram descritos. Essas e outras implantações estão dentro do escopo das reivindicações a seguir.

### **REIVINDICAÇÕES**

1. Método de decodificação de dados de vídeo usando um buffer de imagem decodificada, DPB, particionado em um ou mais dos sub-DPBs, o método **caracterizado** pelo fato de que compreende:

receber (900), de um fluxo de bits de vídeo codificado, uma pluralidade de conjuntos de camadas de saída, em que cada conjunto de camada de saída da pluralidade de conjuntos de camada de saída inclui uma respectiva pluralidade de subcamadas que são camadas escaláveis temporais de um fluxo de bits escalável temporal, em que cada respectiva pluralidade de subcamadas inclui uma primeira subcamada e uma ou mais subcamadas diferentes da primeira subcamada;

decodificar cada respectiva pluralidade de subcamadas da pluralidade de conjuntos de camada de saída;

armazenar (902) a pluralidade de cada respectiva pluralidade decodificada de subcamadas em um ou mais sub-DPBs;

receber (904), a partir do fluxo de bits de vídeo codificado, um primeiro respectivo elemento de sintaxe para cada conjunto de camada de saída da pluralidade de conjunto de camada de saída, em que cada primeiro respectivo elemento de sintaxe é indicativo se os parâmetros de sub-DPB são ou não sinalizados para mais de uma subcamada do respectivo conjunto de camada de saída;

condicional ao primeiro elemento de sintaxe respectivo recebido para um determinado conjunto de camadas de saída sendo indicativo que os parâmetros de sub-DPB são sinalizados para mais de uma subcamada do conjunto de camadas de saída específico, receber, do fluxo de bits de vídeo codificado, um segundo respectivo elemento de sintaxe para cada subcamada da uma ou mais subcamadas diferentes da

primeira subcamada do conjunto de camada de saída específico, em que cada segundo respectivo elemento de sintaxe é indicativo se os parâmetros de sub-DPB são ou não sinalizados para a respectiva subcamada da uma ou mais subcamadas diferentes da primeira subcamada do conjunto de camada de saída específica;

receber, do fluxo de bits de vídeo codificado, um respectivo conjunto de parâmetros de sub-DPB para a primeira subcamada de cada respectiva pluralidade de subcamadas do conjunto de camadas de saída específica;

para cada respectiva subcamada da uma ou mais subcamadas diferentes da primeira subcamada de um conjunto de camadas de saída, dependendo do valor do respectivo segundo elemento de sintaxe recebido para a subcamada, inferir um respectivo conjunto de parâmetros de sub-DPB a partir dos parâmetros de sub-DPB para uma subcamada anterior ou receber um respectivo conjunto de parâmetros de sub-DPB do fluxo de bits de vídeo codificado; e

realizar (908) um processo de gerenciamento de sub-DPB no um ou mais sub-DPBs de acordo com cada respectivo o conjunto de parâmetros de sub-DPB recebidos.

2. Método, de acordo com a reivindicação 1, **caracterizado** pelo fato de que cada respectivo conjunto de parâmetros de sub-DPB para a primeira subcamada de cada respectiva pluralidade de subcamadas inclui:

um respectivo terceiro elemento de sintaxe indicativo de um número máximo de imagens de reordenação permitidas para o um ou mais sub-DPBs; e

um respectivo quarto elemento de sintaxe indicativo de uma latência máxima permitida para o um ou mais sub-DPBs.

3. Método, de acordo com a reivindicação 1, **caracterizado** pelo fato de que receber o respectivo

conjunto de parâmetros de sub-DPB compreende receber o respectivo conjunto de parâmetros de sub-DPB em um conjunto de parâmetro de vídeo - VPS.

4. Método, de acordo com a reivindicação 1, **caracterizado** pelo fato de que compreende adicionalmente:

aplicar, com base no primeiro respectivo de elemento de sintaxe que indica que os parâmetros de sub-DPB não são sinalizados para cada subcamada de um segundo conjunto de camada de saída específica da pluralidade de conjuntos de camada de saída, o respectivo conjunto de parâmetros de sub-DPB da segunda camada de saída específica para todas as subcamadas do segundo conjunto de camada de saída específica.

5. Método, de acordo com a reivindicação 1, **caracterizado** pelo fato de que o processo de gerenciamento de sub-DPB inclui pelo menos um dentre marcar imagens como não usadas para referência, marcar imagens para saída, ou remover imagens do sub-DPB, e em que a realização do processo de gerenciamento de sub-DPB compreende realizar o processo de gerenciamento de sub-DPB em um nível de unidade de acesso com o uso dos respectivos conjuntos recebidos de parâmetros de sub-DPB.

6. Método, de acordo com a reivindicação 5, **caracterizado** pelo fato de que realizar o processo de gerenciamento de sub-DPB no nível da unidade de acesso compreende:

emitir imagens com base no número de unidades de acesso em um DPB que contém pelo menos uma imagem marcada conforme necessária para a saída;

marcar imagens como não usadas para referência em um nível por camada; e

remover imagens do sub-DPB em um nível por camada.

7. Método de codificação de dados de vídeo usando um buffer de imagem decodificada, DPB, particionado em um ou mais dos sub-DPBs, o método **caracterizado** pelo fato de que compreende:

reconstruir (800) uma pluralidade de conjuntos de camada de saída de um fluxo de bits de vídeo codificado, em que cada conjunto de camada de saída da pluralidade de conjuntos de camada de saída inclui uma respectiva pluralidade de subcamadas que são camadas escaláveis temporais de um fluxo de bits escalável temporal, em que cada respectiva pluralidade de subcamadas inclui uma primeira subcamada e uma ou mais subcamadas diferentes da primeira subcamada, em que reconstruir a pluralidade de conjuntos de camada de saída inclui reconstruir cada respectiva pluralidade de subcamadas da pluralidade de conjuntos de camada de saída;

armazenar (802) cada respectiva pluralidade reconstruída de subcamadas em um ou mais sub-DPBs;

gerar (804) um primeiro respectivo elemento de sintaxe para cada conjunto de camada de saída da pluralidade de conjunto de camada de saída, em que cada primeiro respectivo elemento de sintaxe é indicativo se os parâmetros de sub-DPB são ou não sinalizados para mais de uma subcamada do respectivo conjunto de camada de saída;

condicional ao primeiro elemento de sintaxe respectivo gerado para um conjunto de camadas de saída específico sendo indicativo de que os parâmetros de sub-DPB são sinalizados para mais de uma subcamada do conjunto de camadas de saída específico, gerar um segundo respectivo elemento de sintaxe para cada subcamada da uma ou mais subcamadas diferente da primeira subcamada do conjunto de camada de saída específico, em que cada segundo respectivo elemento de sintaxe é indicativo se os parâmetros de sub-



DPB são ou não sinalizados para a respectiva subcamada da uma ou mais subcamadas diferentes da primeira subcamada do conjunto de camada de saída específica;

gerar um respectivo conjunto de parâmetros de sub-DPB para a primeira subcamada de cada respectiva pluralidade de subcamadas e para cada respectiva subcamada da uma ou mais subcamadas diferentes da primeira subcamada de cada respectiva pluralidade de subcamadas;

realizar (808) um processo de gerenciamento de sub-DPB no um ou mais sub-DPBs de acordo com cada respectivo o conjunto de parâmetros de sub-DPB gerados; e

gerar (810), para a primeira subcamada de cada respectiva pluralidade de subcamadas, cada respectivo conjunto de parâmetros de sub-DPB no fluxo de bits de vídeo codificado;

para cada respectiva subcamada de uma ou mais subcamadas diferentes da primeira subcamada de um conjunto de camadas de saída, dependendo do valor do respectivo segundo elemento de sintaxe gerado para a subcamada, gerar um respectivo conjunto de parâmetros de sub-DPB no fluxo de bits de vídeo codificado ou não gerar um respectivo conjunto de parâmetros de sub-DPB no fluxo de bits de vídeo codificado.

8. Aparelho configurado para decodificar dados de vídeo usando um buffer de imagem decodificada, DPB, porcionado em uma pluralidade de sub-DPBs, o aparelho **caracterizado** pelo fato de que compreende:

um ou mais dispositivos de memória que compreendem um ou mais sub-DPBs configurados para armazenar dados de vídeo; e

um decodificador de vídeo configurado para:

receber, de um fluxo de bits de vídeo codificados, uma pluralidade de conjuntos de camadas de

saída, em que cada conjunto de camada de saída da pluralidade de conjuntos de camada de saída inclui uma respectiva pluralidade de subcamadas que são camadas escaláveis temporais de um fluxo de bits escalável temporal, em que cada respectiva pluralidade de subcamadas inclui uma primeira subcamada e uma ou mais subcamadas diferentes da primeira subcamada;

decodificar cada respectiva pluralidade de subcamadas da pluralidade de conjuntos de camada de saída;

armazenar cada respectiva pluralidade decodificada de subcamadas no um ou mais sub-DPBs;

receber, do fluxo de bits de vídeo codificado, um primeiro respectivo elemento de sintaxe para cada conjunto de camada de saída da pluralidade de conjuntos de camada de saída, em que cada primeiro respectivo elemento de sintaxe é indicativo se os parâmetros de sub-DPB são ou não sinalizados por mais de uma subcamada do respectivo conjunto de camadas de saída;

condicional ao primeiro elemento de sintaxe respectivo recebido para um conjunto de camadas de saída específico sendo indicativo de que os parâmetros de sub-DPB são sinalizados para mais de uma subcamada do conjunto de camadas de saída específico, receber, do fluxo de bits de vídeo codificado, um segundo respectivo elemento de sintaxe para cada subcamada da uma ou mais subcamadas diferentes da primeira subcamada do conjunto de camada de saída específico, em que cada segundo respectivo elemento de sintaxe é indicativo se os parâmetros de sub-DPB são sinalizados para a respectiva subcamada da uma ou mais subcamadas diferentes da primeira subcamada do conjunto de camada de saída específica;

receber, a partir fluxo de bits codificado, um respectivo conjunto de parâmetros de sub-DPB para a

primeira subcamada de cada respectiva pluralidade de subcamadas;

para cada respectiva subcamada da uma ou mais subcamadas diferentes da primeira subcamada de um conjunto de camadas de saída, dependendo do valor do respectivo segundo elemento de sintaxe recebido para a subcamada, inferir um respectivo conjunto de parâmetros de sub-DPB a partir dos parâmetros de sub-DPB para uma subcamada anterior ou receber um respectivo conjunto de parâmetros de sub-DPB do fluxo de bits de vídeo codificado; e

realizar um processo de gerenciamento de sub-DPB no um ou mais sub-DPBs de acordo com cada respectivo o conjunto de parâmetros de sub-DPB recebidos.

9. Aparelho, de acordo com a reivindicação 8, **caracterizado** pelo fato de que cada respectivo conjunto de parâmetros de sub-DPB para a primeira subcamada de cada respectiva pluralidade de subcamadas inclui:

um respectivo terceiro elemento de sintaxe indicativo de um número máximo de imagens de reordenação permitidas para o um ou mais sub-DPBs; e

um respectivo quarto elemento de sintaxe indicativo de uma latência máxima permitida para o um ou mais sub-DPBs.

10. Aparelho, de acordo com a reivindicação 8, **caracterizado** pelo fato de que o decodificador de vídeo é configurado para receber o respectivo conjunto de parâmetros de sub-DPB em um conjunto de parâmetro de vídeo, VPS.

11. Aparelho, de acordo com a reivindicação 8, **caracterizado** pelo fato de que o decodificador de vídeo é adicionalmente configurado para:

aplicar, com base no primeiro respectivo de elemento de sintaxe que indica que os parâmetros de sub-DPB

não são sinalizados para cada subcamada de um segundo conjunto de camada de saída específica, o respectivo conjunto de parâmetros de sub-DPB da segunda camada de saída específica definida para todas as subcamadas do segundo conjunto de camada de saída específica.

12. Aparelho, de acordo com a reivindicação 8, **caracterizado** pelo fato de que o processo de gerenciamento de sub-DPB inclui pelo menos um dentre marcar imagens como não usadas para referência, marcar imagens para saída, ou remover imagens do sub-DPB, e em que realizar o processo de gerenciamento de sub-DPB compreende realizar o processo de gerenciamento de sub-DPB em um nível de unidade de acesso com o uso dos respectivos conjuntos recebidos de parâmetros de sub-DPB.

13. Aparelho, de acordo com a reivindicação 12, **caracterizado** pelo fato de que o decodificador de vídeo é adicionalmente configurado para:

emitir imagens com base no número de unidades de acesso em um DPB que contém pelo menos uma imagem marcada conforme necessário para a saída;

marcar imagens como não usadas para referência em um nível por camada; e

remover imagens do sub-DPB em um nível por camada.

14. Aparelho configurado para codificar dados de vídeo usando um buffer de imagem decodificada, DPB, particionando em uma pluralidade de sub-DPBs, o aparelho **caracterizado** pelo fato de que compreende:

um ou mais dispositivos de memória que compreendem um ou mais sub-DPBs configurados para armazenar dados de vídeo; e

um codificador de vídeo configurado para:

reconstruir uma pluralidade de conjuntos de

camada de saída de um fluxo de bits de vídeo codificado, em que cada conjunto de camada de saída da pluralidade de conjuntos de camada de saída inclui uma respectiva pluralidade de subcamadas que são camadas escaláveis temporais de um fluxo de bits escalável temporal, em que cada respectiva pluralidade de subcamadas inclui uma primeira subcamada e uma ou mais subcamadas diferentes da primeira subcamada, em que reconstruir a pluralidade de conjuntos de camada de saída inclui reconstruir cada respectiva pluralidade de subcamadas da pluralidade de conjuntos de camada de saída;

armazenar cada respectiva pluralidade reconstruída de subcamadas em um ou mais sub-DPBs;

gerar um primeiro respectivo elemento de sintaxe para cada conjunto de camada de saída da pluralidade de conjunto de camada de saída, em que cada primeiro respectivo elemento de sintaxe é indicativo se os parâmetros de sub-DPB são ou não sinalizados para cada subcamada de cada respectivo conjunto de camada de saída;

condicional ao primeiro elemento de sintaxe respectivo gerado para um conjunto de camadas de saída específico, sendo indicativo de que os parâmetros de sub-DPB são sinalizados para mais de uma subcamada do conjunto de camadas de saída específico, gerar um segundo respectivo elemento de sintaxe para cada subcamada das uma ou mais subcamadas diferentes da primeira subcamada do conjunto de camada de saída específico, em que cada segundo respectivo elemento de sintaxe é indicativo se os parâmetros de sub-DPB são ou não sinalizados para a respectiva subcamada da uma ou mais subcamadas diferentes da primeira subcamada do conjunto de camada de saída específica;

gerar um respectivo conjunto de parâmetros de sub-DPB para a primeira subcamada de cada respectiva

pluralidade de subcamadas e para cada respectiva subcamada da uma ou mais subcamadas diferentes da primeira subcamada de cada respectiva pluralidade de subcamadas;

realizar um processo de gerenciamento de sub-DPB no um ou mais sub-DPBs de acordo com cada respectivo o conjunto de parâmetros de sub-DPB gerados;

gerar, para a primeira subcamada de cada respectiva pluralidade de subcamadas, cada respectivo conjunto de parâmetros de sub-DPB no fluxo de bits de vídeo codificado; e

para cada respectiva subcamada da uma ou mais subcamadas diferentes da primeira subcamada de um conjunto de camadas de saída, dependendo do valor do respectivo segundo elemento de sintaxe gerado para a subcamada, gerar um respectivo conjunto de parâmetros de sub-DPB no fluxo de bits de vídeo codificado ou não gerar um respectivo conjunto de parâmetros de sub-DPB no fluxo de bits de vídeo codificado.

15. Memória **caracterizada** pelo fato de que compreende instruções armazenadas na mesma, as instruções sendo executadas por um computador para realizar o método conforme definido em qualquer uma das reivindicações 1 a 7.

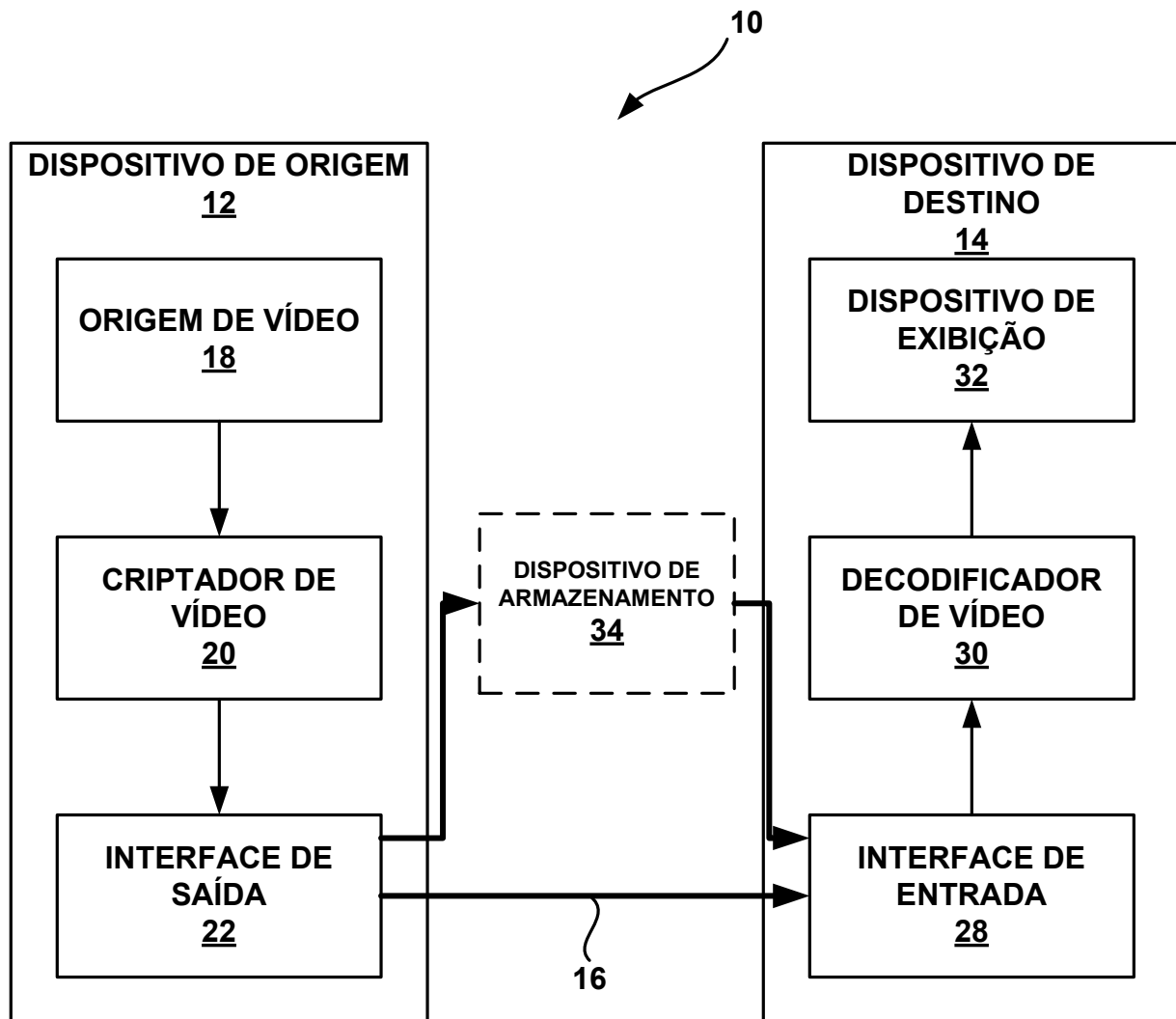
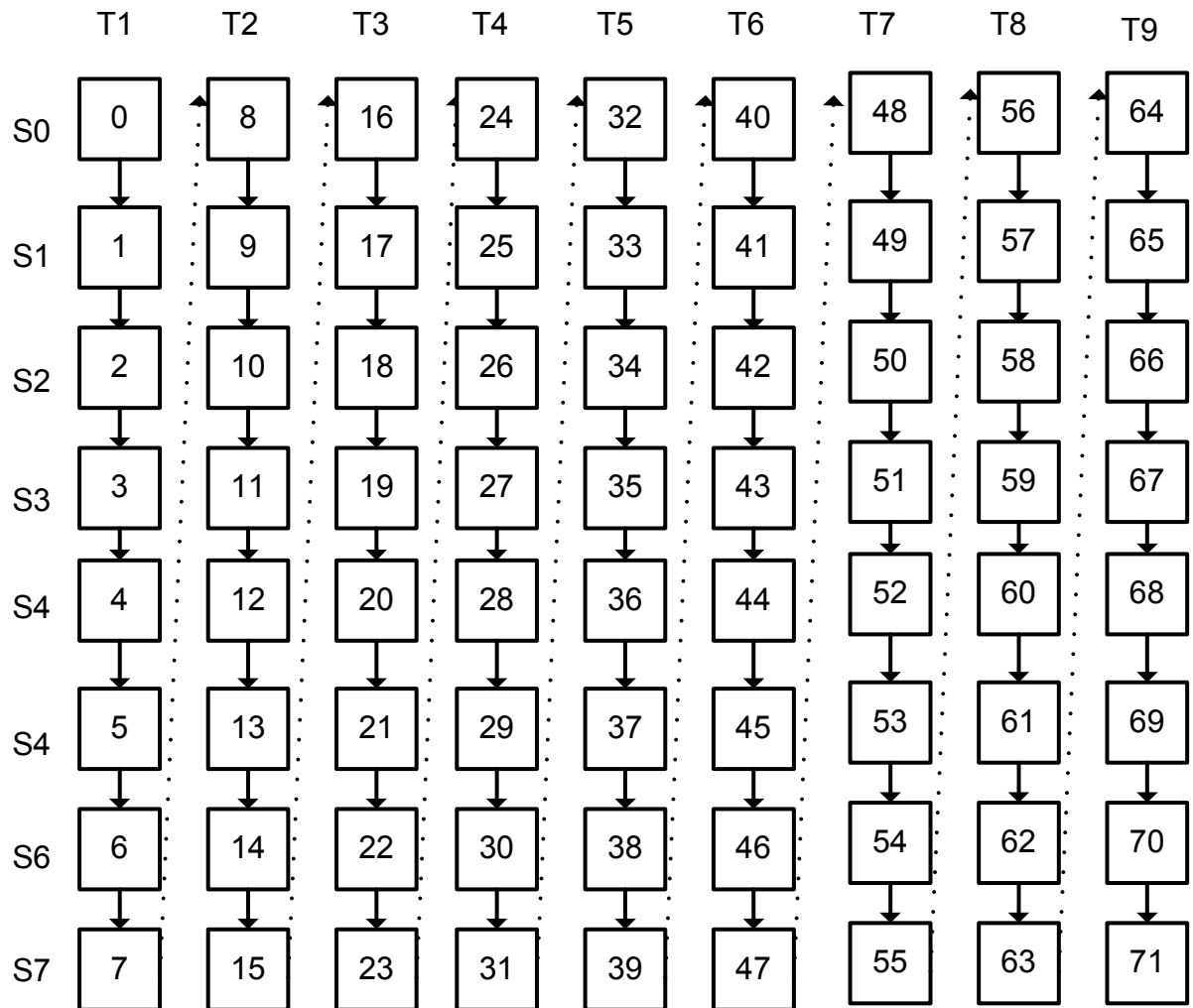


FIG. 1

**FIG. 2**



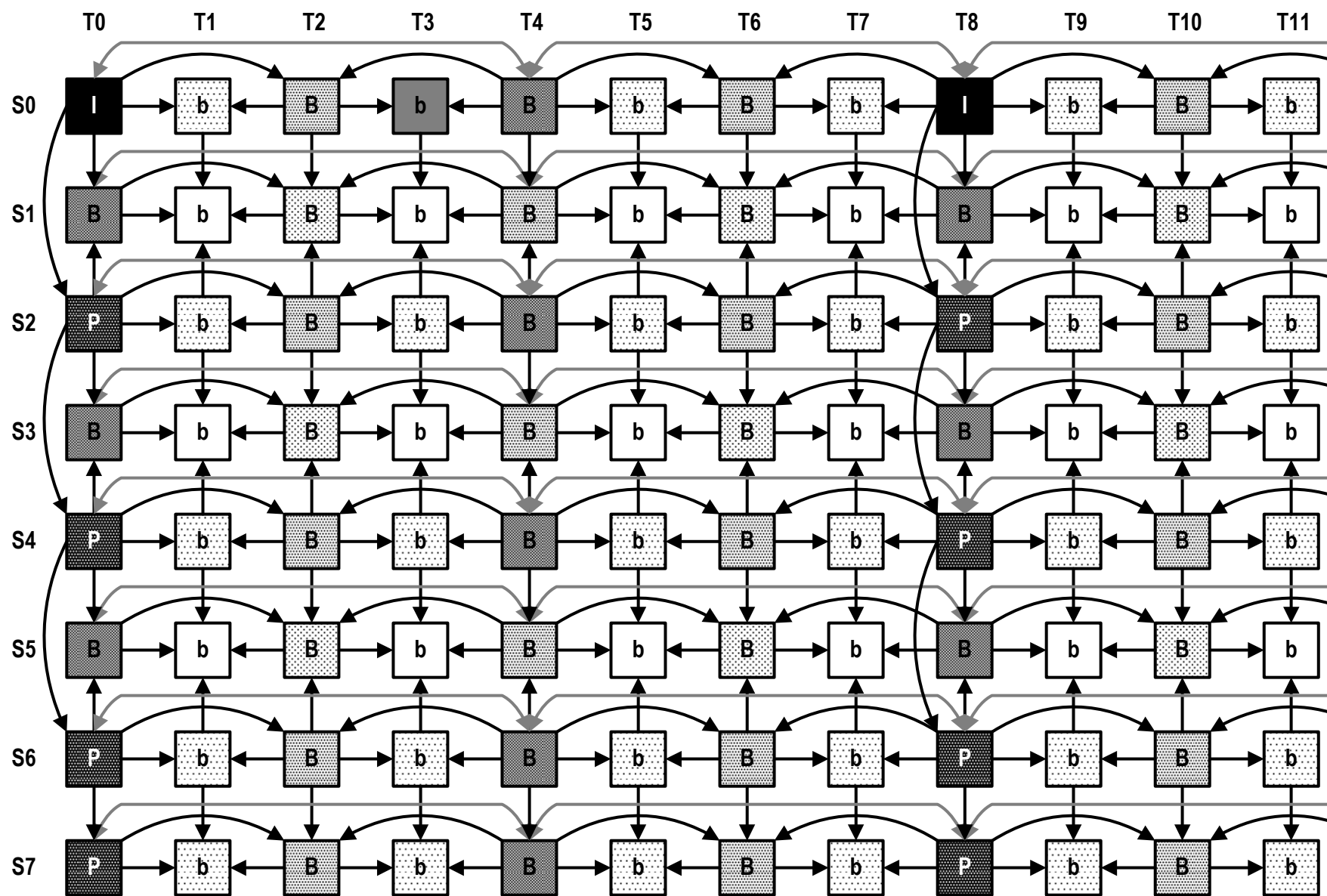


FIG. 3

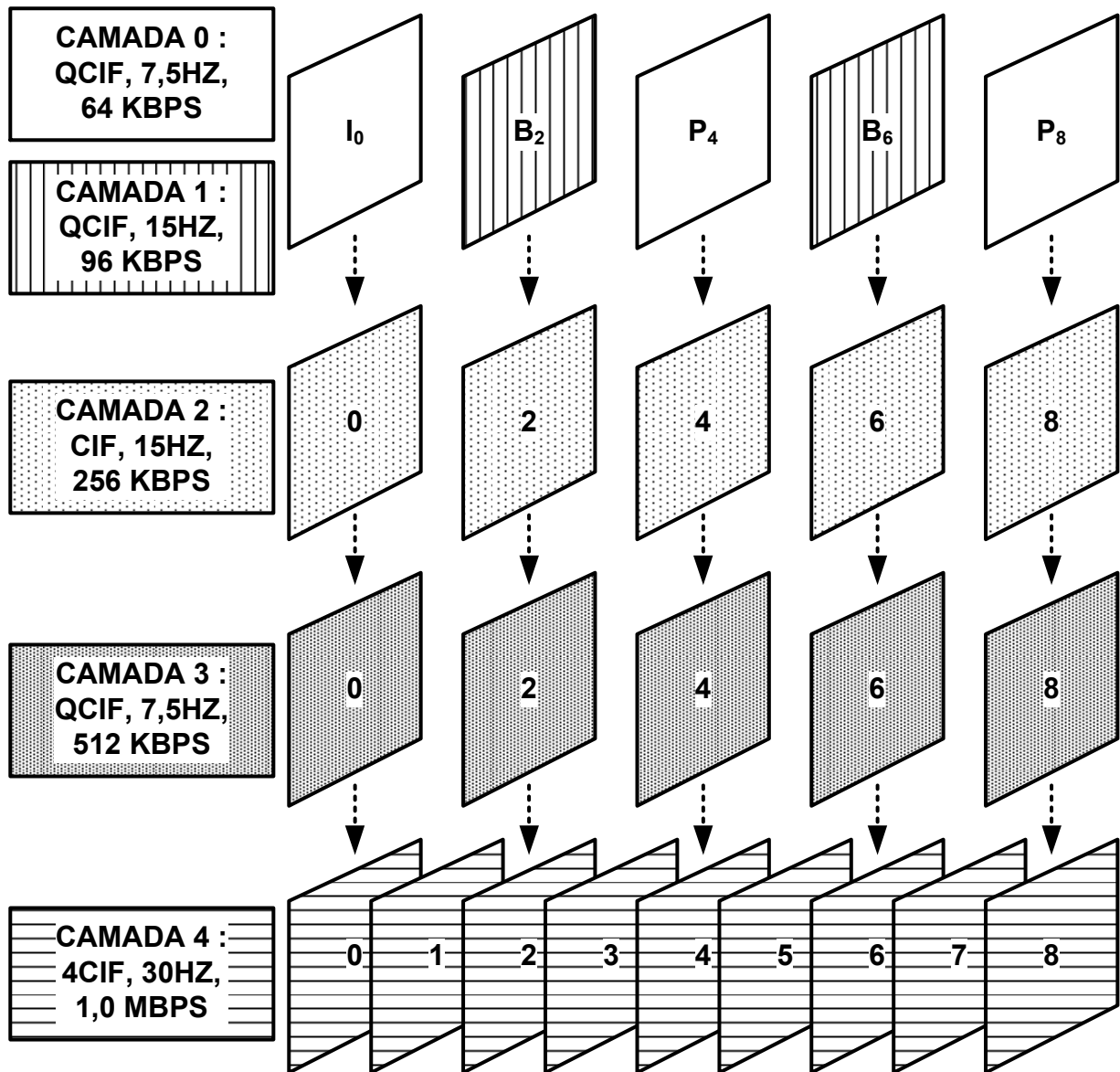
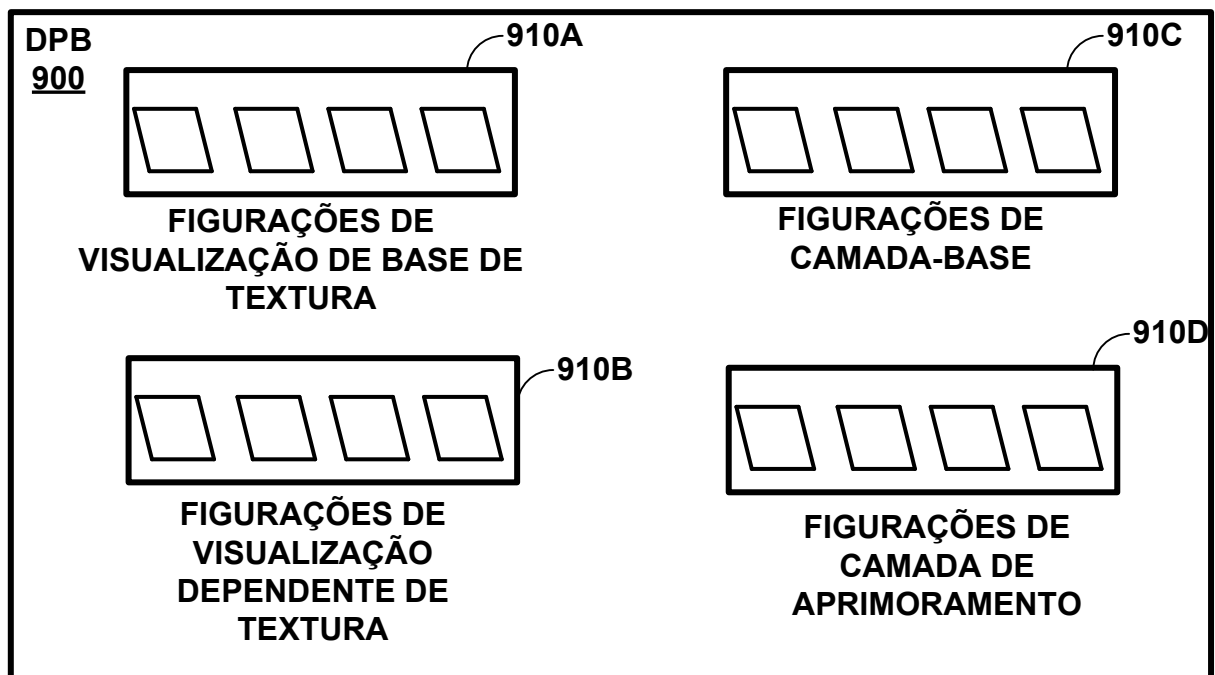
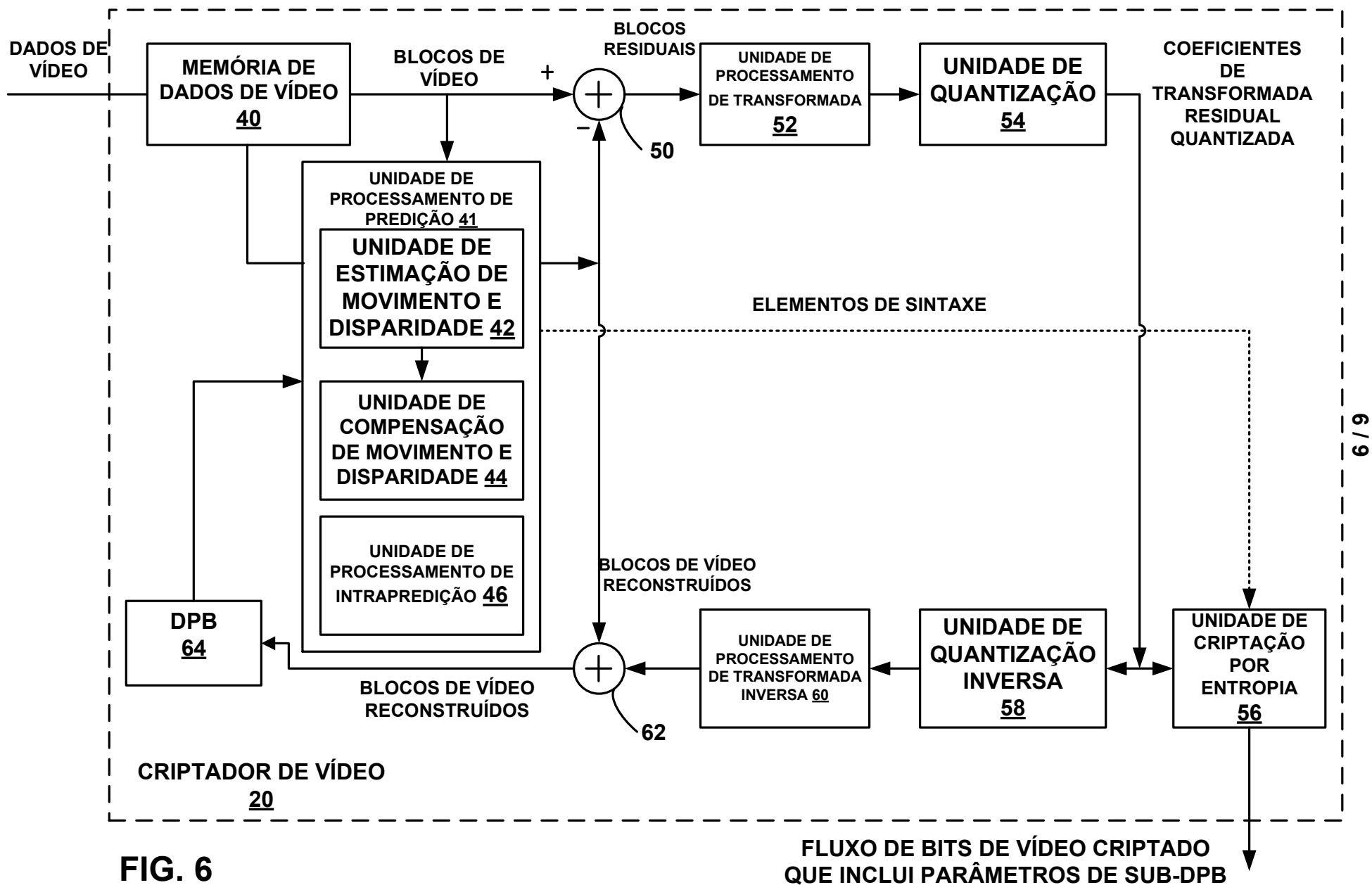


FIG. 4



**FIG. 5**



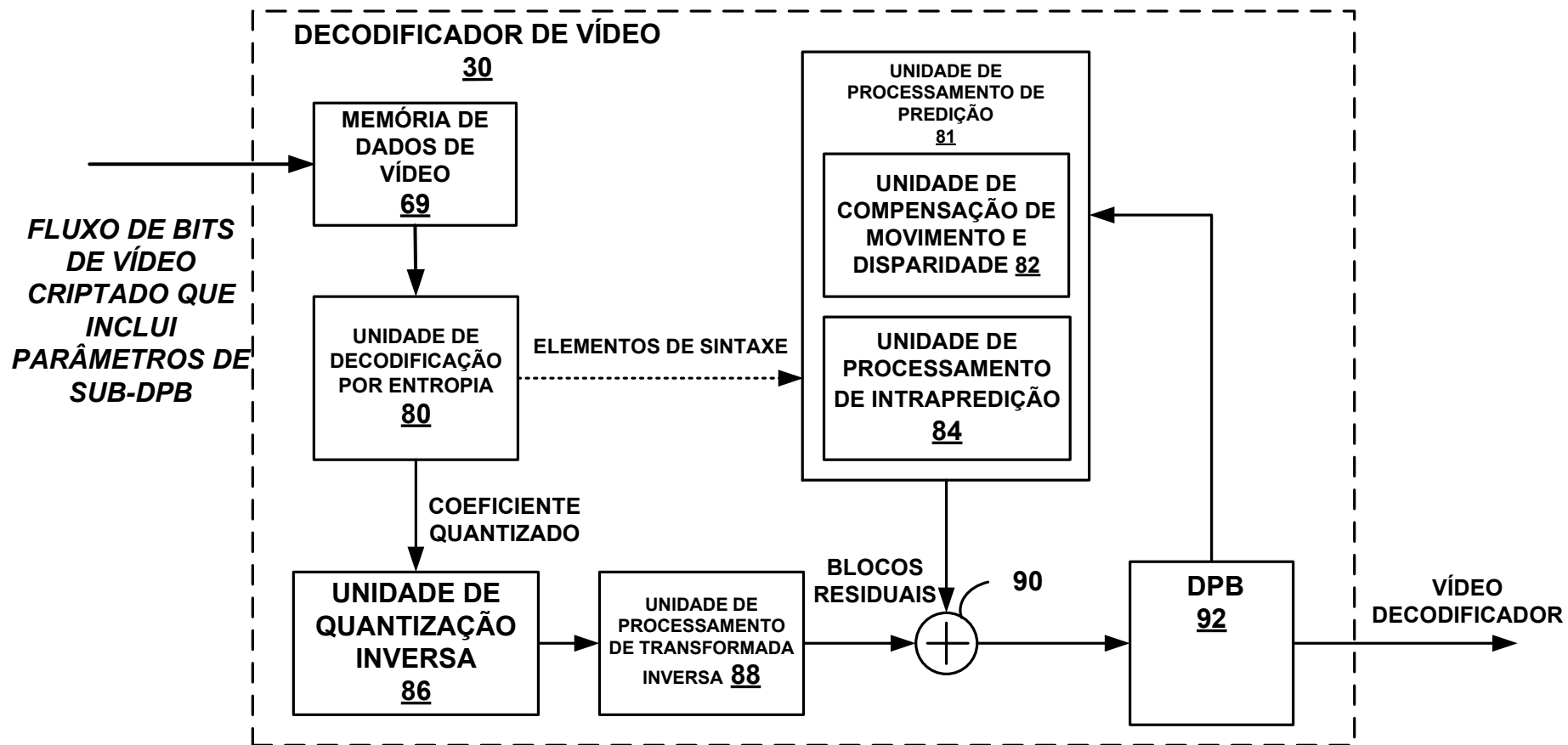


FIG. 7

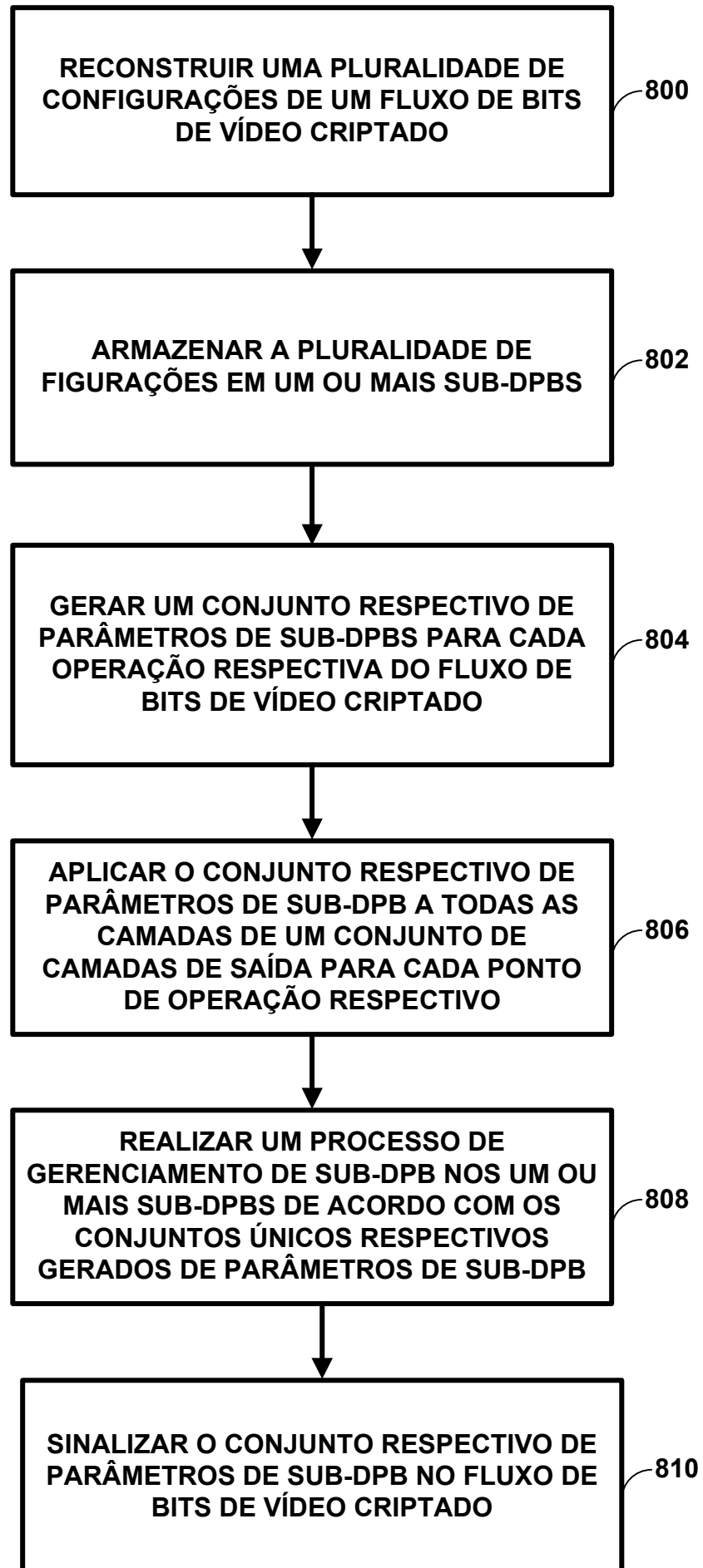


FIG. 8

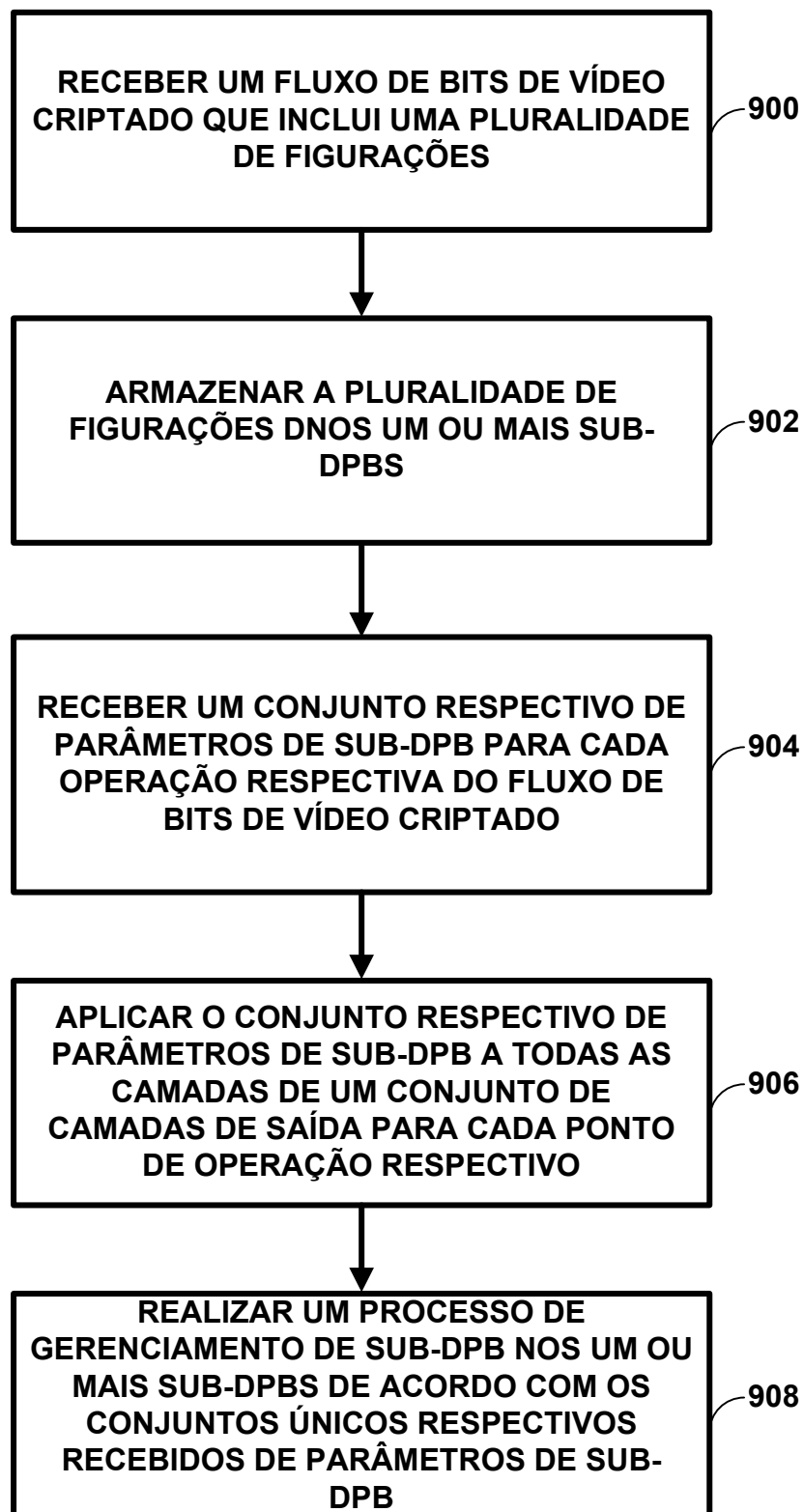


FIG. 9