



República Federativa do Brasil

Ministério do Desenvolvimento, Indústria,
Comércio e Serviços

Instituto Nacional da Propriedade Industrial

(11) BR 112016027428-8 B1

(22) Data do Depósito: 22/05/2015

(45) Data de Concessão: 14/11/2023

(54) Título: INICIALIZAÇÃO DE PALETA DE PREDITOR EM CODIFICAÇÃO DE VÍDEO COM BASE EM PALETA

(51) Int.Cl.: H04N 19/105; H04N 19/157; H04N 19/176; H04N 19/186.

(52) CPC: H04N 19/105; H04N 19/157; H04N 19/176; H04N 19/186.

(30) Prioridade Unionista: 03/10/2014 US 62/059,659; 09/06/2014 US 62/009,772; 17/10/2014 US 62/065,526; 20/06/2014 US 62/015,261; 21/05/2015 US 14/719,260; (...).

(73) Titular(es): QUALCOMM INCORPORATED.

(72) Inventor(es): VADIM SEREGIN; RAJAN LAXMAN JOSHI; MARTA KARCZEWICZ; WEI PU; JOEL SOLE ROJALS.

(86) Pedido PCT: PCT US2015032262 de 22/05/2015

(87) Publicação PCT: WO 2015/179814 de 26/11/2015

(85) Data do Início da Fase Nacional: 22/11/2016

(57) Resumo: INICIALIZAÇÃO DE PALETA DE INDICADOR EM CODIFICAÇÃO DE VÍDEO COM BASE EM PALETA. A presente invenção se refere, em um exemplo, a um método de processamento de dados de vídeo que inclui determinar uma primeira paleta para um primeiro bloco de dados de vídeo que está localizado em uma primeira fileira de blocos, gerar uma paleta de indicador para construir pelo menos uma segunda paleta de pelo menos um segundo bloco de dados de vídeo na primeira fileira de blocos codificados, reinicializar a paleta de indicador para determinar uma terceira paleta de um terceiro bloco de dados de vídeo que está localizado em uma segunda fileira de blocos, em que a reinicialização da paleta de indicador compreende reinicializar a paleta de indicador com base nas uma ou mais entradas de paleta da primeira paleta ou uma paleta de indicador inicial gerada após a codificação do primeiro bloco, determinar a terceira paleta do terceiro bloco com base na paleta de indicador reinicializada e codificar o terceiro bloco com o uso da terceira paleta.

"INICIALIZAÇÃO DE PALETA DE PREDITOR EM CODIFICAÇÃO DE
VÍDEO COM BASE EM PALETA"

[0001] Este pedido reivindica o benefício de Pedido Provisório U.S. nº 62/002.717, depositado no dia 23 de maio de 2014, Pedido Provisório U.S. nº 62/009.772, depositado no dia 9 de junho de 2014, Pedido Provisório U.S. nº 62/015.261, depositado no dia 20 de junho de 2014, Pedido Provisório U.S. nº 62/019.223, depositado no dia 30 de junho de 2014, Pedido Provisório U.S. nº 62/059.659, depositado no dia 3 de outubro de 2014 e Pedido Provisório U.S. nº 62/065.526, depositado no dia 17 de outubro de 2014, cujos conteúdos integrais são incorporados ao presente documento a título de referência.

CAMPO DA TÉCNICA

[0002] Esta revelação refere-se à codificação e decodificação 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 livro eletrônico, câmeras digitais, dispositivos de gravação digitais, reprodutores de mídia digital, dispositivos de video game, consoles de video game, celular ou telefones de rádio-satélite, conhecidos como "telefones inteligentes", dispositivos de teleconferência por vídeo, dispositivos de transmissão contínua de vídeo e similares. Os dispositivos de vídeo digital implantam técnicas de compressão de vídeo,

tais como aquelas 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, codificar, decodificar e/ou armazenar informações de vídeo digital de maneira mais eficiente implantando-se tais técnicas de compressão de vídeo.

[0004] As técnicas de compressão de vídeo realizam predição espacial (intraimagem) e/ou predição temporal (interimagem) para reduzir ou remover a redundância inerente em sequências de vídeo. Para codificação de vídeo com base em bloco, uma fatia de vídeo (isto é, um quadro de vídeo ou uma porção de um quadro de vídeo) pode ser particionada em blocos de vídeo. Os blocos de vídeo em uma fatia intracodificada (I) de uma imagem são codificados com o uso de predição espacial em relação às amostras de referência em blocos próximos na mesma imagem. Os blocos de vídeo em uma fatia intercodificada (P ou B) de uma imagem podem usar predição espacial em relação às amostras de referência em blocos vizinhos na mesma predição de imagem ou temporal em relação às amostras de referência em outras imagens de referência. As figurações podem ser denominadas como quadros, e as figurações de referência podem ser denominadas como quadros de referência.

[0005] A predição espacial ou temporal resulta em um bloco predito para um bloco a ser codificado. Os dados residuais representam diferenças de pixel entre o bloco original a ser codificado e o bloco predito. Um bloco

intercodificado é codificado de acordo com um vetor de movimento que aponta para um bloco de amostras de referência que formam o bloco preditivo e os dados residuais indicam a diferença entre o bloco codificado e o bloco preditivo. Um bloco intracodificado é codificado 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, os quais os mesmos podem ser quantizados. O coeficiente de transformada quantizado pode ser disposto inicialmente em uma matriz bidimensional e varrido a fim de produzir um vetor monodimensional de coeficiente de transformada, e a codificação de entropia pode ser aplicada para conseguir ainda mais compressão.

SUMÁRIO DA INVENÇÃO

[0006] As técnicas desta revelação se referem à codificação de vídeo com base em paleta. Por exemplo, em codificação com base em paleta, um codificador de vídeo (um codificador de vídeo ou decodificador de vídeo) pode formar uma "paleta" como uma tabela de cores para representar os dados de vídeo da área particular (por exemplo, um determinado bloco). A codificação com base em paleta pode ser especialmente útil para áreas de codificação de dados de vídeo que têm um número relativamente pequeno de cores. Em vez da codificação dos valores de pixel atuais (ou seus residuais), o codificador de vídeo pode a codificação dos valores de índice para um ou mais dos pixels que relacionam os pixels com entradas na paleta que representa as cores dos pixels. As técnicas descritas nesta revelação podem incluir técnicas para várias combinações de um ou mais dentre

predizer entradas de paleta, codificar execuções de índices de paleta e várias outras técnicas de codificação de paleta.

[0007] Em um exemplo, um método de processamento de dados de vídeo inclui determinar uma primeira paleta para um primeiro bloco de dados de vídeo que está localizado em uma primeira fileira de blocos, em que a primeira paleta compreende uma ou mais entradas de paleta em que cada uma inclui um índice de paleta que está associado a um valor de cor para a codificação do bloco de dados de vídeo, gerar uma paleta de preditor para construir pelo menos uma segunda paleta de pelo menos um segundo bloco de dados de vídeo na primeira fileira de blocos codificados após o primeiro bloco em que a paleta de preditor inclui pelo menos uma entrada de paleta a partir de um ou mais blocos da primeira fileira diferentes do primeiro bloco, reinicializar a paleta de preditor para determinar uma terceira paleta de um terceiro bloco de dados de vídeo que está localizado em uma segunda fileira de blocos, em que a reinicialização da paleta de preditor compreende reinicializar a paleta de preditor com base nas uma ou mais entradas de paleta da primeira paleta ou uma paleta de preditor inicial gerada após a codificação do primeiro bloco, determinar a terceira paleta do terceiro bloco com base na paleta de preditor reinicializada e codificar o terceiro bloco com o uso da terceira paleta.

[0008] Em um outro exemplo, um dispositivo para o processamento de dados de vídeo inclui uma memória configurada para armazenar um primeiro bloco de dados de vídeo, pelo menos um segundo bloco de dados de vídeo e um terceiro bloco de dados de vídeo e um ou mais processadores

configurada para determinar uma primeira paleta para o primeiro bloco de dados de vídeo que está localizado em uma primeira fileira de blocos, em que a primeira paleta compreende uma ou mais entradas de paleta em que cada uma inclui um índice de paleta que está associado a um valor de cor para a codificação do bloco de dados de vídeo, gerar uma paleta de preditor para construir pelo menos uma segunda paleta do pelo menos um segundo bloco de dados de vídeo na primeira fileira de blocos codificados após o primeiro bloco, em que a paleta de preditor inclui pelo menos uma entrada de paleta a partir de um ou mais blocos da primeira fileira diferentes do primeiro bloco, reinicializar a paleta de preditor para determinar uma terceira paleta do terceiro bloco de dados de vídeo que está localizado em a segunda fileira de blocos, em que a reinicialização da paleta de preditor compreende reinicializar a paleta de preditor com base nas uma ou mais entradas de paleta da primeira paleta ou uma paleta de preditor inicial gerada após a codificação do primeiro bloco, determinar a terceira paleta do terceiro bloco com base na paleta de preditor reinicializada e codificar o terceiro bloco com o uso da terceira paleta.

[0009] Em um outro exemplo, um aparelho para o processamento de dados de vídeo inclui meios para determinar uma primeira paleta para um primeiro bloco de dados de vídeo que está localizado em uma primeira fileira de blocos, em que a primeira paleta compreende uma ou mais entradas de paleta em que cada uma inclui um índice de paleta que está associado a um valor de cor para a codificação do bloco de dados de vídeo, meios para gerar uma paleta de preditor para construir pelo menos uma segunda paleta de pelo menos um

segundo bloco de dados de vídeo na primeira fileira de blocos codificados após o primeiro bloco, em que a paleta de preditor inclui pelo menos uma entrada de paleta a partir de um ou mais blocos da primeira fileira diferentes do primeiro bloco, meios para reinicializar a paleta de preditor para determinar uma terceira paleta de um terceiro bloco de dados de vídeo que está localizado em a segunda fileira de blocos, em que a reinicialização da paleta de preditor compreende reinicializar a paleta de preditor com base nas uma ou mais entradas de paleta da primeira paleta ou uma inicial gerada após a codificação do primeiro bloco, meios para determinar a terceira paleta do terceiro bloco com base na paleta de preditor reinicializada e meios para codificar o terceiro bloco com o uso da terceira paleta.

[0010] Em um outro exemplo, um meio legível por computador não transitório tem instruções armazenadas no mesmo que, quando executadas, fazem com que um ou mais processadores determinem uma primeira paleta para um primeiro bloco de dados de vídeo que está localizado em uma primeira fileira de blocos, em que a primeira paleta compreende uma ou mais entradas de paleta em que cada uma inclui um índice de paleta que está associado a um valor de cor para a codificação do bloco de dados de vídeo, gerar uma paleta de preditor para construir pelo menos uma segunda paleta de pelo menos um segundo bloco de dados de vídeo na primeira fileira de blocos codificados após o primeiro bloco, em que a paleta de preditor inclui pelo menos uma entrada de paleta a partir de um ou mais blocos da primeira fileira diferentes do primeiro bloco, reinicializar a paleta de preditor para determinar uma terceira paleta de um terceiro

bloco de dados de vídeo que está localizado em a segunda fileira de blocos, em que a reinicialização da paleta de preditor compreende reinicializar a paleta de preditor com base nas uma ou mais entradas de paleta da primeira paleta ou uma paleta de preditor inicial gerada após a codificação do primeiro bloco, determinar a terceira paleta do terceiro bloco com base na paleta de preditor reinicializada e codificar o terceiro bloco com o uso da terceira paleta.

[0011] Em um outro exemplo, um método de processamento de dados de vídeo inclui determinar um valor de execução que indica um comprimento de execução de uma execução de um índice de paleta de um bloco de dados de vídeo, em que o índice de paleta está associado a um valor de cor em uma paleta de color valores para a codificação do bloco de dados de vídeo, determinar um contexto para a codificação adaptativa ao contexto que representa o valor de execução com base no índice de paleta e codificar os dados que representam o valor de execução de um fluxo de bits com o uso do contexto determinado.

[0012] Em um outro exemplo, um dispositivo para o processamento de dados de vídeo inclui uma memória configurada para armazenar um bloco de dados de vídeo e um ou mais processadores configurados para determinar um valor de execução que indica um comprimento de execução de uma execução de um índice de paleta do bloco de dados de vídeo, em que o índice de paleta está associado a um valor de cor em uma paleta de color valores para a codificação do bloco de dados de vídeo, determinar um contexto para a codificação adaptativa ao contexto que representa o valor de execução com base no índice de paleta e codificar os dados que

representam valor de execução a partir de um fluxo de bits com o uso do contexto determinado.

[0013] Em um outro exemplo, um aparelho para o processamento de dados de vídeo inclui meios para determinar um valor de execução que indica um comprimento de execução de uma execução de um índice de paleta de um bloco de dados de vídeo, em que o índice de paleta está associado a um valor de cor em uma paleta de codificação adaptativa ao contexto que representa o valor de execução com base no índice de paleta; e meios para a codificação dos dados que representam o valor de execução a partir de um fluxo de bits com o uso do contexto determinado.

[0014] Em um outro exemplo, um meio legível por computador não transitório tem instruções armazenadas no mesmo que, quando executadas, fazem com que um ou mais processadores determinem um valor de execução que indica um comprimento de execução de uma execução de um índice de paleta de um bloco de dados de vídeo, em que o índice de paleta está associado a um valor de cor em uma paleta de color valores para a codificação do bloco de dados de vídeo, determinar um contexto for codificação adaptativa ao contexto que representa o valor de execução com base no índice de paleta e codificar os dados que representam o valor de execução a partir de um fluxo de bits com o uso do contexto determinado.

[0015] Em um outro exemplo, um método de processamento de dados de vídeo inclui determinar os índices de paleta de uma primeira fileira de um bloco de dados de vídeo, em que os índices de paleta correspondem a uma paleta de uma ou mais cores para a codificação do bloco de dados de

vídeo e em que os índices de paleta da primeira fileira incluem um ou mais índices que estão associados a um valor de cor na paleta e um elemento de sintaxe que não está associado a um valor de cor na paleta e codificar uma execução de índices de paleta de uma segunda fileira do bloco de dados de vídeo em relação aos índices de paleta da primeira fileira, em que a execução inclui o elemento de sintaxe que não está associado a um valor de cor na paleta.

[0016] Em um outro exemplo, um dispositivo para o processamento de dados de vídeo inclui uma memória configurada para armazenar um bloco de dados de vídeo e um ou mais processadores configurada para determinar índices de paleta de uma primeira fileira do bloco de dados de vídeo, em que os índices de paleta correspondem a uma paleta de uma ou mais cores para a codificação do bloco de dados de vídeo e em que os índices de paleta da primeira fileira incluem um ou mais índices que estão associados a um valor de cor na paleta e um elemento de sintaxe que não está associado a um valor de cor na paleta e codificar uma execução de índices de paleta de uma segunda fileira do bloco de dados de vídeo em relação aos índices de paleta da primeira fileira, em que a execução inclui os um ou mais índices que estão associados a um valor de cor na paleta e o elemento de sintaxe que não está associado a um valor de cor na paleta.

[0017] Em um outro exemplo, um aparelho para o processamento de dados de vídeo inclui meios para determinar índices de paleta de uma primeira fileira de um bloco de dados de vídeo, em que os índices de paleta correspondem a uma paleta de uma ou mais cores para a codificação do bloco de dados de vídeo e em que os índices de paleta da primeira

fileira incluem um ou mais índices que estão associados a um valor de cor na paleta e um elemento de sintaxe que não está associado à segunda fileira do bloco de dados de vídeo em relação aos índices de paleta da primeira fileira, em que a execução inclui os um ou mais índices que estão associados a um valor de cor na paleta e o elemento de sintaxe que não está associado a um valor de cor na paleta.

[0018] Em um outro exemplo, um meio legível por computador não transitório tem instruções armazenadas no mesmo que, quando executadas, fazem com que um ou mais processadores determinem os índices de paleta de uma primeira fileira de um bloco de dados de vídeo, em que os índices de paleta correspondem a uma paleta de uma ou mais cores para a codificação do bloco de dados de vídeo e em que os índices de paleta da primeira fileira incluem um ou mais índices que estão associados a um valor de cor na paleta e um elemento de sintaxe que não está associado a um valor de cor na paleta e codifique uma execução de índices de paleta de uma segunda fileira do bloco de dados de vídeo em relação aos índices de paleta da primeira fileira, em que a execução inclui os um ou mais índices que estão associados a um valor de cor na paleta e o elemento de sintaxe que não está associado a um valor de cor na paleta.

[0019] Os detalhes de um ou mais exemplos da revelação são estabelecidos nos desenhos anexos e na descrição abaixo. Outros objetivos, aspectos e vantagens ficarão evidentes a partir da descrição, dos desenhos e das reivindicações.

BREVE DESCRIÇÃO DOS DESENHOS

[0020] A Figura 1 é um diagrama de blocos que

ilustra um sistema de codificação de vídeo exemplificativo que pode utilizar as técnicas descritas nesta revelação.

[0021] A Figura 2 é um diagrama de blocos que ilustra um codificador de vídeo exemplificativo que pode implantar as técnicas descritas nesta revelação.

[0022] A Figura 3 é um diagrama de blocos que ilustra um decodificador de vídeo exemplificativo que pode implantar as técnicas descritas nesta revelação.

[0023] A Figura 4 é um diagrama conceitual que ilustra um exemplo de determinação de entradas de paleta para codificação de vídeo com base em paleta, consistente com as técnicas desta revelação.

[0024] A Figura 5 é um diagrama conceitual que ilustra pequenos blocos (tiles) exemplificativos de acordo com o padrão de Codificação de Vídeo de Alta Eficiência (HEVC).

[0025] A Figura 6 é um diagrama conceitual que ilustra frentes de onda para o processamento paralelo de frente de onda (WPP).

[0026] A Figura 7 é um diagrama conceitual que ilustra um exemplo de paleta de preditor redefinido na presença de múltiplos pequenos blocos e frentes de onda, consistentes com as técnicas desta revelação.

[0027] A Figura A Figura 8 é um diagrama conceitual que ilustra um exemplo de determinação de índices para uma paleta para um bloco de pixels, consistente com as técnicas desta revelação.

[0028] A Figura 9 é um fluxograma que ilustra um processo exemplificativo para determinar uma paleta de preditor, consistente com as técnicas desta revelação.

[0029] A Figura 10 é um fluxograma que ilustra um processo exemplificativo para a codificação de um valor de execução de uma execução de índices de paleta, consistente com técnicas desta revelação.

[0030] A Figura 11 é um fluxograma que ilustra um processo exemplificativo para a codificação de uma execução de índices codificados maiores do que um modo de codificação de paleta, consistente com as técnicas desta revelação.

DESCRIÇÃO DETALHADA

[0031] Os aspectos desta revelação são direcionados às técnicas para codificação de vídeo e compressão de dados de vídeo. Em particular, esta revelação descreve técnicas para codificar com base em paleta de dados de vídeo. Em codificação de vídeo tradicional, presume-se que as imagens sejam tom contínuo e espacialmente suaves. Com base nessas suposições, várias ferramentas foram desenvolvidas tais como transformada com base em bloco, filtragem, e outras ferramentas de codificação e tais ferramentas têm mostrado bom desempenho para vídeos de conteúdo natural.

[0032] Entretanto, em aplicações do tipo desktop remoto, trabalho colaborativo e visor sem fio, o conteúdo de tela gerada por computador pode ser o conteúdo dominante a ser comprimido. Esse tipo de conteúdo tende a ter tom distinto, linhas acentuadas de recurso e limites de objeto de alto contraste presentes. A suposição de tom contínuo e lisura pode não mais ser aplicada e, então, as técnicas de codificação de vídeo tradicional podem ser ineficazes na compressão do conteúdo.

[0033] Esta revelação descreve a codificação com base em paleta, que pode ser particularmente adequada para codificar de conteúdo gerado por tela ou outro conteúdo em que uma ou mais ferramentas de codificação tradicionais são ineficazes. As técnicas para codificar com base em paleta de dados de vídeo podem ser usadas com uma ou mais outras técnicas de codificação, tais como técnicas para codificar interpreditiva ou codificação intrapreditiva. Por exemplo, conforme descrito em maiores detalhes abaixo, um codificador ou decodificador, ou codificador-decodificador combinado (codec), pode ser configurado para realizar codificação inter e intrapreditiva, bem como codificação com base em paleta.

[0034] Em alguns exemplos, as técnicas de codificação com base em paleta podem ser configuradas para uso com um ou mais padrões de codificação de vídeo. Por exemplo, a Codificação de Vídeo de Alta Eficácia (HEVC) é um novo padrão de codificação de vídeo que é desenvolvido pela Equipe de Colaboração Conjunta em Codificação de Vídeo (JCT-VC) do Grupo de Especialistas de Codificação de Vídeo de ITU-T (VCEG) e do Grupo de Especialistas de Filme Cinematográfico de ISO/IEC (MPEG). Um rascunho de relatório descritivo de texto de HEVC recente é descrito em Bross et al., "High Efficiency Video Coding (HEVC) Text Specification Draft 10 (for FDIS & Consent)", JCVCLI003_v13, 12º Encontro de JCT-VC de ITU-T SG16 WP 3 e ISO/IEC JCT 1/SC 29/WG 11 , 14 a 23 Janeiro de 2013 ("HEVC Draft 10").

[0035] Em relação à estrutura de HEVC, como um exemplo, as técnicas de codificação com base em paleta podem ser configuradas para serem usadas como um modo de unidade

de controle (CU). Em outros exemplos, as técnicas de codificação com base em paleta podem ser configuradas para ser usadas como um modo de PU na estrutura de HEVC. Consequentemente, todos os processos revelados a seguir descritos no contexto de um modo de CU podem, adicional ou alternativamente, se aplicar à PU. No entanto, esses exemplos com base em HEVC não seriam considerados a uma restrição ou limitação da codificação com base em paleta técnicas descritas no presente documento, tais como técnicas que podem ser aplicadas para funcionar de modo independente ou como parte de outros existentes ou ainda a serem sistemas/padrões desenvolvidos. Nesses casos, a unidade para codificar de paleta pode ser blocos quadrados, blocos retangulares ou mesmo regiões com formato não retangular.

[0036] Em codificação com base em paleta, pode-se considerar que uma área particular de dados de vídeo tenha um número relativamente pequeno de cores. Um codificador de vídeo (um codificador de vídeo ou decodificador de vídeo) pode codificar chamado de "paleta" como uma tabela de cores para representar os dados de vídeo da área particular (por exemplo, um determinado bloco). Cada pixel pode estar associado a uma entrada na paleta que representa a cor do pixel. Por exemplo, o codificador de vídeo pode codificar um índice que relaciona o valor de pixel ao valor apropriado na paleta.

[0037] No exemplo acima, o codificador de vídeo 20 pode codificar um bloco de dados de vídeo pela determinação de uma paleta for o bloco, localização de uma entrada na paleta para representar o valor de cada pixel, e codificação da paleta com valores de índice para os pixels

que relacionam o valor de pixel à paleta. Um decodificador de vídeo pode obter, a partir de um fluxo de bits codificado, uma paleta para um bloco, bem como valores de índice para os pixels do bloco. O decodificador de vídeo pode relacionar os valores de índice dos pixels às entradas da paleta para reconstruir os valores de pixel do bloco. Os pixels (e/ou valores de índice relacionados que indicam um valor de pixel) podem, em geral, ser referidos como amostras.

[0038] Presume-se que as amostras no bloco sejam processadas (por exemplo, varridas) com o uso de ordem de varredura por rastreamento horizontal. Por exemplo, o codificador de vídeo pode converter um bloco bidimensional de valores de índice em uma matriz unidirecional pela varredura dos índices de paleta com o uso de uma ordem de varredura por rastreamento horizontal. Do mesmo modo, o decodificador de vídeo pode reconstruir um bloco de índices com o uso da ordem de varredura por rastreamento horizontal. Consequentemente, esta revelação pode se referir a uma amostra anterior como uma amostra que precede a amostra que é atualmente codificada no bloco na ordem de varredura. Deve ser apreciado que as varreduras diferentes de uma varredura por rastreamento horizontal, tal como ordem de varredura por rastreamento vertical, também podem ser aplicáveis. O exemplo acima se destina a fornecer uma descrição geral de codificação com base em paleta.

[0039] Uma paleta inclui tipicamente entradas numeradas por um índice e que representam pelo menos um dentre componente de cor (por exemplo, pelo menos em um componente de RGB, YUV ou similares), valores ou intensidades. Tanto um codificador de vídeo quanto um

decodificador de vídeo determinam o número de entradas de paleta, valores de componente de cor para cada entrada de paleta e a ordenação exata das entradas de paleta para o bloco atual. Nesta revelação, presume-se que cada entrada de paleta especifica os valores para todos os componentes de cor de uma amostra. Entretanto, os conceitos desta revelação são aplicáveis para o uso de uma paleta separada para cada componente de cor.

[0040] Em alguns exemplos, uma paleta pode ser composta com o uso de informações dos blocos anteriormente codificados ou paletas anteriormente codificadas. Ou seja, uma paleta pode conter entradas de paleta preditas a partir da(s) paleta(s) usadas para codificar o(s) bloco(s) anterior(es). Por exemplo, conforme descrito no documento de submissão padrão Wei Pu et al, "AHG10: Suggested Software for Palette Coding based on RExt6.0", JCTVC-Q0094, Valência, ES, 27 de março a 4 de abril de 2014 (doravante no presente documento JCTVC-Q0094), uma paleta pode incluir entradas que são copiadas de uma paleta de preditor. Uma paleta de preditor pode incluir entradas de paleta a partir dos blocos anteriormente codificados com o uso de modo de paleta ou outras amostras reconstruídas. A paleta de preditor pode usar toda a ou parte de uma paleta anteriormente codificada, ou pode ser composta por entradas de diversas paletas anteriormente codificadas.

[0041] Em alguns exemplos, para cada entrada na paleta de preditor, um sinalizador binário pode ser codificado para indicar se a entrada associada ao sinalizador é copiada para a paleta atual (por exemplo, indicada pelo sinalizador = 1). A coluna de sinalizadores binários pode

ser referida como o vetor de predição de paleta binário. A paleta para a codificação de um bloco atual também pode incluir inúmeras de novas entradas de paleta, que podem ser explicitamente codificadas (por exemplo, separadamente do vetor de predição de paleta). Uma indicação do número de novas entradas também pode ser codificada. Uma soma das entradas preditas e das novas entradas pode indicar o tamanho de paleta total para o bloco.

[0042] Conforme proposto JCTVC-Q0094, cada amostra em um bloco codificado com um modo de codificação com base em paleta pode ser codificada com o uso de um dos três modos de paleta, conforme apresentado abaixo:

[0043] • Modo Fuga: nesse modo, o valor de amostra não está incluído em uma paleta como uma entrada de paleta e o valor de amostra quantizado é sinalizado explicitamente para todos os componentes de cor. É similar à sinalização das novas entradas de paleta, enquanto para novas entradas de paleta, os valores de componente de cor não são quantizados.

[0044] • Modo CopyFromTop (também referido como modo CopyAbove): nesse modo, o índice de entrada de paleta para a amostra atual é copiado da amostra localizada diretamente acima em um bloco.

[0045] • Modo Valor (também referido como modo de índice): nesse modo, o valor do índice de entrada de paleta é explicitamente sinalizado.

[0046] Conforme descrito no presente documento, um índice de entrada de paleta pode ser referido como um índice de paleta ou índice simples. Esses termos podem ser usados de modo intercambiável para descrever as técnicas

desta revelação. Além disso, conforme descrito em maiores detalhes abaixo, um índice de paleta pode ter um ou mais valores de intensidade ou de cor associados. Por exemplo, um índice de paleta pode ter um valor de intensidade ou de cor associado único associado a um componente de intensidade ou de cor único de um pixel (por exemplo, um componente Vermelho de dados de RGB, um componente Y de dados YUV, ou similares). Em outro exemplo, um índice de paleta pode ter múltiplos valores de intensidade ou de cor associados. Em alguns casos, a codificação com base em paleta pode ser aplicada para codificar o vídeo monocromo. Consequentemente, o "valor de cor" pode, em geral, se referir a qualquer componente de cor ou sem cor usada para gerar um valor de pixel.

[0047] Para os modos CopyFromTop e Valor, um valor de execução (que também pode ser referido simplesmente como execução) também pode ser sinalizado. Um valor de execução pode indicar um número de amostras consecutivas (por exemplo, uma execução de amostras) em uma ordem de varredura em particular em um bloco codificado por paleta que são codificadas em conjunto. Em alguns casos, a execução de amostras também pode ser denominada como uma execução de índices de paleta, devido ao fato de que cada amostra da execução tem um índice associado a uma paleta.

[0048] Um valor de execução pode indicar uma execução de índices de paleta que são codificados com o uso do mesmo modo de codificação de paleta. Por exemplo, em relação ao modo Valor, um codificador de vídeo (um codificador de vídeo ou decodificador de vídeo) pode codificar um índice de paleta (também referido como um valor de índice de paleta ou valor de índice simples) e um valor

de execução que indica inúmeras amostras consecutivas em uma ordem de varredura que têm o mesmo índice de paleta e que estão sendo codificadas com o índice de paleta. Em relação ao modo CopyFromTop, o codificador de vídeo pode codificar uma indicação de que um índice para o valor de amostra atual ser copiado com base em um índice de uma amostra vizinha acima (por exemplo, uma amostra que está posicionada acima da amostra que é atualmente codificada em um bloco) e um valor que indica inúmeras amostras consecutivas em uma ordem de varredura que também copiam um índice de paleta de uma amostra vizinha acima e que estão sendo codificadas com o índice de paleta. Consequentemente, nos exemplos acima, uma execução de índices de paleta se refere a uma execução de índices de paleta que tem o mesmo valor ou uma execução de índices de paleta que são copiados a partir dos índices de paleta vizinha acima.

[0049] Consequentemente, a execução pode especificar, para um determinado modo, o número de amostras subsequentes que pertence ao mesmo modo. Em alguns casos, a sinalização de um índice e um valor de execução pode ser similar à codificação de comprimento de execução. Em um exemplo para propósitos de ilustração, uma coluna de índices consecutivos de um bloco pode ser 0, 2, 2, 2, 2, 5 (por exemplo, em que cada índice corresponde a uma amostra no bloco). Nesse exemplo, um codificador de vídeo pode codificar a segunda amostra (por exemplo, o primeiro valor de índice de dois) com o uso de modo Valor. Após a codificação de um índice que é igual a 2, o codificador de vídeo pode codificar uma execução de três, que indica que as três amostras subsequentes também têm o mesmo valor de índice de dois. De

uma maneira similar, a codificação de uma execução de quatro valores de índice após a codificação de um índice com o uso de modo CopyFromTop pode indicar que um total de cinco índices é copiado a partir dos índices correspondentes na fileira acima da posição de amostra que é atualmente codificada.

[0050] Conforme descrito em maiores detalhes abaixo, um codificador de vídeo (por exemplo, um codificador de vídeo e um decodificador de vídeo) pode codificar ou decodificar os dados que indicam se uma amostra é codificada como uma amostra de fuga em uma base por amostra. As amostras de fuga (também referidas como pixels de fuga) podem ser amostras (ou pixels) de um bloco que não tem uma cor correspondente representada em uma paleta para a codificação do bloco. Consequentemente, as amostras de fuga não podem ser reconstruídas com o uso de uma entrada de cor (ou valor de pixel) de uma paleta. Em vez disso, os valores de cor para amostras de fuga são sinalizados em um fluxo de bits de modo separado dos valores de cor da paleta. Em geral, a codificação de uma amostra com o uso de "modo Fuga" podem, em geral, se referir a codificação de uma amostra de um bloco que não tem uma cor correspondente representada em uma paleta para a codificação do bloco. Conforme notado acima, tais amostras podem ser referidas como amostras de fuga ou pixels de fuga.

[0051] Em alguns exemplos, o codificador de vídeo pode codificar um sinalizador para cada amostra que indica se a amostra é codificada como uma amostra de fuga (essa técnica pode ser referida como sinalização de fuga explícita, conforme descrito em maiores detalhes abaixo em

relação ao valor de índice para uma paleta indicar que uma amostra particular é codificada como uma amostra de fuga (essa técnica pode ser referida como sinalização de fuga implícita, conforme descrito em maiores detalhes em relação ao exemplo da Figura 1).

[0052] As técnicas descritas nesta revelação podem incluir técnicas para várias combinações dentre prever entradas de paleta, codificar execuções de índices de paleta e uma variedade de outras técnicas de codificação de paleta. Conforme descrito em maiores detalhes abaixo, as técnicas desta revelação podem, em alguns casos, aprimorar a eficiência e aprimorar a taxa de bits ao codificar os dados de vídeo com o uso de um modo de paleta.

[0053] Por exemplo, certos aspectos desta revelação são direcionados às técnicas para prever entradas de paleta para um bloco de dados de vídeo. Em alguns exemplos, uma paleta de preditor pode ser redefinida para um bloco na borda à esquerda de uma figuração. Isto é, ao gerar uma paleta para o bloco na borda à esquerda da figuração, a paleta de preditor pode ser redefinida para zero (por exemplo, a paleta de preditor não tem entradas e a paleta atual não é predita com o uso de uma paleta de preditor). O indicador pode ser redefinido, dessa maneira, devido ao fato de que uma paleta de preditor pode incluir entradas de paleta que pertencem principalmente às paletas de blocos localizadas no lado direito da figuração após decodificar uma linha (por exemplo, assumir uma varredura por rastreamento da esquerda para a direita). Consequentemente, mediante a codificação do primeiro bloco da fileira a seguir (o bloco na borda mais à esquerda da figuração), a paleta de preditor

pode incluir cores de blocos que estão localizadas relativamente distantes do bloco que é atualmente codificado. Desse modo, a paleta de preditor não pode ser muito eficaz na predição de uma paleta atual para o bloco atual (por exemplo, as cores de pixels no lado à esquerda de uma figuração podem diferir das cores de pixels no lado direito da figuração).

[0054] No entanto, em alguns exemplos, a redefinição de uma paleta de preditor pode levar às perdas de codificação. Por exemplo, sem uma paleta de preditor, um codificador de vídeo (um codificador de vídeo 20 ou decodificador de vídeo) pode codificar os dados que indicam todas as entradas de uma paleta (por exemplo, todos os índices de paleta e valores de cor relacionada) em um fluxo de bits. Essa pode ser uma quantidade relativamente grande de dados em relação aos dados associada às entradas de paleta preditas. Consequentemente, a redefinição de uma paleta de preditor pode afetar adversamente a taxa de bits de dados de vídeo codificados com a codificação com base em paleta.

[0055] De acordo com os aspectos desta revelação, uma paleta de preditor para construir uma paleta para um bloco de dados de vídeo em uma primeira linha pode ser reinicializada com base em um ou mais blocos de uma outra linha. Por exemplo, um codificador de vídeo pode determinar uma primeira paleta para um primeiro bloco de dados de vídeo que está localizado em uma primeira fileira de blocos. Mediante a codificação um bloco em uma segunda fileira, o codificador de vídeo pode reinicializar a paleta de preditor para determinar uma paleta do bloco na segunda fileira com

base nas entradas da primeira paleta. De acordo com alguns aspectos desta revelação, a fileira de blocos pode ser uma fileira de blocos de um certo tamanho (por exemplo, uma fileira de unidade de árvore de codificação (CTUs), conforme descrito abaixo). O comprimento de fileira pode representar uma largura de figuração nas unidades de bloco selecionado e o número de fileiras pode representar a altura de figuração nas unidades de bloco selecionado.

[0056] Em alguns exemplos, a paleta de preditor reinicializada inclui entradas de uma paleta de um ou mais blocos que estão posicionados relativamente perto do bloco que é atualmente codificado. Consequentemente, a paleta de preditor pode incluir entradas que têm uma maior probabilidade de estar incluída em uma paleta do bloco que é atualmente codificada (por exemplo, em relação a uma paleta de preditor com base em blocos localizadas na direção contrária do bloco atual ou uma paleta de preditor que foi redefinida). Dessa maneira, as técnicas desta revelação podem aumentar a eficiência de codificação, devido ao fato de que o codificador de vídeo pode determinar a paleta para o bloco atual com o uso da paleta de preditor em vez da codificação de as entradas da paleta no fluxo de bits.

[0057] Outros aspectos desta revelação são direcionados à codificação (isto é, codificar ou decodificar) de um valor de execução que indica um comprimento de execução de uma execução de índices de paleta. Por exemplo, conforme notado acima, uma execução pode especificar, para um índice que é atualmente codificado com um determinado modo, o número de amostras subsequentes que são codificadas com o índice atual com o uso do mesmo modo.

[0058] Em alguns exemplos, os dados que indicam o valor de execução podem ser codificados com o uso de uma técnica de codificação de contexto adaptativo, tal como Codificação Aritmética Binária Adaptativa de Contexto (CABAC), codificação de comprimento variável adaptativa ao contexto (CAVLC) ou uma outra técnica de codificação adaptativa ao contexto. Por exemplo, um codificador de vídeo (um codificador de vídeo ou um decodificador de vídeo) pode selecionar a modelo de probabilidade ou "modelo de contexto" que opera no contexto para codificar símbolos associados a um bloco de dados de vídeo. Isto é, o modelo de contexto (Ctx) pode ser um índice ou desvio que é aplicado para selecionar um dentre uma pluralidade de diferentes contextos, cada um dos quais pode corresponder a um modelo de probabilidade particular.

[0059] Em alguns exemplos, um único contexto pode ser usado para codificar os dados que indicam um valor de execução. Por exemplo, um valor de execução binarizado pode incluir um primeiro bin que indica se o valor de execução é maior do que zero, um segundo bin que indica se o valor de execução é maior do que uma, um terceiro bin que indica se o valor de execução é maior do que dois e quaisquer outros bins necessários para representar o valor de execução. Dessa maneira, o mesmo contexto pode ser o mesmo modelo de probabilidade para codificar múltiplos bins que podem criar um atraso entre sucessivos ciclos de codificação. Ademais, a correlação de bins de um valor de execução não pode ser suficiente para garantir o tempo e recursos computacionais associados à atualização do modelo de probabilidade.

[0060] De acordo com os aspectos desta

revelação, o contexto pode ser selecionado com base no valor de índice do valor de execução que é codificado. Por exemplo, um codificador de vídeo pode determinar um valor de execução que indica um comprimento de execução de uma execução de um índice de paleta de um bloco de dados de vídeo. O codificador de vídeo também pode determinar um contexto para dados de codificação adaptativa ao contexto que representam o valor de execução com base no valor de índice de paleta. As técnicas podem aprimorar a eficiência de codificação.

[0061] Em alguns exemplos, o índice de paleta usado para derivar os contextos para codificar de execução pode ser um índice de paleta que é usado para recuperar um valor de cor a partir da paleta. Em outros exemplos, o índice de paleta usado para derivar os contextos para codificar de execução pode ser um índice de paleta analisado, isto é, o índice de paleta sinalizado em um fluxo de bits (que pode ser diferente do índice de paleta que é usado para acessar as cores de paleta, conforme descrito em maiores detalhes abaixo).

[0062] Em alguns exemplos, três contextos podem ser usados para codificar um valor de execução. Em tais exemplos, de acordo com os aspectos desta revelação, um codificador de vídeo pode selecionar um primeiro contexto para codificar o valor de execução com base no índice que é maior do que zero. O codificador de vídeo pode selecionar um primeiro contexto para codificar o valor de execução com base no índice que é maior do que zero. O codificador de vídeo pode selecionar um segundo contexto para codificar o valor de execução com base no índice que é maior do que um. O codificador de vídeo pode selecionar um terceiro contexto

para codificar o valor de execução com base no índice que é maior do que dois. Em relação aos três exemplos de bin codificado de contexto descrito acima, o codificador de vídeo pode selecionar os três contextos para a codificação de qualquer combinação dos três bins. Enquanto o exemplo acima é descrito em relação à definição de três contextos associados às três características do valor de índice, deve ser entendido que as técnicas descritas no mesmo podem ser estendidas para a definição de outros números de contextos com base em outras características do valor de índice.

[0063] Outros aspectos desta revelação são direcionados às execuções de codificação de índices de paleta que são codificadas com o uso de mais do que um modo de paleta. Em geral, os índices que são codificados com o uso de diferentes modos de paleta não podem ser codificados na mesma execução. Em um exemplo para propósitos ilustrativos, uma execução codificada com o uso do modo CopyFromTop não pode incluir quaisquer índices que são codificados como nas amostras de fuga. Dessa maneira, um pixel codificado como uma amostra de fuga pode terminar uma execução, que pode impactar adversamente a eficiência de codificação devido às execuções relativamente mais curtas para um determinado bloco.

[0064] De acordo com os aspectos desta revelação, uma execução pode incluir amostras que são codificadas com o uso de mais do que um modo de paleta. Por exemplo, um codificador de vídeo pode determinar índices de paleta de uma primeira fileira de um bloco de dados de vídeo, em que os índices de paleta incluem um ou mais índices que estão associados a um valor de cor na paleta e um elemento

de sintaxe que não está associado a um valor de cor na paleta (que pode ser referido como um índice de paleta apesar de o elemento de sintaxe não corresponder necessariamente a um valor de índice). O codificador de vídeo também pode codificar uma execução de índices de paleta de uma segunda fileira do bloco de dados de vídeo em relação aos índices de paleta da primeira fileira, em que a execução inclui os um ou mais índices que estão associados a um valor de cor na paleta e o elemento de sintaxe que não está associado a um valor de cor na paleta.

[0065] Em alguns exemplos, a execução de índices de paleta pode incluir pixels que são codificados tanto com o modo CopyFromTop quanto como amostras de fuga, por exemplo, com o uso do modo Fuga. Por exemplo, os índices que estão associados a um valor de cor na paleta podem ser codificados com o modo CopyFromTop e o elemento de sintaxe que não está associado a um valor de cor na paleta pode ser codificado como uma amostra de fuga. Em alguns exemplos, conforme descrito em maiores detalhes abaixo, o elemento de sintaxe pode estar associado a um índice que não está associado a um valor de cor na paleta. Em alguns exemplos, os valores de pixels codificados como amostras de fuga podem ser sinalizados após a execução de índices e não se exige que os pixels codificados como amostras de fuga sejam iguais. Em outros exemplos, uma amostra de fuga pode ser representada pelo modo CopyFromTop e uma amostra de fuga pode estar incluída no mesmo grupo de pixels juntamente com as amostras codificadas sem fuga (por exemplo, pixels para os quais os valores de color são representados na paleta), em que o grupo de pixels é identificado por um valor de execução. Dessa

maneira, as técnicas podem ser usadas para aumentar o comprimento de execuções, que pode aprimorar a eficiência de codificação.

[0066] A Figura 1 é um diagrama de blocos que ilustra um sistema de codificação de vídeo exemplificativo 10 que pode utilizar as técnicas desta revelação. Conforme usado no presente documento, o termo "codificador de vídeo" se refere genericamente tanto a codificadores de vídeo quanto a decodificadores de vídeo. Nesta revelação, os termos "codificação de Vídeo" ou "codificação" podem se referir genericamente a codificação de vídeo e decodificação de vídeo. O codificador de vídeo 20 e o decodificador de vídeo 30 do sistema de codificação de vídeo 10 representa exemplos de dispositivos que podem ser configurados para realizar técnicas para codificação de vídeo com base em paleta de acordo com vários exemplos descritos nesta revelação. Para diversos blocos de dados de vídeo, tais como CUs ou PUs na codificação de HEVC, com uso de codificação com base em paleta ou codificação sem base em paleta. Os modos de codificação não com base em paleta podem se referir a vários modos de codificação temporais interpreditivo ou modos de codificação espaciais intrapreditivo, tais como vários modos de codificação especificado por HEVC Draft 10.

[0067] Conforme mostrado na Figura 1, o sistema de codificação de vídeo 10 inclui um dispositivo de origem 12 e um dispositivo de destino 14. O dispositivo de origem 12 gera dados de vídeo codificados. Consequentemente, o dispositivo de origem 12 pode ser referido como um dispositivo de codificação de vídeo ou um aparelho de codificação de vídeo. O dispositivo de destino 14 pode

decodificar os dados de vídeo codificados gerados pelo dispositivo de origem 12. Consequentemente, o dispositivo de destino 14 pode ser referido como um dispositivo de decodificação de vídeo ou um aparelho de decodificação de vídeo. O dispositivo de origem 12 e o dispositivo de destino 14 podem ser exemplos de dispositivos de codificação de vídeo ou aparelhos de codificação de vídeo.

[0068] O dispositivo de origem 12 e o dispositivo de destino 14 podem compreender uma ampla faixa de dispositivos, incluindo computadores do tipo desktop, dispositivos de computação móveis, computadores do tipo notebook (por exemplo, computadores do tipo laptop), computadores do tipo tablet, decodificadores de sinais, aparelhos de telefone como os conhecidos por telefones "inteligentes", televisões, câmeras, dispositivos de exibição, reprodutores de mídia digital, consoles de vídeo game, computadores de bordo para carros, ou similares.

[0069] O dispositivo de destino 14 pode receber dados de vídeo codificados provenientes do dispositivo de origem 12 por meio de um canal 16. O canal 16 pode compreender uma ou mais mídias ou dispositivos com capacidade para mover os dados de vídeo codificados do dispositivo de origem 12 para o dispositivo de destino 14. Em um exemplo, o canal 16 pode compreender um ou mais meios de comunicação que permitem que o dispositivo de origem 12 transmita dados de vídeo codificados diretamente para o dispositivo de destino 14 em tempo real. Nesse exemplo, o dispositivo de origem 12 pode modular os dados de vídeo codificados de acordo com um padrão de comunicação, como um protocolo de comunicação sem fio, e pode transmitir os dados de vídeo modulados para o

dispositivo de destino 14. Os um ou mais meios de comunicação podem incluir meios de comunicação sem fio e/ou com fio, como um espectro de radiofrequência (RF) ou uma ou mais linhas de transmissão físicas. Os um ou mais meios de comunicação podem fazer parte de uma rede com base em pacote, como uma rede de área local, uma rede de área ampla ou uma rede global (por exemplo, a Internet). Os um ou mais meios de comunicação podem incluir roteadores, comutadores, estações-base ou outro equipamento que facilite a comunicação a partir do dispositivo de origem 12 até o dispositivo de destino 14.

[0070] Em outro exemplo, o canal 16 pode incluir um meio de armazenamento que armazena os dados de vídeo codificados gerados pelo dispositivo de origem 12. Nesse exemplo, o dispositivo de destino 14 pode acessar o meio de armazenamento, por exemplo, por meio de acesso de disco ou acesso de cartão. Os discos de armazenamento, DVDs, CD-ROMs, memória rápida ou outros meios de armazenamento digital adequados para armazenar dados de vídeo codificados.

[0071] Em um exemplo adicional, um canal 16 pode incluir um servidor de arquivos ou outro dispositivo de armazenamento intermediário que armazene os dados de vídeo codificados gerados pelo dispositivo de origem 12. Nesse exemplo, o dispositivo de destino 14 pode acessar os dados de vídeo codificados armazenados no servidor de arquivos ou outro dispositivo de armazenamento intermediário por meio de transmissão contínua ou transferência por download. O servidor de arquivos pode ser um tipo de servidor com capacidade para armazenar dados de vídeo codificados e transmitir os dados de vídeo codificados para o dispositivo

de destino 14. Os servidores de arquivos exemplificativos incluem servidores da web (por exemplo, para um site da web), servidores de protocolo de transferência de arquivo (FTP), dispositivos de armazenamento fixado à rede (NAS) e unidades de disco local.

[0072] O dispositivo de destino 14 pode acessar os dados de vídeo codificados através de uma conexão de dados padrão, como uma conexão de Internet. Os tipos exemplificativos de conexões de dados podem incluir canais sem fio (por exemplo, conexões Wi-Fi), conexões com fio (por exemplo, (DSL)), modem a cabo, etc.), ou combinações de ambos que sejam adequadas para acessar dados de vídeo codificados armazenados em um servidor de arquivos. A transmissão de dados de vídeo codificados a partir do servidor de arquivos pode ser uma transmissão de transferência, uma transmissão de transferência por download ou uma combinação das mesmas.

[0073] As técnicas desta revelação não estão necessariamente limitadas a aplicações ou definições sem fio. As técnicas podem ser aplicadas à codificação de vídeo em apoio a uma variedade de aplicações de multimídia, como difusões de televisão sem fio, transmissões de televisão a cabo, transmissões de televisão por satélite, transmissões contínuas de vídeo, por exemplo, por meio da Internet, codificação de dados de vídeo para armazenamento em um meio de armazenamento de dados, decodificação de dados de vídeo armazenados em um meio de armazenamento de dados, ou outras aplicações. Em alguns exemplos, o sistema de codificação de vídeo 10 pode ser configurado para suportar transmissão de vídeo unidirecional ou bidirecional para suportar aplicações tais como transmissão contínua de vídeo, reprodução de vídeo,

difusão de vídeo e/ou telefonia por vídeo.

[0074] O sistema de codificação de vídeo 10 ilustrado na Figura 1 é meramente um exemplo e as técnicas desta revelação podem se aplicar a definições de codificação de vídeo (por exemplo, codificação de vídeo ou decodificação de vídeo) que não incluem necessariamente qualquer comunicação de dados entre os dispositivos de codificação e decodificação. Em outros exemplos, os dados são recuperados de uma memória local, transmitidos de modo contínuo por uma rede, ou similares. Um dispositivo de codificação de vídeo pode codificar e armazenar dados para a memória, e/ou um dispositivo de decodificação de vídeo pode recuperar e decodificar dados provenientes da memória. Em muitos exemplos, a codificação e a decodificação são realizadas por dispositivos que não se comunicam um com o outro, mas simplesmente codificam dados para a memória e/ou recuperam e decodificam dados provenientes da memória.

[0075] No exemplo da Figura 1, o dispositivo de origem 12 inclui uma fonte de vídeo 18, um codificador de vídeo 20 e uma interface de saída 22. Em alguns exemplos, a interface de saída 22 pode incluir um modulador/demodulador (modem) e/ou um transmissor. A fonte de vídeo 18 pode incluir um dispositivo de captura de vídeo, por exemplo, uma câmera de vídeo, um arquivo de vídeo que contenha dados de vídeo capturados anteriormente, uma interface de alimentação de vídeo para receber dados de vídeo a partir de um fornecedor de conteúdo de vídeo e/ou um sistema de gráficos de computador para gerar dados de vídeo, ou uma combinação de tais fontes de dados de vídeo.

[0076] O codificador de vídeo 20 pode codificar

dados de vídeo provenientes da fonte de vídeo 18. Em alguns exemplos, o dispositivo de origem 12 transmite diretamente os dados de vídeo codificados para o dispositivo de destino 14 por meio da interface de saída 22. Em outros exemplos, os dados de vídeo codificados também podem ser armazenados em um meio de armazenamento ou um servidor de arquivos para acesso posterior pelo dispositivo de destino 14 para decodificação e/ou reprodução.

[0077] No exemplo da Figura 1, o dispositivo de destino 14 inclui uma interface de entrada 28, um decodificador de vídeo 30 e um dispositivo de exibição 32. Em alguns exemplos, a interface de entrada 28 inclui um receptor e/ou um modem. A interface de entrada 28 pode receber os dados de vídeo codificados pelo canal 16. O dispositivo de exibição 32 pode ser integrado ou externo ao dispositivo de destino 14. Em geral, o dispositivo de visor 32 exibe os dados de vídeo decodificados. O dispositivo de exibição 32 pode compreender uma variedade de dispositivos de exibição, 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.

[0078] O codificador de vídeo 20 e o decodificador de vídeo 30, cada um, podem ser implantados como qualquer um dentre uma variedade de conjuntos de circuitos adequados, como um ou mais microprocessadores, processadores de sinal digital (DSPs), circuitos integrados de aplicação específica (ASICs), arranjos de portas programáveis em campo (FPGAs), lógica discreta, hardware ou quaisquer combinações dos mesmos. Se as técnicas forem implantadas parcialmente em software, um dispositivo pode

armazenar instruções para o software em um meio de armazenamento legível por computador não transitório adequado e pode executar as instruções em hardware com o uso de um ou mais processadores para realizar as técnicas desta revelação. Qualquer um dos supracitados (que inclui hardware, software, uma combinação de hardware e software, etc.) pode ser considerado como sendo um ou mais processadores. Cada um dentre o codificador de vídeo 20 e o decodificador de vídeo 30 pode estar incluído em um ou mais codificadores ou decodificadores, um dos quais pode ser integrado como parte de um codificador/decodificador (CODEC) de vídeo combinado.

[0079] Esta revelação pode se referir, de modo geral, ao codificador de vídeo 20 "que sinaliza" ou "que transmite" certas informações a outro dispositivo, como o decodificador de vídeo 30. O termo "que sinaliza" ou "que transmite" pode se referir, de modo geral, à comunicação de elementos de sintaxe e/ou outros dados usados para decodificar os dados de vídeo comprimidos. Tal comunicação pode ocorrer em tempo real ou proximamente real. De modo alternativo, tal comunicação pode ocorrer por uma extensão de tempo, como pode ocorrer quando se armazena elementos de sintaxe em um meio de armazenamento legível por computador em um fluxo de bits codificado no momento de codificação, pode, então, ser recuperado por um dispositivo de decodificação em qualquer momento após ser armazenado nesse meio.

[0080] Em alguns exemplos, o codificador de vídeo 20 e o decodificador de vídeo 30 operam de acordo com um padrão de compressão de vídeo, tal como padrão HEVC

mencionado acima e descrito em HEVC Draft 10. Além do padrão HEVC de base, há esforços contínuos para produzir codificação de vídeo escalonável, codificação de vídeo multivista, e extensões de codificação em 3D para HEVC. Além disso, a codificação com base em modos de paleta, por exemplo, conforme descrito nesta revelação, pode ser fornecida para extensão do padrão HEVC. Em alguns exemplos, as técnicas descritas nesta revelação para codificar com base em paleta podem ser aplicadas aos codificadores e decodificadores configurados para a operação de acordo com outros padrões de codificação de vídeo, tal como o padrão ITU-T-H.264/AVC ou padrões futuros. Consequentemente, a aplicação de um modo de codificação com base em paleta para a codificação de unidades de codificação (CUs) ou unidades de predição (PUs) em um codec HEVC é descrita para propósitos exemplificativos.

[0081] Em HEVC WD10 e outros padrões de codificação de vídeo, uma sequência de vídeo inclui tipicamente uma série de figurações. As imagens também podem ser chamadas de "quadros". A figuração pode incluir três matrizes de amostra, denotadas S_L , S_{Cb} e S_{Cr} . A S_L é uma matriz bidimensional (isto é, um bloco) de amostras de luma. A S_{Cb} é uma matriz bidimensional de amostras de croma Cb. S_{Cr} é uma matriz bidimensional de amostras de croma Cr. As amostras de croma também podem ser chamadas no presente documento de amostras de "croma". Em outros exemplos, uma figuração pode ser monocromática e pode incluir apenas uma matriz de amostras de luma.

[0082] A fim de gerar uma representação codificada de uma imagem, o codificador de vídeo 20 pode gerar um conjunto de unidades de árvore de codificação

(CTUs). Cada uma das CTUs pode ser um bloco de árvore de codificação de amostras luma, dois blocos de árvore de codificação correspondentes de amostras croma e estruturas de sintaxe usadas para codificar as amostras dos blocos de árvore de codificação. Um bloco de árvore de codificação pode ser um bloco $N \times N$ de amostras. Uma CTU também pode ser denominada de "bloco de árvore" ou de uma "unidade de codificação maior" (LCU). As CTUs de HEVC podem amplamente ser a CTU que não se limita necessariamente a um tamanho particular e pode incluir uma ou mais unidades de codificação (CUs). Uma fatia pode incluir um número inteiro de CTUs ordenadas consecutivamente na varredura de rastreamento.

[0083] A fim de gerar uma CTU codificada, o codificador de vídeo 20 pode realizar, recursivamente, partição de árvore quadrática nos blocos de árvore de codificação de uma CTU para dividir os blocos de árvore de codificação em blocos de codificação, por conseguinte, o nome "unidades de árvore de codificação". Um bloco de codificação é um bloco $N \times N$ de amostras. Uma CU pode ser um bloco de codificação de amostras luma e dois blocos de codificação correspondentes de amostras croma de uma figuração que tem uma matriz de amostra luma, uma matriz de amostra Cb e uma matriz de amostra Cr e estruturas de sintaxe usadas para codificar as amostras dos blocos de codificação. O codificador de vídeo 20 pode particionar um bloco de codificação de uma CU em um ou mais blocos de predição. Um bloco de predição pode ser um bloco retangular (isto é, quadrado ou não quadrado) de amostras no qual a mesma predição é aplicada. Uma unidade de predição (PU) de uma CU pode ser um bloco de predição de amostras luma, dois blocos

de predição correspondentes de amostras croma de uma figuração e estruturas de sintaxe usadas para prever o bloco de predição amostras. O codificador de vídeo 20 pode gerar blocos de luma, de Cb e de Cr preditivos para blocos de predição de luma, de Cb e de Cr de cada PU da CU.

[0084] O codificador de vídeo 20 pode usar intrapredição ou interpredição para gerar os blocos preditivos para uma PU. Se o codificador de vídeo 20 usar intrapredição para gerar os blocos preditivos de uma PU, o codificador de vídeo 20 pode gerar blocos preditivos da PU com base nas amostras decodificadas da figuração associada à PU.

[0085] Se o codificador de vídeo 20 usa interpredição para gerar os blocos preditivos de uma PU, o codificador de vídeo 20 pode gerar os blocos preditivos da PU com base em amostras decodificadas de uma ou mais figurações diferentes da figuração associada à PU. O codificador de vídeo 20 pode usar unipredição ou bipredição para gerar os blocos preditivos de uma PU. Quando o codificador de vídeo 20 utiliza monopredição para gerar os blocos preditivos para uma PU, a PU pode ter um único vetor de movimento (MV). Quando o codificador de vídeo 20 utiliza bipredição para gerar os blocos preditivos para uma PU, a PU pode ter dois MVs.

[0086] Após o codificador de vídeo 20 gerar blocos luma, de Cb e de Cr preditivos para uma ou mais PUs de uma CU, o codificador de vídeo 20 pode gerar um bloco residual luma para uma CU. Cada amostra no bloco residual de luma da CU indica uma diferença entre uma amostra de luma em um dentre os blocos preditivos de luma da CU e uma amostra

correspondente no bloco de codificação de luma original da CU. Além disso, o codificador de vídeo 20 pode gerar um bloco residual de Cb para uma CU. Cada amostra no bloco residual de Cb da CU pode indicar uma amostra correspondente no bloco de codificação de Cb original da CU. O codificador de vídeo 20 também pode gerar um bloco residual de Cr para uma CU. Cada amostra no bloco residual de Cr da CU pode indicar uma diferença entre uma amostra de Cr em um dos blocos de Cr preditivos da CU e uma amostra correspondente no bloco de codificação de Cr original da CU.

[0087] Além disso, o codificador de vídeo 20 pode usar a partição de árvore quadrática para decompor os blocos residuais luma, de Cb e de Cr de uma CU em um ou mais blocos de transformada luma, de Cb e de Cr. Um bloco de transformada pode ser um bloco retangular de amostras no qual a mesma transformada é aplicada. Uma unidade de transformada (TU) de um CU pode ser um bloco de transformada de amostras luma, dois blocos de transformada correspondentes de amostras croma e estruturas de sintaxe usadas para transformar as amostras de bloco de transformada. Assim, cada TU de uma CU pode estar associada a um bloco de transformada de luma, um bloco de transformada de Cb e um bloco de transformada de Cr. O bloco de transformada de luma associado à TU pode ser um sub-bloco do bloco residual de luma da CU. O bloco de transformada de Cb pode ser um sub-bloco do bloco residual de Cb da CU. O bloco de transformada de Cr pode ser um sub-bloco do bloco residual de Cr da CU.

[0088] O codificador de vídeo 20 pode aplicar uma ou mais transformadas a um bloco de transformada de luma de uma TU para gerar um bloco de coeficiente de luma para a

TU. Um bloco de coeficiente pode ser uma matriz bidimensional de coeficientes de transformada. Um coeficiente de transformada pode ser uma quantidade escalar. O codificador de vídeo 20 pode aplicar uma ou mais transformadas a um bloco de transformada de Cb de uma TU para gerar um bloco de coeficiente de Cb para a TU. O codificador de vídeo 20 pode aplicar uma ou mais transformadas a um bloco de transformada de Cr de uma TU para gerar um bloco de coeficiente de Cr para a TU.

[0089] Após gerar um bloco de coeficiente (por exemplo, um bloco de coeficiente de luma, um bloco de coeficiente de Cb ou um bloco de coeficiente de Cr), o codificador de vídeo 20 pode quantizar o bloco de coeficiente. Em geral, a quantização se refere a um processo no qual os coeficientes de transformada são quantizados para reduzir possivelmente a quantidade de dados usada para representar os coeficientes de transformada, o que fornece uma compressão adicional. Após o codificador de vídeo 20 quantizar um bloco de coeficiente, o codificador de vídeo 20 pode codificar por entropia elementos de sintaxe que indicam os coeficientes de transformada quantizados. Por exemplo, o codificador de vídeo 20 pode realizar Codificação Aritmética Binária Adaptativa ao Contexto (CABAC) nos elementos de sintaxe que indicam os coeficientes de transformada quantizados.

[0090] Em relação ao CABAC, como um exemplo, o codificador de vídeo 20 e o decodificador de vídeo 30 podem selecionar um modelo de probabilidade (também referido como um modelo de contexto) para codificar símbolos associados a um bloco de dados de vídeo com base no contexto. Por exemplo,

um modelo de contexto contextos, cada um dos quais pode corresponder a um modelo de probabilidade particular. Consequentemente, um modelo de probabilidade diferente é tipicamente definido para cada contexto. Após codificar ou decodificar o bin, o modelo de probabilidade é adicionalmente atualizado com base em um valor do bin para refletir as estimativas de probabilidade mais atuais para o bin. Por exemplo, um modelo de probabilidade pode ser mantido como um estado em uma máquina de estado finito. Cada estado particular pode corresponder a um valor de probabilidade específico. O próximo estado, que corresponde a uma atualização do modelo de probabilidade, pode depender do valor do bin atual (por exemplo, o bin que é atualmente codificado). Consequentemente, a seleção de um modelo de probabilidade pode ser influenciada pelos valores dos bins anteriormente codificados, devido ao fato de que os valores indicam, pelo menos em parte, a probabilidade do bin que tem um determinado valor. O processo de codificação de contexto descrito acima pode, em geral, ser referido como um modo de codificação adaptativa ao contexto.

[0091] Consequentemente, o codificador de vídeo 20 pode codificar um símbolo alvo com o uso de um modelo de probabilidade. De outro modo, o decodificador de vídeo 30 pode analisar um símbolo alvo com o uso do modelo de probabilidade. Em alguns exemplos, o codificador de vídeo 20 pode codificar elementos de sintaxe com o uso de uma combinação de codificação não adaptativa ao contexto e adaptativa ao contexto. Por exemplo, o codificador de vídeo 20 codificar pelo contexto bins pela seleção de um modelo de probabilidade ou "modelo de contexto" que opera no contexto

para codificar os bins. Em contrapartida, o codificador de vídeo 20 pode codificar por desvio bins desviando-se ou omitindo-se o processo de codificação aritmética regular ao codificar os bins. Em tais exemplos, o codificador de vídeo 20 pode usar um modelo de probabilidade fixo para desviar o código dos bins. Isto é, os bins codificados por desvio não incluem atualizações de probabilidade ou contexto.

[0092] O codificador de vídeo 20 pode emitir um fluxo de bits que inclui os elementos de sintaxe codificador por entropia. O fluxo de bits pode incluir uma sequência de bits que forma uma representação de imagens codificadas e dados associados. O fluxo de bits pode compreender uma sequência de unidades de camada de abstração de rede (NAL). Cada uma das unidades de NAL inclui um cabeçalho de unidade de NAL e encapsula uma carga útil de sequência de byte bruto (RBSP). O cabeçalho de unidade de NAL pode incluir um elemento de sintaxe que indica um código de tipo de unidade de NAL. O código do tipo unidade de NAL especificado pelo cabeçalho de unidade de NAL de uma unidade de NAL indica o tipo da unidade de NAL. Uma RBSP pode ser uma estrutura de sintaxe que contém uma quantidade em número inteiro de bytes que são encapsulados dentro de uma unidade de NAL. Em alguns exemplos, uma RBSP inclui zero bits.

[0093] Tipos diferentes de unidades de NAL podem encapsular tipos diferentes de RBSPs. Por exemplo, um primeiro tipo de unidade de NAL pode encapsular uma RBSP para um ajuste de parâmetro de figuração (PPS), um segundo tipo de unidade de NAL pode encapsular uma RBSP para uma fatia codificada, um terceiro tipo de unidade de NAL pode encapsular uma RBSP para as SEI, e assim por diante. As

unidades de NAL que encapsulam as RBSPs para dados de codificação de vídeo (em oposição às RBSPs para os ajustes de parâmetros e mensagens de SEI) podem ser denominadas de camada de codificação de vídeo (VCL) unidades de NAL.

[0094] O decodificador de vídeo 30 pode receber um fluxo de bits gerado pelo codificador de vídeo 20. Além disso, o decodificador de vídeo 30 pode analisar o fluxo de bits para decodificar elementos de sintaxe do fluxo de bits. O decodificador de vídeo 30 pode reconstruir as figurações dos dados de vídeo com base, pelo menos em parte, nos elementos de sintaxe decodificados do fluxo de bits. O processo para reconstruir os dados de vídeo pode ser, em geral, recíproco ao processo realizado pelo codificador de vídeo 20. Por exemplo, o decodificador de vídeo 30 pode usar MVs de PUs para determinar os blocos de predição para as PUs de uma CU atual. Além disso, o decodificador de vídeo 30 pode quantizar, inversamente, os blocos de coeficiente de transformada associados às TUs da CU atual. O decodificador de vídeo 30 pode realizar transformadas inversas nos blocos de coeficiente de transformada para reconstruir os blocos de transformada associados às TUs da CU atual. O decodificador de vídeo 30 pode reconstruir os blocos de codificação da CU atual adicionando-se as amostras dos blocos preditivos para as PUs da CU atual às amostras correspondentes dos blocos de transformada das TUs da CU atual. Através da reconstrução dos blocos de codificação para cada CU de uma imagem, o decodificador de vídeo 30 pode reconstruir a imagem.

[0095] O HEVC contém diversas propostas para tornar o codec mais favorável a ser paralelo, que inclui pequenos blocos e processamento paralelo de frente de onda

(WPP). HEVC WD10 define pequenos blocos como um número inteiro de blocos de árvore de codificação que ocorrem simultaneamente em uma coluna e uma fileira, em ordem consecutivamente em uma varredura por rastreamento de bloco de árvore de codificação do pequeno bloco. A divisão de cada figuração em pequenos blocos é uma partição. O número de pequenos blocos e a localização de seus limites podem ser definidos para toda sequência ou alterados de figuração a figuração. Os limites de pequeno bloco, de modo similar aos limites de fatia, quebram as dependências de análise e de predição de modo que um pequeno bloco possa ser processado independentemente, mas os filtros in-loop (desvio adaptativo de amostra (SAO) e desbloqueio) ainda podem cruzar os limites de pequeno bloco. HEVC WD10 também especifica algumas restrições sobre a relação entre fatias e pequenos blocos.

[0096] Em alguns exemplos, o codificador de vídeo 20 e o decodificador de vídeo 30 podem ser configurados para realizar a codificação com base em paleta. Por exemplo, na codificação com base em paleta, ao invés de realizar as técnicas de codificação de intrapreditivo ou interpredictivo descritas acima, o codificador de vídeo 20 e o decodificador de vídeo 30 podem codificar a então denominada paleta como uma tabela de cores ou valores de pixels que representa os dados de vídeo de uma área particular (por exemplo, um bloco determinado). Cada pixel pode estar associado a uma entrada na paleta que representa a cor do pixel. Por exemplo, o codificador de vídeo 20 e o decodificador de vídeo 30 podem codificar um índice que relaciona

[0097] No exemplo acima, o codificador de vídeo 20 pode codificar um bloco de dados de vídeo pela

determinação de uma paleta for o bloco, localização de uma entrada na paleta para representar o valor de cada pixel, e codificação da paleta com valores de índice para os pixels que relacionam o valor de pixel à paleta. O decodificador de vídeo 30 pode obter, a partir de um fluxo de bits codificado, uma paleta para um bloco, bem como valores de para os pixels do bloco. O decodificador de vídeo 30 pode relacionar os valores de índice dos pixels às entradas da paleta para reconstruir os valores de pixel do bloco.

[0098] Em alguns exemplos, o codificador de vídeo 20 e o decodificador de vídeo 30 podem ser configurados para construir uma paleta com o uso de entradas de paleta a partir de um ou mais blocos anteriormente codificados. Por exemplo, o codificador de vídeo 20 e o decodificador de vídeo 30 podem ser configurados para copiar uma ou mais entradas de paleta para a codificação de um bloco atual a partir de uma paleta de preditor que inclui entradas de paleta a partir de blocos anteriormente codificados com o uso do modo de paleta ou outras amostras reconstruídas. Para cada entrada na paleta de preditor, o codificador de vídeo 20 e o decodificador de vídeo 30 podem ser configurados para codificar um sinalizador binário para indicar se a entrada associada ao sinalizador é copiada para a paleta atual (por exemplo, indicada pelo sinalizador = 1). A coluna de sinalizadores binários pode ser referida como o vetor de predição de paleta binário. A paleta para a codificação de um bloco atual também pode incluir inúmeras de novas entradas de paleta, que podem ser explicitamente codificadas (por exemplo, separadamente do vetor de predição de paleta).

[0099] Conforme notado acima, o codificador de

vídeo 20 e o decodificador de vídeo 30 podem usar inúmeros modos de codificação de paleta diferentes para codificar índices de uma paleta. Por exemplo, o codificador de vídeo 20 e o decodificador de vídeo 30 podem usar um modo Fuga, um modo CopyFromTop (também referido como modo Cópia Acima), ou um modo Valor (também referido como modo índice) para codificar de índices de um bloco. Em geral, a codificação de uma amostra com o uso de "modo Fuga" podem, em geral, se referir a codificação de uma amostra de um bloco que não tem uma cor correspondente representada em uma paleta para a codificação do bloco. Conforme notado acima, tais amostras podem ser referidas como amostras de fuga ou pixels de fuga.

[0100] Como um outro exemplo, em um terceiro experimento principal de codificação de conteúdo de tela, o subteste B.6, conforme descrito em Yu-Wen Huang et al, "Description of Screen Content Core Experiment 3 (SCCE3): Palette Mode", JCTVC-Q1123, Valência, ES, 27 de março a 4 de abril de 2014 (doravante no presente documento Q1123), um outro modo foi introduzido no software liberado por Canon em 26 de maio de 2014. O macro para esse pode ser definido como "CANON_NEW_RUN_LAST_TRANSITION" e pode ser referido no presente documento como modo de execução de transição. A Execução de Transição pode ser similar ao modo Valor de modo que o codificador de vídeo 20 ou decodificador de vídeo 30 possa codificar um valor de índice seguido por uma execução que especifica o número de amostras subsequentes que têm o mesmo índice de paleta.

[0101] Uma diferença entre modo Valor e o modo de Execução de Transição é que o valor de índice do modo de Execução de Transição não é sinalizado no fluxo de bits. De

preferência, o codificador de vídeo 20 e o decodificador de vídeo 30 podem interferir no valor de índice para o modo de Execução de Transição. Conforme descrito no presente documento, inferir um valor pode se referir à determinação de um valor sem a referência à sintaxe dedicada que representa o valor que é codificada em um fluxo de bits. Isto é, o codificador de vídeo 20 e o decodificador de vídeo 30 podem inferir um valor sem a codificação de um elemento de sintaxe dedicado para o valor em um fluxo de bits. O índice inferido pode ser referido como um índice de transição.

[0102] Em alguns exemplos, pode haver duas maneiras de sinalização dos modos de paleta. Uma primeira técnica para sinalizar os modos de paleta pode ser referida como sinalização de fuga explícita. Por exemplo, em JCTVC-Q0094, se o macro "PLT REJVIOVE fuga FLAG" for zero, o codificador de vídeo 20 pode explicitamente codificar um sinalizador de fuga para cada amostra de um bloco para indicar se uma amostra que é codificada em um bloco é codificada no modo Fuga. Se a amostra não for codificada com o modo Fuga, o codificador de vídeo 20 pode codificar os dados adicionais para indicar se o modo é CopyFromTop ou Valor. Em alguns casos, os dados adicionais podem ser um sinalizador, referido no presente documento como um sinalizador SPoint (por exemplo, um valor de sinalizador SPoint de zero pode indicar o modo CopyFromTop e um valor de sinalizador SPoint de um pode indicar o modo Valor ou vice versa).

[0103] Consequentemente, com a sinalização de fuga explícita, o sinalizador SPoint pode ser usado para

indicar um tipo de execução particular para uma execução de valores de pixel associada ao modo indicado. Por exemplo, o codificador de vídeo 20 pode codificar um sinalizador SPoint para indicar se o índice que é atualmente codificado e uma execução de valores de índice subsequentes que é codificada em uma execução são codificados com o uso de modo CopyFromTop ou modo Valor. O codificador de vídeo 20 não codifica o sinalizador de fuga (por exemplo, "PLT_REMOVE_ESCAPE_FLAG") e o sinalizador SPoint (quando for necessário) para as amostras de execução subsequentes. Isto é, o codificador de vídeo 20 e o decodificador de vídeo 30 podem inferir os valores do sinalizador de fuga e sinalizador SPoint para amostras incluídas em uma execução. Por exemplo, o codificador de vídeo 20 e o decodificador de vídeo 30 podem determinar o valor do sinalizador de fuga e sinalizador SPoint para amostras incluídas em uma execução sem a referência à sintaxe dedicada que represente tais valores no fluxo de bits.

[0104] Uma segunda técnica para sinalizar os modos de paleta pode ser referida como sinalização de fuga implícita. Por exemplo, se o macro "PLT_REMOVE_ESCAPE_FLAG" a partir do aumento do número de entradas de paleta de uma paleta por uma acomodar um índice especial para a paleta que não corresponde a qualquer um dos índices do bloco. Em alguns exemplos, o codificador de vídeo 20 e o decodificador de vídeo 30 podem incluir o índice adicional como o último índice de paleta na paleta aumentada para um determinado bloco. O índice adicional pode ser usado como uma indicação de uma amostra de fuga.

[0105] Nessa sinalização de fuga implícita, o

codificador de vídeo 20 pode codificar, para um valor de amostra particular de um bloco, em que os dados representam o índice adicional para indicar que a amostra adicional é codificada com o uso de modo Fuga. O codificador de vídeo 20 também pode codificar o valor de cor da amostra de fuga. Consequentemente, no caso de sinalização de fuga implícita, há apenas dois modos possíveis (por exemplo, modo CopyFromTop ou modo Valor) para ser sinalizado com o uso de sintaxe explícita. Por exemplo, apenas o sinalizador SPoint pode ser sinalizado para se distinguir entre os modos. Se uma amostra for codificada no modo Valor e o índice para o modo Valor for igual ao índice de fuga (por exemplo, o índice adicional notado acima para a paleta), o codificador de vídeo 20 e o decodificador de vídeo 30 podem inferir a amostra para ser codificada no modo Fuga. Nesse caso, nenhuma execução é sinalizada. Ao usar a sinalização de fuga implícita com o modo de Execução de Transição, o sinalizador SPoint pode assumir os valores 0 (por exemplo, modo Valor), 1 (por exemplo, modo CopyFromTop) ou 2 (por exemplo, modo de Execução de Transição).

[0106] As técnicas descritas nesta revelação podem incluir técnicas para várias combinações de um ou mais dentre predição de paleta e sinalização de valores de índice com sinalização de fuga implícita. Por exemplo, os aspectos desta revelação se referem à redefinição de uma paleta de preditor. Por exemplo, a redefinição de uma paleta de preditor pode incluir definir o tamanho da paleta de preditor para zero, de modo que a paleta de preditor não possa ser usada para construir uma paleta para um bloco que é atualmente codificado. Nesse caso, um vetor de predição de

paleta binária não é sinalizado em um fluxo de bits e nenhuma entrada de paleta é predita a partir da paleta de preditor.

[0107] No documento JCTVC-Q0094, uma paleta de preditor é redefinida para um bloco de árvore de codificação (CTB) na borda à esquerda da figuração (por exemplo, no começo de uma fileira de CTBs em uma figuração ("fileira de CTB")). A racional para redefinir a paleta de preditor no começo de uma fileira de CTB pode ser o dobro. Em primeiro lugar, após a codificação d uma linha de CTBs, a paleta de preditor pode incluir entradas de paleta que pertencem principalmente às paletas de blocos localizadas no lado direito da figuração. Consequentemente, a paleta de preditor não pode ser muito eficaz na predição de uma paleta atual para um primeiro CTB de uma fileira de CTB, que pode ser posicionada no lado à esquerda da figuração (por exemplo, as cores de pixels no lado à esquerda de a figuração podem diferir das frentes de onda que estão sendo usadas para a codificação de uma figuração particular, além dos estado de CABAC, seria necessário propagar as informações sobre a paleta de preditor entre as frentes de onda.

[0108] Por exemplo, uma frente de onda pode ser uma unidade de dados de vídeo para a codificação e pode ser codificada em paralelo com outras frentes de onda. Por exemplo, duas abordagens de paralelização estão incluídas em HEVC, incluindo o processamento paralelo de frente de onda (WPP) e Frente de Onda Sobreposta (OWF). O WPP permite criar partições de figuração que podem ser processadas em paralelo sem efetuar as altas perdas de codificação. Em fileiras de WPP de CTBs são processados em paralelo enquanto preserva todas as dependências de codificação. A OWF permite sobrepor

a execução de figurações consecutivas com o uso de frentes de onda.

[0109] No entanto, em alguns exemplos, a redefinição de uma paleta de preditor pode levar às perdas de codificação. Por exemplo, sem uma paleta de preditor, o codificador de vídeo 20 e o decodificador de vídeo 30 podem codificar os dados que indicam todas as entradas de uma paleta (por exemplo, todos os valores de índice e valores de cor relacionada) em um fluxo de bits. Esses dados de paleta podem ser uma quantidade relativamente grande de dados em relação às entradas de paleta preditas, que podem ser indicadas no fluxo de bits com o uso de um único sinalizador. Consequentemente, a redefinição de uma paleta de preditor pode afetar adversamente a taxa de bits de dados de vídeo codificados com a codificação com base em paleta.

[0110] De acordo com os aspectos desta revelação, o codificador de vídeo 20 e o decodificador de vídeo 30 podem ser configurados para desativar a redefinição de uma paleta de preditor com base no número de frentes de onda que é usado para codificar uma figuração que é um ou zero (isto é, nenhuma frente de onda). Isto é, o codificador de vídeo 20 e o decodificador de vídeo 30 podem ser configurados para não redefinir uma paleta de preditor quando frentes de onda de um ou zero são usadas para codificar a figuração. No entanto, o codificador de vídeo 20 e o decodificador de vídeo 30 podem ser configurados para permitir a redefinição de uma paleta de preditor quando duas ou mais frentes de onda são usadas para codificar uma figuração. Dessa maneira, o codificador de vídeo 20 e o decodificador de vídeo 30 não precisam propagar as

informações sobre uma paleta de preditor entre as frentes de onda, embora ainda evite as perdas de codificação no único caso de frente de onda.

[0111] De acordo com os aspectos desta revelação, quando múltiplos pequenos blocos e múltiplas frentes de onda são usados para codificar uma figuração (por exemplo, codificador de vídeo 20 e decodificador de vídeo 30 usam dois ou mais pequenos blocos e duas ou mais frentes de onda para codificar a figuração), o codificador de vídeo 20 e o decodificador de vídeo 30 podem ser configurado para redefinir a paleta de preditor no começo de cada fileira de CTB de pequeno bloco. Uma fileira de CTB de pequeno bloco podem incluir CTBs a partir de uma fileira de CTB que pertence a um pequeno bloco particular. Deve ser entendido que uma "fileira" pode ser orientada vertical ou horizontalmente, dependendo de uma ordem de varredura particular usada para codificar os blocos. Uma fileira também pode ser referida no presente documento como uma linha e os termos podem ser usados de modo intercambiável no presente documento.

[0112] As técnicas descritas acima também podem ser aplicadas a uma tabela de transição de paleta. Por exemplo, o documento de submissão padrão Yu-Chen Sun et al., "CE6 Test C.2 Transition Copy Mode," JCTVC-S0078, Estrasburgo, FR, 17 a 24 de outubro (JCTVC-S0078) descreve um modo de cópia de transição. A cópia de transição também pode ser referida como um modo de execução de cópia de transição (execução de TC). NO modo de execução de TC, um decodificador grava os padrões de índice de cor adjacentes e usa os padrões para predizer os índices a serem

codificados. No modo de execução de TC, como um exemplo, um pixel atual, C, tem um índice de cor piloto igual a um. A partir de uma área causal de pixels codificados, um codificador de vídeo (tal como codificador de vídeo 20 ou decodificador de vídeo 30) pode determinar que um índice de cor para a direita de um índice de cor que é igual a um está propenso a ser dois (por exemplo, assumindo uma varredura da esquerda para a direita). Esse padrão de transição de um a dois pode ser gravado em uma tabela de cópia de transição (tabela de TC) para aperfeiçoar a eficiência de predição de índice de cor.

[0113] No documento JCTVC-S0078, uma tabela de TC é redefinido quando uma CU atual que é codificada for a primeira CU da fatia, ou a CU atual herda a tabela de TC da CU anterior quando a CU atual for uma não primeira CU da fatia. Essa propriedade de herança pode ser referida como "propagação de tabela de TC". A tabela de TC pode ser mantida e atualizada à medida que os pixels na CU são codificados.

[0114] Em alguns exemplos, no modo de cópia de transição, duas tabelas de transição são mantidas, uma para fileiras com índices pares e as outras para fileiras com índices ímpares. As contribuições anteriores descritas com o uso de uma única tabela de transição. Independentemente, em geral, conforme notado acima, uma tabela de transição de uma CU codificada de paleta anterior (ou bloco) pode ser usada para inicializar a tabela de transição para uma CU atual (ou bloco) codificado no modo de paleta.

[0115] De acordo com os aspectos desta revelação, o conceito de redefinir a paleta no começo de um primeiro CTB em uma fileira de CTB ou o primeiro CTB em um

pequeno bloco em uma fileira de CTB quando as múltiplas frentes de onda estiverem sendo usadas, conforme descrito acima, pode ser estendido para redefinir a tabela (ou tabelas) de transição de paleta também naquelas localizações. Por exemplo, o codificador de vídeo 20 e o decodificador de vídeo 30 podem ser configurados para definir as todas as entradas em uma tabela de transição de paleta para zero (ou para qualquer outro índice de paleta fixo, ou qualquer outro padrão) para o primeiro CTB em uma fileira de CTB ou o primeiro CTB em um pequeno bloco em uma fileira de CTB quando as múltiplas frentes de onda estiverem sendo usadas (ou, equivalentemente, as CUs codificadas de primeira paleta dentro dos CTBs). De acordo com os aspectos desta revelação, a redefinição da paleta de preditor e paleta tabelas de transição pode ser generalizada para quaisquer outros dados relacionados à paleta que podem ser propagados de uma CU codificada de paleta para uma outra.

[0116] Em um outro exemplo, o codificador de vídeo 20 e o decodificador de vídeo 30 podem ser configurados para codificar um sinalizador de um conjunto de parâmetros de vídeo (VPS), conjunto de parâmetros de sequência (SPS), conjunto de parâmetros de figuração (PPS), cabeçalho de fatia, em um nível de pequeno bloco, em um nível de bloco ou em qualquer lugar que indique se uma paleta de preditor redefinido (e/ou redefinição de outras informações tais como as tabelas de transição de paleta notadas acima) é realizada no começo de cada fileira de CTB. Em exemplos nos quais o sinalizador é sinalizado em um nível de pequeno bloco, o sinalizador pode indicar se a paleta de preditor redefinido é realizada no começo de cada fileira de CTB de pequeno bloco

(por exemplo, no começo de cada fileira de CTBs em um pequeno bloco). Alternativamente, para o primeiro bloco codificado com o uso do modo de paleta em uma fileira de CTBs, um sinalizador pode ser sinalizado para indicar se a paleta de preditor redefinido é realizada.

[0117] Em um outro exemplo, um sinalizador (por exemplo, em que um sinalizador pode ser um ou mais elementos de sintaxe) pode ser sinalizado para cada bloco codificado com o uso do modo de paleta para indicar se a paleta de preditor redefinido deve ser realizada. Em alguns exemplos, essa técnica pode ter uma vantagem de permitir que o codificador de vídeo 20 e o decodificador de vídeo 30 tenham capacidade para selecionar entre as técnicas que têm todas as entradas de paleta explicitamente sinalizadas ou técnicas que usam uma paleta de preditor para prever algumas das entradas de paleta. Em alguns exemplos, o sinalizador valor pode ser inferido. Por exemplo, o codificador de vídeo 20 e o decodificador de vídeo 30 podem ser configurados para inferir o valor do sinalizador para cada frente de onda, de modo que o valor inferido indique que a paleta de preditor é redefinida. Alternativa ou adicionalmente, o codificador de vídeo 20 e o decodificador de vídeo 30 podem ser configurados para inferir o valor do sinalizador para o primeiro bloco de uma fatia ou pequeno bloco, de modo que o valor inferido indique que a paleta de preditor é redefinida no começo de cada fatia e/ou pequeno bloco.

[0118] Em um outro exemplo, em vez de sinalizar um sinalizador para indicar uma paleta de preditor redefinido, o codificador de vídeo 20 e o decodificador de vídeo 30 podem ser configurados para redefinir uma paleta de

preditor de acordo com a regra ou conjunto de regras. Por exemplo, tanto o codificador de vídeo 20 quanto o decodificador de vídeo 30 podem ser configurados para aplicar a regra ou o conjunto de regras. Em um exemplo, o codificador de vídeo 20 e o decodificador de vídeo 30 podem ser configurados para aplicar a seguinte regra para determinar a possibilidade de redefinir a paleta de preditor: se a paleta no bloco atual não tiver entradas preditas a partir de uma paleta de preditor, a paleta de preditor é redefinida (por exemplo, a paleta de preditor é ajustada para ter um comprimento de zero) para esse bloco atual, de modo que apenas a paleta atual possa ser usada como um indicador para os seguintes, mas não as entradas de paleta

[0119] No exemplo acima, codificador de vídeo 20 pode controlar a redefinição de paleta. Por exemplo, quando a paleta tiver que ser redefinida, o codificador de vídeo 20 pode simplesmente não usar quaisquer entradas da paleta de preditor. Uma vantagem potencial de implantação desse processo é que nenhum elemento de sintaxe novo é adicionado e a redefinição de paleta pode ser mais granular e controlada de modo mais flexível. Por exemplo, a flexibilidade/granularidade pode ser alcançada devido à capacidade de alterar as condições sob as quais a redefinição de paleta de preditor é realizada.

[0120] Em alguns exemplos, o codificador de vídeo 20 não pode usar a paleta de preditor em pelo menos um dos casos a seguir: para a primeira LCU em cada fileira de LCU, quando a frente de onda codificação for ativada; para o primeiro bloco em um pequeno bloco ou para o primeiro bloco em uma fatia.

[0121] De acordo com os aspectos desta revelação, uma paleta de preditor pode ser reinicializada. Por exemplo, a reinicialização da paleta de preditor pode, em geral, se referir à substituição de entradas da paleta de preditor por novas entradas. Por exemplo, após a codificação d um bloco de índices, uma paleta de preditor pode incluir inúmeras entradas para predizer uma paleta do próximo bloco a ser codificado. A reinicialização da paleta de preditor pode incluir remover o número de entradas e adicionar novas entradas à paleta de preditor para predizer a paleta do próximo bloco a ser codificado.

[0122] Em alguns exemplos, de acordo com os aspectos desta revelação, no começo de cada linha de CTB, o codificador de vídeo 20 e o decodificador de vídeo 30 podem ser configurados para reinicializar uma paleta de preditor para determinar uma paleta atual de um CTB atual com o uso de entradas de paleta de um primeiro CTB da linha de CTB acima do CTB atual, se a linha de CTB atual não for a primeira linha de CTB na figuração/fatia/pequeno bloco. Nesse caso, conforme descrito em mais detalhes em relação ao exemplo da Figura 5 abaixo, após finalizar a codificação do primeiro CTB e antes da codificação de a segunda CTB em uma linha de CTB acima (visto que o primeiro CTB não é a primeira linha de CTB na figuração/fatia/pequeno bloco), o codificador de vídeo 20 e o decodificador de vídeo 30 podem ser configurados para usar a paleta de preditor para reinicializar a paleta de preditor no começo da linha de CTB atual. Se a linha de CTB atual for o primeiro um na figuração/fatia/pequeno bloco, o codificador de vídeo 20 e o decodificador de vídeo 30 podem ser configurados para definir o número de indicador entradas

de paleta (por exemplo, conforme descrito no documento JCTVC-Q0094) e não predizer quaisquer entradas de paleta ao determinar a paleta para o CTB atual.

[0123] Em um outro exemplo, o conceito de reinicializar uma paleta de preditor pode ser generalizado. Por exemplo, o codificador de vídeo 20 e o decodificador de vídeo 30 podem ser configurados para gerar uma paleta de preditor após a codificação de N CTBs de uma linha de CTBs acima de um CTB que é atualmente codificado e antes de começar a codificação de $N + 1^{\text{ésimo}}$ CTB na linha de CTB acima do CTB que é atualmente codificado (desde que a linha de CTBs não seja a primeira linha de CTB na figuração/fatia/pequeno bloco). O codificador de vídeo 20 e o decodificador de vídeo 30 podem ser configurados para reinicializar a paleta de preditor para determinar a paleta do CTB que é atualmente codificado com o uso da paleta de preditor gerada. O CTB que é atualmente codificado pode ser o primeiro CTB de uma linha de CTB atual (desde que o CTB atual não seja a primeira linha de CTB na figuração/fatia/pequeno bloco). Aqui, N pode estar entre um e o número de CTBs na linha de CTB. Para alinhar conceitualmente essa técnica de reinicialização de paleta de preditor com codificação de frente de onda, N pode ser selecionado como sendo menor que ou igual a dois. Dessa maneira, o mesmo processo de reinicialização de paleta de preditor pode ser usado independentemente do número de frentes de onda usado para codificar uma figuração.

[0124] Conforme notado acima, um CTB também pode ser referido como um LCU. Em alguns exemplos, a reinicialização de paleta de preditor pode ser realizada

para o primeiro CTB de uma linha de blocos na qual um modo de paleta é usado para predizer qualquer bloco do CTB (por exemplo, em casos em que uma variedade de modos de predição (modos inter ou intra) ou modos de paleta pode ser usada para codificar um CTB. Embora os exemplos acima sejam descritos em relação a CTBs para propósitos ilustrativos, deve ser entendido que as técnicas podem ser estendidas como qualquer tipo de bloco de dados de vídeo.

[0125] De acordo com os aspectos desta revelação, o processo de reinicialização de paleta descrito acima pode ser estendido para tabelas de transição de paleta. Por exemplo, o codificador de vídeo 20 e o decodificador de vídeo 30 podem ser configurados para reinicializar as tabelas de transição de paleta no começo de uma fileira atual de CTB com o uso das tabelas de transição de paleta geradas após a codificação de N CTBs e antes de começar a codificar o N+1^{ésimo} CTB na linha de CTB acima do CTB atual (desde que o CTB atual não seja a primeira linha de CTB na figuração/fatia/pequeno bloco). Aqui, N pode estar entre um e o número de CTBs na linha de CTB. Por exemplo, o codificador de vídeo 20 e o decodificador de vídeo 30 podem ser configurados para selecionar um valor de N que é menor do que ou igual a dois.

[0126] Em alguns exemplos, o codificador de vídeo 20 e o decodificador de vídeo 30 podem ser configurados para copiar o tamanho das tabelas de transição além das próprias tabelas de transição de paleta. De fato, esse processo de reinicializar as informações de codificação de paleta (por exemplo, informações de paleta de preditor, informações de tabela de transição de paleta ou similares) pode ser generalizado para quaisquer outros dados

relacionados à paleta que são propagados a partir de uma CU codificada de paleta para a próxima CU codificada de paleta. Por exemplo, o codificador de vídeo 20 e o decodificador de vídeo 30 podem ser configurados para reinicializar os dados relacionados à paleta para um CTB atual com base em dados de paleta gerados após a codificação dos N CTBs e antes de começar a codificar o N+1^{ésimo} CTB na linha de CTB acima do CTB atual.

[0127] Em alguns exemplos, o codificador de vídeo 20 e o decodificador de vídeo 30 podem ser configurados para usar a reinicialização para todos os CTBs aplicáveis (por exemplo, CTBs inicialmente codificados em uma linha de CTBs). Em outros exemplos, o codificador de vídeo 20 e o decodificador de vídeo 30 podem ser configurados para poder realizar apenas as técnicas de reinicialização quando o número de frentes de onda que é usado para codificar uma figuração ou pequeno bloco for dois ou mais.

[0128] De acordo com os aspectos desta revelação, quando pequenos blocos e frentes de onda estão sendo usados juntos para codificar de uma figuração particular, o codificador de vídeo 20 e o decodificador de vídeo 30 podem ser configurados para aplicar as técnicas de reinicialização a pequenos blocos apesar de os pequenos blocos serem figurações separadas. Por exemplo, o codificador de vídeo 20 e o decodificador de vídeo 30 podem ser configurados para aplicar o processo de reinicialização descrito acima para um primeiro CTB de pequeno bloco se o primeiro CTB de pequeno bloco não for a primeira linha de CTB no pequeno bloco, conforme descrito em maiores detalhes em relação à Figura 7 abaixo.

[0129] Outros aspectos desta revelação se referem à sinalização de amostras de fuga em exemplos nos quais a sinalização de fuga implícita é usada. Por exemplo, em JCTVC-Q0094, para a técnica de fuga implícita descrita acima, o tamanho da paleta é aumentado em um e o último índice na paleta expandida é atribuído para indicar se uma amostra é codificada como uma amostra de fuga. Se a codificação binária truncada for usada para codificar os índices, o último índice pode usar mais de um bit do que o índice zero. De modo similar, se uma outra estratégia de codificação de comprimento variável for usada, a diferença em comprimentos pode ser ainda maior.

[0130] Tipicamente, para certos tamanhos de bloco, pode haver mais amostras que são codificados como amostras de fuga do que blocos de outros tamanhos. Para incentivar essa relação para aprimorar a codificação de índices, o codificador de vídeo 20 e o decodificador de vídeo 30 podem ser configurados para atribuir o índice zero de uma paleta ao índice usado para indicar uma amostra de fuga para blocos que são propensos a terem mais pixels codificados como uma amostra de fuga. Para os outros blocos que podem ter relativamente menos amostras codificadas como amostras de fuga, o codificador de vídeo 20 e o decodificador de vídeo 30 podem ser configurados para poderem seguir o esquema atual de atribuição do índice mais alto a partir da paleta expandida para indicar uma amostra de fuga.

[0131] Em um exemplo para propósitos ilustrativos, o codificador de vídeo 20 e o decodificador de vídeo 30 podem ser configurados para atribuir um índice zero ao índice usado para indicar uma amostra de fuga para

tamanhos de bloco 32×32 e abaixo. Dessa maneira, o codificador de vídeo 20 e o decodificador de vídeo 30 podem ser configurados para atribuir o índice mais alto (por exemplo, numericamente maior dos valores de índice) ao índice usado para indicar uma amostra de fuga para blocos de 64×64 e maiores.

[0132] Em alguns exemplos, o codificador de vídeo 20 e o decodificador de vídeo 30 podem ser configurados para implantar a mesma regra de atribuição de índice de fuga para cada tamanho de bloco *a-priori*. Em um outro exemplo, uma regra de atribuição de índice de fuga pode ser sinalizada explicitamente em um conjunto de parâmetros de vídeo (VPS), conjunto de parâmetros de sequência (SPS), conjunto de parâmetros de figuração (PPS), cabeçalho de fatia, em um nível de pequeno bloco ou em qualquer lugar.

[0133] Outros aspectos desta revelação se referem à sinalização de amostras de fuga em exemplos nos quais a sinalização de fuga explícita é usada. No case de sinalização de fuga explícita, o modo Fuga pode ser sinalizado com um sinalizador. De acordo com os aspectos desta revelação, o codificador de vídeo 20 e o decodificador de vídeo 30 podem ser configurados para contextualizar adaptativamente a codificação dos sinalizadores de fuga com base em tamanho de bloco. Por exemplo, os contextos podem ser projetados para codificar sinalizadores de fuga a fim de explorar a relação que, para certos tamanhos de bloco, pode haver mais amostras codificadas com o uso de uma amostra de fuga em relação a outros tamanhos de bloco. Nesse caso, o contexto para o sinalizador pode depender do tamanho de bloco. Adicionalmente, em alguns exemplos, um tamanho de

paleta pode ter alguma correlação com a quantidade/número de amostras de fuga a ser sinalizado na CU. Portanto, o contexto pode depender do tamanho de paleta ou se o tamanho de paleta é igual ao tamanho máximo de paleta (ou não igual ao tamanho máximo de paleta).

[0134] Outros aspectos desta revelação se referem aos valores de execução de sinalização para execuções de índices codificados com o uso de modo de paleta Valor. Por exemplo, no documento de apresentação padrão Guillaume Laroche et al., "AHG10: Run Coding for Palette Mode", JCTVC-Q0066, Valência, ES, 27 de março a 4 de abril de 2014 (JCTVC-Q0066), é proposto que, para cada bloco codificado com o uso de uma paleta, um limite de índice possa ser sinalizado. Se o índice que é codificado no modo Valor for maior do que ou igual ao limite de índice, nenhuma execução é sinalizada para o modo Valor. Esta revelação pode se referir a esse método como "Execução Limitada".

[0135] De acordo com os aspectos desta revelação, modificações podem ser feitas no processo descrito em Q0066. Em um primeiro exemplo, os primeiros índices indicados pelo elemento de sintaxe `num_skip`, uma execução é sempre sinalizada. Em alguns exemplos, o elemento de sintaxe `num_skip` é igual a dois, então, a execução é sempre sinalizada para índices zero e um. O valor indicado pelo elemento de sintaxe `num_skip` pode ser conhecido tanto pelo codificador de vídeo 20 quanto pelo decodificador de vídeo 30 *a-priori* (por exemplo, predeterminado nos codificadores de vídeo), ou pode ser sinalizado explicitamente no conjunto de parâmetros de vídeo (VPS), conjunto de parâmetros de sequência (SPS), conjunto de

parâmetros de figuração (PPS), cabeçalho de fatia, em um nível de pequeno bloco ou em qualquer lugar.

[0136] Em um segundo exemplo, se o tamanho de paleta estiver abaixo de um limite de tamanho de paleta particular, o limite de índice não é sinalizado e a execução é sinalizada para cada índice. O limite de tamanho de paleta pode ser conhecido tanto pelo codificador de vídeo 20 quanto pelo decodificador de vídeo 30 *a-priori* ou pode ser sinalizado explicitamente no conjunto de parâmetros de vídeo (VPS), conjunto de parâmetros de sequência (SPS), conjunto de parâmetros de figuração (PPS), cabeçalho de fatia, em um nível de pequeno bloco ou em qualquer lugar.

[0137] Em um terceiro exemplo, um sinalizador pode ser sinalizado para cada bloco em que o tamanho de paleta é maior do que o limite de tamanho de paleta, para indicar se o limite de índice é sinalizado. O sinalizador pode ser sinalizado apenas para certos tamanhos de bloco e/ou tamanhos de paleta. Se o limite de índice não for sinalizado, a execução é sinalizada para todos os índices. Em um exemplo alternativo, em vez de sinalizar um sinalizador explícito, um limite de índice de um tamanho de paleta é sinalizado para indicar que a execução é sinalizada para todos os índices.

[0138] Em um quarto exemplo, a binarização truncada (Por exemplo, codificação binária truncada conforme descrito no Pedido Provisório nº US 62/002.054, depositado em 22 de maio de 2014) pode ser usada para codificar o limite de índice para um bloco se o tamanho da paleta estiver acima do limite de tamanho de paleta. Se um sinalizador explícito como no terceiro exemplo acima for sinalizado, o valor máximo

para binarização truncada é (tamanho de paleta atual - num_skip). Se nenhum sinalizador explícito estiver sendo enviado, o valor máximo para binarização truncada pode ser (tamanho de paleta atual - num_skip + 1).

[0139] Em um quinto exemplo, um valor de execução pode ser codificado após a redistribuição, a reordenação e o remapeamento de índice de paleta. Por exemplo, algumas técnicas de codificação de paleta podem ser usadas para agrupar índices para ter execuções de índices mais longas. Em um exemplo, o codificador de vídeo 20 e o decodificador de vídeo 30 podem ser configurados para realizar uma varredura adaptativa de índices ou aplicar uma transformada de Burrows Wheeler.

[0140] Em um sexto exemplo, o limite de índice pode ser sinalizado como (tamanho de paleta - limite de índice) em casos nos quais o sinalizador explícito descrito acima em relação ao terceiro exemplo é sinalizado. De modo similar, o limite de índice pode ser sinalizado como (tamanho de paleta + 1 - limite de índice) em casos nos quais tal sinalizador explícito não é sinalizado. Em alguns exemplos, o codificador de vídeo 20 e o decodificador de vídeo 30 podem ser configurados para codificar o limite de índice ou (tamanho de paleta - limite de índice) com base no tamanho de bloco e/ou no tamanho de paleta. Em alguns exemplos, uma estratégia similar pode ser usada em casos nos quais um sinalizador explícito.

[0141] As técnicas dos seis exemplos descritos acima podem ser implantadas por codificador de vídeo 20 e decodificador de vídeo 30 individualmente ou em combinação. Além disso, as técnicas podem ser implantadas seletivamente

dependendo do tamanho de bloco e/ou tamanho de paleta. Tal dependência de tamanho de bloco e/ou de tamanho de paleta pode ser conhecida tanto pelo codificador de vídeo 20 quanto pelo decodificador de vídeo 30 *a-priori*, ou pode ser sinalizada explicitamente no conjunto de parâmetros de vídeo (VPS), conjunto de parâmetros de sequência (SPS), conjunto de parâmetros de figuração (PPS), cabeçalho de fatia, em um nível de pequeno bloco ou em qualquer lugar.

[0142] Em contrapartida a JCTVC-Q0066, de acordo com os aspectos desta revelação, as técnicas descritas acima podem ser normativas. Por exemplo, isso significa que não é permitido que um fluxo de bits válido (por exemplo, um fluxo de bits que é decodificável por um decodificador que se conforma a padrão particular) tenha um índice para o qual a execução não é sinalizada para ocorrer em posições consecutivas e codificado com o uso de modo Valor. Isto é, duas posições consecutivas que têm o mesmo valor de índice não podem ser codificadas com o uso do modo Valor. Essa restrição pode ser tratada como uma restrição normativa em um fluxo de bits. Isto é, um fluxo de bits de conformação não deve conter quaisquer dois ou mais pixels consecutivos codificados com o modo Valor que tem valores de índice idênticos para os quais uma execução não é sinalizada. Isso permite a aplicação de todas as remoções de redundância descritas no Pedido Provisório nº U.S. 62/002.054, depositado no dia 22 de maio de 2014, bem como a remoção dos denominados valores de índice impossíveis descritos em C. Gisquet et al., "AHG10: Palette Index Coding," JCTVC-Q0064, Valência, ES, 27 de março a 4 de abril de 2014 (JCTVC-Q0064).

[0143] Se o modo de Execução de Transição também

for usado, a técnica de Execução Limitada também pode ser estendida para o modo de Execução de Transição. Nesse caso, se o modo for Execução de Transição ou Valor e o índice de paleta (para modo Valor) ou índice de paleta inferido (para o modo de Execução de Transição) estiver acima do limite de índice, nenhuma execução é sinalizada e a mesma restrição de fluxo de bits normativa pode ser aplicada conforme descrito acima.

[0144] Outros aspectos desta revelação se referem à codificação de um valor de execução que indica um comprimento de execução de uma execução de valores de índice. No exemplo, as técnicas podem ser usadas para codificar um valor de execução de um índice codificado com o uso de modo Valor.

[0145] Por exemplo, em alguns exemplos, três sinalizadores codificados de contexto podem ser usados para codificar um valor de execução no modo Valor. Em tais exemplos, os três sinalizadores codificados de contexto podem incluir um sinalizador maior que zero que indica se o valor de execução é maior do que zero, um sinalizador maior que um que indica se o valor de execução é maior do que uma e um sinalizador maior que dois que indica se o valor de execução é maior do que dois. Em alguns exemplos, o codificador de vídeo 20 e o decodificador de vídeo 30 podem ser configurados para codificar cada um dos três sinalizadores com o uso de um único contexto.

[0146] No entanto, com o uso de um único contexto pode exigir que o codificador de vídeo 20 e o decodificador de vídeo 30 atualizem o modelo de probabilidade associado ao contexto após a codificação de cada um dos

sinalizadores. Além disso, o processo de atualização de probabilidade acima pode introduzir o atraso no processo de codificação. Em um exemplo para propósitos ilustrativos, três bins usam o mesmo modelo de contexto (por exemplo, `ctx(0)`) para propósitos de codificação adaptativa ao contexto. Dessa maneira, um primeiro bin pode usar `ctx(0)` para determinar um modelo de probabilidade para a codificação. O valor do primeiro bin influencia o modelo de probabilidade associada a `ctx(0)`. Consequentemente, uma atualização de probabilidade é realizada antes da codificação de o segundo bin com `ctx(0)`. Uma atualização de probabilidade adicional também é realizada antes da codificação de o terceiro bin com `ctx(0)`. Dessa maneira, a atualização de probabilidade pode introduzir o atraso no ciclo de codificação.

[0147] Ademais, a correlação entre o primeiro bin, o segundo bin e o terceiro bin de um valor de execução pode ser insuficiente para garantir o tempo e recursos computacionais associados à atualização do modelo de probabilidade. Isto é, um benefício potencial de codificação adaptativa ao contexto é a capacidade para adaptar um modelo de probabilidade com base em bins anteriormente codificados (determinado o mesmo contexto). Se o valor de um primeiro bin, no entanto, tiver pouca influência sobre o valor de um bin subsequente, pode haver pouco ganho de eficiência associado à atualização de probabilidade. Consequentemente, os bins que exibem uma baixa correlação não podem se beneficiar da codificação adaptativa ao contexto tanto quanto os bins que têm correlações relativamente mais altas.

[0148] De acordo com os aspectos desta

revelação, o codificador de vídeo 20 e o decodificador de vídeo 30 podem ser configurados para determinar um contexto para a codificação de um valor de execução com base no índice do valor de execução. Adicional ou alternativamente, o codificador de vídeo 20 e o decodificador de vídeo 30 podem ser configurados para determinar um contexto para a codificação de um valor de execução com base em um tamanho de bloco de um bloco que é atualmente codificado ou um tamanho de paleta de uma paleta para o bloco que é atualmente codificado. Em alguns exemplos, os aspectos desta revelação incluem o uso de múltiplos contextos para codificar um valor de execução.

[0149] Em um exemplo, de acordo com os aspectos desta revelação, o codificador de vídeo 20 e o decodificador de vídeo 30 podem ser configurados para usar sete contextos para a codificação de um valor de execução. Por exemplo, sete contextos podem ser definidos para a codificação de um sinalizador que indica que um valor de execução é maior do que zero. Esse exemplo é ilustrado na Tabela 1 abaixo:

TABELA 1- SETE CONTEXTOS PARA A EXECUÇÃO DE CODIFICAÇÃO COM BASE NO ÍNDICE

Índice	Contexto de execução
0	0
1	1
2	2
3	3
[4, 7]	4
[8, 15]	5

[16, 31]	6
----------	---

[0150] Embora o exemplo de Tabela 1 ilustre sete contextos, em outros exemplos, um número alternativo de contextos pode ser definido com base em um valor de índice da execução que é codificada. A Tabela 1 pode assumir que o elemento de sintaxe MAX_PLT_SIZE é 32.

[0151] Em um outro exemplo, o codificador de vídeo 20 e o decodificador de vídeo 30 podem ser configurados para usar dois contextos para codificar os valores de execução para o modo Valor com base no valor de índice. Por exemplo, para um índice de modo Valor 0, o codificador de vídeo 20 e o decodificador de vídeo 30 podem ser configurados para usar o contexto de execução (0) para codificar os dados que indicam o valor de execução. Dessa maneira, para todos os outros índices de Valor, o codificador de vídeo 20 e o decodificador de vídeo 30 podem ser configurados para usar o contexto de execução (1) para codificar os dados que indicam o valor de execução.

[0152] Em um outro exemplo, o codificador de vídeo 20 e o decodificador de vídeo 30 podem ser configurados para usar dois contextos para codificar os valores de execução no modo Valor com base no valor de índice. Nesses exemplos, para os índices de Valor 0 e 1, o codificador de vídeo 20 e o decodificador de vídeo 30 podem ser configurados para usar o contexto de execução (0) para codificar os dados que indicam o valor de execução. Para os índices de Valor maior do que 1, o codificador de vídeo 20 e o decodificador de vídeo 30 podem ser configurados para usar o contexto de execução (1) para codificar os dados que indicam o valor de

execução.

[0153] Ainda em um outro exemplo, o codificador de vídeo 20 e o decodificador de vídeo 30 podem ser configurados para usar três contextos para codificar os valores de execução no modo Valor com base no valor de índice. Por exemplo, para o índice de Valor 0, o codificador de vídeo 20 e o decodificador de vídeo 30 podem ser configurados para usar o contexto de execução (0) para a codificação dos dados que indicam a execução. Para os índices de Valor de 1 e 2, o codificador de vídeo 20 e o decodificador de vídeo 30 podem ser configurados para usar o contexto de execução (1) para codificar os dados que indicam a execução. Para os índices de Valor maior do que 2, o codificador de vídeo 20 e o decodificador de vídeo 30 podem ser configurados para usar o contexto de execução (2) para codificar os dados que indicam o valor de execução.

[0154] Ainda em um outro exemplo, três contextos podem ser usados para codificar os valores de execução no modo Valor com base no valor de índice. Por exemplo, para os índices de Valor 0 e 1, o codificador de vídeo 20 e o decodificador de vídeo 30 podem ser configurados para usar o contexto de execução (0) para codificar os dados que indicam o valor de execução. Para os índices de Valor na faixa de 2 a 5, inclusive, o codificador de vídeo 20 e o decodificador de vídeo 30 podem ser configurados para usar o contexto de execução (1) para codificar os dados que indicam o valor de execução. Para os índices de Valor maior do que 5, o codificador de vídeo 20 e o decodificador de vídeo 30 podem ser configurados para usar o contexto de execução (2) para codificar os dados que indicam o valor de

execução.

[0155] Em um outro exemplo, quatro contextos podem ser usados para codificar os valores de execução no modo Valor que dependem (por exemplo, com base em) do valor de índice. Por exemplo, para o índice de Valor 0, o codificador de vídeo 20 e o decodificador de vídeo 30 podem ser configurados para usar o contexto de execução (0) para codificar os dados que indicam o valor de execução. Para os índices de Valor 1 e 2, o codificador de vídeo 20 e o decodificador de vídeo 30 podem ser configurados para usar o contexto de execução (1) para codificar os dados que indicam o valor de execução. Para os índices de Valor na faixa de 3 a 6, inclusive, o codificador de vídeo 20 e o decodificador de vídeo 30 podem ser configurados para usar o contexto de execução (2) para codificar os dados que indicam o valor de execução. Para os índices de Valor maior do que 6, o codificador de vídeo 20 e o decodificador de vídeo 30 podem ser configurados para usar o contexto de execução (3) para codificar os dados que indicam que o valor de execução pode ser usado. Em alguns exemplos, qualquer combinação das técnicas acima pode ser usada para determinar o contexto.

[0156] As técnicas descritas acima podem ser usadas para codificar qualquer combinação de bins de um valor de execução binarizado. Por exemplo, conforme notado acima, um valor de execução binarizado pode incluir um primeiro bin que indica se o valor de execução é maior do que zero, um segundo bin que indica se o valor de execução é maior do que uma, um terceiro bin que indica se o valor de execução é maior do que dois e quaisquer outros bins necessários para

representar o valor de execução. De acordo com os aspectos desta revelação, o codificador de vídeo 20 e o decodificador de vídeo 30 podem ser configurados para selecionar os contextos definidos acima para a codificação do primeiro bin do valor de execução binarizado, o segundo bin do valor de execução binarizado, o terceiro bin do valor de execução binarizado ou qualquer combinação dos mesmos. As técnicas também podem ser aplicadas às outras binarizações de um valor de execução.

[0157] De acordo com os aspectos desta revelação, o contexto pode ter como base um valor de índice atual ou um valor de índice ajustado. Por exemplo, em um exemplo, os contextos de execução podem depender do elemento de sintaxe `adjusted_palette_index` em vez de `syntax_palette_index`, em que o elemento de sintaxe `adjusted_palette_index` pode indicar um índice ajustado com base nas condições a seguir:

[0158] Se `(adjusted_palette_index >= adjustedRefIndex)`, então, `palette_index =`

[0159] De outro modo, `(palette_index = adjusted_palette_index)`

[0160] conforme descrito em Wei Pu et al., "SCCE3: Test B.12- Binarization of Escape Sample and Palette Index," JCTVC-R0065, Sapporo, JP, 30 de junho a 9 de julho de 2014 (JCTVC-R0065). Isto é, de acordo com os aspectos desta revelação, o codificador de vídeo 20 e o decodificador de vídeo 30 podem ser configurados para determinar o contexto para a codificação de um valor de execução com base em um índice de paleta sinalizado em um fluxo de bits (que pode ser associado ao elemento de sintaxe `adjusted_palette_index`)

mais que o índice de paleta atualmente usado para codificar uma amostra em um bloco, em exemplos em que um índice ajustado de paleta é sinalizado no fluxo de bits.

[0161] Em um outro exemplo, se um código binário truncado for usado na codificação do índice, o codificador de vídeo 20 e o decodificador de vídeo 30 podem ser configurados para selecionar os contextos de execução com base nos primeiros "k" bits da sintaxe de índice (por exemplo, conforme descrito, por exemplo, na Seção 9.3.3.x de JCTVC-R0065). Por exemplo, com o uso do código binário truncado, o elemento de sintaxe para representar um índice de paleta pode usar k bits ou k+1 bits, com base no valor relativo do índice de paleta para a variável u, descrita abaixo (conforme descrito na Seção 9.3.3.x de JCTVC-R0065)):

[0162] A entrada para esse processo é uma solicitação para uma binarização de TB para um elemento de sintaxe com valor synVal, cMax. A saída desse processo é a binarização de TB do elemento de sintaxe.

[0163] Seja $n = cMax + 1$, $k = \text{piso}(\log_2(n))$ de modo que $2^k n \leq 2^{k+1}$ e seja $u = 2^{k+1} - n$.

[0164] Se $\text{synVal} < u$, a coluna de bin de TB é especificada pela representação binária de synVal com comprimento k.

[0165] De outro modo, a coluna de bin de TB é especificada pela representação binária de $\text{synVal} + u$ com comprimento $k + 1$.

[0166] Nesse exemplo, os contextos de execução dependem apenas dos primeiros k bits decodificados. Consequentemente, por exemplo, o decodificador de vídeo 30 não precisa esperar para decodificar o k+1 bit possível antes

de determinar os contextos.

[0167] Em um outro exemplo, o codificador de vídeo 20 e o decodificador de vídeo 30 podem ser configurados para usar múltiplos contextos com base no valor de índice, o tamanho da paleta de um bloco que é atualmente codificado e/ou um tamanho do bloco que é atualmente codificado. Se o modo de Execução Limitada descrito acima for usado para codificar um bloco, o codificador de vídeo 20 e o decodificador de vídeo 30 podem ser configurados para selecionar o contexto com base no valor de índice, o tamanho do bloco e/ou o limite de índice acima do qual uma execução não é sinalizada.

[0168] Se o modo de Execução de Transição também for usado, essa técnica também pode ser estendida para o modo de Execução de Transição. Nesse caso, para o modo de Execução de Transição, o codificador de vídeo com base no valor de índice, o tamanho do bloco e/ou o limite de índice acima do qual uma execução não é sinalizada. Nesse caso, os conjuntos separados de contextos podem ser usados para as execuções que correspondem ao modo Valor e modo de Execução de Transição.

[0169] Outros aspectos desta revelação se referem às execuções de codificação de valores de índice que são codificados com o uso de mais de um modo de paleta. Por exemplo, no modo CopyFromTop descrito no documento JCTVC-Q0094, não é possível copiar uma amostra codificada como uma amostra de fuga ou um sinalizador de fuga da linha. Um exemplo de duas fileiras de índices é mostrado na Tabela 2 abaixo:

TABELA 2- ÍNDICES DE EXEMPLO

Linha acima	1	2	2	4	ESC	5	1
Linha atual	1	2	2	4	ESC	5	2

[0170] em que os valores numéricos representam os índices e ESC representa uma amostra codificada como uma amostra de fuga. Dessa maneira, o começo da amostra mais à esquerda no exemplo acima, como por JCTVC-Q0094, codificador de vídeo 20 pode sinalizar o modo CopyFromTop. O codificador de vídeo 20 também pode sinalizar um valor de execução de três, visto que os valores ou sinalizadores ou valores de fuga de cópia não são permitidos. Consequentemente, uma amostra codificada como uma amostra de fuga termina a execução.

[0171] O documento de submissão padrão Jianqing Zhu et al., "AHG10: Modified Copy Above Mode for Palette Based Coding", JCTVC-Q0174, Valência, ES, 27 de março a 4 de abril de 2014 (JCTVC-Q0174) descreve permitir a cópia de valores de fuga (por exemplo, os valores de pixel atual) em uma execução. No entanto, essa técnica exige que os valores associados às amostras de fuga sejam idênticos.

[0172] De acordo com os aspectos desta revelação, o codificador de vídeo 20 e o decodificador de vídeo 30 podem ser configurados para codificar os dados que indicam mais do que um modo de paleta em uma execução de índices. Por exemplo, o codificador de vídeo 20 e o decodificador de vídeo 30 podem ser configurados para copiar os sinalizadores de fuga (na sinalização de fuga explícita)

ou um índice de fuga (na sinalização de fuga implícita) com outros índices que têm valores de color associados em uma paleta. Em relação ao exemplo de Tabela 2 acima, o codificador de vídeo 20 pode ser configurada para sinalizar o modo CopyFromTop seguido pelos dados que indicam um valor de execução de seis. Essa técnica pode alcançar a vantagem de aumento do tamanho de execuções, que pode resultar na compressão de dados aprimorada.

[0173] De acordo com os aspectos desta revelação, durante a execução de índices codificados em relação aos índices de uma outra fileira se uma amostra da execução tiver uma amostra localizada diretamente acima de uma amostra que é codificada como uma amostra de fuga, o codificador de vídeo 20 e o decodificador de vídeo 30 podem ser configurados para inferir que uma amostra é codificada como uma amostra de fuga. Dessa maneira, o valor atual de uma amostra pode ser sinalizado após a execução.

[0174] Em alguns exemplos, o codificador de vídeo 20 e o decodificador de vídeo 30 podem ser configurados para não usar o modo CopyFromTop para uma amostra se a amostra vizinha acima (por exemplo, uma amostra localizada diretamente acima de uma amostra que é codificada) for codificada como uma amostra de fuga. Em exemplos em que se permite que as amostras codificadas como amostras de fuga estejam incluídas em uma execução associada ao modo de paleta CopyFromTop (conforme descrito acima) e algumas redundâncias associadas à sinalização do modo Fuga podem ser removidas. Em um primeiro exemplo, de acordo com os aspectos desta revelação, se uma amostra que segue uma execução de índices tiver uma amostra vizinha acima que é codificada como uma

amostra de fuga, o codificador de vídeo 20 e o decodificador de vídeo 30 podem ser configurados para não codificar a amostra a seguir como uma amostra de fuga. Dessa maneira, a amostra não pode ser codificada como uma amostra de fuga, devido ao fato de que estaria, de outro modo, incluída na execução CopyFromTop. Dessa maneira, o codificador de vídeo 20 e o decodificador de vídeo 30 podem ser configurados para omitir a codificação de um sinalizador de fuga ou um índice de fuga, devido ao fato de que o codificador de vídeo 20 e o decodificador de vídeo 30 podem ser configurados para inferir que uma amostra não é codificada como uma amostra de fuga.

[0175] Pelas mesmas razões do exemplo descrito acima, o codificador de vídeo 20 e o decodificador de vídeo 30 podem ser configurados para não codificar uma amostra com o uso do modo CopyFromTop após uma execução CopyFromTop. Consequentemente, o codificador de vídeo 20 e o decodificador de vídeo 30 podem ser configurados para omitir a codificação de um elemento de sintaxe relacionado ao modo (por exemplo, um sinalizador que se diferencia entre modos Valor ou CopyFromTop), e o codificador de vídeo 20 e o decodificador de vídeo 30 podem ser configurados para inferir que uma amostra é codificada no modo Valor.

[0176] Em um segundo exemplo, se a última amostra incluída na execução CopyFromTop é uma amostra de fuga e a amostra a seguir é codificada com o modo Valor, o codificador de vídeo 20 e o decodificador de vídeo 30 podem ser configurados para ajustar o índice do modo Valor (bem como o número de índices possíveis máximo que pode ser usado na binarização truncada para codificar o índice de Valor)

pela verificação do valor de índice da amostra vizinha acima da amostra a seguir. Nesse caso, se o pixel vizinho acima tiver um índice, então, o codificador de vídeo 20 e o decodificador de vídeo 30 podem ser configurados para reduzir o índice de Valor por um se o valor do índice for maior do que o índice acima. O codificador de vídeo 20 e o decodificador de vídeo 30 podem ser configurados para reduzir o decodificador máximo 30 que pode verificar o valor de índice decodificado para determinar se o índice é maior ou igual ao índice vizinho acima e, se for o caso, aumentar o valor decodificado por um.

[0177] A Figura 2 é um diagrama de blocos que ilustra um codificador de vídeo exemplificativo 20 que pode implantar as técnicas desta revelação. A Figura 2 é fornecida para propósitos de explicação e não deve ser considerada como limitante das técnicas conforme amplamente exemplificado e descrito nesta revelação. Para propósitos de explicação, esta revelação descreve o codificador de vídeo 20 no contexto de codificação de HEVC. Entretanto, as técnicas desta revelação podem ser aplicáveis a outros padrões ou métodos de codificação.

[0178] O codificador de vídeo 20 representa um exemplo de um dispositivo que pode ser configurado para realizar técnicas para codificação de vídeo com base em paleta de acordo com vários exemplos descritos nesta revelação. Por exemplo, o codificador de vídeo 20 pode ser configurado para codificar seletivamente diversos blocos de dados de vídeo, tais como CUs ou PUs na codificação de HEVC, com o uso de codificação com base em paleta ou codificação sem base em paleta. Os modos de codificação não com base em

paleta podem se referir a vários modos de codificação temporais interpreditivo ou modos de codificação espaciais intrapreditivo, tais como vários modos de codificação especificado por HEVC Draft 10. O codificador de vídeo 20, em um exemplo, pode ser configurado para gerar uma paleta que tem entradas que indicam valores de pixel, selecionar valores de pixel em uma paleta para representarem valores de pixels de pelo menos alguns locais de pixel em um bloco de dados de vídeo e sinalizar informações que associam pelo menos alguns dos locais de pixel no bloco de dados de vídeo com entradas na paleta que correspondem, respectivamente, aos valores de pixel selecionados na paleta. As informações sinalizadas podem ser usadas pelo decodificador de vídeo 30 para decodificar os dados de vídeo.

[0179] No exemplo da Figura 2, o codificador de vídeo 20 inclui uma unidade de processamento de predição 100, uma memória de dados de vídeo 101, uma unidade de geração residual 102, uma unidade de processamento de transformada 104, uma unidade de quantização 106, uma unidade de quantização inversa 108, uma unidade de processamento de transformada inversa 110, uma unidade de reconstrução 112, uma unidade de filtro 114, um buffer de imagem decodificada 116 e uma unidade de codificação por entropia 118. A unidade de processamento de predição 100 inclui uma unidade de processamento de interpredição 120 e uma unidade de processamento de intrapredição 126. A unidade de processamento de interpredição 120 inclui uma unidade estimativa de movimento e uma unidade de compensação de movimento (não mostrada). O codificador de vídeo 20 também inclui uma unidade de codificação com base em paleta 122

configurada para realizar vários aspectos das técnicas de codificação com base em paleta descritas nesta revelação. Em outros exemplos, o codificador de vídeo 20 pode incluir mais, poucos ou diferentes componentes funcionais.

[0180] A memória de dados de vídeo 101 pode armazenar dados de vídeo a serem codificados pelos componentes de codificador de vídeo 20. Os dados de vídeo armazenados na memória de dados de vídeo 101 podem ser obtidos, por exemplo, a partir da fonte de vídeo 18. O buffer de imagem decodificada 116 pode ser uma memória de figuração de referência que armazena dados de vídeo de referência para uso em dados de vídeo de codificação pelo codificador de vídeo 20, por exemplo, em modos de intra- ou inter-codificação. A memória de dados de vídeo 101 e o buffer de imagem decodificada 116 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 101 e o buffer de imagem decodificada 116 podem ser fornecidos pelo menos dispositivo de memória ou dispositivos de memória separados. Em vários exemplos, a memória de dados de vídeo 101 pode estar no chip com outros componentes de codificador de vídeo 20 ou fora do chip em relação àqueles componentes.

[0181] O codificador de vídeo 20 pode receber dados de vídeo. O codificador de vídeo 20 pode codificar cada CTU em uma fatia de uma figuração dos dados de vídeo. Cada uma das CTUs pode estar associada aos blocos de árvore de codificação luma igualmente dimensionados (CTBs) e CTBs

correspondentes da figuração. Como parte da codificação de uma CTU, a unidade de processamento de predição 100 pode realizar o particionamento de árvore quadrática para dividir os CTBs da CTU em blocos progressivamente menores. O menor bloco pode ser blocos de codificação de CUs. Por exemplo, a unidade de processamento de predição 100 pode separar um CTB associado a uma CTU em quatro sub-blocos igualmente dimensionados, separar um ou mais dos sub-blocos em quatro sub-blocos igualmente dimensionados e assim por diante.

[0182] O codificador de vídeo 20 pode codificar as CUs de uma CTU para gerar representações codificadas das CUs (isto é, CUs codificadas). Como parte da codificação de uma CU, a unidade de processamento de predição 100 pode particionar os blocos de codificação associados à CU entre uma ou mais PUs da CU. Dessa forma, cada PU pode ser associada a um bloco de predição de luma e os blocos de predição de croma correspondentes. O codificador de vídeo 20 e o decodificador de vídeo 30 podem suportar as PUs que têm vários tamanhos. Conforme indicado acima, o tamanho de uma CU pode se referir ao tamanho do bloco de codificação luma da CU e o tamanho de uma PU pode se referir ao tamanho de um bloco de predição luma da PU. Presumindo-se que o tamanho de uma CU particular seja $2N \times 2N$, o codificador de vídeo 20 e o decodificador de vídeo 30 podem suportar tamanhos de PU de $2N \times 2 \times N$ ou $N \times N$ para intrapredição e tamanhos simétricos de PU de $2N \times 2N$, $2N \times N$, $N \times 2N$, $N \times N$ ou similar para interpredição. O codificador de vídeo 20 e o decodificador de vídeo 30 também podem suportar particionamento assimétrico para tamanhos de PU de $2 \times nU$, $2N \times nD$, $nL \times 2N$ e $nR \times 2N$ para interpredição.

[0183] A unidade de processamento de

interpriedição 120 pode gerar dados preditivos para uma PU através da realização de interpriedição em cada PU de uma CU. Os dados preditivos para a PU podem incluir blocos preditivos da PU e informações de movimento para a PU. A unidade de interpriedição 121 pode realizar diferentes operações para uma PU de uma CU que depende de se a PU está em uma fatia I, uma fatia P ou uma fatia B. Em uma fatia I, todas as PUs são intrapreditivas. Por conseguinte, se a PU estiver em uma fatia I, a unidade de interpriedição 121 não realiza interpriedição na PU. Dessa forma, para blocos codificados no modo I, o bloco predito é formado com uso de predição espacial a partir de blocos vizinhos anteriormente codificados dentro do mesmo quadro.

[0184] Se uma PU estiver em uma fatia P, a unidade de estimativa de movimento da unidade de processamento de interpriedição 120 pode buscar as figurações de referência em uma lista de figurações de referência (por exemplo, "RefPicList0") para uma região de referência para a PU. A região de referência para a PU pode ser uma região, dentro de uma figuração de referência, que contém blocos de amostra que correspondem mais proximamente aos blocos de predição da PU. A unidade de estimativa de movimento pode gerar um índice de referência que indica uma posição em RefPicList0 da figuração de referência que contém a região de referência para a PU. Adicionalmente, a unidade de estimativa de movimento 122 pode gerar um MV que indica um deslocamento espacial entre um bloco de codificação da PU e uma localidade de referência associada à região de referência. Por exemplo, o MV pode ser um vetor bidimensional que fornece um desvio a partir das coordenadas na figuração

decodificada atual para as coordenadas em uma figuração de referência. A unidade estimativa de movimento pode emitir o índice de referência e o MV como as informações de movimento da PU. A unidade de compensação de movimento de unidade de processamento de interpriedição 120 pode gerar os blocos preditivos da PU com base nas amostras atuais ou interpoladas na localização de referência indicada pelo vetor de movimento da PU.

[0185] Se uma PU estiver em uma fatia B, a unidade estimativa de movimento pode realizar monopredição ou bipredição para a PU. Para realizar monopredição para a PU, a unidade estimativa de movimento pode pesquisar as figurações de referência de RefPicList0 ou uma segunda lista de figuração de referência ("RefPicList1") para uma região de referência para a PU. A unidade estimativa de movimento pode emitir, como as informações de movimento da PU, um índice de referência que indica uma posição em RefPicList0 ou RefPicList1 da figuração de referência que contém a região de referência, um MV que indica um deslocamento espacial entre um bloco de predição da PU e uma localização de referência associada à região de referência e um ou mais indicadores de direção de predição que indicam se a figuração de referência está em RefPicList0 ou gera os blocos de preditivos da PU com base pelo menos em parte nas amostras interpoladas ou atuais da região de referência indicada pelo vetor de movimento da PU.

[0186] Para realizar a interpriedição bidirecional para uma PU, a unidade estimativa de movimento pode pesquisar as figurações de referência em RefPicList0 para uma região de referência para a PU e também pode

pesquisar as figurações de referência em RefPicList1 para uma outra região de referência para a PU. A unidade estimativa de movimento pode gerar índices de figuração de referência que indicam posições em RefPicList0 e RefPicList1 das figurações de referência que contêm as regiões de referência. Além disso, a unidade estimativa de movimento pode gerar MVs que indicam deslocamentos espaciais entre a localização de referência associada às regiões de referência e um bloco de amostra da PU. As informações de movimento da PU podem incluir os índices de referência e os MVs da PU. A unidade de compensação de movimento pode gerar os blocos preditivos da PU com base pelo menos em parte nas amostras atuais ou interpoladas nas regiões de referência indicadas pelo vetor de movimento da PU.

[0187] De acordo com vários exemplos desta revelação, o codificador de vídeo 20 pode ser configurado para realizar codificação com base em paleta. Em relação à estrutura de HEVC, como um exemplo, as técnicas de codificação com base em paleta podem ser configuradas para serem usadas como um modo de unidade de controle (CU). Em outros exemplos, as técnicas de codificação com base em paleta podem ser configuradas para ser usadas como um modo de PU na estrutura de HEVC. Consequentemente, todos os processos revelados descritos no presente documento (através desta revelação) no contexto de um modo de CU podem, adicional ou alternativamente, se aplicar à PU. No entanto, esses exemplos com base em HEVC não seriam considerados a uma restrição ou limitação da codificação com base em paleta técnicas descritas no presente documento, tais como técnicas que podem ser aplicadas para funcionar de modo independente

ou como parte de outros existentes ou ainda a serem sistemas/padrões desenvolvidos. Nesses casos, a unidade para codificar de paleta pode ser blocos quadrados, blocos retangulares ou mesmo regiões com formato não retangular.

[0188] A unidade de codificação com base em paleta 122, por exemplo, pode realizar decodificação com base em paleta quando um modo de codificação com base em paleta é selecionado, por exemplo, para uma CU ou PU. Por exemplo, a unidade de codificação com base em paleta 122 pode ser configurada para gerar uma paleta que tem entradas que indicam valores de pixels, selecionar valores de pixels em uma paleta para representar valores de pixels de pelo menos algumas posições de um bloco de dados de vídeo e informações de sinal se associam pelo menos algumas das posições do bloco de dados de vídeo às entradas na paleta que correspondem, respectivamente, aos valores de pixels selecionados. Embora várias funções sejam descritas conforme sendo realizadas pela unidade de codificação com base em paleta 122, algumas ou todas as tais funções podem ser realizadas por outras unidades de processamento ou uma combinação de diferentes unidades de processamento.

[0189] De acordo com os aspectos desta revelação, a unidade de codificação com base em paleta 122 pode ser configurada para realizar qualquer combinação das técnicas para a codificação de paleta descrita no presente documento. Por exemplo, conforme descrito em relação às Figuras 4 a 7 abaixo, de acordo com os aspectos desta revelação, a unidade de codificação com base em paleta 122 pode ser configurada para reinicializar uma paleta de preditor para construir uma paleta para um bloco de dados de

vídeo com base em um ou mais blocos de uma outra linha. Por exemplo, conforme descrito no presente documento, a unidade de codificação com base em paleta 122 pode determinar uma primeira paleta para um primeiro bloco de dados de vídeo que está localizado em uma primeira fileira de blocos. A unidade de codificação com base em paleta 122 também pode gerar uma paleta de preditor ao codificar um ou mais outros blocos na primeira fileira. Mediante a codificação de um bloco em uma segunda fileira, a unidade de codificação com base em paleta 122 pode reinicializar a paleta de preditor para determinar uma paleta do bloco na segunda fileira com base nas entradas da primeira paleta.

[0190] Em um outro exemplo, a unidade de codificação com base em paleta 122 pode ser configurada para codificar uma execução de índices que inclui amostras codificadas com o uso de mais de um modo de paleta. Por exemplo, a unidade de codificação com base em paleta 122 pode determinar os valores de índice de uma primeira fileira de um bloco de dados de vídeo, em que os valores de índice incluem um ou mais índices que estão associados a um valor de cor na paleta e um elemento de sintaxe que não está associado a um valor de cor na paleta. A unidade de codificação com base em paleta 122 também pode codificar uma execução de valores de índice de uma segunda fileira do bloco de dados de vídeo em relação aos valores de índice da primeira fileira, em que a execução inclui os um ou mais índices que estão associados a um valor de cor na paleta e o elemento de sintaxe que não está associado a um valor de cor na paleta.

[0191] A unidade de processamento de

intrapredição 126 pode gerar dados preditivos para uma PU realizando-se a intrapredição na PU. Os dados preditivos para a PU podem incluir blocos preditivos para a PU e vários elementos de sintaxe. A unidade de processamento de intrapredição 126 pode realizar intrapredição em PUs em fatias I, fatias P e fatias B.

[0192] Para realizar a intrapredição em uma PU, a unidade de processamento de intrapredição 126 pode usar múltiplos modos de intrapredição para gerar múltiplos conjuntos de dados preditivos para a PU. unidade de processamento de intrapredição 126 pode usar amostras a partir dos blocos de amostra de PUs vizinhas para gerar um bloco preditivo para uma PU. As PUs vizinhas podem estar acima, acima e à direita, acima e à esquerda ou à esquerda da PU, assumindo que uma unidade de processamento 126 pode usar vários números de modos de intrapredição, por exemplo, 33 modos de intrapredição direcionais. Em alguns exemplos, o número de modos de intrapredição pode depender do tamanho da região associada à PU.

[0193] A unidade de processamento de predição 100 pode selecionar os dados preditivos para as PUs de uma CU de entre os dados preditivos gerados pela unidade de processamento de interpredição 120 para as PUs ou os dados preditivos gerados pela unidade de processamento de intrapredição 126 para as PUs. Em alguns exemplos, a unidade de processamento de predição 100 seleciona os dados preditivos para as PUs da CU com base nas métricas de taxa/distorção dos conjuntos de dados preditivos. Os blocos preditivos dos dados preditivos selecionados podem ser referidos no presente documento como blocos preditivos

selecionados.

[0194] A unidade de geração residual 102 pode gerar, com base nos blocos de codificação luma, Cb e Cr de uma CU e os blocos preditivos luma selecionados, Cb e Cr das PUs da CU, os blocos residuais luma, Cb e Cr da CU. Por exemplo, a unidade de geração residual 102 pode gerar os blocos residuais da CU de modo que cada amostra nos blocos residuais tenha um valor igual a uma diferença entre uma amostra em um bloco de codificação da CU e uma amostra correspondente em um bloco preditivo selecionado correspondente de uma PU da CU.

[0195] A unidade de processamento de transformada 104 pode realizar particionamento de árvore quadrática para separar os blocos residuais associados a uma CU em blocos de transformada associados às TUs da CU. Dessa forma, uma TU pode ser associada a um bloco de transformada de luma e dois blocos de transformada de croma. Os tamanhos e posições dos blocos de transformada luma e croma de TUs de uma CU podem ou não podem ter base nos tamanhos e posições de blocos de predição das PUs da CU. Uma estrutura de árvore quadrática conhecida como uma "árvore quadrática residual" (RQT) pode incluir nós associados a cada uma das regiões. As TUs de uma CU podem corresponder aos nós-folha da RQT.

[0196] A unidade de processamento de transformação 104 pode gerar blocos de coeficiente de transformação para cada TU de uma CU através da aplicação de uma ou mais transformações aos blocos de transformação da TU. A unidade de processamento de transformação 104 pode aplicar várias transformações a um bloco de transformação associado a uma TU. Por exemplo, uma unidade de processamento

de transformada 104 pode aplicar uma transformada discreta de cosseno (DCT), uma transformada direcional ou uma transformada conceitualmente similar a um bloco de transformada. Em alguns exemplos, a unidade de processamento de transformada 104 não aplica transformadas a um bloco de transformada. Em tais exemplos, o bloco de transformada pode ser tratado como um bloco de coeficiente de transformada.

[0197] A unidade de quantização 106 pode quantizar os coeficientes de transformada em um bloco de coeficiente. O processo de quantização pode reduzir a profundidade de bit associada ao mesmo coeficiente de transformada ou a todos os coeficientes de transformada. Por exemplo, um coeficiente de transformada de n-bits pode ser arredondado para baixo para um coeficiente de transformada de m-bits durante a quantização, em que n é maior que m. A unidade de quantização 106 pode quantizar um bloco de coeficientes associado a uma TU de uma CU com base em um valor de parâmetro de quantização (QP) associado à CU. O codificador de vídeo 20 pode ajustar o grau de quantização aplicado aos blocos de coeficiente associados a uma CU através do ajuste do valor de QP associado à CU. A quantização pode apresentar perda de informações, desse modo, os coeficientes de transformada quantizados podem ter precisão menor que aqueles originais.

[0198] A unidade de quantização inversa 108 e a unidade de processamento de transformada inversa 110 pode aplicar quantização inversa e transformadas inversas a um bloco de coeficiente, respectivamente, para reconstruir um bloco residual a partir do bloco de coeficiente. A unidade de reconstrução 112 pode adicionar o bloco residual

reconstruído às amostras correspondentes a partir de um ou mais blocos preditivos gerados pela unidade de processamento de predição 100 para produzir um bloco de transformada reconstruído associado a uma TU. Através da reconstrução dos blocos de transformada para cada TU de uma CU, dessa maneira, o codificador de vídeo 20 pode reconstruir os blocos de codificação da CU.

[0199] A unidade de filtro 114 pode realizar uma ou mais operações de desbloqueio para reduzir os artefatos de bloqueio nos blocos de codificação associados a uma CU. O buffer de imagem decodificada 116 pode armazenar os blocos de codificação reconstruídos após a unidade de filtro 114 realizar as uma ou mais operações de desbloqueio nos blocos de codificação reconstruídos. A unidade de processamento de interpredição 120 pode utilizar uma imagem de referência que contém os blocos de codificação reconstruídos para realizar a interpredição em PUs de outras imagens. Além disso, uma unidade de processamento de intrapredição 126 pode usar blocos de codificação reconstruídos no buffer de imagem decodificada 116 para realizar intrapredição em outras PUs na mesma imagem como da CU.

[0200] A unidade de codificação por entropia 118 pode receber dados a partir de outros componentes funcionais do codificador de vídeo 20. Por exemplo, a unidade de codificação por entropia 118 pode receber blocos de coeficiente a partir da unidade de quantização 106 e pode receber elementos de sintaxe a partir da unidade de processamento de predição 100. A unidade de codificação por entropia 118 pode realizar uma ou mais operações de

codificação por entropia nos dados para gerar dados codificados por entropia. Por exemplo, a unidade de codificação por entropia 118 pode realizar uma operação de codificação de comprimento variável adaptativo ao contexto (CAVLC), uma operação de CABAC, uma operação de codificação de comprimento de variável para variável (V2V), uma operação de codificação aritmética binária adaptativa a contexto com base em sintaxe (SBAC), uma operação de codificação de Particionamento por Entropia com Intervalo de Probabilidade (PIPE), uma operação de codificação de Exponencial de Golomb ou um outro tipo de operação de codificação por entropia nos dados.

[0201] Para realizar CABAC, a unidade de codificação por entropia 118 pode selecionar um modelo de contexto para aplicar a um certo contexto para codificar os símbolos a serem transmitidos. De acordo com os aspectos desta revelação, a unidade de codificação por entropia 118 pode ser configurada para usar o contexto com base no valor de índice do valor de execução que é codificado. Por exemplo, a unidade de codificação por entropia 118 pode ser configurada para determinar um valor de execução que indica um comprimento de execução de uma execução de um índice de paleta de um bloco de dados de vídeo. A unidade de codificação por entropia 118 pode ser configurada para determinar um contexto para os dados de codificação adaptativa ao contexto que representa o valor de execução com base no valor de índice de paleta.

[0202] O codificador de vídeo 20 pode emitir um fluxo de bits que inclui dados codificados por entropia gerados pela unidade de codificação por entropia 118. Por

exemplo, o fluxo de bits pode incluir dados que representam uma RQT para uma CU.

[0203] A Figura 3 é um diagrama de blocos que ilustra um decodificador de vídeo exemplificativo 30 que é configurado para implementar as técnicas desta revelação. A Figura 3 é fornecida para propósitos de explicação e não está limitada às técnicas conforme amplamente exemplificado e descrito nesta revelação. Para propósitos de explicação, esta revelação descreve o decodificador de vídeo 30 no contexto de codificação de HEVC. Entretanto, as técnicas desta revelação podem ser aplicáveis a outros padrões ou métodos de codificação.

[0204] O codificador de vídeo 30 representa um exemplo de um dispositivo que pode ser configurado para realizar técnicas para codificação de vídeo com base em paleta de acordo com vários exemplos descritos nesta revelação. Por exemplo, o codificador de vídeo 30 pode ser configurado para decodificar seletivamente diversos blocos de dados de vídeo, tais como CUs ou PUs na codificação de HEVC, com uso de codificação com base em paleta ou codificação sem base em paleta. Os modos de codificação não com base em paleta podem se referir a vários modos de codificação temporais interpreditivo ou modos de codificação espaciais intrapreditivo, tais como vários modos de codificação especificado por HEVC Draft 10. O decodificador de vídeo 30, em um exemplo, pode ser configurado para gerar uma paleta que tem entradas que indicam valores de pixel, receber informações que associam pelo menos algumas localizações de pixel em um bloco de dados de vídeo com entradas na paleta, selecionar os valores de pixel na paleta

com base nas informações e reconstruir os valores de pixel do bloco com base nos valores de pixel selecionados na paleta.

[0205] No exemplo da Figura 3, o decodificador de vídeo 30 inclui uma unidade de decodificação por entropia 150, memória de dados de vídeo 151, uma unidade de processamento de predição 152, uma unidade de quantização inversa 160 e um buffer de imagem decodificada 162. A unidade de processamento de predição 152 inclui uma unidade de compensação de movimento 164 e uma unidade de processamento de intrapredição 166. O decodificador de vídeo 30 também inclui uma unidade de decodificação com base em paleta 165 configurada para realizar vários aspectos das técnicas de codificação com base em paleta descritas nesta revelação. Em outros exemplos, o decodificador de vídeo 30 pode incluir mais, poucos ou diferentes componentes funcionais.

[0206] A memória de dados de vídeo 151 pode armazenar dados de vídeo, tal como um fluxo de bits de vídeo codificado, a ser decodificado pelos componentes de decodificador de vídeo 30. Os dados de vídeo armazenados na memória de dados de vídeo 151 podem ser obtidos, por exemplo, a partir da mídia legível por computador 16, por exemplo, a partir de uma fonte de vídeo local, tal como uma câmera, através de comunicação de rede sem fio ou cabeada de dados de vídeo ou acessando-se a mídia de armazenamento de dados físicos. A memória de dados de vídeo 151 pode formar um buffer de imagem codificada (CPB) que armazena dados de vídeo codificados a partir de um fluxo de bits de vídeo codificado. O buffer de imagem decodificada 162 pode ser uma memória de figuração de referência que armazena os dados de vídeo de

referência para uso em dados de vídeo de decodificação pelo decodificador de vídeo 30, por exemplo, em modos de intra ou intercodificação. A memória de dados de vídeo 151 e o buffer de imagem decodificada 162 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 151 e o buffer de imagem decodificada 162 podem ser fornecidos pelo menos dispositivo de memória ou dispositivos de memória separados. Em vários exemplos, a memória de dados de vídeo 151 pode estar no chip com outros componentes de decodificador de vídeo 30, ou fora do chip em relação àqueles componentes.

[0207] Um buffer de imagem codificada (CPB) 79 pode receber e armazenar dados de vídeo codificados (por exemplo, unidades de NAL) de um fluxo de bits. A unidade de decodificação por entropia 150 pode receber dados de vídeo codificados (por exemplo, unidades de NAL) a partir do CPB e analisar as unidades de NAL para decodificar os elementos de sintaxe. A unidade de decodificação por entropia 150 pode decodificar por entropia os elementos de sintaxe codificados por entropia nas unidades de NAL.

[0208] De acordo com os aspectos desta revelação, a unidade de decodificação por entropia 150 pode ser configurada para usar o contexto com base no valor de índice do valor de execução que é codificado. Por exemplo, a unidade de decodificação por entropia 150 pode ser configurada para determinar um valor de execução que indica um comprimento de execução de uma execução de um índice de

paleta de um bloco de dados de vídeo. A unidade de decodificação por entropia 150 pode ser configurada para determinar um contexto para os dados de codificação adaptativa ao contexto que representa o valor de execução com base no valor de índice de paleta.

[0209] A unidade de processamento de predição 152, a unidade de quantização inversa 154, a unidade de processamento de transformada inversa 156, a unidade de reconstrução 158 e a unidade de filtro 160 podem gerar dados de vídeo decodificados com base em elementos de sintaxe extraídos do fluxo de bits.

[0210] As unidades de NAL do fluxo de bits podem incluir unidades de NAL de fatia codificada. Como parte de decodificação do fluxo de bits, a unidade de decodificação por entropia 150 pode extrair e decodificar por entropia elementos de sintaxe a partir das unidades de NAL de fatia codificada. Cada uma das fatias codificadas pode incluir um cabeçalho de fatia e dados de fatia. O cabeçalho de fatia pode conter elementos de sintaxe pertencentes a uma fatia. Os elementos de sintaxe no cabeçalho de fatia podem incluir um elemento de sintaxe que identifica um PPS associado a uma figuração que contém a fatia.

[0211] Além de decodificar os elementos de sintaxe a partir do fluxo de bits, o decodificador de vídeo 30 pode realizar uma operação de reconstrução em uma CU não particionada. Para realizar a operação de reconstrução em uma CU não particionada, o decodificador de vídeo 30 pode realizar uma operação de reconstrução em cada TU da CU. Através da realização da operação de reconstrução para cada TU da CU, o decodificador de vídeo 30 pode reconstruir os

blocos residuais da CU.

[0212] Como parte de realização de uma operação de reconstrução em uma TU de uma CU, a unidade de quantização inversa 154 pode inverter e quantizar, isto é, desquantizar, os blocos de coeficiente associados à TU. A unidade de quantização inversa 154 pode usar um valor de QP associado a CU da TU para determinar um grau de quantização e, de modo similar, para aplicar um grau de quantização inversa para unidade de quantização inversa 154. Isto é, a razão de comparação, isto é, a razão do número de bits usada para representar sequência original e a compactada, pode ser controlada através do ajuste do valor do QP usado ao quantizar os coeficientes de transformada. A razão de comparação também pode depender do método de codificação de entropia empregado.

[0213] Após a unidade de quantização inversa 154 quantizar inversamente um bloco de coeficiente, a unidade de processamento de transformada inversa 156 pode aplicar uma ou mais transformadas inversas ao bloco de coeficiente a fim de gerar um bloco residual associado à TU. Por exemplo, a unidade de processamento de transformada inversa 156 pode aplicar uma DCT inversa, uma transformada de número inteiro inversa, uma transformada de Karhunen-Loeve inversa (KLT), uma transformada rotacional inversa, uma transformada direcional inversa ou uma outra transformada inversa ao bloco de coeficiente.

[0214] Se uma PU for codificada com uso de intrapredição, a unidade de processamento de intrapredição 166 pode realizar a intrapredição para gerar blocos preditivos para a PU. Os blocos Cb e Cr de intrapredição

para a PU com base nos blocos de predição de PUs espacialmente vizinhas. A unidade de processamento de intrapredição 166 pode determinar o modo de intrapredição para a PU com base em um ou mais elementos de sintaxe decodificados a partir do fluxo de bits.

[0215] A unidade de processamento de predição 152 pode construir uma primeira lista de imagem de referência (RefPicList0) e uma segunda imagem de referência (RefPicList1) com base nos elementos de sintaxe extraídos do fluxo de bits. Além disso, se uma PU for codificada com uso de interpredição, a unidade de decodificação por entropia 150 pode extrair as informações de movimento para a PU. A unidade de compensação de movimento 164 pode determinar, com base nas informações de movimento da PU, uma ou mais regiões de referência para a PU. A unidade de compensação de movimento 164 pode gerar, com base em blocos de amostra nos um ou mais blocos de referência para a PU, os blocos luma, Cb e Cr para a PU.

[0216] A unidade de reconstrução 158 pode usar os blocos de transformada luma, Cb e Cr associados às TUs de uma CU e os blocos luma, Cb e Cr preditivos das PUs da CU, isto é, dados de intrapredição ou dados de interpredição, caso seja aplicável, para reconstruir os blocos de codificação luma, Cb e Cr) da CU. Por exemplo, a unidade de reconstrução 158 pode adicionar amostras dos blocos de transformada luma, Cb e Cr às amostras correspondentes dos blocos preditivos luma, Cb e Cr preditivos para reconstruir os blocos de codificação luma, Cb e Cr da CU.

[0217] A unidade de filtro 160 pode realizar uma operação de de-blocagem para reduzir os artefatos de

característica de bloco associados aos blocos de codificação de luma, Cb e Cr da CU. O decodificador de vídeo 30 pode armazenar os blocos de codificação de luma, Cb e Cr da CU no buffer de imagem decodificada 162. O buffer de imagem decodificada 162 pode fornecer figurações de referência para compensação, intrapredição e apresentação de movimento subsequente em um dispositivo de visor, tal como dispositivo de visor 32 da Figura 1. Por exemplo, o decodificador de vídeo 30 pode realizar, com base nos blocos de luma, Cb e Cr no buffer de imagem decodificada 162, operações de intrapredição ou interpredição em PUs de outras CUs.

[0218] Conforme notado acima, o decodificador de vídeo 30 pode ser configurado para realizar a codificação com base em paleta. A unidade de decodificação com base em paleta 165, por exemplo, pode realizar decodificação com base em paleta quando um modo de decodificação com base em paleta é selecionado, por exemplo, para uma CU ou PU. Por exemplo, a unidade de decodificação com base em paleta 165 pode ser configurada para gerar uma paleta que tem entradas que indicam valores de pixel, receber as informações que associam pelo menos algumas localizações de pixel em um bloco de dados de vídeo com entradas na paleta, selecionar valores de pixel na paleta com base nas informações e reconstruir os valores de pixel do bloco com base em ser realizada pela unidade de decodificação com base em paleta 165, algumas ou todas as tais funções podem ser realizadas por outras unidades de processamento ou uma combinação de diferentes unidades de processamento.

[0219] A unidade de decodificação com base em paleta 165 pode receber as informações de modo de codificação

de paleta e realizar as operações acima quando as informações de modo de codificação de paleta indicarem que o modo de codificação de paleta se aplica ao bloco. Quando as informações de modo de codificação de paleta indicarem que o modo de codificação de paleta não se aplica ao bloco ou quando outras informações de modo indicarem o uso de um modo diferente, a unidade de decodificação com base em paleta 165 decodifica o bloco de dados de vídeo com uso de um modo de codificação não com base em modo de paleta, por exemplo, tal como um modo de codificação intrapreditivo ou interpreditivo de HEVC. O bloco de dados de vídeo pode ser, por exemplo, uma CU ou PU gerada de acordo com um processo de codificação de HEVC.

[0220] De acordo com os aspectos desta revelação, a unidade de decodificação com base em paleta 165 pode ser configurada para realizar qualquer combinação das técnicas para a codificação de paleta descrita no presente documento. Por exemplo, conforme descrito em relação às Figuras 4 a 7 abaixo, de acordo com os aspectos desta revelação, a unidade de decodificação com base em paleta 165 pode ser configurada para reinicializar uma paleta de preditor para construir uma paleta para um bloco de dados de vídeo com base em um ou mais blocos de uma outra linha. Por exemplo, conforme descrito no presente documento, a unidade de decodificação com base em paleta 165 pode determinar uma primeira paleta para um primeiro bloco de dados de vídeo que está localizado em uma primeira fileira de blocos. A unidade de decodificação com base em paleta 165 também pode gerar uma paleta de preditor ao decodificar um ou mais outros blocos na primeira fileira. Mediante a codificação de um

bloco em uma segunda fileira, a unidade de decodificação com base em paleta 165 pode reinicializar a paleta de preditor para determinar uma paleta do bloco na segunda fileira com base nas entradas da primeira paleta.

[0221] Em outro exemplo, conforme descrito em relação à Figura 8 abaixo, a unidade de decodificação com base em paleta 165 pode ser configurada para decodificar uma execução de índices que inclui amostras codificadas com o uso de mais de um modo de paleta. Por exemplo, a unidade de decodificação com base em paleta 165 pode determinar os valores de índice de uma primeira fileira de um bloco de dados de vídeo, em que os valores de índice incluem um ou mais índices que estão associados a um valor de cor na paleta e um elemento de sintaxe que não está associado a um valor de cor na paleta. A unidade de decodificação com base em paleta 165 também pode decodificar uma execução de valores de índice de uma segunda fileira do bloco de dados de vídeo em relação aos valores de índice da primeira fileira, em que a execução inclui os um ou mais índices que estão associados a um valor de cor na paleta e o elemento de sintaxe que não está associado a um valor de cor na paleta.

[0222] A Figura 4 é um diagrama conceitual que ilustra um exemplo da determinação de uma paleta. A Figura 4 inclui uma figuração 178 que tem uma primeira unidade de codificação (CU) 180 que é associada a primeiras paletas 184 e uma segunda CU 188 que é associada a segundas paletas 192. Conforme descrito em maiores detalhes abaixo e de acordo com as técnicas desta revelação, as segundas paletas 192 têm como base as primeiras paletas 184. A figuração 178 também inclui o bloco 196 codificado com um modo de codificação de

intrapredição e o bloco 200 que é codificado com um modo de codificação de interpredição.

[0223] As técnicas da Figura 4 são descritas no contexto de codificador de vídeo 20 (Figura 1 e Figura 2) e decodificador de vídeo 30 (Figura 1 e Figura 3) e em relação ao padrão HEVC de codificação de vídeo para propósitos explicativos. No entanto, deve ser entendido que as técnicas desta revelação não são limitadas dessa maneira e, podem ser aplicadas por outros processadores e/ou dispositivos de codificação de vídeo em outros padrões e/ou processos de codificação de vídeo.

[0224] Em geral, uma paleta se refere a diversos valores de pixels que são dominantes e/ou representativos para uma CU que é atualmente codificada, CU 188 no exemplo da Figura 4. As primeiras paletas 184 e segundas paletas 192 são mostradas conforme incluindo múltiplas paletas. Em alguns exemplos, de acordo com os aspectos desta revelação, um codificador de vídeo (tal como codificador de vídeo 20 ou decodificador de vídeo 30) pode codificar as paletas de modo separado para cada componente de cor de uma CU. Por exemplo, o codificador de vídeo 20 pode codificar uma paleta para um componente de luma (Y) de uma CU, outra paleta para um componente de croma (U) da CU e ainda outra paleta para o componente de croma (V) da CU. Nesse exemplo, as entradas da paleta Y podem representar valores Y de pixels da CU, as entradas da paleta U pode representar valores U de pixels da CU e as entradas da paleta V podem representar valores V de pixels da CU.

[0225] Em outros exemplos, o codificador de vídeo 20 pode codificar uma paleta única para todos os

componentes de cor de uma CU. Nesse exemplo, o codificador de vídeo 20 pode codificar uma paleta que tem uma i -ésima entrada que é um valor triplo, que inclui Y_i , U_i e V . Nesse caso, a paleta inclui valores para cada um dos componentes dos pixels. Consequentemente, a representação de paletas 184 e 192 como um conjunto de paletas que tem múltiplas paletas individuais é simplesmente um exemplo e não se destina a ser limitante.

[0226] No exemplo da Figura 4, as primeiras paletas 184 incluem três entradas 202-206 que têm valor de índice de entrada 1, valor de índice de entrada 2 e valor de índice de entrada 3, respectivamente. As entradas 202-206 referem os valores de índice aos valores de pixels que incluem valor de pixel A, valor de pixel B e valor de pixel C, respectivamente. Conforme descrito no presente documento, em vez da codificação de o pixel atual pode usar a codificação com base em paleta para codificar os pixels do bloco com o uso dos índices 1-3. Isto é, para cada posição de pixel da primeira CU 180, o codificador de vídeo 20 pode codificar um valor de índice para o pixel, em que o valor de índice é associado a um valor de pixel em uma ou mais das primeiras paletas 184. O decodificador de vídeo 30 pode obter os valores de índice a partir de um fluxo de bits e pode reconstruir os valores de pixels com uso dos valores de índice e uma ou mais das primeiras paletas 184. Dessa forma, as primeiras paletas 184 são transmitidas por codificador de vídeo 20 em um fluxo de bits de dados de vídeo codificados para uso pelo decodificador de vídeo 30 na decodificação com base em paleta.

[0227] Em alguns exemplos, o codificador de

vídeo 20 e o decodificador de vídeo 30 podem determinar segundas paletas 192 com base nas primeiras paletas 184. Por exemplo, o codificador de vídeo 20 e/ou o decodificador de vídeo 30 podem localizar um ou mais blocos a partir dos quais as paletas preditivas, nesse exemplo, as primeiras paletas 184, são determinadas. A combinação de entradas que são usadas para propósitos de predição pode ser referida como uma paleta de preditor.

[0228] No exemplo da Figura 4, as segundas paletas 192 incluem três entradas 208 a 212 que têm valor de índice de entrada 1, valor de índice de entrada 2 e valor de índice de entrada 3, respectivamente. As entradas 208-212 referem os valores de índice aos valores de pixels que incluem valor de pixel A, valor de pixel B e valor de pixel D, respectivamente. Nesse exemplo, o codificador de vídeo 20 pode codificar um ou mais elementos de sintaxe que indicam quais entradas das primeiras paletas 184 (que representam uma paleta preditiva, embora a paleta preditiva possa incluir entradas de diversos blocos) estão incluídas nas segundas paletas 192.

[0229] No exemplo da Figura 4, os um ou mais elementos de sintaxe são ilustrados como um vetor 216. O vetor 216 tem diversos bins associados (ou bits), em que cada bin indica se a paleta de preditor associado àquele bin é usado para prever uma entrada da paleta atual. Por exemplo, o vetor 216 indica que as primeiras duas entradas das primeiras paletas 184 (202 e 204) estão incluídas nas segundas paletas 192 (um valor de "1" no vetor 216), enquanto a terceira entrada das primeiras paletas 184 não está incluída nas segundas paletas 192 (um valor de "0" no vetor

216). No exemplo da Figura 4, o vetor é um vetor Boolean. O vetor pode ser referido como um vetor de predição de paleta.

[0230] Em alguns exemplos, conforme notado acima, o codificador de vídeo 20 e o decodificador de vídeo 30 podem determinar uma paleta de preditor (que também pode ser referida como uma tabela de paleta de preditor ou lista de paleta de preditor) ao realizar a predição de paleta. A paleta de preditor pode incluir entradas a partir das paletas de um ou mais blocos vizinhos que são usados para predizer uma ou mais entradas de uma paleta para a codificação de um bloco atual. O codificador de vídeo 20 e o decodificador de vídeo 30 podem construir a lista da mesma forma. O codificador de vídeo 20 e o decodificador de vídeo 30 podem codificar os dados (tal como vetor 216) para indicar quais entradas da paleta de preditor devem ser copiadas para uma paleta para codificar um bloco atual,

[0231] De acordo com os aspectos desta revelação, uma paleta de preditor (tal como a paleta de preditor associada ao vetor 216) pode ser reinicializada para a geração de uma paleta associada a certos blocos em uma figuração e/ou pequeno bloco de dados de vídeo. Por exemplo, conforme descrito em mais detalhes em relação aos exemplos das Figuras 5 a 7 abaixo, uma paleta de preditor para construir uma paleta para um bloco de dados de vídeo em uma primeira linha pode ser reinicializada com base em um ou mais blocos de uma outra linha. Em um exemplo, o codificador de vídeo 20 e o decodificador de vídeo 30 podem determinar uma primeira paleta para um primeiro bloco de dados de vídeo que está localizado em uma primeira fileira de blocos. O codificador de vídeo 20 e o decodificador de vídeo 30 também

podem gerar uma paleta de preditor ao codificar um ou mais outros blocos na primeira fileira. Mediante a codificação de um bloco em uma segunda fileira, o codificador de vídeo 20 e o decodificador de vídeo 30 podem reinicializar a paleta de preditor para determinar uma paleta do bloco na segunda fileira com base nas entradas da primeira paleta.

[0232] A Figura 5 é um diagrama conceitual que ilustra pequenos blocos exemplificativos de acordo com o padrão de Codificação de Vídeo de Alta Eficiência (HEVC). O HEVC contém diversas propostas para tornar o codec mais favorável a ser paralelo, que inclui pequenos blocos e processamento paralelo de frente de onda (WPP). HEVC WD10 define pequenos blocos como um número inteiro de CTBs que ocorrem simultaneamente em uma coluna e uma fileira, em ordem consecutivamente em uma varredura de rastreamento de CTB do pequeno bloco. A divisão de cada figuração em pequenos blocos pode ser chamada de partição. Pequenos blocos em uma figuração são ordenados consecutivamente na varredura de rastreamento de pequeno bloco da figuração, conforme mostrado na Figura 5.

[0233] Por exemplo, a Figura 5 ilustra uma ordem de codificação de CTB exemplificativa para uma figuração 220 que é particionada em múltiplos pequenos blocos 222A, 222B, 222C, 222D, 222E, 222F, 222G, 222H, e 222I (coletivamente, "pequenos blocos 222"), com limites de pequeno bloco indicados pelas linhas espessas. Cada bloco quadrado na figuração 220 representa um bloco de pixel associado a um CTB. Os números nos blocos de pixel indicam as posições dos CTBs correspondentes (por exemplo, LCUs) em uma ordem de codificação de pequeno bloco para a figuração 220.

[0234] Conforme ilustrado no exemplo da Figura 5, CTBs em pequeno bloco 222A são codificados primeiro, seguido por CTBs em pequeno bloco 222B, seguido por CTBs em pequeno bloco 222C, seguido por CTBs em pequeno bloco 222D, seguido por CTBs em pequeno bloco 222E, seguido por CTBs em pequeno bloco 222F, seguido por CTBs em pequeno bloco 222G, seguido por CTBs em pequeno bloco 222H, seguido por CTBs em pequeno bloco 222I. A ordem de processamento de pequenos blocos 222 é, em geral, ilustrada pelos números incluídos nos pequenos blocos 222. No interior de cada um dentre os pequenos blocos 222, os CTBs são codificados de acordo com a ordem de varredura de rastreamento.

[0235] O número de pequenos blocos e a localização de seus limites podem ser definidos para toda sequência ou alterados de figuração a figuração. Os limites de pequeno bloco, de modo semelhante aos limites de fatia, quebram as dependências de análise e de predição de modo que um pequeno bloco possa ser processado independentemente. No entanto, em alguns exemplos, filtros in-loop (por exemplo, filtros de desbloqueio e de deslocamento adaptável de amostra) podem ainda atravessar os limites de pequeno bloco.

[0236] O uso de pequenos blocos pode aprimorar o paralelismo, devido ao fato de que nenhuma (ou relativamente pouca) comunicação é necessária entre os processadores ou núcleos de processador para decodificação por entropia e reconstrução de compensação de movimento. Além disso, pequenos blocos podem exibir uma eficácia de codificação relativamente melhor quando comparados às fatias, devido ao fato dos pequenos blocos permitirem formatos de partição de figuração que contêm amostras com

correlação potencialmente mais alta que as fatias. Pequenos blocos também podem reduzir a suspensão de cabeçalho de fatia.

[0237] De acordo com os aspectos desta revelação, o codificador de vídeo 20 e o decodificador de vídeo 30 podem reinicializar uma paleta de preditor para determinar uma paleta de um CTB com base em uma paleta de uma outra CTB. Em um exemplo para propósitos ilustrativos, o codificador de vídeo 20 e o decodificador de vídeo 30 podem ser configurados para reinicializar uma paleta de preditor para determinar uma paleta atual de um CTB atual no começo de cada linha de CTB. O codificador de vídeo 20 e o decodificador de vídeo 30 podem reinicializar a paleta de preditor com o uso de entradas de paleta de um primeiro CTB da linha de CTB acima do CTB atual,

[0238] Em um exemplo para propósitos ilustrativos, após a codificação d CTB (4) (de pequeno bloco 222A) com o uso de codificação com base em paleta, o codificador de vídeo 20 e o decodificador de vídeo 30 podem ser configurados para gerar uma paleta de preditor para determinar uma paleta de CTB (5) (de pequeno bloco 222A). Além disso, o codificador de vídeo 20 e o decodificador de vídeo 30 podem ser configurados para atualizar a paleta de preditor após a codificação d cada um dentre CTB (5), CTB (6), e CTB (7) com o uso de uma codificação com base em modo de paleta. Isto é, o codificador de vídeo 20 e o decodificador de vídeo 30 podem substituir ou adicionar uma ou mais entradas da paleta de preditor após a codificação d cada um dentre CTB (5), CTB (6), e CTB (7).

[0239] De acordo com os aspectos desta

revelação, ao codificar o CTB (8) da linha de CTB abaixo da linha de CTB que inclui CTB (4) (assumindo a ordem de varredura de rastreamento), o codificador de vídeo 20 pode reinicializar a paleta de preditor antes de gerar uma paleta para a codificação de CTB (8). Por exemplo, em vez do uso da paleta de preditor conforme anteriormente atualizado com base em CTB (7), o codificador de vídeo 20 e o decodificador de vídeo 30 podem reinicializar uma paleta de preditor com base em entradas de uma paleta associada ao primeiro CTB da fileira de CTB vizinha acima, isto é, CTB (4). Dessa maneira, a paleta de preditor para a geração da paleta para o CTB (8) pode ser mais propensa a ter cores similares àsquelas usadas em CTB (8), devido ao fato de que CTB (4) está localizado espacialmente mais próximo do CTB (4) do que do CTB (7).

[0240] Conforme notado acima, o codificador de vídeo 20 e o decodificador de vídeo 30 podem reinicializar uma paleta de preditor com base em entradas de uma paleta do primeiro CTB da fileira de CTB vizinha acima, isto é, CTB (4). De acordo com alguns exemplos, o codificador de vídeo 20 e o decodificador de vídeo 30 podem ser configurados para reinicializar uma paleta de preditor para o CTB (8) (por exemplo, a paleta de preditor usada para gerar uma paleta para a codificação de amostras de CTB (8) pela substituição de entradas da paleta de preditor com entradas a partir da paleta usada para codificar o CTB (4)).

[0241] Em um outro exemplo, de acordo com os aspectos desta revelação, o codificador de vídeo 20 e o decodificador de vídeo 30 podem ser configurados para reinicializar a paleta de preditor com base em uma paleta de preditor associada a CTB (4) que inclui entradas a partir da

paleta de CTB (4). Por exemplo, conforme notado acima, o codificador de vídeo 20 e o decodificador de vídeo 30 podem gerar uma paleta de preditor (ou atualizar uma paleta de preditor anteriormente gerada) após a codificação do CTB (4). O codificador de vídeo 20 e o decodificador de vídeo 30 podem incluir entradas de CTB (4) na paleta de preditor gerada ou atualizada. Nesse caso, o codificador de vídeo 20 e o decodificador de vídeo 30 podem reinicializar a paleta de preditor com base em entradas da paleta de preditor após a codificação de CTB (4) e antes da codificação de o CTB (5). Dessa maneira, o codificador de vídeo 20 e o decodificador de vídeo 30 podem ser configurados para reinicializar a paleta de preditor de CTB (8) após concluir a codificação de CTB (4) e antes da codificação de CTB (5).

[0242] Embora o exemplo acima seja descrito em relação ao CTB (4) e CTB (5), o codificador de vídeo 20 e o decodificador de vídeo 30 podem ser configurados para aplicar técnicas generalizadas de uma maneira similar. Por exemplo, em relação ao exemplo de reinicializar a paleta de preditor para o CTB (8), o codificador de vídeo 20 e o decodificador de vídeo 30 podem ser configurados para gerar e/ou atualizar uma paleta de preditor após a codificação d N CTBs da linha de CTBs acima do CTB (8) (por exemplo, a linha vizinha acima de CTBs, que inclui o CTB (4)). O codificador de vídeo 20 e o decodificador de vídeo 30 podem ser configurados para reinicializar a paleta de preditor para o CTB (8) com base na paleta de preditor gerada (ou atualizada) e antes de começar a codificar o N+1^{enésimo} CTB na linha de CTB acima do CTB (4). Consequentemente, em alguns exemplos, a paleta de preditor usada para a reinicialização pode incluir entradas

que são diferentes (ou além das) das entradas da paleta para o CTB (4).

[0243] Em um exemplo, o processo de reinicialização acima para o CTB (4) pode ser realizado ao em vez do CTB (5). Por exemplo, a paleta de preditor gerada após CTB (5) pode ser usada, o processo de inicialização pode ser o mesmo CTB usado para o processo de inicialização de contexto de CABAC aplicado para frentes de onda, que harmonizariam potencialmente a reinicialização de paleta com o processo de inicialização de contexto.

[0244] Em alguns exemplos, o codificador de vídeo 20 e o decodificador de vídeo 30 podem ser configurados para não reinicializar uma paleta de preditor para um CTB inicial que está localizado na linha superior (assumindo uma varredura de rastreamento de topo para fundo) de uma figuração ou pequeno bloco. Por exemplo, o codificador de vídeo 20 e o decodificador de vídeo 30 podem ser configurados para não reinicializar uma paleta de preditor para o CTB (0) (de pequeno bloco 222A), devido ao fato de que não há CTBs localizados acima do CTB (0).

[0245] Embora os exemplos da Figura 5 sejam descritas em relação aos CTBs para propósitos ilustrativos, deve ser entendido que as técnicas podem ser estendidas para qualquer tipo de bloco de dados de vídeo.

[0246] A Figura 6 é um diagrama conceitual que ilustra frentes de onda para o processamento paralelo de frente de onda (WPP). HEVC define uma técnica WPP. Quando se permite o WPP, cada fileira de CTU de uma figuração é uma partição separada. Em comparação às fatias e pequenos blocos, no entanto, nenhuma dependência de codificação é quebrada

nos limites de fileira de CTU. Adicionalmente, as probabilidades de CABAC são propagadas a partir da segunda CTU da fileira anterior, para reduzir adicionalmente as perdas de codificação. Além disso, WPP não altera a ordem de varredura de rastreamento regular. Devido ao fato de que as dependências não são quebradas, a perda de distorção de taxa de um fluxo de bits de WPP é pequena, em comparação a um fluxo de bits não paralelo.

[0247] Quando se permite WPP, inúmeros processadores até o número de fileiras de CTU podem funcionar em paralelo ao processo da (ou linhas) fileira de CTU. As dependências de frente de onda, no entanto, não permitem que todas as fileiras de CTU comecem a decodificação no começo da figuração. Consequentemente, as fileiras de CTU não podem concluir a decodificação ao mesmo tempo no final da figuração. Isso introduz ineficiências de paralelização que se tornam mais evidentes quando um número mais alto de processadores é usado. A Figura 6 ilustre como o WPP processa as fileiras de CTBs em paralelo, em que cada fileira começa com a probabilidades de CABAC disponível após o processamento da segunda CTB da fileira acima.

[0248] De acordo com os aspectos desta revelação, conforme descrito em maiores detalhes abaixo, o codificador de vídeo 20 e o decodificador de vídeo 30 podem determinar a possibilidade de redefinir uma paleta de preditor com base nas frentes de onda ilustradas na Figura 6.

[0249] A Figura 7 é um diagrama conceitual que ilustra um exemplo de paleta de preditor redefinido na presença de múltiplos pequenos blocos e frentes de onda,

consistentes com as técnicas desta revelação. No exemplo da Figura 7, linhas contínuas relativamente mais finas podem indicar o pequeno bloco. Por exemplo, a fileira de CTB 232 indica uma fileira de pequeno bloco de CTBs que inclui os dois CTBs mais superiores de pequeno bloco 230.

[0250] Algumas técnicas desta revelação incluem redefinir uma paleta de preditor com base em inúmeras frentes de onda e/ou inúmeros pequenos blocos que são usados para codificar uma figuração, tal como figuração 234. Por exemplo, de acordo com os aspectos desta revelação, o codificador de vídeo 20 e o decodificador de vídeo 30 podem redefinir uma paleta de preditor para os primeiros CTBs no pequeno bloco (tal como o CTB à esquerda superior de pequeno bloco 230) quando múltiplas frentes de onda e múltiplos pequenos blocos são usados para codificar a figuração 234 (tais como os pequenos blocos e frentes de onda mostrados nas Figuras 5 e 6, respectivamente). No exemplo da Figura 7, os círculos contínuos incluídos dentro dos CTBs indicam um primeiro CTB de pequeno bloco dos pequenos blocos para os quais a redefinição de paleta é realizada. O codificador de vídeo 20 e o decodificador de vídeo 30 podem redefinir uma paleta ajustando-se o tamanho de paleta de preditor para zero e não prevendo-se qualquer entrada de paleta de uma paleta que é usada para codificar um bloco atual.

[0251] A Figura 8 é um diagrama conceitual que ilustra um exemplo de determinação de índices para uma paleta para um bloco de pixels, consistente com as técnicas desta revelação. Por exemplo, a Figura 5 inclui um mapa 240 de índices que relaciona as respectivas posições de pixels associadas aos valores de índice a uma entrada de paletas

244. Por exemplo, o índice 1 é associado ao Valor A, o índice 2 é associado ao Valor B e o índice 3 é associado ao Valor C. Além disso, quando amostras de fuga são indicadas com o uso de sinalização de fuga implícita, o codificador de vídeo 20 e o decodificador de vídeo 30 também podem adicionar um índice adicional às paletas 244, ilustrado na Figura 5 como índice 4, que pode indicar que as amostras de mapa 240 associadas ao índice 4 são amostras de fuga. Nesse caso, o codificador de vídeo 20 pode codificar (e o decodificador de vídeo 30 pode obter, a partir de um fluxo de bits codificado) uma indicação de um valor de pixel real (ou sua versão quantizada) para uma posição no mapa 240 se o valor de pixel não estiver incluído nas paletas 244.

[0252] Em alguns exemplos, o codificador de vídeo 20 e o decodificador de vídeo 30 podem ser configurados para codificar um mapa adicional que indica quais posições de pixel são associadas a valores de índice. Por exemplo, assume-se que a (i, j) entrada no mapa corresponde à (i, j) posição de uma CU. O codificador de vídeo 20 pode codificar um ou mais elementos de sintaxe para cada entrada do mapa (isto é, cada posição de pixel) que indica se a entrada tem um valor de índice associado. Por exemplo, o codificador de vídeo 20 pode codificar um sinalizador que tem um valor de um para indicar que o valor de pixel no local (i, j) na CU é um dentre os valores nas paletas 244.

[0253] O codificador de vídeo 20 também pode, em tal exemplo, codificar um índice de paleta (mostrado para permitir que o decodificador de vídeo reconstrua o valor de pixel. Em casos em que as paletas 244 incluem uma única entrada e um valor de pixel associado, o codificador de vídeo

20 pode omitir a sinalização do valor de índice. O codificador de vídeo 20 pode codificar o sinalizador para ter um valor de zero para indicar que o valor de pixel no local (i, j) na CU não é um dentre os valores nas paletas 244. Nesse exemplo, o codificador de vídeo 20 também pode codificar uma indicação do valor de pixel para uso pelo decodificador de vídeo 30 na reconstrução do valor de pixel. Em alguns casos, o valor de pixel pode ser codificado de uma maneira com perdas.

[0254] O valor de um pixel em uma posição de uma CU pode fornecer uma indicação de valores de um ou mais outros pixels em outras posições da CU. Por exemplo, pode haver uma probabilidade relativamente alta de que as posições de pixel vizinhas de uma CU terão o mesmo valor de pixel ou podem ser mapeadas para o mesmo valor de índice (no caso de codificação com perdas, em que um valor maior do que um valor de pixel pode ser mapeado para um único valor de índice). Consequentemente, o codificador de vídeo 20 pode codificar um ou mais elementos de sintaxe que indicam os inúmeros valores de índice consecutivos em uma determinada ordem de varredura que tem o mesmo índice. Conforme notado acima, os índices de codificação dessa maneira podem ser referidos como o modo de paleta de Valor.

[0255] Conforme notado acima, as execuções podem ser usadas em conjunto com um modo CopyFromTop ou modo Valor. Em um exemplo para propósitos ilustrativos, considere as fileiras 264 e 268 do mapa 240. Presumindo-se uma direção de varredura da esquerda para a direita horizontal, a fileira 264 inclui três valores de índice de "1", dois valores de índice de "2" e três valores de índice de "3". A fileira 268

inclui cinco valores de índice de "1", dois valores de índice de "3" e uma amostra que não está incluída nas paletas 244 (representada pelo índice 4, embora um sinalizador de fuga no nível de amostra possa ser usado para sinalização de fuga explícita), a qual pode ser denominada como uma amostra de fuga.

[0256] Nesse exemplo, o codificador de vídeo 20 pode usar o modo CopyFromTop para codificar dados para a fileira 268. Por exemplo, o codificador de vídeo 20 pode codificar um ou mais elementos de sintaxe que indicam que a primeira posição da fileira 268 (a posição mais à esquerda da fileira 268) é a mesma que a primeira posição da fileira 264. O codificador de vídeo 20 também pode codificar um ou mais elementos de sintaxe que indicam que a próxima execução de duas entradas consecutivas na direção de varredura na fileira 268 são as mesmas que na primeira posição da fileira 264.

[0257] Após codificar os um ou mais elementos de sintaxe que indicam a primeira posição da fileira 264 e a execução de duas entradas (observada acima), o codificador de vídeo 20 pode codificar as quartas e quintas posições na fileira 268 (da esquerda para a direita) com o uso do modo Valor. Por exemplo, o codificador de vídeo 20 pode codificar um ou mais elementos de sintaxe que indicam um valor de 1 para a quarta posição e um ou mais elementos de sintaxe que indicam uma execução 1 (por exemplo, modo Valor). Logo, o codificador de vídeo 20 codifica essas duas posições sem referência a outra linha.

[0258] De acordo com os aspectos desta revelação, o codificador de vídeo 20 e o decodificador de

vídeo 30 codificam pelo contexto um valor de execução para um índice codificado com o uso de modo Valor com base no valor do índice. Por exemplo, conforme descrito no presente documento, o codificador de vídeo 20 e o decodificador de vídeo 30 podem usar um modo adaptativo ao contexto de um processo de codificação de CABAC para codificar os dados que indicam um valor de execução. De acordo com os aspectos desta revelação, o codificador de vídeo 20 e o decodificador de vídeo 30 podem ser configurados para determinar um contexto para a codificação dos dados com base no índice da execução.

[0259] No exemplo da Figura 8, o codificador de vídeo 20 e o decodificador de vídeo 30 codificam um índice de 1 para a quarta posição de mapa notada acima 40. O codificador de vídeo 20 e o decodificador de vídeo 30 também podem binarizar um valor de execução de 1 para a quarta posição de mapa 240. Em um exemplo, o codificador de vídeo 20 e o decodificador de vídeo 30 podem gerar um sinalizador que indica que o valor de execução é maior do que zero, mas não maior do que um (que pode ser indicado com o uso de um sinalizador separado).

[0260] De acordo com os aspectos desta revelação, o codificador de vídeo 20 e o decodificador de vídeo 30 podem selecionar um contexto para a codificação do sinalizador maior que zero com base no valor de índice de 1. Por exemplo, de acordo com alguns aspectos, o codificador de vídeo 20 e o decodificador de vídeo 30 podem definir um conjunto de contextos para a codificação do valor de execução. O conjunto de contextos podem incluir um contexto, dois contextos, os contextos, quatro contextos, ou mais contextos.

[0261] Em um exemplo, o codificador de vídeo 20 e o decodificador de vídeo 30 podem ser configurados para selecionar o contexto de execução (0) com base no índice que é igual a zero. Dessa maneira, o codificador de vídeo 20 e o decodificador de vídeo 30 podem ser configurados para selecionar o contexto de execução (1) com base no índice que é qualquer valor diferente de zero. Com esse exemplo, o codificador de vídeo 20 e o decodificador de vídeo 30 selecionam um contexto de execução de (1) com base no índice que é 1 para a quarta posição de mapa 240.

[0262] Em um outro exemplo, o codificador de vídeo 20 e o decodificador de vídeo 30 podem ser configurados para selecionar o contexto de execução (0) com base no índice que é igual à zero. O codificador de vídeo 20 e o decodificador de vídeo 30 podem ser configurados para selecionar o contexto de execução (1) com base no índice que é um. O codificador de vídeo 20 e o decodificador de vídeo 30 podem ser configurados para selecionar o contexto de execução (2) com base no índice que é qualquer valor maior do que um. Com esse exemplo, o codificador de vídeo 20 e o decodificador de vídeo 30 selecionam um contexto de execução de (1) com base no índice que é 1 para a quarta posição de mapa 240.

[0263] Em um outro exemplo, o codificador de vídeo 20 e o decodificador de vídeo 30 podem ser configurados para selecionar o contexto de execução (0) com base no índice que é igual à zero. O codificador de vídeo 20 e o decodificador de vídeo 30 podem ser configurados para selecionar o contexto de execução (1) com base no índice que é um ou dois. O codificador de vídeo 20 e o decodificador de

vídeo 30 podem ser configurados para selecionar o contexto de execução (2) com base no índice que é maior do que dois. Com esse exemplo, o codificador de vídeo 20 e o decodificador de vídeo 30 selecionam um contexto de execução de (1) com base no índice que é 1 para a quarta posição de mapa 240.

[0264] De acordo com os aspectos desta revelação, o contexto pode ter como base um valor de índice atual ou um valor de índice ajustado. Por exemplo, de acordo com os aspectos desta revelação, o codificador de vídeo 20 e o decodificador de vídeo 30 podem ser configurados para determinar o contexto para a codificação de um valor de execução com base em um índice de paleta sinalizado em um fluxo de bits (que pode ser associado ao elemento de sintaxe `adjusted_palette_index`) maior do que o índice de paleta atualmente usado para codificar uma amostra em um bloco, em exemplos em que um índice ajustado de paleta é sinalizado no fluxo de bits.

[0265] Após a codificação da execução associada à quarta posição de mapa 240, o codificador de vídeo 20 pode, então, codificar a primeira posição que tem um valor de índice de 3 na fileira 268 com o uso de modo `CopyFromTop` em relação à fileira superior 264. Por exemplo, o codificador de vídeo 20 pode sinalizar um modo `CopyFromTop` e uma execução de 1. Consequentemente, o codificador de vídeo 20 pode selecionar entre os valores de índice ou pixel de codificação de uma linha em relação aos outros valores da linha, por exemplo, com o uso de uma execução, os valores de índice ou pixel de codificação de uma linha em relação aos valores de uma outra linha (ou coluna), ou uma combinação dos mesmos. O codificador de vídeo 20 pode, em alguns exemplos, realizar

uma otimização de taxa/distorção para fazer a seleção.

[0266] O codificador de vídeo 20 pode, então, decodificar a amostra de fuga para a amostra final da fileira 268 (da esquerda para a direita), a qual não está incluída nas primeiras paletas 244. Por exemplo, o codificador de vídeo 20 pode codificar a posição final da fileira 268 como uma amostra de fuga. Isto é, o codificador de vídeo 20 pode codificar uma indicação de que a posição final da fileira 268 é em uma amostra de fuga (por exemplo, índice 4), assim como uma indicação do valor de amostra. O decodificador de vídeo 30 pode obter a sintaxe descrita acima de um fluxo de bits codificado e reconstruir a fileira 268 com o uso de tal sintaxe.

[0267] Conforme notado acima, pode haver duas ou mais técnicas para codificar uma indicação de como a amostra codificada como uma amostra de fuga. Por exemplo, com a sinalização de fuga explícita, o codificador de vídeo 20 e o decodificador de vídeo 30 podem codificar um sinalizador de fuga explícito para cada posição de amostra de mapa 240. Se uma amostra particular (tal como a amostra de fileira final 268) indicar o valor de cor para a amostra particular. Se a amostra não for codificada como uma amostra de fuga, o codificador de vídeo 20 e o decodificador de vídeo 30 podem codificar os dados adicionais para indicar se o modo é CopyFromTop ou Valor, tal como um sinalizador SPoint.

[0268] Com sinalização de fuga implícita, o codificador de vídeo 20 e o decodificador de vídeo 30 podem adicionar um índice adicional a paletas 244 (índice de entrada 4). O codificador de vídeo 20 e o decodificador de vídeo 30 podem usar o índice adicional a paletas 244 para

indicar que uma amostra é codificada como uma amostra de fuga. O índice adicional, no entanto, não tem um valor de cor associado. Em vez disso, o codificador de vídeo 20 e o decodificador de vídeo 30 também codificam valores de cor para cada amostra que é associada ao índice adicional. Se a amostra não for codificada como uma amostra de fuga, o codificador de vídeo 20 e o decodificador de vídeo 30 podem codificar os dados para indicar se o modo é CopyFromTop ou Valor, tal como um sinalizador SPoint.

[0269] De acordo com os aspectos desta revelação, o codificador de vídeo 20 e o decodificador de vídeo 30 podem ser configurados para codificar uma execução de índices com o uso de modo CopyFromTop que inclui uma ou mais amostras codificadas como amostras de fuga. Em alguns exemplos, as amostras codificadas como amostras de fuga podem ser referidas como "índices" incluídos na execução, apesar das amostras codificadas visto que as amostras de fuga que não tem um índice que mapeia para um valor de cor de uma paleta (por exemplo, o índice adicional é adicionado à paleta para propósitos de indicação de amostras codificadas como amostras de fuga, mas não tem um valor de cor correspondente). Por exemplo, o codificador de vídeo 20 e o decodificador de vídeo 30 podem ser configurados para copiar os sinalizadores de fuga (na sinalização de fuga explícita) ou um índice de fuga (na sinalização de fuga implícita) dentro de uma execução de outros índices que têm valores de color associados a uma paleta que é copiada com modo de paleta CopyFromTop. Tanto os sinalizadores de fuga quanto o índice de fuga podem ser referidos como um elemento de sintaxe que não está associado a um valor de cor na paleta,

devido ao fato de que nem o sinalizador nem o índice de fuga estão incluídos em uma paleta.

[0270] No presente documento, de acordo com os aspectos desta revelação, o codificador de vídeo 20 e o decodificador de vídeo 30 podem ser configurado para inferir que uma amostra atual incluída em uma execução associada ao modo de paleta CopyFromTop (por exemplo, uma amostra que é codificada em relação a uma amostra de uma outra fileira) é codificada como uma amostra de fuga se a amostra vizinha acima da amostra atual for codificada como uma amostra de fuga. O codificador de vídeo 20 e o decodificador de vídeo 30 também podem codificar os valores de fuga relacionados para todas as amostras codificadas como amostra de fuga após a execução.

[0271] Em relação à Figura 8, a terceira amostra de fileira 270 é codificada como uma amostra de fuga e a terceira amostra de fileira 272 é codificada como uma amostra de fuga (assumindo um código da esquerda para a direita de uma primeira amostra de fileira 270 com o uso de modo CopyFromTop. Além disso, o codificador de vídeo 20 e o decodificador de vídeo 30 podem codificar um valor de execução que indica um comprimento de execução de 15 para o modo CopyFromTop. A execução inclui tanto a terceira amostra de fileira 270 quanto a terceira amostra de fileira 272. Consequentemente, o codificador de vídeo 20 e o decodificador de vídeo 30 podem codificar os dados que indicam um valor de fuga (por exemplo, um valor de cor) para a terceira amostra de fileira 270. O codificador de vídeo 20 e o decodificador de vídeo 30 também podem codificar os dados que indicam um valor de fuga (por exemplo, um valor de cor) para a terceira

amostra de fileira 272, que pode ser diferente do valor de fuga de fileira 270.

[0272] A Figura 9 é um fluxograma que ilustra um processo exemplificativo para determinar uma paleta de preditor, consistente com as técnicas desta revelação. O processo da Figura 9 é, em geral, descrito como sendo realizado por um codificador de vídeo, que pode incluir o codificador de vídeo 20, o decodificador de vídeo 30 ou uma variedade de outros processadores.

[0273] No exemplo da Figura 9, o codificador de vídeo pode determinar uma primeira paleta para um primeiro bloco de dados de vídeo que está localizado em uma primeira fileira de blocos (290). Embora descritas em relação às "fileiras" para propósitos ilustrativos, deve ser entendido que as técnicas também podem ser aplicadas às colunas de blocos em exemplos em que o codificador de vídeo varre os blocos na ordem de coluna. Em qualquer caso, a primeira paleta para o primeiro bloco pode incluir uma ou mais entradas de paleta em que cada uma tem um índice de paleta que está associado a um valor de cor para a codificação do bloco.

[0274] O codificador de vídeo também gera uma paleta de preditor de geração ou construção de pelo menos uma paleta de pelo menos um segundo bloco que está localizado na primeira fileira (292). Por exemplo, o codificador de vídeo pode gerar uma paleta de preditor com base nas entradas da primeira paleta após a codificação do primeiro bloco. O codificador de vídeo também pode atualizar a paleta de preditor para cada bloco subsequentemente codificado da primeira fileira que é codificada com o uso de codificação

com base em paleta.

[0275] De acordo com os aspectos desta revelação, o codificador de vídeo pode reinicializar uma paleta de preditor para determinar uma terceira paleta de um terceiro bloco que está localizado em uma segunda fileira de blocos (294). O terceiro bloco pode ser o primeiro bloco que é codificado na segunda fileira de blocos. O terceiro bloco pode ser posicionado imediatamente abaixo do primeiro bloco.

[0276] O codificador de vídeo pode reinicializar uma paleta de preditor com base nas entradas da paleta do primeiro bloco. De acordo com alguns exemplos, o codificador de vídeo pode ser configurado para reinicializar uma paleta de preditor pela substituição de entradas do indicador configurado para reinicializar uma paleta de preditor com o uso da paleta de preditor que é gerada após a codificação do primeiro bloco (por exemplo, antes da codificação de um segundo bloco na primeira fileira) ou uma paleta de preditor atualizada conforme atualizado após a codificação d um ou mais outros blocos da primeira fileira.

[0277] O codificador de vídeo pode determinar a terceira paleta com base na paleta de preditor reinicializada (296). Por exemplo, o codificador de vídeo pode determinar quais entradas da paleta de preditor copiam para a terceira paleta e copiam as entradas relevantes. Em alguns exemplos, conforme descrito no presente documento, o codificador de vídeo pode determinar as entradas da paleta de preditor para copiar com base em um vetor de predição.

[0278] O codificador de vídeo pode, então, codificar/decodificar o bloco com o uso da terceira paleta

(298). Por exemplo, em exemplos em que o codificador de vídeo opera como um codificador de vídeo (tal como, codificador de vídeo 20), o codificador de vídeo pode codificar os dados que representam os índices do terceiro bloco com o uso da paleta determinada em um fluxo de bits codificado. Em exemplos em que o codificador de vídeo opera como um decodificador de vídeo (tal como decodificador de vídeo 30), o decodificador de vídeo pode determinar os valores de amostra para os índices do terceiro bloco com o uso da terceira paleta.

[0279] A Figura 10 é um fluxograma que ilustra um processo exemplificativo para a codificação de um valor de execução de uma execução de índices de paleta, consistente com técnicas desta revelação. O processo da Figura 10 é, em geral, descrito como sendo realizado por um codificador de vídeo, que pode incluir o codificador de vídeo 20, o decodificador de vídeo 30 ou uma variedade de outros processadores.

[0280] No exemplo da Figura 10, o codificador de vídeo determina um valor de execução que indica um comprimento de execução de uma execução de um índice de paleta (310). Por exemplo, o codificador de vídeo pode codificar uma indicação de um índice de paleta, por exemplo, com o uso de um modo de paleta de Valor. O codificador de vídeo também pode determinar o número de amostras consecutivas em uma ordem de varredura que tem o mesmo valor do índice codificado. O número de amostras consecutivas pode ser o valor de execução.

[0281] O codificador de vídeo também determina o contexto para a codificação adaptativa ao contexto dos

dados determinados que indicam o valor de execução com base no índice de paleta (312). Por exemplo, o codificador de vídeo pode determinar um ou mais contextos para a codificação de CABAC do valor de execução determinado com base no índice de paleta. Em alguns exemplos, o codificador de vídeo pode selecionar a partir de uma pluralidade de contextos definido para a codificação do valor de execução.

[0282] O codificador de vídeo pode, então, codificar/decodificar os dados que indicam o valor de execução com o uso do contexto determinado (314). Por exemplo, em exemplos em que o codificador de vídeo os dados que indicam o valor de execução em um fluxo de bits codificado. Em exemplos em que o codificador de vídeo opera como um decodificador de vídeo (tal como, decodificador de vídeo 30), o decodificador de vídeo pode analisar (decodificar) os dados que indicam o valor de execução de um fluxo de bits codificado.

[0283] A Figura 11 é um fluxograma que ilustra um processo exemplificativo para a codificação de uma execução de índices codificados maiores do que um modo de codificação de paleta, consistente com as técnicas desta revelação. O processo da Figura 11 é, em geral, descrito como sendo realizado por um codificador de vídeo, que pode incluir o codificador de vídeo 20, o decodificador de vídeo 30 ou uma variedade de outros processadores.

[0284] O codificador de vídeo determina índices de uma primeira fileira de um bloco que inclui índices associados a uma cor de uma paleta e um ou mais elementos de sintaxe não associados a uma cor da paleta (320). Por exemplo, o codificador de vídeo pode determinar amostras

para as quais os valores de índice são codificados e amostras codificadas no modo de paleta de Fuga.

[0285] O codificador de vídeo também pode codificar/decodificar uma execução de índices de uma segunda fileira em relação aos valores de índice da primeira fileira (322). Por exemplo, em exemplos em que o codificador de vídeo opera como um codificador de vídeo (tal como, codificador de vídeo 20), o codificador de vídeo pode codificar uma indicação do modo CopyFromTop e uma indicação do comprimento de execução da execução. Em exemplos em que o codificador de vídeo opera como um decodificador de vídeo (tal como decodificador de vídeo 30), o decodificador de vídeo pode decodificar uma indicação do modo CopyFromTop e uma indicação do comprimento de execução da execução. Em qualquer caso, a execução inclui ambos os índices que têm cores associadas e os um ou mais elementos de sintaxe não associados a uma cor da paleta, por exemplo, amostras codificadas tanto no modo CopyFromTop quanto as amostras de fuga.

[0286] O codificador de vídeo pode, então, codificar/decodificar os dados que indicam os valores de color para os elementos de sintaxe não associados a uma cor da paleta (324). Por exemplo, em exemplos em que o codificador de vídeo opera como um codificador de vídeo (tal como, codificador de vídeo 20), o codificador de vídeo pode codificar os dados que indicam os valores de color das amostras representadas pelos elementos de sintaxe em um fluxo de bits codificado. Em exemplos em que o codificador de vídeo opera como um decodificador de vídeo (tal como, decodificador de vídeo 30), o decodificador de vídeo pode decodificar os dados que indicam os valores de color das amostras

representadas pelos elementos de sintaxe de um fluxo de bits.

[0287] Deve ser entendido que todas as técnicas descritas no presente documento podem ser usadas individualmente ou em combinação. Esta revelação inclui diversos métodos de sinalização que podem alterar dependendo de certos fatores tais como o tamanho do bloco, tamanho da paleta, fatia do codificador e decodificador *a-priori* ou podem ser sinalizados explicitamente no conjunto de parâmetros de vídeo (VPS), conjunto de parâmetros de sequência (SPS), conjunto de parâmetros de figuração (PPS), cabeçalho de fatia, em um nível de pequeno bloco ou em qualquer lugar.

[0288] 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). Além disso, em certos exemplos, os atos ou eventos podem ser realizados de modo concorrente, por exemplo, através de processamento de múltiplos encadeamentos, processamento interrompido ou em múltiplos processadores, em vez de sequencialmente. Adicionalmente, embora certos aspectos desta revelação sejam descritos como realizados por um único módulo ou unidade para propósitos de clareza, deve ser entendido que as técnicas desta revelação podem ser realizadas por uma combinação de unidades ou módulos associados a um codificador de vídeo.

[0289] Certos aspectos desta revelação foram

descritos em relação ao padrão de HEVC de desenvolvimento para propósitos de ilustração. No entanto, as técnicas descritas nesta revelação podem ser úteis para outros processos de codificação de vídeo, que incluem outro padrão ou processos de codificação de vídeo de proprietário ainda não desenvolvidos.

[0290] As técnicas descritas acima podem ser realizadas pelo codificador de vídeo 20 (Figuras 1 e 2) e/ou decodificador de vídeo 30 (Figuras 1 e 3), ambos os quais podem ser referidos de modo geral como um codificador de vídeo. De modo semelhante, a codificação de vídeo pode se referir a codificação de vídeo ou decodificação de vídeo, conforme aplicável.

[0291] Embora as combinações particulares de vários aspectos das técnicas sejam descritas acima, essas combinações são fornecidas meramente para ilustrar exemplos das técnicas descritas nesta revelação. Portanto, as técnicas desta revelação não devem ser limitadas a essas combinações exemplificativas e podem abranger qualquer combinação concebível dos vários aspectos das técnicas descritas nesta revelação.

[0292] 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 um meio legível por computador e executadas por uma unidade de processamento com base em hardware. Os meios legíveis por computador podem incluir meios de armazenamento legíveis por computador, que correspondem a um meio tangível como meios

de armazenamento de dados ou meios de comunicação que incluem qualquer meio que facilite a transferência de um programa de computador proveniente de um lugar para outro, por exemplo, de acordo com um protocolo de comunicação. Dessa maneira, o meio legível por computador que não é transitório ou (2) um meio de comunicação tal como uma onda portadora ou sinal. Os meios de armazenamento de dados podem ser quaisquer meios disponíveis que possam ser acessados 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 um meio legível por computador.

[0293] A título de exemplo, e não de limitação, tais meios 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 outro meio que possa ser usado 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 um meio 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, no entanto, que os meios de armazenamento legíveis por computador e os meios de armazenamento de dados não incluem conexões, ondas portadoras, sinais ou outros meios transitórios, mas são, em vez disso, direcionadas para meios não transitórios e tangíveis. O disco magnético e o disco óptico, conforme usados 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, em geral, reproduzem os dados de modo magnético, enquanto os discos ópticos reproduzem os dados de modo óptico com lasers. As combinações dos supracitadas também devem ser abrangidas pelo escopo de meios legíveis por computador.

[0294] As instruções podem ser executadas por um ou mais processadores, tais como um ou mais processadores de sinal digital (DSPs), microprocessadores de propósito geral, circuitos integrados de aplicação específica (ASICs), arranjos lógicos programáveis em campo (FPGAs) ou outro conjunto de circuitos lógico discreto ou integrado equivalente. Consequentemente, o termo "processador", conforme usado no presente documento, pode se referir a qualquer uma das estruturas supracitadas ou a 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 codificar e decodificar ou incorporados em um codec combinado. Também, as técnicas podem ser totalmente implantadas em um ou mais circuitos ou elementos lógicos.

[0295] As técnicas desta revelação podem ser

implantadas em uma ampla variedade de ICs (por exemplo, um conjunto de chips). Vários componentes, módulos ou unidades são descritos nesta revelação para enfatizar os aspectos funcionais de 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 interoperativas, incluindo um ou mais processadores conforme descrito acima, em conjunto com software e/ou firmware adequados.

[0296] Vários exemplos foram descritos. Essas e outras implantações estão dentro do escopo das reivindicações a seguir.

REIVINDICAÇÕES

1. Método para processar dados de vídeo, o método caracterizado pelo fato de que compreende:

determinar uma primeira paleta para um primeiro bloco de dados de vídeo que é o bloco inicial localizado em uma primeira fileira de blocos, em que a primeira paleta compreende uma ou mais entradas de paleta, cada uma incluindo um índice de paleta que está associado a um valor de cor para codificar o bloco de dados de vídeo;

gerar uma paleta de preditor para construir uma pluralidade de segundas paletas de uma pluralidade de segundos blocos de dados de vídeo na primeira fileira de blocos codificados após o primeiro bloco, em que a paleta de preditor inclui pelo menos uma entrada de paleta de um ou mais blocos da primeira fileira diferente do primeiro bloco, os um ou mais blocos sendo codificados entre o primeiro bloco e o segundo bloco para o qual a paleta de preditor é construída;

reinicializar, com base em um terceiro bloco de dados de vídeo que está localizado em uma segunda fileira de blocos, a paleta de preditor para determinar uma terceira paleta do terceiro bloco de dados de vídeo que está localizado na segunda fileira de blocos, em que reinicializar a paleta de preditor compreende remover todas as entradas da paleta de preditor e incluir pelo menos uma dentre as uma ou mais entradas de paleta da primeira paleta ou uma paleta de preditor inicial gerada após codificar o primeiro bloco, em que a primeira fileira de blocos está posicionada acima da segunda fileira de blocos em uma imagem e em que o primeiro bloco está posicionado acima do terceiro e um bloco para a

direita do terceiro bloco;

determinar a terceira paleta do terceiro bloco com base na paleta de preditor reinicializada; e

codificar o terceiro bloco com o uso da terceira paleta, em que a codificação compreende codificar ou decodificar.

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

antes de codificar o terceiro bloco, codificar o primeiro bloco e a pluralidade de segundos blocos; e

gerar a paleta de preditor inicial após codificar o primeiro bloco e um bloco inicial da pluralidade de segundos blocos que seguem imediatamente o primeiro bloco na ordem de codificação e antes de codificar qualquer outro bloco dentre a pluralidade de segundos blocos.

3. Método, de acordo com a reivindicação 1, caracterizado pelo fato de que reinicializar a paleta de preditor com base nas uma ou mais entradas de paleta da primeira paleta ou na paleta de preditor inicial compreende copiar a uma ou mais entradas da primeira paleta ou entradas da paleta de preditor inicial para a paleta de preditor.

4. Método, de acordo com a reivindicação 1, caracterizado pelo fato de que o primeiro bloco compreende um primeiro bloco de árvore codificado (CTB) e o terceiro bloco compreende um terceiro CTB.

5. Método, de acordo com a reivindicação 1, caracterizado pelo fato de que a primeira fileira de blocos compreende uma fileira superior de blocos de um pequeno bloco, e em que o método compreende adicionalmente:

determinar que uma paleta de preditor para

predizer a primeira paleta tem zero entradas; e

determinar a primeira paleta sem predizer as uma ou mais entradas da primeira paleta.

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

após codificar o terceiro bloco, determinar diversas frentes de onda para codificar uma pluralidade de terceiras fileiras de blocos, em que a pluralidade de terceiras fileiras de blocos é dividida em uma pluralidade de pequenos blocos;

com base no número de frentes de onda que é maior do que um, redefinir uma paleta de preditor associada a cada bloco inicial de cada um dentre a pluralidade de fileiras para zero.

7. Método, de acordo com a reivindicação 6, caracterizado pelo fato de que compreende adicionalmente:

com base no número de frentes de onda sendo um ou zero, não redefinir a paleta de preditor associada a cada bloco inicial de cada uma dentre a pluralidade de fileiras para zero.

8. Método, de acordo com a reivindicação 1, caracterizado pelo fato de que codificar compreende codificar e em que codificar o terceiro bloco com o uso da terceira paleta compreende:

determinar respectivos valores de índice para uma ou mais amostras do terceiro bloco, em que os respectivos valores de índice identificam uma entrada da terceira paleta; e

codificar os valores de índice em um fluxo de bits codificado.

9. Método, de acordo com a reivindicação 1, caracterizado pelo fato de que codificar compreende decodificar e em que decodificar o terceiro bloco com o uso da terceira paleta compreende:

obter, a partir de um fluxo de bits codificado, respectivos valores de índice para uma ou mais amostras do terceiro bloco, em que os respectivos valores de índice identificam uma entrada da terceira paleta; e

determinar valores para as uma ou mais amostras correspondendo-se um ou mais dentre os respectivos valores de índice a pelo menos uma entrada da paleta.

10. Aparelho para processar dados de vídeo, caracterizado pelo fato de que o aparelho compreende:

meios para determinar uma primeira paleta para um primeiro bloco de dados de vídeo que é o bloco inicial localizado em uma primeira fileira de blocos, em que a primeira paleta compreende uma ou mais entradas de paleta, cada uma incluindo um índice de paleta que está associado a um valor de cor para codificar o bloco de dados de vídeo;

meios para gerar uma paleta de preditor para construir uma pluralidade de segundas paletas de uma pluralidade de segundos blocos de dados de vídeo na primeira fileira de blocos codificados após o primeiro bloco, em que a paleta de preditor inclui pelo menos uma entrada de paleta de um ou mais blocos da primeira fileira diferente do primeiro bloco, os um ou mais blocos sendo codificados entre o primeiro bloco e o segundo bloco para o qual a paleta de preditor é construída;

meios para reinicializar, com base em um terceiro bloco de dados de vídeo que está localizado em uma segunda

fileira de blocos, a paleta de preditor para determinar uma terceira paleta do terceiro bloco de dados de vídeo que está localizado na segunda fileira de blocos, em que reinicializar a paleta de preditor compreende remover todas as entradas da paleta de preditor e incluir pelo menos uma dentre as uma ou mais entradas de paleta da primeira paleta ou uma paleta de preditor inicial gerada após codificar o primeiro bloco, sendo que a primeira fileira de blocos está posicionada acima da segunda fileira de blocos em uma imagem e em que o primeiro bloco está posicionado acima do terceiro bloco e um bloco para a direita do terceiro bloco;

meios para determinar a terceira paleta do terceiro bloco com base na paleta de preditor reinicializada; e

meios para codificar o terceiro bloco usando a terceira paleta.

11. Aparelho, de acordo com a reivindicação 10, caracterizado pelo fato de que compreende adicionalmente:

meios para codificar antes de codificar o terceiro bloco, o primeiro bloco e a pluralidade de segundos blocos;

meios para gerar a paleta de preditor inicial após codificar o primeiro bloco e um bloco da pluralidade de segundos blocos que seguem imediatamente o primeiro bloco na ordem de codificação e antes de codificar qualquer outro bloco dentre a pluralidade de segundos blocos.

12. Aparelho, de acordo com a reivindicação 10, caracterizado pelo fato de que os meios para codificar compreendem um codificador.

13. Aparelho, de acordo com a reivindicação 10, caracterizado pelo fato de que os meios para codificar

compreendem um decodificador.

14. Memória legível por computador caracterizada pelo fato de que possui instruções nela armazenadas que, quando executadas, fazem com que um computador realize o método conforme definido em qualquer uma das reivindicações 1 a 9.

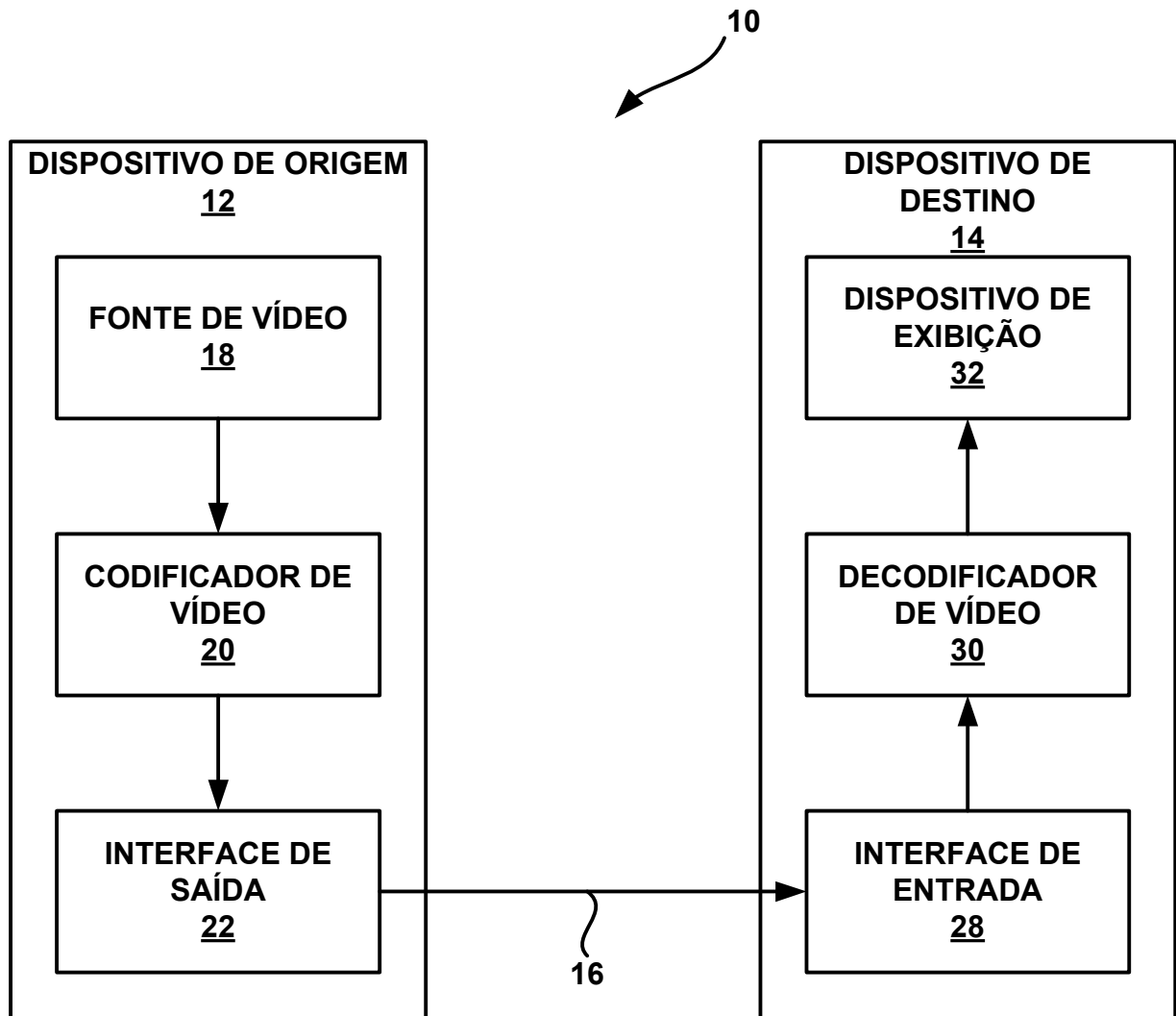
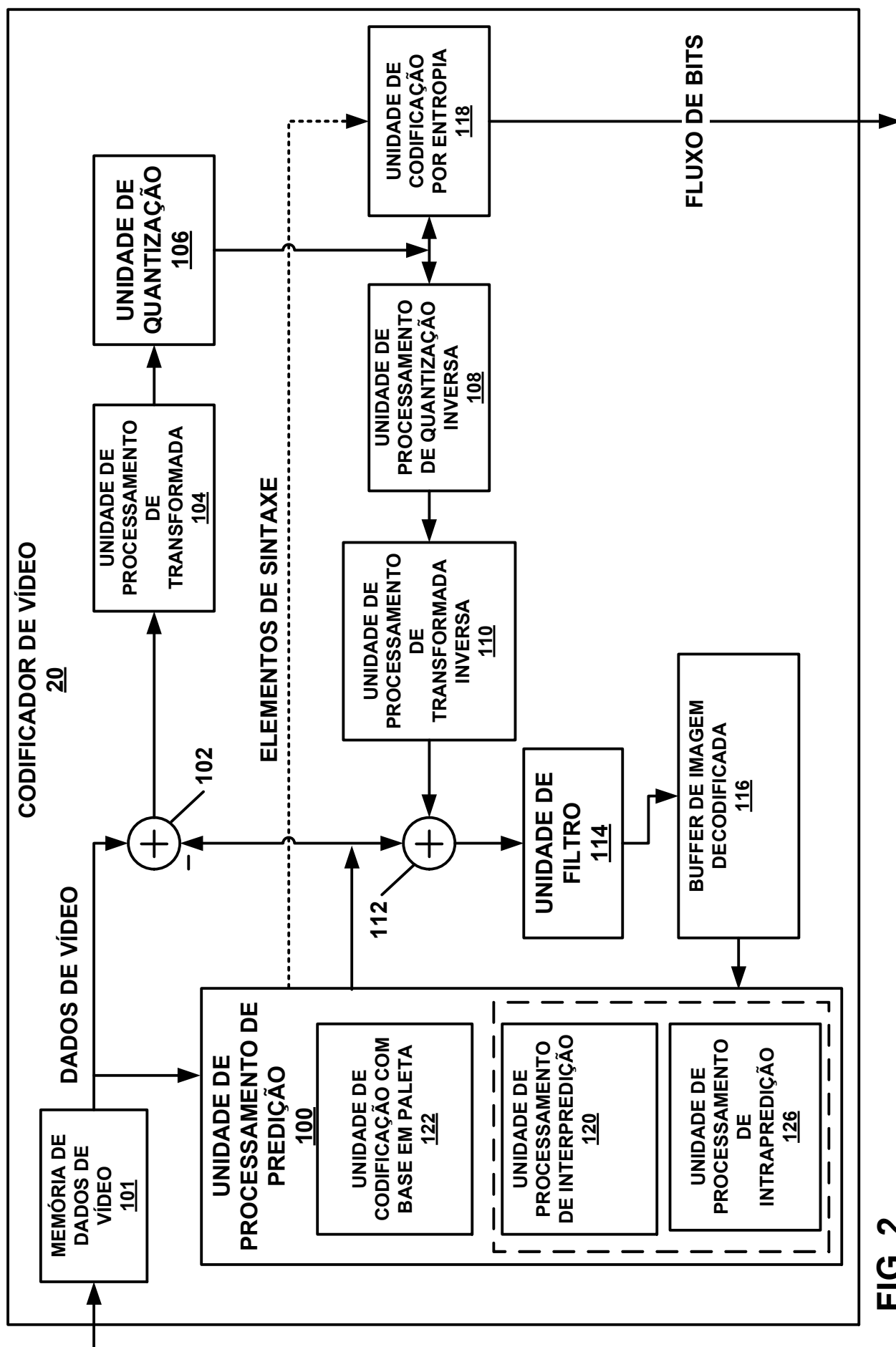


FIG. 1



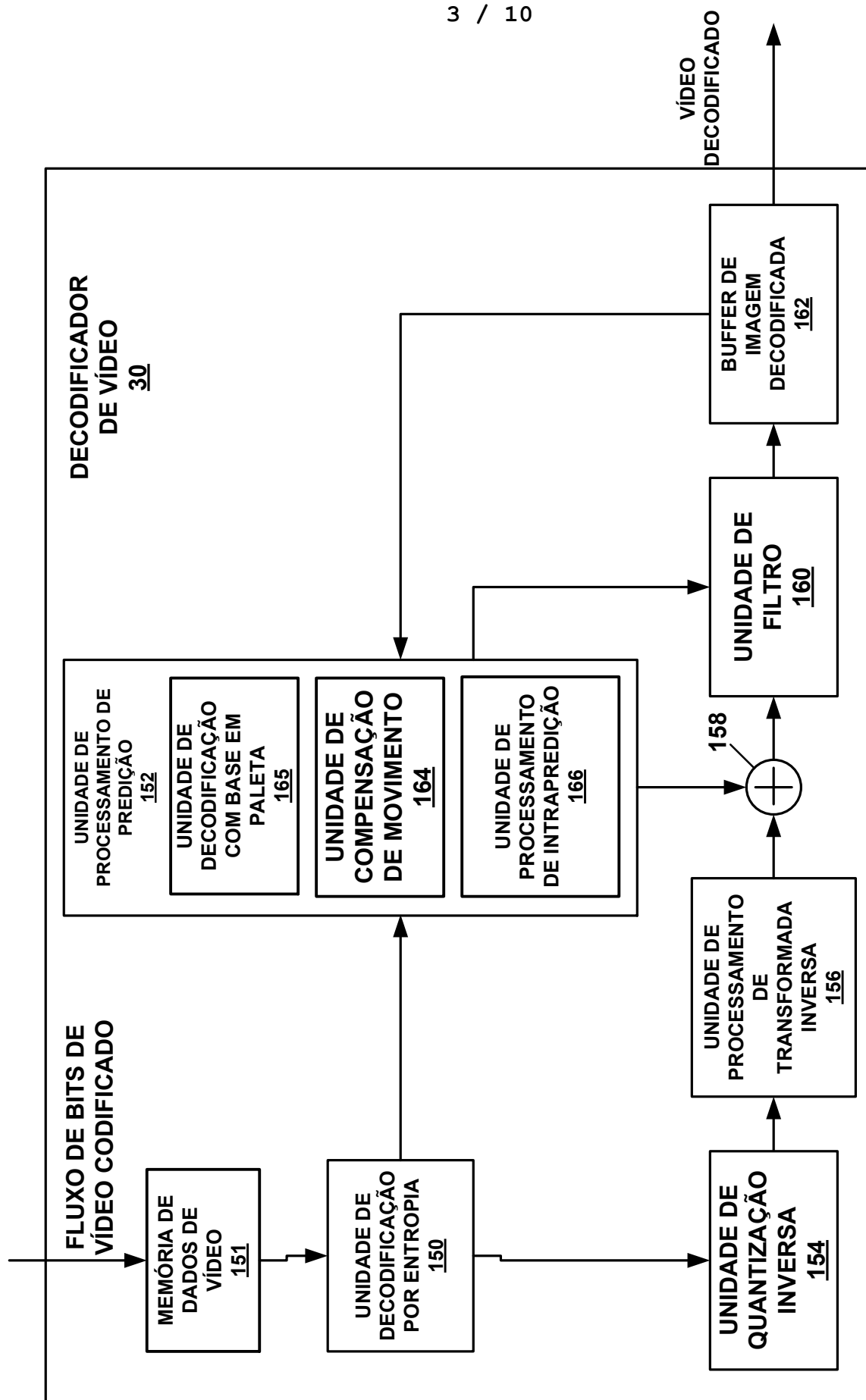


FIG. 3

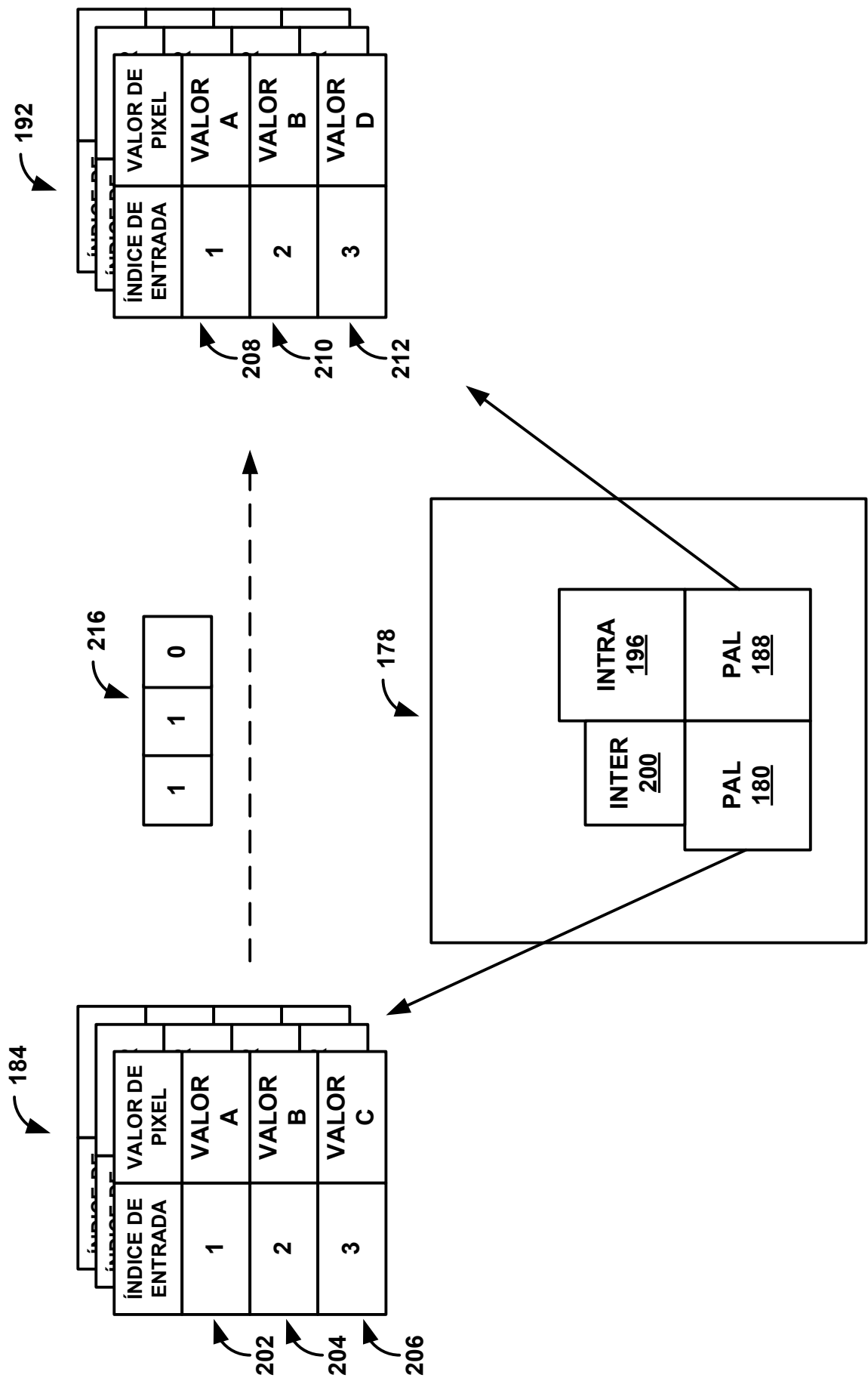


FIG. 4

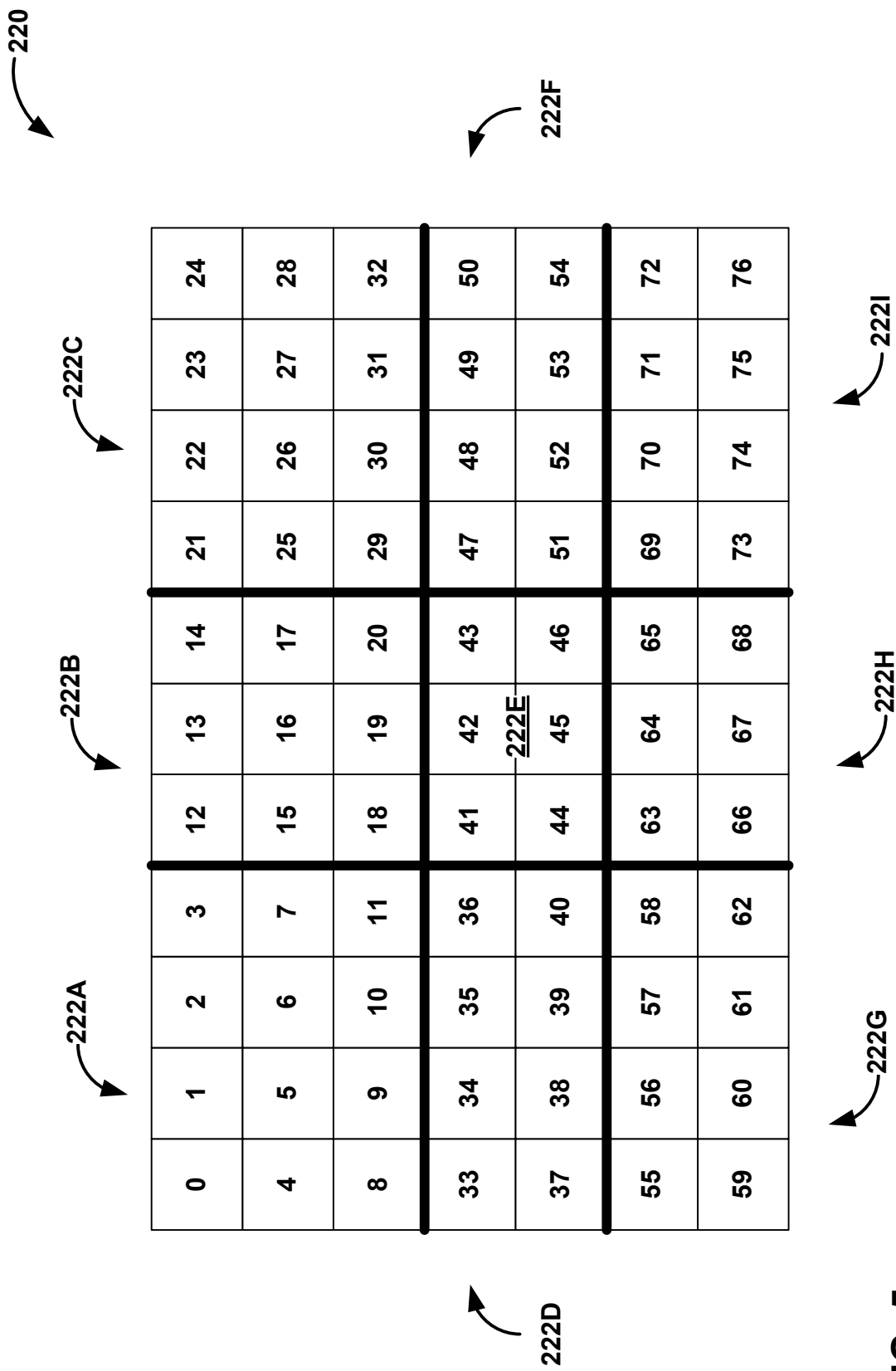
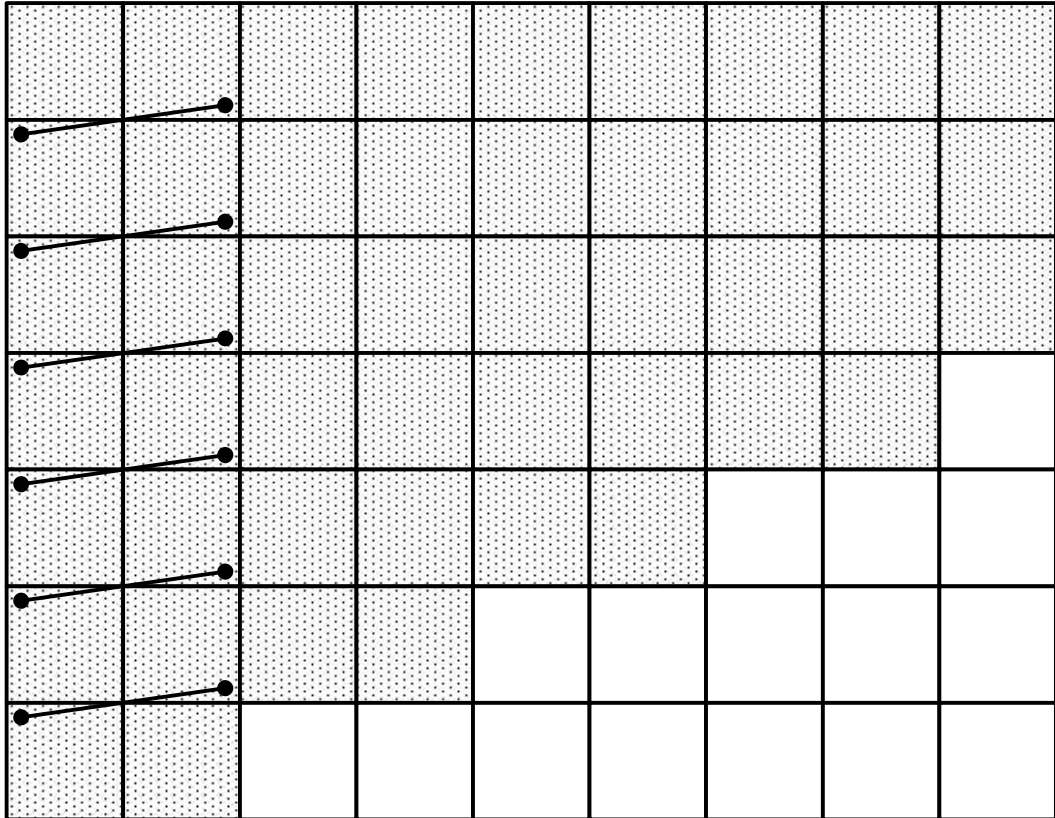


FIG. 5

**FIG. 6**

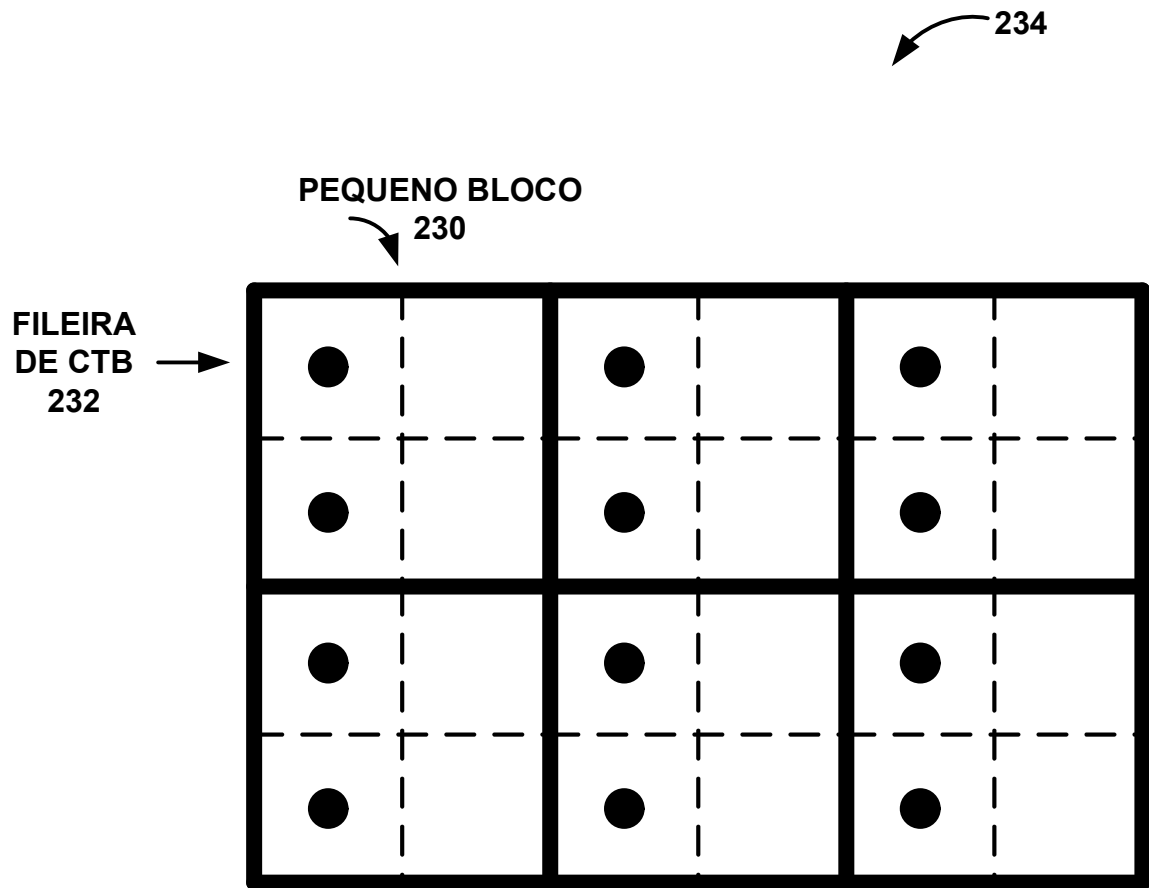


FIG. 7

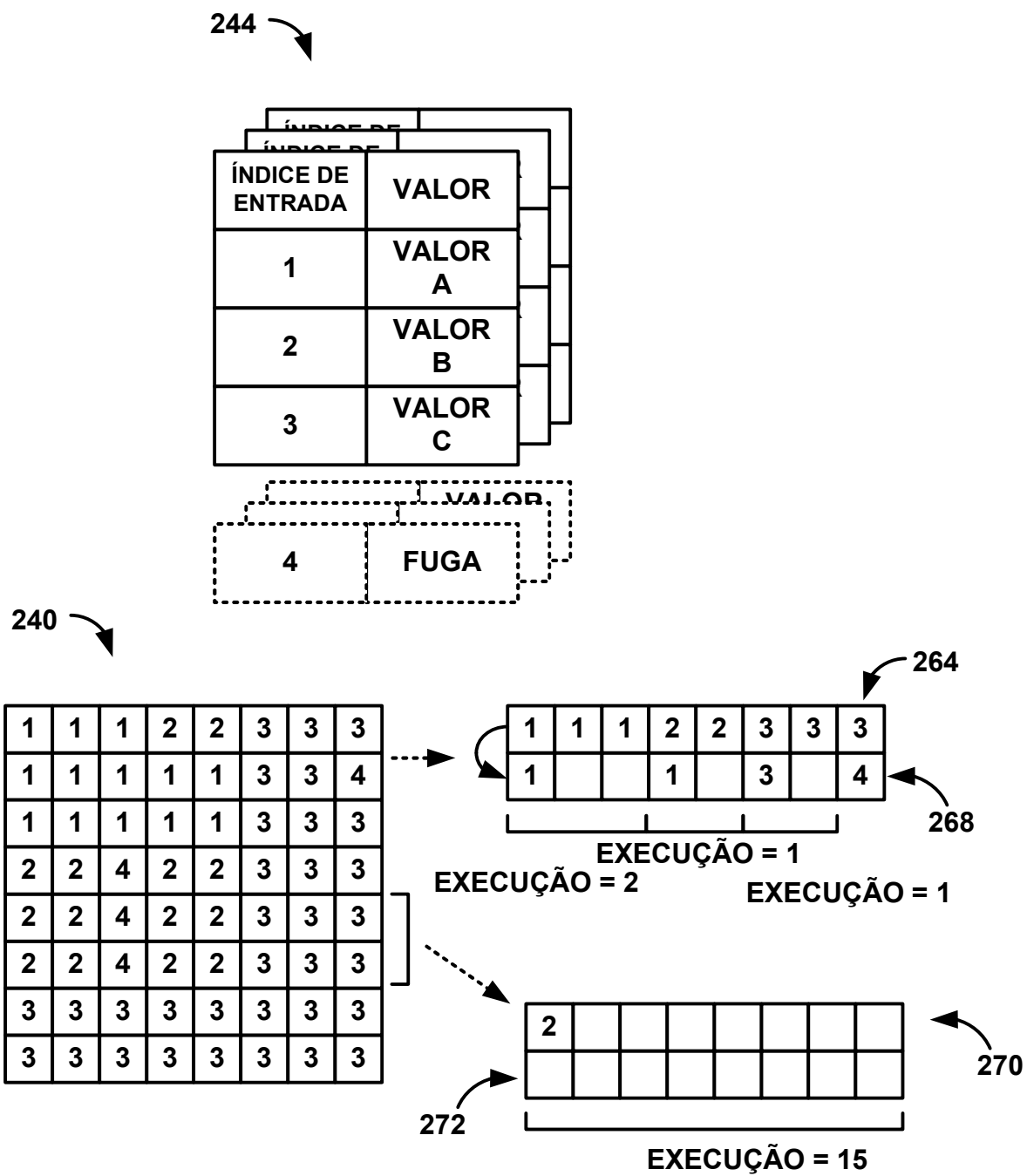
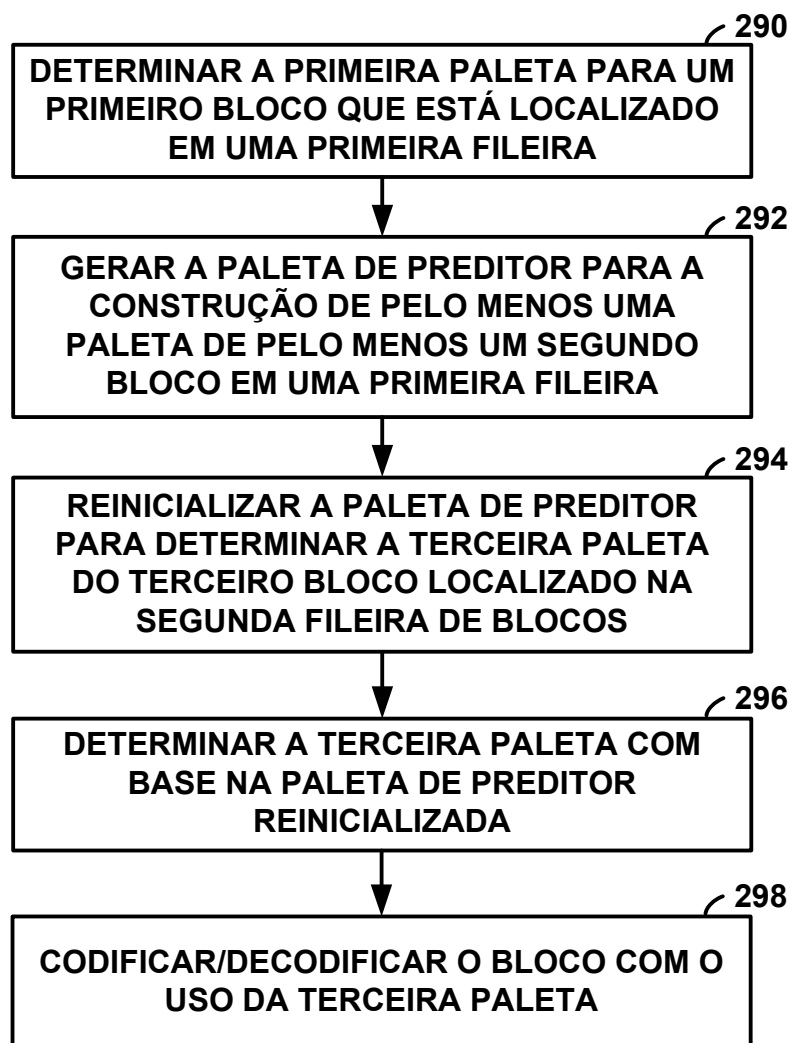


FIG. 8

**FIG. 9**

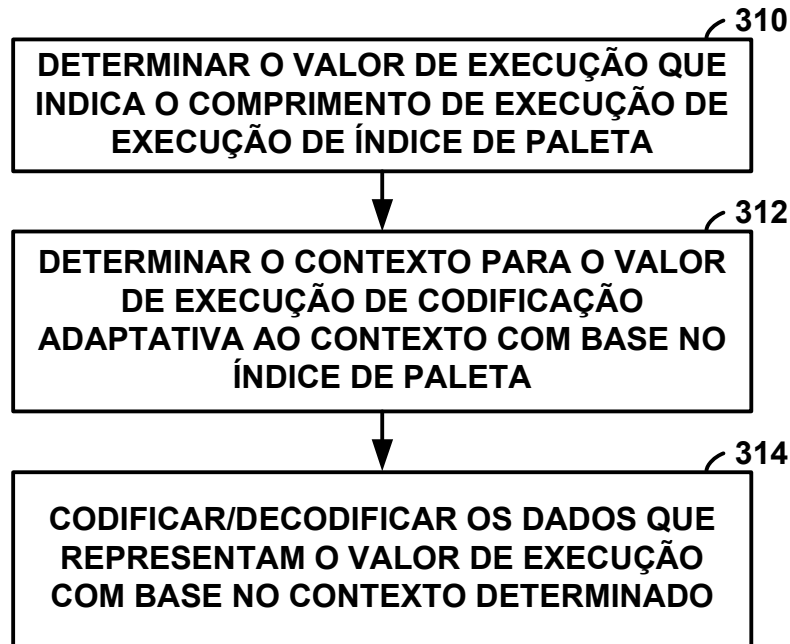


FIG. 10

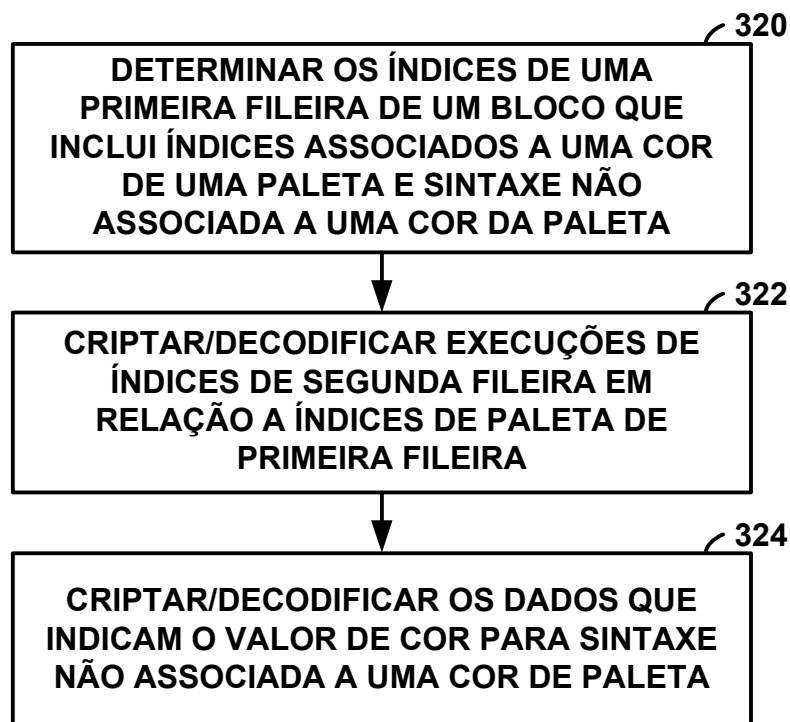


FIG. 11