



República Federativa do Brasil

Ministério do Desenvolvimento, Indústria,
Comércio e Serviços

Instituto Nacional da Propriedade Industrial

(11) BR 112017002185-4 B1

(22) Data do Depósito: 07/08/2015

(45) Data de Concessão: 14/11/2023

(54) Título: MÉTODO PARA A COMPACTAÇÃO DE VÍDEO DE ENLACE DE EXIBIÇÃO, DISPOSITIVO DE CODIFICAÇÃO DE VÍDEO E MEMÓRIA LEGÍVEL POR COMPUTADOR

(51) Int.Cl.: H04N 19/152; H04N 19/174; H04N 19/124.

(30) Prioridade Unionista: 06/08/2015 US 14/820,404; 08/08/2014 US 62/035,349.

(73) Titular(es): QUALCOMM INCORPORATED.

(72) Inventor(es): VIJAYARAGHAVAN THIRUMALAI; NATAN HAIM JACOBSON; RAJAN LAXMAN JOSHI.

(86) Pedido PCT: PCT US2015044252 de 07/08/2015

(87) Publicação PCT: WO 2016/022938 de 11/02/2016

(85) Data do Início da Fase Nacional: 02/02/2017

(57) Resumo: Trata-se de um sistema e método para compactação de fluxo de exibição revelados. Em um exemplo, é fornecido um método que inclui a codificação de um bloco dentro de uma fatia de dados de vídeo. A fatia pode compreender um ou mais blocos dos dados de vídeo. O método inclui adicionalmente armazenar bits correspondentes ao bloco em um armazenamento temporário, determinar informações indicativas de uma posição do bloco dentro da fatia, e determinar a plenitude do armazenamento temporário com base, pelo menos em parte, na posição do bloco dentro da fatia.

**"MÉTODO PARA A COMPACTAÇÃO DE VÍDEO DE ENLACE DE EXIBIÇÃO,
DISPOSITIVO DE CODIFICAÇÃO DE VÍDEO E MEMÓRIA LEGÍVEL POR
COMPUTADOR"**

CAMPO DA TÉCNICA

[0001] Esta revelação refere-se ao campo de compactação e codificação de vídeo, e particularmente uma compactação de vídeo para transmissão através de enlaces de exibição, como a compactação de vídeo de enlace de exibição.

ANTECEDENTES

[0002] As capacidades de vídeo digital podem ser incorporadas em uma ampla faixa de visores, incluindo televisões digitais, assistentes pessoais digitais (PDAs), computadores do tipo laptop, monitores do tipo desktop (de mesa), câmeras digitais, dispositivos de gravação digitais, reprodutores de mídias digitais, dispositivos de vídeo game, consoles de vídeo game, telefones celulares ou de radiossatélite, dispositivos de teleconferência por vídeo e similares. OS enlaces de exibição são usados para conectar visores a dispositivos-fonte adequados. Os requisitos de largura de banda dos enlaces de exibição são proporcionais à resolução dos visores, e, portanto, visores de alta resolução exigem enlaces de exibição banda larga grande. Alguns enlaces de exibição não têm a largura de banda para suportar visores de alta resolução. A compactação de vídeo pode ser usada para reduzir os requisitos de largura de banda de modo que os enlaces de exibição largura de banda inferior podem ser usados para fornecer vídeo digital a visores de alta resolução.

[0003] Outros tentaram utilizar compactação de imagem nos dados de pixel. Entretanto, tais esquemas não são algumas vezes visualmente sem perda ou podem ser difíceis e dispendiosos para implantar em dispositivos de exibição convencionais.

[0004] A Associação de Padrões Eletrônicos de Vídeo (VESA) desenvolveu Compactação de Fluxo de Exibição (DSC) como um padrão para compactação de vídeo de enlace de exibição. O conjunto de procedimentos de compactação de vídeo de enlace de exibição como DSC, deve fornecer, dentre outras coisas, qualidade de figuração que é visualmente sem perda (isto é, figurações que têm um nível de qualidade de modo que os usuários não possam perceber que a compactação está ativa). O conjunto de procedimentos de compactação de vídeo de enlace de exibição também deve fornecer um esquema que é fácil e não dispendioso para implantar em tempo real com hardware convencional. SUMÁRIO

[0005] Os sistemas, métodos e dispositivos desta revelação têm, cada um, diversos aspectos inovadores, sendo que nenhum dos quais é unicamente responsável pelos atributos desejáveis revelados no presente documento.

[0006] Em um aspecto, é fornecido um método de compactação de vídeo de enlace de exibição que compreende a codificação de um bloco dentro de uma fatia de dados de vídeo, sendo que a fatia compreende um ou mais blocos dos dados de vídeo, armazenar bits correspondentes ao bloco em um armazenamento temporário, determinar informações indicativas de uma posição do bloco dentro da fatia e determinar a plenitude do armazenamento temporário com base, pelo menos em parte, na posição do bloco dentro da fatia.

[0007] Em outro aspecto, um aparelho para compactação de vídeo de enlace de exibição compreende um processador configurado para codificar um bloco de uma fatia de dados de vídeo, sendo que a fatia compreende um ou mais blocos dos dados de vídeo, um armazenamento temporário configurado para armazenar bits correspondentes ao bloco, e um processador acoplado de modo operacional ao dispositivo de codificação de vídeo e ao armazenamento temporário. O

processador é configurado adicionalmente para determinar as informações indicativas de uma posição do bloco dentro da fatia e determinar a plenitude do armazenamento temporário com base, pelo menos em parte, na posição do bloco dentro da fatia.

[0008] Em outro aspecto, uma mídia de armazenamento legível por computador não transitória contém instruções que, quando executadas, levam um processador de um dispositivo a codificar um bloco dentro de uma fatia de dados de vídeo, sendo que a fatia compreende um ou mais blocos de dados de vídeo, armazenar bits correspondentes ao bloco em um armazenamento temporário, determinar informações indicativas de uma posição do bloco dentro da fatia e determinar a plenitude do armazenamento temporário com base, pelo menos em parte, na posição do bloco dentro da fatia.

[0009] Em outro aspecto, é fornecido um dispositivo de codificação de vídeo que inclui meios para codificar um bloco dentro de uma fatia de dados de vídeo, sendo que a fatia compreende um ou mais blocos dos dados de vídeo, meios para armazenar bits correspondentes ao bloco em um armazenamento temporário, meios para determinar informações indicativas de uma posição do bloco dentro da fatia, e meios para determinar a plenitude do armazenamento temporário com base, pelo menos em parte, na posição do bloco dentro da fatia. BREVE DESCRIÇÃO DOS DESENHOS

[0010] A Figura 1A é um diagrama de blocos que ilustra um sistema de decodificação e criptação de vídeo exemplificativo que pode utilizar conjuntos de procedimentos de acordo com os aspectos descritos nesta revelação.

[0011] A Figura 1B é um diagrama de blocos que ilustra outro sistema decodificação e criptação de vídeo exemplificativo que pode realizar conjuntos de procedimentos de acordo com os aspectos descritos nesta revelação.

[0012] A Figura 2A é um diagrama de blocos que ilustra um exemplo de um encriptador de vídeo que pode implantar conjuntos de procedimentos de acordo com os aspectos descritos nesta revelação.

[0013] A Figura 2B é um diagrama de blocos que ilustra um exemplo de um decodificador de vídeo que pode implantar conjuntos de procedimentos de acordo com os aspectos descritos nesta revelação.

[0014] A Figura 3 é um gráfico que mostra um processo exemplificativo para criptar fatias de dados.

[0015] A Figura 4 ilustra um conjunto de procedimentos exemplificativo para ajustar a plenitude de armazenamento temporário (BF) durante a criptação de uma fatia.

[0016] A Figura 5 é um fluxograma que ilustra um método exemplificativo para fornecer saída de taxa de bits constante (CB_R) em um armazenamento temporário.

DESCRIÇÃO DETALHADA

[0017] Em geral, esta revelação se refere a métodos para o aprimoramento de conjuntos de procedimentos de compactação de vídeo como aqueles utilizados em compactação de vídeo de enlace de exibição. Mais especificamente, a presente revelação se refere a sistemas e métodos para fornecer saída e/ou entrada de taxa de bits constante (CB_R) de um armazenamento temporário (por exemplo, um armazenamento temporário de taxas) para compactação de vídeo de enlace de exibição.

[0018] Na tecnologia de exibição convencional, compactação de 3: 1 conforme fornecido pela solução de Compactação de Fluxo de Exibição (DSC) v1.0 pela Associação de Padrões Eletrônicos de Vídeo (VESA), que é um exemplo de compactação de vídeo de enlace de exibição, pode ser insuficiente para movimentar os requisitos de mercado móveis

futuros, especificamente para requisitos associados a visores de alta resolução como resolução 4K (também chamada de 4K). Portanto, para satisfazer demandas futuras, é desejável desenvolver uma metodologia para incorporação em, por exemplo, um padrão de DSC de próxima geração que fornece compactação de 4: 1 e acima. Nesse contexto, é desejável gerenciar o controle de taxa na determinação do desempenho da metodologia de DSC. Um propósito de controle de taxa é determinar um conjunto de parâmetros de codificação, como um parâmetro de quantização (QP), um modo de codificação etc., que satisfaz restrições instantâneas e médias sobre a taxa enquanto maximiza o desempenho de taxa-distorção.

[0019] Os bits gastos na criptação (isto é, os bits utilizados para criptar) de cada bloco de dados de vídeo podem variar substancialmente com base nas propriedades (por exemplo, tamanho, número de bits, etc.) do bloco. Portanto, um armazenamento temporário pode ser usado como uma parte do mecanismo de controle de taxa a fim de suavizar as variações de taxa no fluxo de bits de saída. No modelo de armazenamento temporário CB_R , os bits podem ser removidos do armazenamento temporário em uma taxa constante durante a transmissão de dados através de um enlace físico com fio. Nesse modelo de armazenamento temporário, se o encriptador adiciona bits demais em relação aos bits removidos do armazenamento temporário, o número de bits no armazenamento temporário pode exceder a capacidade do armazenamento temporário, causando estouro. Por outro lado, o encriptador deveria adicionar bits em uma taxa suficiente a fim de impedir o estouro negativo. No lado de decodificador, os bits são adicionados ao armazenamento temporário em uma taxa constante e o decodificador remove um número de bits variável para cada bloco. Para assegurar a decodificação adequada, o armazenamento temporário não deveria estourar negativamente

ou estourar em qualquer ponto durante a decodificação do fluxo de bits compactado.

[0020] Deixar uma variável BufferCurrentSize representar o número de bits atualmente armazenado dentro de um armazenamento temporário e uma variável BufferMaxSize representar um tamanho (isto é, uma capacidade) do armazenamento temporário, isto é, o número de bits máximo geral que pode ser armazenado no armazenamento temporário. A "plenitude" do armazenamento temporário (também referido como plenitude de armazenamento temporário (BF)) pode ser calculada como mostrado na Equação 1 abaixo. BF representam uma porcentagem da capacidade de um armazenamento temporário ser usado para armazenamento de bits em um ponto particular no tempo.

$$BF = ((BufferCurrentSize * 100) / BufferMaxSize) \quad (\text{Equação 1})$$

[0021] ou, com o uso de:

$$BF = (((BufferCurrentSize * 100) + (BufferMaxSize \gg 1)) / BufferMaxSize) \quad (\text{Equação 2})$$

[0022] Entretanto, em determinadas modalidades, é desejável para o número de bits contidos (isto é, armazenados) em um armazenamento temporário no fim de uma fatia de dados de vídeo, isto é, após a codificação de todos os blocos na fatia ser menor do que BufferMaxSize. Deixar uma variável maxBufferBitsAtSliceEnd representar um número máximo desejado de bits no armazenamento temporário no fim da fatia. Em um exemplo, para evitar o estouro negativo e/ou estouro, pode ser ideal para o número de bits em um armazenamento temporário no fim de uma fatia igualar-se a maxBufferBitsAtSliceEnd, que pode ser menor do que BufferMaxSize. Nessas modalidades, maxBufferBitsAtSliceEnd seria usado no lugar de BufferMaxSize no cálculo de BF no

fim de uma fatia, e as equações 1 e/ou 2 descritas acima não seriam ideais.

[0023] Nessas modalidades, é desejado ajustar o BF dependendo (isto é, com base em) de uma posição de codificação de um dado bloco dentro de uma fatia. Mais precisamente, o BF é linearmente reduzido em uma taxa constante após a codificação de um número fixo (isto é, um número predeterminado ou configurável) de blocos em uma fatia, de tal forma que o fim da fatia, BF seja 100% se $\text{BufferCurrentSize} = \text{maxBufferBitsAtSliceEnd}$.

[0024] Embora determinadas modalidades sejam descritas no presente documento no contexto do padrão de DSC, uma pessoa da habilidade comum na técnica entenderia que os sistemas e os métodos revelados no presente documento podem ser aplicáveis a qualquer padrão de codificação de vídeo adequado. Por exemplo, as modalidades reveladas no presente documento podem ser aplicáveis a um ou mais dentre os seguintes padrões: O Setor de Padronização de Telecomunicações (ITU-T) da União Internacional de Telecomunicações (ITU) H.261, o Visual do Grupo de Especialistas em Imagem em Movimento - 1 (MPEG-1) da Organização Internacional de Normalização/Comissão Eletrotécnica Internacional (ISO/IEC), o Visual ITU-T H.262 ou ISO/IEC MPEG-2 , ITU-T H.263, o Visual ISO/IEC MPEG-4, ITU-T H.264 (também conhecido como ISO/IEC MPEG-4 AVC), Codificação de Vídeo de Alta Eficiência (HEVC) e quaisquer extensões para tais padrões. Além disso, os conjuntos de procedimentos descritos nesta revelação podem se tornar parte de padrões desenvolvidos no futuro. Em outras palavras, os conjuntos de procedimentos descritos nesta revelação podem ser aplicáveis aos padrões de codificação de vídeo previamente desenvolvidos, aos padrões de codificação de vídeo atualmente em desenvolvimento e aos futuros padrões de

codificação de vídeo.

PADRÕES DE CODIFICAÇÃO DE VÍDEO

[0025] Uma imagem digital, tal como uma imagem de vídeo, uma imagem de TV, uma imagem estática ou uma imagem gerada por um gravador de vídeo ou um computador, pode incluir pixels ou amostras dispostos em linhas horizontais e verticais. O número de pixels em uma única imagem é tipicamente de dezenas de milhares. Cada pixel contém, tipicamente, informações de luminância e de crominância. Sem compactação, a quantidade pura de informações a serem transportadas de um encriptador de imagem para um decodificador de imagem tornaria a transmissão de imagem em tempo real impraticável. Para reduzir a quantidade de informações a serem transmitidas, diversos métodos de compactação diferentes, tais como JPEG, MPEG e padrões H.263, foram desenvolvidos.

[0026] Os padrões de codificação de vídeo incluem ITU-T H.261, ISO/IEC MPEG-1 Visual, ITU-T H.262 ou ISO/IEC MPEG-2 Visual, ITU-T H.263, ISO/IEC MPEG-4 Visual, ITU-T H.264 (também conhecido como ISO/IEC MPEG-4 AVC), e HEVC que inclui extensões de tais padrões.

[0027] Além disso, um padrão de codificação de vídeo, a saber DSC, foi desenvolvido por VESA. O padrão de DSC é um padrão de compactação de vídeo que pode comprimir o vídeo para transmissão através de enlaces de exibição. Conforme a resolução dos visores aumenta, a largura de banda dos dados de vídeo necessários para acionar os visores aumenta de modo correspondente. Alguns enlaces de exibição podem não ter a largura de banda para transmitir todos os dados de vídeo para o visor para tais resoluções. Consequentemente, o padrão de DSC especifica um padrão de compactação para compactação visualmente sem perda interoperacional através de enlaces de exibição.

[0028] O padrão de DSC é diferente dos outros padrões de codificação de vídeo, como H.264 e HEVC. DSC inclui a compactação intraquadro, mas não inclui a compactação interquadros, significando que as informações temporais podem não ser usadas pelo padrão de DSC na codificação dos dados de vídeo. Em contraste, outros padrões de codificação de vídeo podem empregar a compactação interquadros em seus conjuntos de procedimentos de codificação de vídeo.

SISTEMA DE CODIFICAÇÃO DE VÍDEO

[0029] Vários aspectos dos sistemas, aparelhos e métodos inovadores são descritos mais plenamente doravante com referência aos desenhos anexos. Entretanto, esta revelação pode ser incorporada de formas diferentes e não deve ser interpretada como limitada a qualquer estrutura ou função específica apresentada ao longo de toda esta revelação. Em vez disso, esses aspectos são fornecidos de modo que esta revelação seja minuciosa e completa, e transmita plenamente o escopo da revelação para os indivíduos versados na técnica. Com base nos ensinamentos no presente documento, um indivíduo versado na técnica deve observar que o escopo da revelação está destinado a cobrir qualquer aspecto dos sistemas, aparelhos e métodos inovadores revelados no presente documento, sejam os mesmos implantados independentemente ou em combinação com qualquer outro aspecto da revelação. Por exemplo, um aparelho pode ser implantado ou um método pode ser praticado com o uso de qualquer quantidade dos aspectos apresentados no presente documento. Além disso, o escopo da presente revelação está destinado a cobrir tal aparelho ou método que é praticado com o uso de outra estrutura, funcionalidade ou estrutura e funcionalidade além de ou outras que não sejam os vários aspectos da presente revelação estabelecidos no presente

documento. Deve ser entendido que qualquer aspecto revelado no presente documento pode ser incorporado por um ou mais elementos de uma reivindicação.

[0030] Embora aspectos particulares sejam descritos no presente documento, muitas variações e permutações desses aspectos são abrangidos pelo escopo da revelação. Embora alguns benefícios e vantagens dos aspectos preferenciais sejam mencionados, o escopo da revelação não está destinado a ser limitado a benefícios, usos ou objetivos particulares. Em vez disso, os aspectos da revelação destinam-se a ser amplamente aplicáveis a diferentes tecnologias sem fio, configurações de sistema, redes e protocolos de transmissão, alguns dos quais são ilustrados a título de exemplo nas figuras e na descrição a seguir dos aspectos preferenciais. A descrição detalhada e desenhos são meramente ilustrativos da revelação em vez de limitantes, sendo que o escopo da revelação é definido pelas reivindicações anexas e equivalentes das mesmas.

[0031] Os desenhos anexos ilustram exemplos. Os elementos indicados por números de referência nos desenhos anexos correspondem aos elementos indicados por números de referência semelhantes na descrição a seguir. Nesta revelação, os elementos que têm nomes que começam com palavras ordinais (por exemplo, "primeiro(a)", "segundo(a)", "terceiro(a)" e assim por diante) não implicam necessariamente que os elementos têm uma ordem particular. Em vez disso, tais palavras ordinais são meramente usadas para se referir a elementos diferentes de um tipo igual ou similar.

[0032] A Figura 1A é um diagrama de blocos que ilustra um sistema de codificação de vídeo exemplificativo 10 que pode utilizar os conjuntos de procedimentos de acordo com os aspectos descritos nesta revelação. Conforme usado

descrito no presente documento, o termo "codificador de vídeo" ou "codificador" se refere genericamente tanto a encriptadores de vídeo quanto a decodificadores de vídeo. Nesta revelação, os termos "codificação de vídeo" ou "codificação" podem se referir genericamente à criptação de vídeo e à decodificação de vídeo. Além de encriptadores de vídeo e decodificadores de vídeo, os aspectos descritos no presente pedido podem ser estendidos a outros dispositivos relacionados, como transcodificadores (por exemplo, dispositivos que podem decodificar um fluxo de bits e recriptar outro fluxo de bits) e dispositivos intermediários (por exemplo, dispositivos que podem modificar, transformar e/ou, de outro modo, manipular um fluxo de bits).

[0033] Conforme mostrado na Figura 1A, sistema de codificação de vídeo 10 inclui um dispositivo-fonte 12 (isto é, "dispositivo de codificação de vídeo 12" ou "dispositivo de codificação 12") que gera dados de vídeo criptados a serem decodificados em um momento posterior por um dispositivo de destino 14 (isto é, "dispositivo de codificação de vídeo 14" ou "dispositivo de codificação 14"). No exemplo da Figura 1A, o dispositivo-fonte 12 e o dispositivo de destino 14 constituem dispositivos separados. Nota-se, entretanto, que o dispositivo-fonte e o dispositivo de destino 12 podem estar ou fazer parte do mesmo dispositivo, conforme mostrado no exemplo da Figura 1B

[0034] Novamente com referência à Figura 1A, o dispositivo de fonte 12 e o dispositivo de destinação 14 podem compreender, respectivamente, qualquer dentre uma ampla faixa de dispositivos (também chamados de dispositivos de codificação de vídeo) incluindo computadores do tipo desktop, computadores do tipo notebook (por exemplo, laptop), computadores do tipo tablet, decodificadores de sinais, aparelhos de telefone tal como os chamados telefones

"inteligentes", chamados computadores do tipo pad "inteligente", televisões, câmeras, dispositivos de exibição, reprodutores de mídias digitais, consoles para jogo eletrônico, dispositivo de transmissão contínua de vídeo ou semelhantes. Em várias modalidades, o dispositivo-fonte 12 e o dispositivo de destino 14 podem ser equipados para (isto é, configurados para se comunicar por meio de) comunicação sem fio.

[0035] Os dispositivos de codificação de vídeo 12, 14 do sistema de codificação de vídeo 10 podem ser configurados para se comunicar por meio de redes sem fio e tecnologias de rádio, como portadoras de rede de área ampla sem fio (WWAN) (por exemplo, celular) e/ou rede de área local sem fio (WLAN). Os termos "rede" e "sistema" são frequentemente usados de modo intercambiável. Cada um dos dispositivos de codificação de vídeo 12, 14 pode ser um equipamento de usuário (UE), um dispositivo sem fio, um terminal, uma estação móvel, uma unidade de assinante, etc.

[0036] As portadoras de WWAN podem incluir, por exemplo, redes de comunicação sem fio como Acesso Múltiplo por Divisão de Código (CDMA), Acesso Múltiplo por Divisão de Tempo (TDMA), Acesso Múltiplo por Divisão de Frequência (FDMA), FDMA Ortogonal (OFDMA), FDMA de Única Portadora (SC-FDMA) e outras redes. Uma rede de CDMA pode implantar uma tecnologia a rádio tal como Acesso de Rádio Terrestre Universal (UTRA), CDMA2000, etc. UTRA inclui CDMA de banda larga (WCDMA) e outras variações de CDMA. A CDMA2000 abrange os padrões IS-2000, IS-95 e IS-856. Uma rede de TDMA pode implantar uma tecnologia a rádio como o Sistema Global para Comunicações Móveis (GSM). Uma rede de OFDMA pode implantar uma tecnologia a rádio como UTRA Evoluído (E-UTRA), Banda larga Ultra Móvel (UMB), IEEE 802.11 (Wi-Fi), IEEE 802.16 (WiMAX), IEEE 802.20, Flash-OFDMA, etc. Os UTRA e E-UTRA são

partes do Sistema Universal de Telecomunicação Móvel (UMTS). A Evolução em Longo Prazo (LTE) de 3GPP e a LTE Avançada (LTE-A) são novas liberações de UMTS que usam E-UTRA. O UTRA, o E-UTRA, o UMTS, a LTE, a LTE-A e a GSM são descritos nos documentos de uma organização chamada "Projeto de Parceria de Terceira Geração" (3GPP). O CDMA2000 e o UMB são descritos em documentos de uma organização chamada "Projeto de Parceria de Terceira Geração 2" (3GPP2).

[0037] Os dispositivos de codificação de vídeo 12, 14 do sistema de codificação de vídeo 10 também podem se comunicar um com o outro por meio de uma estação-base de WLAN de acordo com um ou mais padrões, como o padrão IEEE 802.11, incluindo, por exemplo, essas modificações: 802.11a-1999 (chamada comumente de "802.11a"), 802.11b-1999 (chamada comumente de "802.11b"), 802.11g-2003 (chamada comumente de "802.11g"), e assim por diante.

[0038] O dispositivo de destino 14 pode receber, através do enlace 16, os dados de vídeo criptados a ser decodificados. O enlace 16 pode compreender qualquer tipo de mídia ou dispositivo com a capacidade de mover os dados de vídeo criptados a partir do dispositivo-fonte 12 para o dispositivo de destino 14. No exemplo da Figura 1A, o enlace 16 pode compreender um meio de comunicação para habilitar o dispositivo-fonte 12 para transmitir dados de vídeo criptados ao dispositivo de destino 14 em tempo real. Os dados de vídeo criptados podem ser modulados de acordo com um padrão de comunicação, tal como um protocolo de comunicação sem fio, e transmitidos para o dispositivo de destino 14. A mídia de comunicação pode compreender qualquer mídia de comunicação sem fio ou com fio, tal como um espectro de radiofrequência (RF) ou uma ou mais linhas de transmissão físicas. A mídia de comunicação pode formar parte de uma rede com base em pacote, tal como, uma rede de área local,

uma rede de área ampla ou uma rede global como a Internet. A mídia de comunicação pode incluir roteadores, comutadores, estações-base, ou qualquer outro equipamento que possa ser útil que para facilitar a comunicação do dispositivo-fonte 12 com o dispositivo de destino 14.

[0039] No exemplo da Figura 1A, o dispositivo-fonte 12 inclui uma fonte de vídeo 18, um encriptador de vídeo 20 e uma interface de saída 22. Em alguns casos, a interface de saída 22 pode incluir um modulador/demodulador (modem) e/ou um transmissor. No dispositivo-fonte 12, a fonte de vídeo 18 pode incluir uma fonte tal como um dispositivo de captura de vídeo, por exemplo, uma câmera de vídeo, um arquivo de vídeo que contém vídeo previamente capturado, uma interface de alimentação de vídeo para receber vídeo de um provedor de conteúdo de vídeo e/ou um sistema de gráficos de computador para gerar dados de gráficos de computador como o vídeo-fonte, ou uma combinação de tais fontes. Como um exemplo, se a fonte de vídeo 18 for uma câmera de vídeo, o dispositivo-fonte 12 e o dispositivo de destino 14 podem formar os chamados "telefones com câmera" ou "telefones para gravação de vídeo", conforme ilustrado no exemplo da Figura 1B. Entretanto, os conjuntos de procedimentos descritos nesta revelação podem ser aplicáveis à codificação de vídeo em geral e podem ser aplicadas a aplicações com fio e/ou sem fio.

[0040] O vídeo gerado por computador, pré-capturado ou capturado pode ser criptado pelo encriptador de vídeo 20. Os dados de vídeo criptados podem ser transmitidos para o dispositivo de destino 14 através da interface de emissão 22 do dispositivo-fonte 12. Os dados de vídeo criptados também podem (ou, alternativamente) ser armazenados no dispositivo de armazenamento 31 para acesso posterior pelo dispositivo de destino 14 ou outros

dispositivos, para decodificação e/ou reprodução. O encriptador de vídeo 20 ilustrado nas Figuras 1A e 1B pode compreender o encriptador de vídeo 20 ilustrado na Figura 2A, ou qualquer outro encriptador de vídeo descrito no presente documento.

[0041] No exemplo da Figura 1A, o dispositivo de destino 14 inclui uma interface de entrada 28, um decodificador de vídeo 30 e um dispositivo de exibição 32. Em alguns casos, a interface de entrada 28 pode incluir um receptor e/ou um modem. A interface de entrada 28 do dispositivo de destino 14 pode receber os dados de vídeo criptados através do enlace 16 e/ou a partir do dispositivo de armazenamento 31. Os dados de vídeo criptados comunicados através do enlace 16, ou fornecidos no dispositivo de armazenamento 31, podem incluir uma variedade de elementos de sintaxe gerados pelo encriptador de vídeo 20 para uso através de um decodificador de vídeo, tal como o decodificador de vídeo 30, na decodificação dos dados de vídeo. Tais elementos de sintaxe podem estar incluídos com os dados de vídeo criptados transmitidos em uma mídia de comunicação, armazenados em uma mídia de armazenamento ou armazenados em um servidor de arquivo. O decodificador de vídeo 30 ilustrado nas Figuras 1A e 1B pode compreender o decodificador de vídeo 30 ilustrado na Figura 2B, ou qualquer outro decodificador de vídeo descrito no presente documento.

[0042] O dispositivo de exibição 32 pode ser integrado ao dispositivo de destino 14, ou pode ser externo ao mesmo. Em alguns exemplos, o dispositivo de destino 14 pode incluir um dispositivo de exibição integrado e também pode ser configurado para fazer interface com um dispositivo de exibição externo. Em outros exemplos, o dispositivo de destino 14 pode ser um dispositivo de exibição. Em geral, o dispositivo de exibição 32 exibe os dados de vídeo

decodificados a um usuário e pode compreender qualquer um dentre uma variedade de dispositivos de exibição, tais 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.

[0043] Em aspectos relacionados, a Figura 1B mostra um sistema de codificação de vídeo exemplificativo 10' em que o dispositivo-fonte 12 e o dispositivo de destino 14 estão em, ou fazem parte de um dispositivo 11. O dispositivo 11 pode ser um aparelho de telefone, tal como um telefone "inteligente" ou similares. O dispositivo 11 pode incluir um dispositivo controlador/processador 13 (opcionalmente presente) em comunicação operacional com o dispositivo-fonte 12 e o dispositivo de destino 14. O sistema de codificação de vídeo 10' da Figura 1B, e os componentes do mesmo, são, de outro modo, similares ao sistema de codificação de vídeo 10 da Figura 1A, e aos componentes do mesmo.

[0044] O encriptador de vídeo 20 e o decodificador de vídeo 30 podem operar de acordo com um padrão de compactação de vídeo, como DSC. Alternativamente, o encriptador de vídeo 20 e o decodificador de vídeo 30 podem operar de acordo com outros padrões de proprietário ou de indústria, como o padrão ITU-T H.264, referido de modo alternativo como MPEG-4, Parte 10, AVC, HEVC ou extensões de tais padrões. Entretanto, os conjuntos de procedimentos desta revelação não são limitados a qualquer padrão de codificação particular. Outros exemplos de padrões de compactação de vídeo incluem MPEG-2 e ITU-T H.263.

[0045] Embora não mostrado nos exemplos das Figuras 1A e 1B, o encriptador de vídeo 20 e o decodificador de vídeo 30 podem ser, cada um, integrados a um encriptador e decodificador de áudio, e podem incluir unidades de MUX-

DEMUX adequadas, ou outro hardware e software, para manipular a criptação tanto do áudio quanto do vídeo em um fluxo de dados comum ou fluxos de dados separados. Caso aplicável, em alguns exemplos, as unidades MUX-DEMUX podem se conformar ao protocolo multiplexador ITU H.223, ou outros protocolos como protocolo de datagrama de usuário (UDP).

[0046] Cada um dentre o encriptador de vídeo 20 e o decodificador de vídeo 30 pode ser implantado como qualquer um dentre uma variedade de conjuntos de circuitos de encriptador adequados, tal como um ou mais microprocessadores, processadores de sinal digital (DSPs), circuitos integrados específicos para aplicação (ASICs), matrizes de porta programável em campo (FPGAs), lógica discreta, software, hardware, firmware ou quaisquer combinações dos mesmos. Quando os conjuntos de procedimentos são implantados parcialmente em software, um dispositivo pode armazenar instruções para o software em um meio legível por computador não transitório adequado e executar as instruções em hardware com o uso de um ou mais processadores para realizar os conjuntos de procedimentos desta revelação. Cada um dentre o encriptador de vídeo 20 e o decodificador de vídeo 30 pode ser incluído em um ou mais encriptadores ou decodificadores, em que qualquer um dos quais pode ser integrado como parte de um encriptador/decodificador combinado (CODEC) em um respectivo dispositivo.

PROCESSO DE CODIFICAÇÃO DE VÍDEO

[0047] Conforme mencionado brevemente acima, o encriptador de vídeo 20 cripta os dados de vídeo. Os dados de vídeo podem compreender uma ou mais figurações. Cada uma das figurações é uma imagem estática que faz parte de um vídeo. Em alguns casos, uma figuração pode ser referida como um "quadro" de vídeo. Quando o encriptador de vídeo 20 cripta os dados de vídeo (por exemplo, dados de camada de

codificação de vídeo (VCL) e/ou dados de não VCL), o encriptador de vídeo 20 pode gerar um fluxo de bits. O fluxo de bits pode incluir uma sequência de bits que forma uma representação codificada dos dados de vídeo. O fluxo de bits pode incluir figurações codificadas e dados associados. Uma figuração codificada é uma representação codificada de uma figuração. Os dados de VCL podem incluir dados de figuração codificados (isto é, informações associadas a amostras de uma figuração (ou figurações) codificada) e dados de não VCL podem incluir informações de controle (por exemplo, conjuntos de parâmetros e/ou informações de aperfeiçoamento suplementar) associadas à uma ou mais figurações codificadas.

[0048] A fim de gerar o fluxo de bits, o encriptador de vídeo 20 pode realizar operações de criptação em cada figuração nos dados de vídeo. Quando o encriptador de vídeo 20 realiza operações de criptação nas figurações, o encriptador de vídeo 20 pode gerar uma série de figurações codificadas e dados associados. Os dados associados podem incluir um conjunto de parâmetros de codificação como um parâmetro de quantização (QP). Para gerar uma figuração codificada, o encriptador de vídeo 20 pode particionar uma figuração em blocos de vídeo igualmente dimensionados. Um bloco de vídeo pode ser uma matriz bidimensional de amostras. Os parâmetros de codificação podem definir uma opção de codificação (por exemplo, um modo de codificação) para cada bloco dos dados de vídeo. A opção de codificação pode ser selecionada a fim de alcançar um desempenho de taxa-distorção desejado.

[0049] Em alguns exemplos, o encriptador de vídeo 20 pode particionar uma figuração em uma pluralidade de fatias. Cada uma das fatias pode incluir uma região espacialmente distinta em uma imagem (por exemplo, um quadro)

que pode ser decodificado independentemente sem informações do restante das regiões na imagem ou quadro. Cada quadro de imagem ou vídeo pode ser criptado em uma única fatia ou cada quadro de imagem ou vídeo pode ser criptado em várias fatias. Em DSC, o número de bits alocados para criptar cada pode ser substancialmente constante. Como parte da realização de uma operação de criptação em uma figuração, o encriptador de vídeo 20 pode realizar operações de criptação em cada fatia da figuração. Quando o encriptador de vídeo 20 realizar uma operação de encriptação em uma fatia, o encriptador de vídeo 20 pode gerar dados criptados associados à fatia. Os dados criptados associados à fatia podem ser denominados "fatia codificada".

ENCRIPTADOR DE VÍDEO DE DSC

[0050] A Figura 2A é um diagrama de blocos que ilustra um exemplo do encriptador de vídeo 20 que pode implantar conjuntos de procedimentos de acordo com os aspectos descritos nesta revelação. Adicionalmente, o encriptador de vídeo 20 pode ser configurado para realizar alguns ou todos os conjuntos de procedimentos desta revelação. Em alguns exemplos, os conjuntos de procedimentos descritos nesta revelação podem ser compartilhados dentre os vários componentes do encriptador de vídeo 20. Em alguns exemplos, adicional ou alternativamente, um processador (não mostrado) pode ser configurado para realizar alguns ou todos os conjuntos de procedimentos descritos nesta revelação.

[0051] Para propósitos de explicação, esta revelação descreve o encriptador de vídeo 20 no contexto de codificação de DSC. Entretanto, os conjuntos de procedimentos desta revelação podem ser aplicáveis a outros padrões ou métodos de codificação.

[0052] No exemplo da Figura 2A, o encriptador de vídeo 20 inclui uma pluralidade de componentes funcionais.

Os componentes funcionais do encriptador de vídeo 20 incluem um conversor de espaço de cores 105, um armazenamento temporário, 110, um detector de planura 115, um controlador de taxa 120, um componente previsor, quantizador e reconstrutor 125, um armazenamento temporário em linha 130, um histórico de cor indexado 135, um encriptador por entropia 140, um multiplexador de subfluxo 145 e um armazenamento temporário de taxa 150. Em outros exemplos, o encriptador de vídeo 20 pode incluir mais, menos ou diferentes componentes funcionais.

[0053] O conversor de espaço de cores 105 pode converter um espaço de cores de entrada no espaço de cores usado na implantação de codificação. Por exemplo, em uma modalidade exemplificativa, o espaço de cores dos dados de vídeo de entrada está no espaço de cores vermelho, verde e azul (RGB) e a codificação é implantada no espaço de cores de luminância Y, crominância verde Cg e crominância laranja (YCgCo). A conversão de espaço de cores pode ser realizada pelo método (ou métodos) que inclui turnos e adições aos dados de vídeo. Nota-se que os dados de vídeo de entrada em outros espaços de cor podem ser processados e conversões em outros espaços de cor também podem ser realizados.

[0054] Em aspectos relacionados, o encriptador de vídeo 20 pode incluir o armazenamento temporário 110, o armazenamento temporário em linha 130 e/ou o armazenamento temporário de taxa 150. Por exemplo, o armazenamento temporário 110 pode manter os dados de vídeo convertidos de espaço de cores antes de seu uso por outras porções do encriptador de vídeo 20. Em outro exemplo, os dados de vídeo podem ser armazenados no espaço de cores RGB e a conversão de espaço de cor pode ser realizada conforme necessários, visto que os dados convertidos de espaço de cores podem exigir mais bits.

[0055] O armazenamento temporário de taxa 150 pode funcionar como parte do mecanismo de controle de taxa no encriptador de vídeo 20, que será descrito em maiores detalhes abaixo em conexão com o controlador de taxa 120. Os bits gastos na criptação de cada bloco podem variar de modo altamente substancial com base na natureza do bloco. O armazenamento temporário de taxa 150 pode suavizar as variações de taxa no vídeo compactado. Em algumas modalidades, um modelo de armazenamento temporário CB_R é empregue, em que os bits são removidos do armazenamento temporário em um CB_R . No modelo de armazenamento temporário CB_R , se o encriptador de vídeo 20 adiciona bits demais ao fluxo de bits, o armazenamento temporário de taxa 150 pode estourar. Por outro lado, o encriptador de vídeo 20 deve adicionar bits suficientes a fim de evitar o subfluxo do armazenamento temporário de taxa 150.

[0056] No lado de decodificador de vídeo, os bits podem ser adicionados ao armazenamento temporário de taxa 155 do decodificador de vídeo 30 (consulte a Figura 2B que é descrita em maiores detalhes abaixo) em um CB_R e o decodificador de vídeo 30 pode remover números variáveis de bits para cada bloco. Para assegurar a decodificação adequada, o armazenamento temporário de taxa 155 do decodificador de vídeo 30 não deveria "estourar negativamente" ou "estourar" durante a decodificação do fluxo de bits compactado.

[0057] Conforme discutido acima, o BF pode ser definido com base nos valores `BufferCurrentSize` que representam o número de bits atualmente no armazenamento temporário e `BufferMaxSize` que representa o tamanho do armazenamento temporário de taxa 150, isto é, o número de bits máximo que pode ser armazenado no armazenamento temporário de taxa 150 em qualquer ponto no tempo. O BF pode

ser calculado de acordo com as Equações 1 ou 2 acima, mas os métodos adicionais para calcular o BF são descritos abaixo.

[0058] O detector de planura 115 pode detectar mudanças de áreas complexas (isto é, não planas) nos dados de vídeo para áreas planas (isto é, simples ou uniformes) nos dados de vídeo. Os termos "complexo" e "plano" serão usados no presente documento para se referir genericamente à dificuldade para o encriptador de vídeo 20 criptar as respectivas regiões dos dados de vídeo. Portanto, o termo "complexo" conforme usado no presente documento descreve, de modo geral, uma região dos dados de vídeo como sendo complexa para o encriptador de vídeo 20 criptar e pode, por exemplo, incluir dados de vídeo texturizados, frequência espacial alta e/ou outras particularidades que são complexas de criptar. O termo "plano" conforme usado no presente documento descreve, de modo geral, uma região dos dados de vídeo como sendo simples para o encriptador de vídeo 20 criptar e pode, por exemplo, incluir um gradiente suave nos dados de vídeo, frequência espacial baixa e/ou outras particularidades que são simples de criptar. As transições entre as regiões complexas e planas podem ser usadas pelo encriptador de vídeo 20 para reduzir os artefatos de quantização nos dados de vídeo criptados. Especificamente, o controlador de taxa 120 e o componente previsor, quantizador e reconstrutor 125 podem reduzir tais artefatos de quantização quando as transições de regiões complexas para regiões planas forem identificadas.

[0059] O controlador de taxa 120 determina um conjunto de parâmetros de codificação, por exemplo, um QP. O QP pode ser ajustado pelo controlador de taxa 120 com base no BF do armazenamento temporário de taxa 150 e atividade de imagem dos dados de vídeo a fim de maximizar a qualidade de figuração para uma taxa de bits-alvo que assegura que o

armazenamento temporário de taxa 150 não estoure ou estoure negativamente. O controlador de taxa 120 também seleciona uma opção de codificação particular (por exemplo, um modo particular) para cada bloco dos dados de vídeo a fim de alcançar o desempenho de taxa-distorção ideal. O controlador de taxa 120 minimiza a distorção das imagens reconstruídas de modo que a distorção satisfaça a restrição de taxa de bits, isto é, a taxa de codificação real geral se adapta dentro da taxa de bits alvo.

[0060] O componente previsor, quantizador e reconstrutor 125 pode realizar pelo menos três operações de criptação do encriptador de vídeo 20. O componente previsor, quantizador e reconstrutor 125 pode realiza previsão em vários modos diferentes. Um modelo de previsão exemplificativo é uma versão modificada da previsão adaptativa de mediana. A previsão adaptativa de mediana pode ser implantada pelo padrão de JPEG sem Perda (JPEG-LS). A versão modificada de previsão adaptativa mediana que pode ser realizada pelo componente previsor, quantizador e reconstrutor 125 pode permitir a previsão paralela de três valores de amostra consecutivos. Outro modo de previsão exemplificativo é a previsão de bloco. Na previsão de bloco, as amostras são previstas a partir de pixels reconstruídos anteriormente na linha acima ou à esquerda na mesma linha. Em algumas modalidades, o encriptador de vídeo 20 e o decodificador de vídeo 30 podem ambos realizar uma busca idêntica em pixels reconstruídos para determinar os usos de previsão de bloco, e portanto, nenhum bit precisa ser enviado no modo de previsão de bloco. Em outras modalidades, o encriptador de vídeo 20 pode realizar a busca e sinalizar vetores de previsão de bloco no fluxo de bits, de modo que o decodificador de vídeo 30 não precise realizar uma busca separada. Um modo de previsão de ponto intermediário também

pode ser implantado, no qual as amostras são previstas com o uso do ponto intermediário da faixa de componente. O modo de previsão de ponto intermediário pode habilitar a delimitação do número de bits necessário para o vídeo compactado até mesmo na amostra de pior caso.

[0061] O componente previsor, quantizador e reconstrutor 125 também realiza a quantização. Por exemplo, a quantização pode ser realizada por meio de um quantizador de potência de 2 que pode ser implantado com o uso de um comutador. Nota-se que outros conjuntos de procedimentos de quantização podem ser implantados ao invés do quantizador de potência de 2. A quantização realizada pelo componente previsor, quantizador e reconstrutor 125 pode ter por base o QP determinado pelo controlador de taxa 120. Finalmente, o componente previsor, quantizador e reconstrutor 125 também realiza a reconstrução que inclui adicionar o resíduo quantizado inverso ao valor previsto e assegurar que o resultado não caia fora da faixa válida dos valores de amostra.

[0062] Nota-se que as abordagens exemplificativas à previsão, quantização e reconstrução realizada pelo componente previsor, quantizador e reconstrutor 125 são meramente ilustrativas e que outras abordagens podem ser implantadas. Nota-se que o componente previsor, quantizador e reconstrutor 125 pode incluir subcomponente (ou subcomponentes) para realizar a previsão, a quantização e/ou a reconstrução. Nota-se adicionalmente que a previsão, a quantização e/ou a reconstrução pode ser realizada por vários componentes encriptadores separados em vez do componente previsor, quantizador e reconstrutor 125.

[0063] O armazenamento temporário em linha 130 mantém a saída do componente previsor, quantizador e reconstrutor 125 de modo que o componente previsor,

quantizador e reconstrutor 125 e o histórico de cor indexado 135 possa usar os dados de vídeo armazenados em armazenamento temporário. O histórico de cor indexado 135 armazena valores de pixel usados recentemente. Esses valores de pixel usados recentemente podem ser referenciados diretamente pelo encriptador de vídeo 20 por meio de uma sintaxe dedicada.

[0064] O encriptador por entropia 140 cripta os resíduos de previsão e quaisquer outros dados (por exemplo, índices identificados pelo componente predictor, quantizador e reconstrutor 125) recebido a partir do componente predictor, quantizador e reconstrutor 125 com base no histórico de cor indexado 135 e as transições de planura identificadas pelo detector de planura 115. Em alguns exemplos, o encriptador por entropia 140 pode criptar três amostras por relógio por encriptador de subfluxo. O multiplexador de subfluxo 145 pode multiplexar o fluxo de bits com base no esquema de multiplexação de pacote sem cabeçalho. Isso permite que o decodificador de vídeo 30 corra três decodificadores por entropia em paralelo, facilitando a decodificação de três pixels por relógio. O multiplexador de subfluxo 145 pode otimizar a ordem de pacotes de modo que os pacotes possam ser decodificados de modo eficaz pelo decodificador de vídeo 30. Nota-se que abordagens diferentes à codificação por entropia podem ser implantadas, que pode facilitar a decodificação de pixels de potência de 2 por relógio (por exemplo, 2 pixels/relógio ou 4 pixels/relógio).

DECODIFICADOR DE VÍDEO DE DSC

[0065] A Figura 2B é um diagrama de blocos que ilustra um exemplo do decodificador de vídeo 30 que pode implantar conjuntos de procedimentos de acordo com os aspectos descritos nesta revelação. O decodificador de vídeo 30 pode ser configurado para realizar alguns ou todos os conjuntos de procedimentos da presente revelação. Em alguns

exemplos, os conjuntos de procedimentos descritos nesta revelação podem ser compartilhados entre os vários componentes de decodificador de vídeo 30. Em alguns exemplos, adicional ou alternativamente, um processador (não mostrado) pode ser configurado para realizar alguns ou todos os conjuntos de procedimentos descritos nesta revelação.

[0066] Para propósitos de explicação, esta revelação descreve o decodificador de vídeo 30 no contexto de codificação de DSC. Entretanto, os conjuntos de procedimentos desta revelação podem ser aplicáveis a outros padrões ou métodos de codificação.

[0067] No exemplo da Figura 2B, o decodificador de vídeo 30 inclui uma pluralidade de componentes funcionais. Os componentes funcionais do decodificador de vídeo 30 incluem um armazenamento temporário de taxa 155, um demultiplexador de subfluxo 160, um decodificador por entropia 165, um controlador de taxa 170, um componente predictor, quantizador e reconstrutor 175, um histórico de cor indexado 180, um armazenamento temporário em linha 185, e um conversor de espaço de cores 190. Os componentes ilustrados do decodificador de vídeo 30 são análogos aos componentes correspondentes descritos acima em conexão com o encriptador de vídeo 20 na Figura 2A. Como tal, cada um dos componentes do decodificador de vídeo 30 pode funcionar de modo semelhante aos componentes correspondentes do encriptador de vídeo 20 conforme descrito acima.

FATIAS EM DSC

[0068] Conforme notado acima, uma fatia se refere, de modo geral, a uma região espacialmente distinta em uma imagem ou um quadro que pode ser decodificado independentemente sem o uso de informações do restante das regiões na imagem ou quadro. Cada quadro de imagem ou vídeo pode ser criptado em uma única fatia ou cada quadro de imagem

ou vídeo pode ser criptado em várias fatias. Em DSC, o número de bits alocados para criptar cada pode ser substancialmente constante. Uma fatia pode ser composta de múltiplos blocos.

ARMAZENAMENTO TEMPORÁRIO DE CONTROLE DE TAXA

[0069] A Figura 3 é um gráfico que mostra um processo exemplificativo para criptar fatias de dados. Com referência à Figura 3, eixo geométrico vertical 305 representa um número de bits atualmente armazenado no armazenamento temporário de taxa (por exemplo, o armazenamento temporário de taxa 150 no encriptador de vídeo 20 na Figura 2A, ou o armazenamento temporário de taxa 155 do decodificador de vídeo 30 na Figura 2B) e o eixo geométrico horizontal 310 representa o tempo. A Figura 3 mostra uma criptação de uma primeira fatia 315 dos dados de vídeo e uma segunda fatia 320 dos dados de vídeo. A segunda fatia 320 é ilustrada como sendo posicionada abaixo da primeira fatia 315 meramente para mostrar uma sobreposição no tempo que ocorre durante a criptação da primeira fatia 315 e a criptação da segunda fatia 320 e não se pretende indicar que o número de bits na segunda fatia 320 seja menor do que o número de bits na primeira fatia 315.

[0070] Para propósitos ilustrativos, os aspectos da presente revelação serão descritos com referência ao armazenamento temporário de taxa 150 do encriptador de vídeo 20 na Figura 2A; entretanto, será compreendido que tais aspectos também são aplicáveis ao decodificador de vídeo 30 da Figura 2B ou componente(s) do mesmo, incluindo, sem limitação, o armazenamento temporário de taxa 155.

[0071] Nota-se que o armazenamento temporário de taxa 150 pode ter a capacidade de manter (isto é, armazenar) somente um número finito de bits. Conforme discutido acima, uma variável BufferMaxSize pode representar

uma capacidade geral do armazenamento temporário de taxa. Entretanto, a fim de evitar o estouro negativo e/ou estouro no armazenamento temporário de taxa 150, é vantajoso limitar o armazenamento temporário de taxa 150 para um número de bits inferior ao BufferMaxSize. Como tal, os níveis de bit máximo temporário (representados pelas variáveis tempBufferMaxSize e maxBufferBitsAtSliceEnd) podem ser definidos pelo armazenamento temporário de taxa 150 para refletir uma capacidade desejada do armazenamento temporário de taxa 150.

[0072] Em 325, a criptação da primeira fatia 315 começa. Durante a criptação, os bits podem ser armazenados no armazenamento temporário de taxa 150 e os bits anteriormente armazenados ao armazenamento temporário de taxa 150 podem ser removidos do armazenamento temporário de taxa 150. Por exemplo, os bits removidos do armazenamento temporário de taxa 150 podem ser transmitidos (por exemplo, pelo dispositivo-fonte 12, através do enlace 16). Como tal, devido às operações para armazenar bits a e/ou remover bits do armazenamento temporário de taxa 150, o número total de bits no armazenamento temporário de taxa 150 pode aumentar e/ou reduzir em vários pontos de tempo. Em alguns casos, os bits podem ser agrupados juntos para criptação, transmissão, etc. Por exemplo, os bits podem ser transmitidos como pixels (por exemplo, seis bits por pixel) ou blocos (por exemplo, 96 bits por bloco para um tamanho de bloco de 2x8 com 6 bits por pixel).

[0073] Em 330, transmissão de bits armazenados começa. A transmissão pode começar após um atraso um período de atraso de criptação inicial, indicado na Figura 3 como Init_enc_delay. O Init_enc_delay pode ser medido em blocos ou pixels e podem ser escolhido e/ou valor fixo ou podem ser calculado para cada fatia. Em uma modalidade, Init_enc_delay

pode ter por base o `maxBufferBitsAtSliceEnd`. Por exemplo, se `maxBufferBitsAtSliceEnd` for 4128 e `Init_enc_delay` for medido nos blocos, `Init_enc_delay` pode ser 4128 dividido pelo número de bits alvo por bloco (por exemplo, 96). Isso pode ser feito de modo que o armazenamento temporário 150 possa transmitir os blocos restantes a partir da primeira fatia 315 após a primeira fatia 315 foi completamente decodificado.

[0074] Com referência contínua à Figura 3, em 335, criptação da primeira fatia 315 pode ser completada. Isto é, em 335, o número de bits no armazenamento temporário 150 deve ser igual a `maxBufferBitsAtSliceEnd`. Por exemplo, se `maxBufferBitsAtSliceEnd` for 4128, o número de bits no armazenamento temporário 150 deve ser igual 4128 no evento 335, quando a criptação da primeira fatia 315 é completa. Em 340, a transmissão da primeira fatia 315 é completa. Conforme mostrado na Figura 3, o fim da criptação 335 para a primeira fatia 315 pode corresponder ao início da criptação para a segunda fatia 320, enquanto o fim da transmissão 340 para a primeira fatia 315 corresponde ao início de transmissão 350 para a segunda fatia 320. `Init_enc_delay` pode ser selecionado para alinhar os eventos de terminação (por exemplo, fim da criptação e transmissão) da primeira fatia 315 com os eventos iniciais (por exemplo, início da criptação e transmissão) da segunda fatia 320. Isto é, o período `Init_enc_delay` para a segunda fatia 320 pode corresponder ao período de liberação, `Flush_time`, para a primeira fatia 315.

[0075] Em uma modalidade, os bits podem ser removidos do armazenamento temporário 150 em uma taxa constante após `Init_enc_delay`. Ao fazer isso, pode ser assegurado que o número de bits no armazenamento temporário 150 no fim da fatia é igual a `maxBufferBitsAtSliceEnd`. Ademais, os bits podem ser removidos do armazenamento temporário 150 em uma taxa constante durante o `Flush_time`.

Os blocos compactados da segunda fatia 320 são removidos do armazenamento temporário 150 após o período `Init_enc_delay` para a segunda fatia 320 terminar.

CÁLCULO DE BF

[0076] Dado que `maxBufferBitsAtSliceEnd` é menor do que `BufferMaxSize`, é vantajoso ajustar os cálculos de BF com base em quanto de uma fatia foi criptado. A Figura 4 ilustra um método exemplificativo para ajustar BF durante a codificação de uma fatia. Para propósitos ilustrativos, a Figura 4 será descrita com referência ao armazenamento temporário de taxa 150 do encriptador de vídeo 20 na Figura 2A; entretanto, será compreendido que os componentes da Figura 4 também são aplicáveis ao decodificador de vídeo 30 da Figura 2B ou componente(s) do mesmo, incluindo, sem limitação, o armazenamento temporário de taxa 155.

[0077] A fim de calcular BF, uma variável, `tempBufferMaxSize`, pode ser definida, que é indicativa do número máximo temporário de bits para o armazenamento temporário 150 em um ponto específico de uma fatia. Por exemplo, no fim de uma fatia, `tempBufferMaxSize` é igual a `maxBufferBitsAtSliceEnd`. Ainda outra variável, `Deslocamento`, é indicativo de quanto de `BufferMaxSize` está indisponível em um dado ponto no tempo.

[0078] O gráfico na Figura 4 mostra `Deslocamento` 405 ao longo de seu eixo geométrico vertical e a posição (isto é, número) de cada bloco na fatia ao longo de seu eixo geométrico horizontal. Devido ao fato de que os blocos podem ser codificados em ordem, a posição de cada bloco individual pode ser equivalente ao número de blocos codificado (doravante, `numBlocksCoded`). Por exemplo, o valor de `numBlocksCoded` pode variar de 0 ao número total de blocos na fatia, denotado N_T (isto é, `#blocksInSlice`). Em uma modalidade, um número de blocos, denotado N_P , pode ser

codificado antes de BF ser ajustado. Isto é, enquanto numBlocksCoded é menor do que ou igual a N_P , Deslocamento 405 é igual a zero e tempBufferMaxSize é igual a BufferMaxSize, conforme indicado em 420. Em uma modalidade, N_P pode fornecer uma posição limite, de modo que os blocos tenham uma posição no ou abaixo de p pode ter um tempBufferMaxSize igual a BufferMaxSize, enquanto os blocos que têm uma posição acima N_P pode ter um tempBufferMaxSize que é menor do que BufferMaxSize. Conforme mostrado na Figura 4, o valor de tempBufferMaxSize pode reduzir de modo linear com cada bloco subsequente acima N_P . Isto é, um bloco que tem uma posição após N_P pode ter um respectivo tempBufferMaxSize exclusivo que é maior do que o tempBufferMaxSize de quaisquer blocos subsequentes na fatia. Em outras palavras, os blocos que têm uma posição após N_P podem ter um tempBufferMaxSize que é igual a um valor ajustável que diminui para cada bloco entre N_P e N_T , conforme ilustrado na Figura 4.

[0079] Com referência contínua à Figura 4, o número total de blocos em uma fatia exemplificativa é denotado como N_T . Portanto, quando numBlocksCoded for igual a N_T , tempBufferMaxSize deve ser igual ao maxBufferBitsAtSliceEnd; dito de outra forma, Deslocamento 405 deve ser igual a MaxBufferSize menos maxBufferBitsAtSliceEnd. Por exemplo, se maxBufferBitsAtSliceEnd for 4128, tempBufferMaxSize deve ser igual a 4128 em N_T e Deslocamento 405 deve ser igual a MaxBufferSize menos 4128, conforme indicado em 426.

[0080] A fim de alcançar isso, tempBufferMaxSize e Deslocamento 405 podem ser ajustados em uma taxa constante, denotada B_R , entre numBlocksCoded = N_P e numBlocksCoded = N_T . A diferença entre N_P e N_T é denotada como N_R na Figura 3. B_R forma uma linha entre os pontos

(nuniBlocksCoded = N_P , Deslocamento = 0) e (nuniBlocksCoded = N_T , Deslocamento = bufferMaxSize menos maxBufferBitsAtSliceEnd). Por exemplo, se maxBufferBitsAtSliceEnd for 4128, B_R pode ser definido como $(\text{maxBufferSize} - 4128)/N_R$. Os valores do Deslocamento 405 e tempBufferMaxSize para pontos exemplificativos junto a B_R são ilustrados em 422 e 424.

[0081] Desta forma, pontos junto a B_R podem fornecer tempBufferMaxSize correspondente para blocos individuais de uma fatia. O tempBufferMaxSize correspondente pode, então, ser usado para calcular um BF individual para cada bloco na fatia. Em uma modalidade, os blocos em uma fatia podem ser definidos com base em suas posições dentro da fatia.

[0082] Em uma modalidade, as Equações 1 e 2, acima, podem ser usadas para calcular BF para blocos dentro de uma fatia que estão posicionados no ou abaixo de N_P . Por exemplo, para N_P igual a 2, BF pode ser calculado de acordo com as Equações 1 e 2 para o primeiro e o segundo blocos. Para quaisquer blocos restantes, BF pode ser calculado substituindo-se tempBufferMaxSize por BufferMaxSize na Equação 1 e/ou 2. Isto é, BF pode ser calculado com o uso de:

$$BF = ((\text{BufferCurrentSize} * 100) / \text{tempBufferMaxSize})$$

(Equação 3)

[0083] ou, com o uso de:

$$BF = (((\text{BufferCurrentSize} * 100) + (\text{tempBufferMaxSize} \gg 1)) / \text{tempBufferMaxSize}) \text{ (Equação 4)}$$

[0084] Além disso, tempBufferMaxSize pode ser calculado com o uso de:

$\text{tempBufferMaxSize} = \text{BufferMaxSize} - \text{Deslocamento}$ (Equação 5),

[0085] em que

$\text{Deslocamento} = B_R * (\text{numBlocksCoded} - N_P + 1)$ (Equação 6)

[0086] e em que

$B_R = (\text{BufferMaxSize} - \text{tempBufferMaxSize}) / (\text{numBlocksCoded} - N_P)$ (Equação 7)

[0087] Para resumir, o seguinte código exemplificativo pode ser usado no cálculo de BF:

```
numBlocksCoded = 0; tempBufferMaxSize =
BufferMaxSize;
BR = (BufferMaxSize - maxBufferBitsAtSliceEnd) /
(NT - NP);
for i = 1 : NT
{
code i* block
numBlocksCoded += 1;
if(numBlocksCoded >= NP)
{
tempBufferMaxSize = BufferMaxSize - (BR*
(numBlocksCoded - NP + 1));
}
calculate BF
}
```

FLUXOGRAMA EXEMPLIFICATIVO PARA FORNECER CB_R

[0088] Com referência à Figura 5, um procedimento exemplificativo para determinar a plenitude armazenamento temporário em um armazenamento temporário (por exemplo, o armazenamento temporário de taxa 150 no

encriptador de vídeo 20 na Figura 2A, ou o armazenamento temporário de taxa 155 do decodificador de vídeo 30 na Figura 2B) será descrito. Para propósitos ilustrativos, as etapas da Figura 5 serão descritas com referência ao armazenamento temporário de taxa 150 do encriptador de vídeo 20 na Figura 2A. Entretanto, as etapas ilustradas na Figura 5 podem ser realizadas por um encriptador de vídeo (por exemplo, o encriptador de vídeo 20 na Figura 2A), um decodificador de vídeo (por exemplo, o decodificador de vídeo 30 na Figura 2B), ou componente (ou componentes) do mesmo. Por conveniência, o método 500 é descrito como realizado por um codificador de vídeo (também simplesmente denominado como codificador), que pode ser o encriptador de vídeo 20, o decodificador de vídeo 30 ou outro componente.

[0089] O método 500 se inicia no bloco 501. No bloco 505, o processo 500 pode envolver codificar um bloco de dados dentro de uma fatia de dados. Em uma modalidade, a codificação pode ser realizada por um processador. Conforme descrito acima, uma fatia pode ser qualquer região espacialmente distinta em uma imagem ou um quadro que pode ser decodificado independentemente sem o uso de informações do restante das regiões na imagem ou quadro. A fatia pode ser composta de um ou mais blocos e/ou pixels.

[0090] No bloco 510, o processo 500 pode envolver armazenar um ou mais bits correspondentes ao bloco em um armazenamento temporário de taxa 150. Por exemplo, blocos podem ser compactados e os bits correspondentes podem ser temporariamente armazenados no armazenamento temporário de taxa 150. Em uma modalidade, os bits individuais podem ser adicionados ao armazenamento temporário 150 após ser codificados. Em uma modalidade alternativa, múltiplos bits podem ser adicionados ao armazenamento temporário 150 simultaneamente.

[0091] No bloco 515, o processo 500 pode envolver determinar informações indicativas de uma posição dos um ou mais blocos em relação a outros blocos na fatia. Por exemplo, pode ser determinado que um dado bloco é um terceiro bloco codificado, e tal bloco pode ser definido como o terceiro bloco da fatia. Determinando-se a posição do bloco, o número de blocos anteriormente codificados na fatia e/ou o número de blocos na fatia ainda a ser codificada pode ser determinado. Em uma modalidade, a determinação de informações indicativas da posição pode ser realizada por um processador.

[0092] No bloco 520, o processo 500 pode envolver a determinação de BF em uma ou mais posições de bloco. Essa determinação pode ser usada para controle de taxa. Por exemplo, com referência à Figura 4, o posicionamento de um bloco pode indicar o ponto junto a B_R que é associado ao bloco. A fim de determinar isso, pode ser necessário saber $\text{maxBufferBitsAtSliceEnd}$, que pode ser usado no cálculo de B_R . A posição associada junto a B_R pode indicar o tempBufferMaxSize correspondente para o bloco. Então, o BF pode ser calculado dividindo-se BufferCurrentSize pelo tempBufferMaxSize para o bloco. Em uma modalidade, a determinação de BF pode ser realizada por um processador. O processo 500 termina no bloco 525.

[0093] Em uma modalidade, tempBufferMaxSize pode ser ajustado por um valor inteiro, c . Por exemplo, tempBufferMaxSize pode ser calculado com o uso de:

$$\text{tempBufferMaxSize} = \text{BufferMaxSize} - (B_R * (\text{numBlocksCoded} - N_P + c)) \quad (\text{Equação 8})$$

[0094] Em um exemplo, c pode ser igual a 0, 1, -1 ou qualquer outro valor inteiro. Em outra modalidade, B_R pode ser ajustado por um valor inteiro, d . Por exemplo, B_R

pode ser calculado com o uso de:

$$B_R = (\text{BufferMaxSize} - \text{maxBufferBitsAtSliceEnd}) / (N_T - N_P + d)$$

(Equação 9)

[0095] em que d pode ser igual a 0, 1, -1, ou qualquer outro valor inteiro. Ajustes semelhantes às outras variáveis das equações acima também podem ser feitos.

OUTRAS CONSIDERAÇÕES

[0096] As informações e os sinais revelados podem ser representados com uso de qualquer uma dentre uma variedade de tecnologias e conjuntos de procedimentos diferentes. Por exemplo, dados, instruções, comandos, informações, sinais, bits, símbolos e chips que podem ser referenciados por toda a descrição acima podem ser representados por tensões, correntes, ondas eletromagnéticas, partículas ou campos magnéticos, partículas ou campos ópticos ou qualquer combinação dos mesmos.

[0097] Os vários blocos lógicos ilustrativos e etapas de algoritmos descritos em conexão com as modalidades reveladas no presente documento podem ser implantados como hardware eletrônico, software de computador ou combinações de ambos. A fim de ilustrar claramente essa intercambialidade de hardware e software, vários componentes, blocos, e etapas ilustrativos foram descritos acima de modo geral no que diz respeito à sua funcionalidade. Se tal funcionalidade é implantada como hardware ou software depende da aplicação particular e das restrições de projeto impostas sobre o sistema geral. Os indivíduos versados na técnica podem implantar a funcionalidade descrita em modos variáveis para cada aplicação em particular, porém tais decisões de implantação não devem ser interpretadas como causadoras de um afastamento do escopo da presente revelação.

[0098] Os conjuntos de procedimentos descritos no presente documento podem ser implantados em hardware, software, firmware ou em qualquer combinação dos mesmos. Tais conjuntos de procedimentos podem ser implantados em qualquer um dentre uma variedade de dispositivos, tais como computadores de propósito geral, monofones de dispositivo de comunicação sem fio ou dispositivos de circuito integrado que têm múltiplos usos incluindo a aplicação em monofones de dispositivo de comunicação sem fio e outros dispositivos. Quaisquer recursos descritos como dispositivos ou componentes podem ser implantados juntos em um dispositivo lógico integrado ou separadamente como dispositivos lógicos distintos, porém, interoperacionais. Caso sejam implantadas em software, os conjuntos de procedimentos podem ser realizados pelo menos em parte por uma mídia de armazenamento de dados legível por computador que compreende código de programa incluindo instruções que, quando executadas, realizam um ou mais dos métodos descritos acima. A mídia de armazenamento de dados legível por computador pode fazer parte de um produto de programa de computador, que pode incluir materiais de empacotamento. A mídia legível por computador pode compreender mídias de armazenamento de dados ou memória, tais como memória de acesso aleatório (RAM), tal como memória de acesso aleatório dinâmica síncrona (SDRAM), memória de somente leitura (ROM), memória de acesso aleatório não volátil (NVRAM), memória de somente leitura programável eletricamente apagável (EEPROM), memória Flash, mídias de armazenamento de dados magnéticos ou ópticos e similares. Os conjuntos de procedimentos, adicional ou alternativamente, podem ser realizados pelo menos em parte por uma mídia de comunicação legível por computador que transporta ou comunica o código de programa na forma de instruções ou estruturas de dados e que podem ser acessadas, lidas e/ou

executadas por um computador, tal como ondas ou sinais propagados.

[0099] O código de programa pode ser executado por um processador, que pode incluir um ou mais processadores, tais como um ou mais processadores de sinal digital (DSPs), microprocessadores de propósito geral, circuitos integrados específicos para aplicação (ASICs), matrizes lógicas programáveis em campo (FPGAs) ou outro conjunto de circuitos lógico discreto ou integrado equivalente. Tal processador pode ser configurado para realizar qualquer um dos conjuntos de procedimentos descritos nesta revelação. Um processador de propósito geral pode ser um microprocessador, porém alternativamente, o processador pode ser qualquer processador, controlador, microcontrolador ou máquina de estado convencional. Um processador também pode ser implantado como uma combinação de dispositivos de computação, por exemplo, uma combinação de um DSP e um microprocessador, uma pluralidade de microprocessadores, um ou mais microprocessadores em combinação com um núcleo de DSP ou qualquer outra tal configuração. Consequentemente, o termo "processador", conforme usado no presente documento pode se referir a qualquer um dentre a estrutura anterior, qualquer combinação da estrutura anterior ou qualquer outra estrutura ou aparelho adequado para implantar os conjuntos de procedimentos descritos no presente documento. Além disso, em alguns aspectos, a funcionalidade descrita no presente documento pode ser fornecida dentro de software ou hardware dedicados configurados para criptar e decodificar ou incorporados em um encriptador-decodificador de vídeo combinado (CODEC). Além disso, os conjuntos de procedimentos podem ser totalmente implantados em um ou mais circuitos ou elementos lógicos.

[0100] Os conjuntos de procedimentos desta revelação podem ser implantados em uma ampla variedade de dispositivos ou aparelhos, incluindo um monofone sem fio, um circuito integrado (IC) ou um conjunto de ICs (por exemplo, um conjunto de chips). Vários componentes ou unidades são descritos nesta revelação para enfatizar os aspectos funcionais dos dispositivos configurados para realizar os conjuntos de procedimentos revelados, 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 interoperacionais, que incluem um ou mais processadores conforme descrito acima, em conjunto com software e/ou firmware adequados.

[0101] Embora o supracitado tenha sido descrito em conexão com várias modalidades, recursos ou elementos diferentes de uma modalidade podem ser combinados com outras modalidades sem se afastar dos ensinamentos desta revelação. Entretanto, as combinações de recursos entre as respectivas modalidades não são necessariamente limitadas à mesma. Várias modalidades da revelação foram descritas. Essas e outras modalidades estão abrangidas no escopo das reivindicações a seguir.

REIVINDICAÇÕES

1. Método (500) para a compactação de vídeo de enlace de exibição **caracterizado** por compreender:

codificar (505) um bloco dentro de uma fatia de dados de vídeo, sendo que a fatia compreende um ou mais blocos dos dados de vídeo;

armazenar (510) bits correspondentes ao bloco em um armazenamento temporário;

determinar um número de bits armazenados no armazenamento temporário (*BufferCurrentSize*);

determinar uma posição limite (N_P) dentro da fatia, em que cada bloco codificado antes de atingir a posição limite (N_P) está associado a um número máximo temporário de bits que é igual a um primeiro valor e cada bloco codificado após atingir a posição limite (N_P) está associado a um número máximo temporário de bits que é igual a um valor ajustável, sendo que o valor ajustável é menor do que o primeiro valor;

determinar (515) informações indicativas de uma posição do bloco dentro da fatia e um número máximo temporário de bits associados ao bloco (*tempBufferMaxSize*);
e

determinar (520) a plenitude do armazenamento temporário com base, pelo menos em parte, na posição do bloco dentro da fatia, em que determinar a plenitude do armazenamento temporário compreende dividir o número de bits armazenados no armazenamento temporário (*BufferCurrentSize*) pelo número máximo temporário de bits associados ao bloco (*tempBufferMaxSize*).

2. Método, de acordo com a reivindicação 1, **caracterizado** por compreender adicionalmente determinar uma capacidade máxima do armazenamento temporário e um número máximo desejado de bits a serem armazenados no armazenamento temporário após a fatia ser completamente codificada;

em que o primeiro valor é igual à capacidade máxima do armazenamento temporário.

3. Método, de acordo com a reivindicação 2, **caracterizado** por compreender adicionalmente, com base no número máximo desejado de bits a serem armazenados no armazenamento temporário após a fatia ser completamente codificada, diminuir o valor ajustável para cada bloco codificado após atingir a posição limite.

4. Método, de acordo com a reivindicação 1, **caracterizado** pela determinação das informações indicativas da posição do bloco dentro da fatia compreender determinar um número de blocos na fatia que foram codificados.

5. Dispositivo de codificação de vídeo (20) **caracterizado** por compreender:

meios para codificar um bloco dentro de uma fatia de dados de vídeo, sendo que a fatia compreende um ou mais blocos dos dados de vídeo;

meios para armazenar bits correspondentes ao bloco em um armazenamento temporário;

meios para determinar um número de bits armazenados no armazenamento temporário (*BufferCurrentSize*);

meios para determinar uma posição limite (N_P) dentro da fatia, em que cada bloco codificado antes de atingir a posição limite (N_P) está associado a um número máximo temporário de bits que é igual a um primeiro valor e cada bloco codificado após atingir a posição limite (N_P) está associado a um número máximo temporário de bits que é igual a um valor ajustável, sendo que o valor ajustável é menor do que o primeiro valor;

meios para determinar informações indicativas de uma posição do bloco dentro da fatia e um número máximo temporário de bits associados ao bloco (*tempBufferMaxSize*);
e

meios para determinar a plenitude do armazenamento temporário com base, pelo menos em parte, na posição do bloco dentro da fatia, em que determinar a plenitude do armazenamento temporário compreende dividir o número de bits armazenados no armazenamento temporário (*BufferCurrentSize*) pelo número máximo temporário de bits associados ao bloco (*tempBufferMaxSize*).

6. Dispositivo de codificação de vídeo, de acordo com a reivindicação 5, **caracterizado** por compreender adicionalmente meios para determinar uma capacidade máxima do armazenamento temporário e um número máximo desejado de bits a serem armazenados no armazenamento temporário após a fatia ser completamente codificada;

em que o primeiro valor é igual à capacidade máxima do armazenamento temporário.

7. Dispositivo de codificação de vídeo, de acordo com a reivindicação 6, **caracterizado** por compreender adicionalmente meios para diminuir o valor ajustável para cada bloco codificado após atingir a posição limite com base no número máximo desejado de bits a serem armazenados no armazenamento temporário após a fatia ser completamente codificada.

8. Dispositivo de codificação de vídeo, de acordo com a reivindicação 5, **caracterizado** pelos meios para determinar as informações indicativas da posição do bloco dentro da fatia compreenderem meios para determinar um número de blocos na fatia que foram codificados.

9. Memória legível por computador **caracterizada** por compreender instruções nela armazenadas que, quando executadas, fazem com que o computador realize o método conforme definido em qualquer uma das reivindicações 1 a 4.

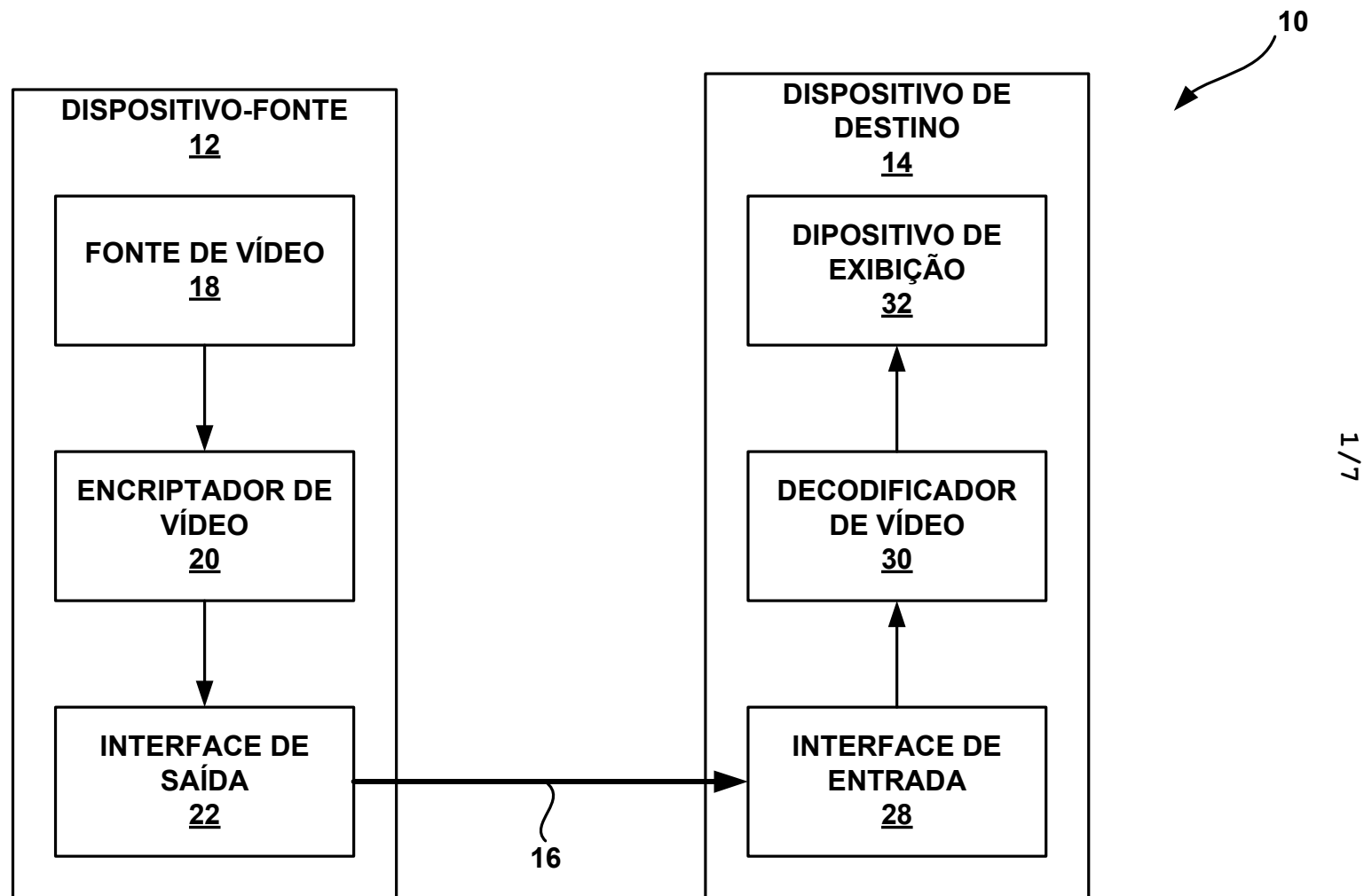


FIG. 1A

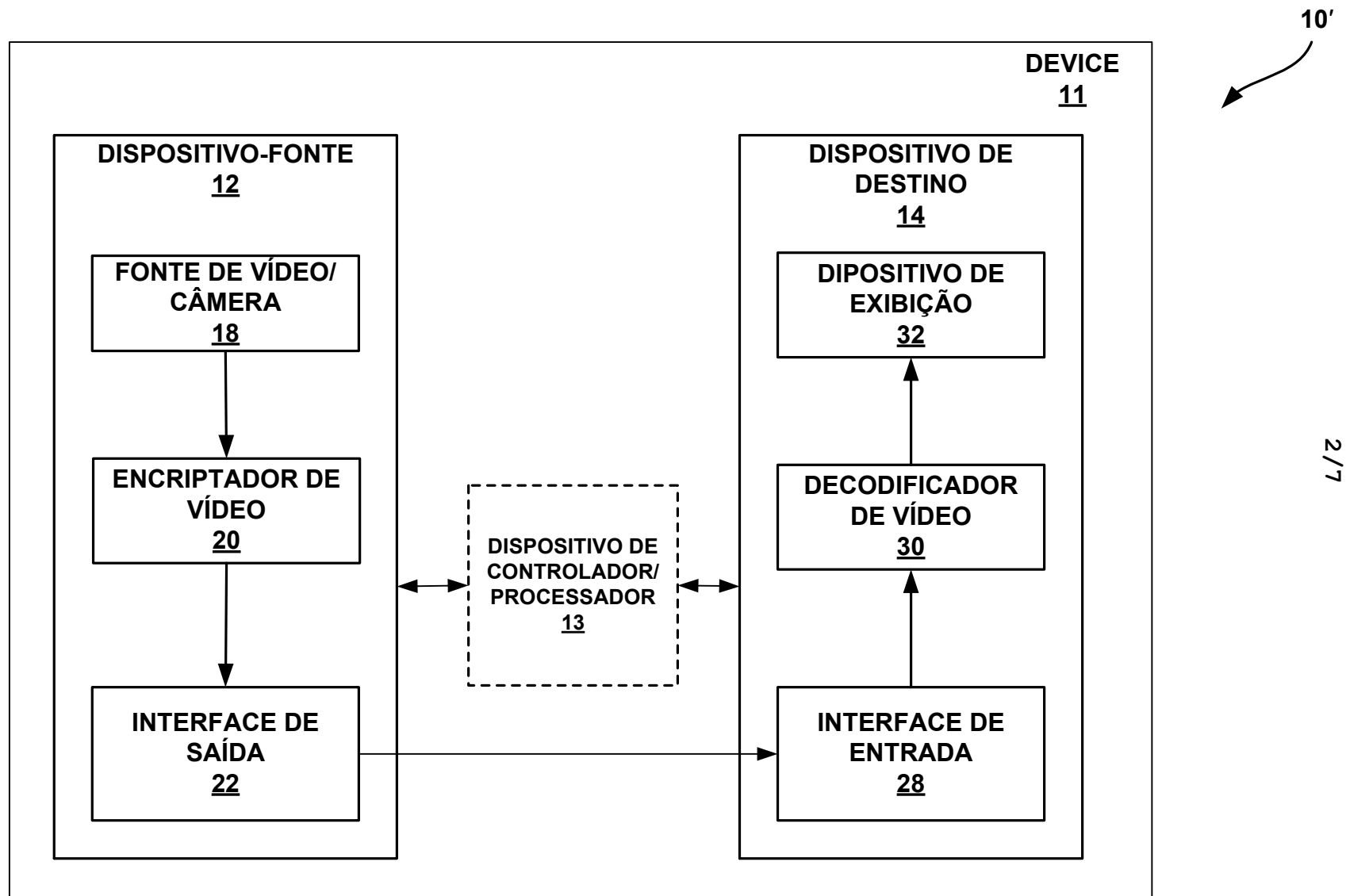


FIG. 1B

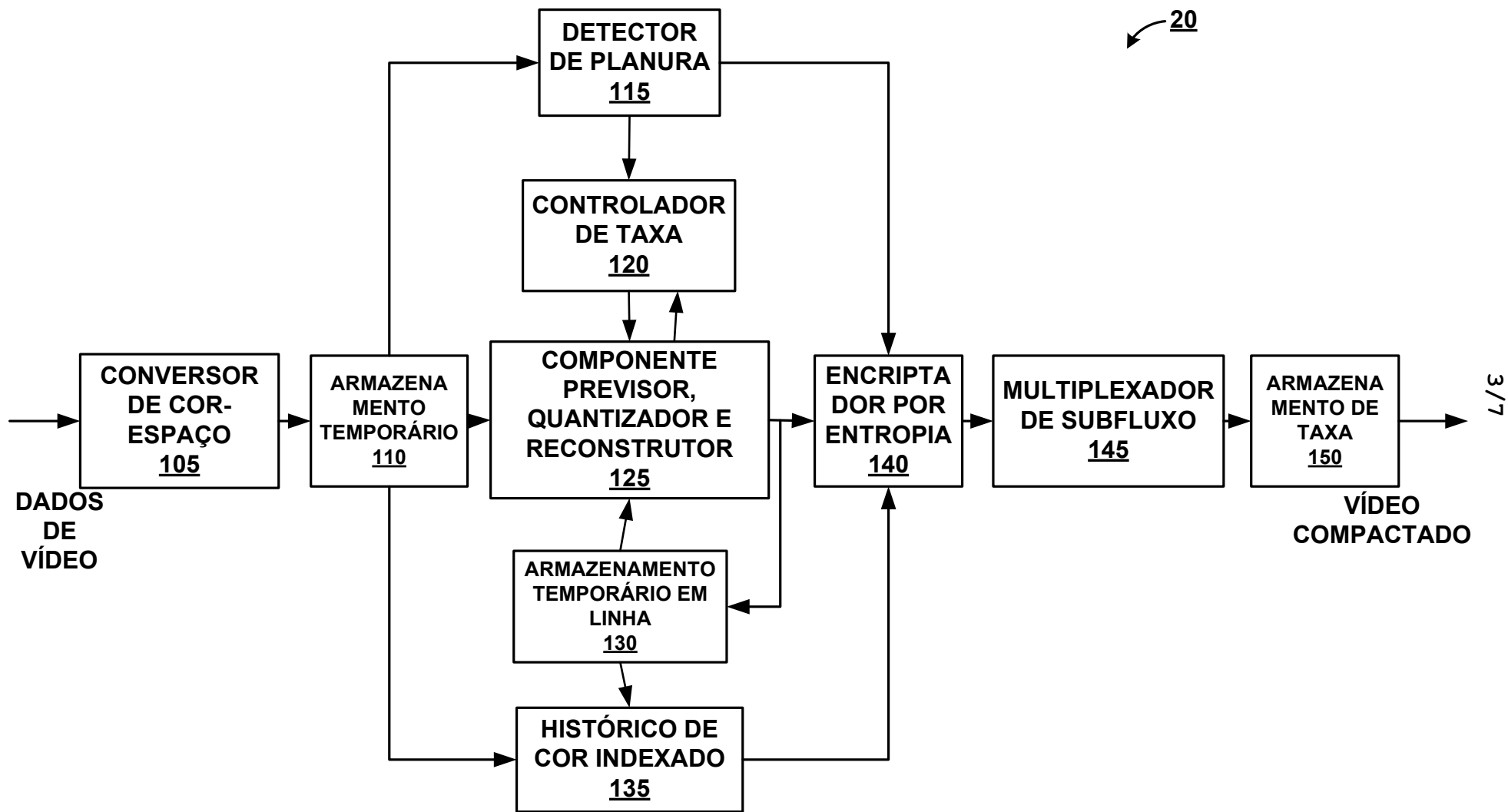


FIG. 2A

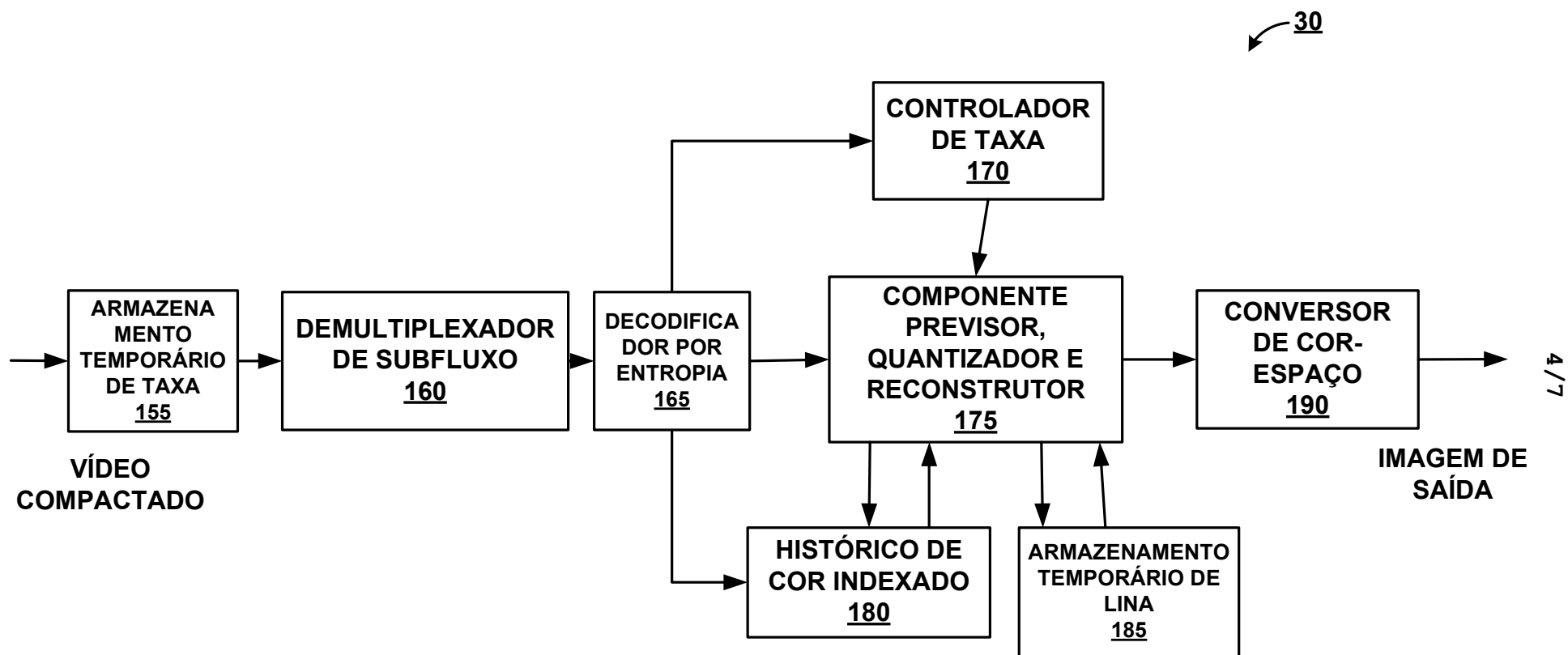


FIG. 2B

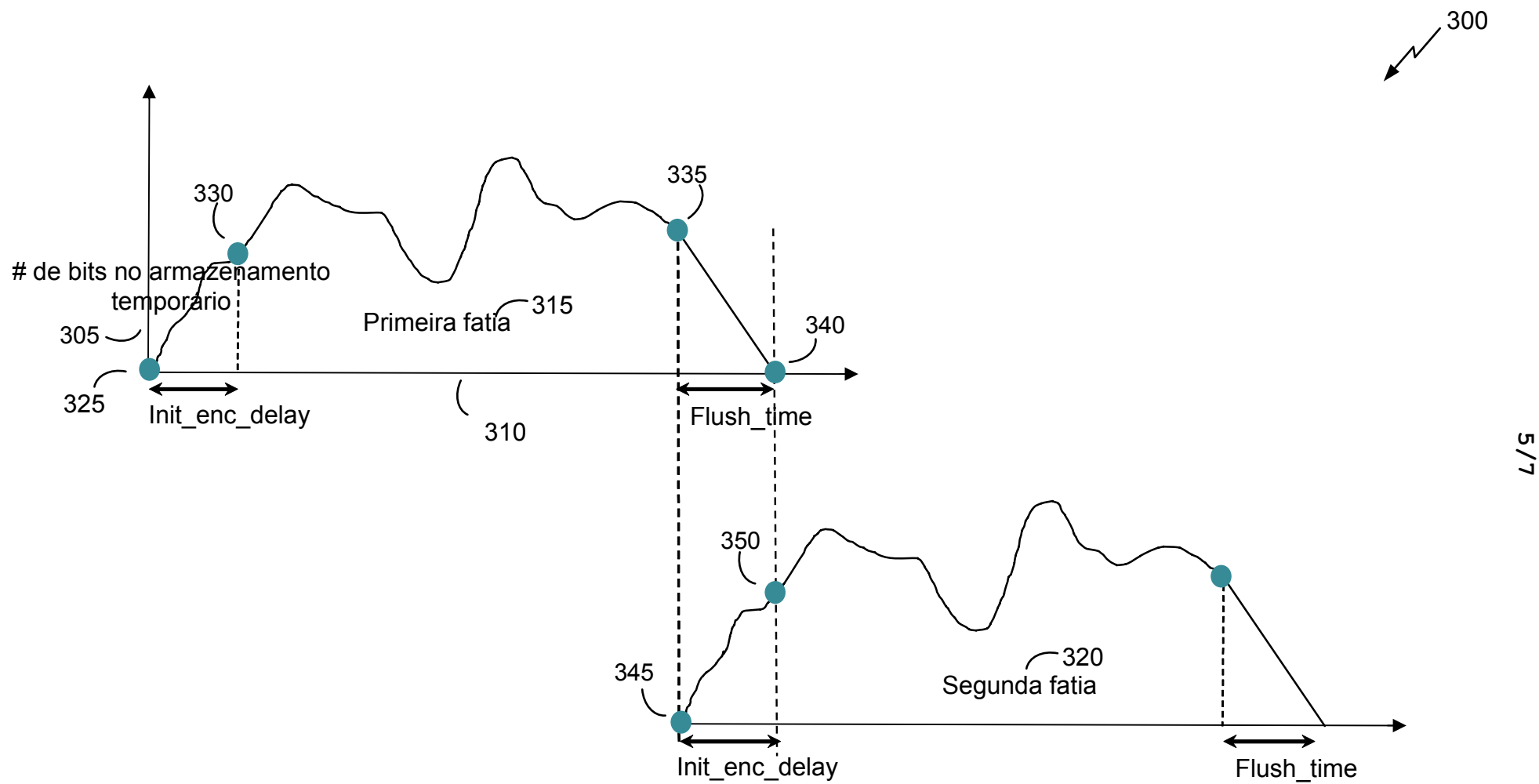


FIG. 3

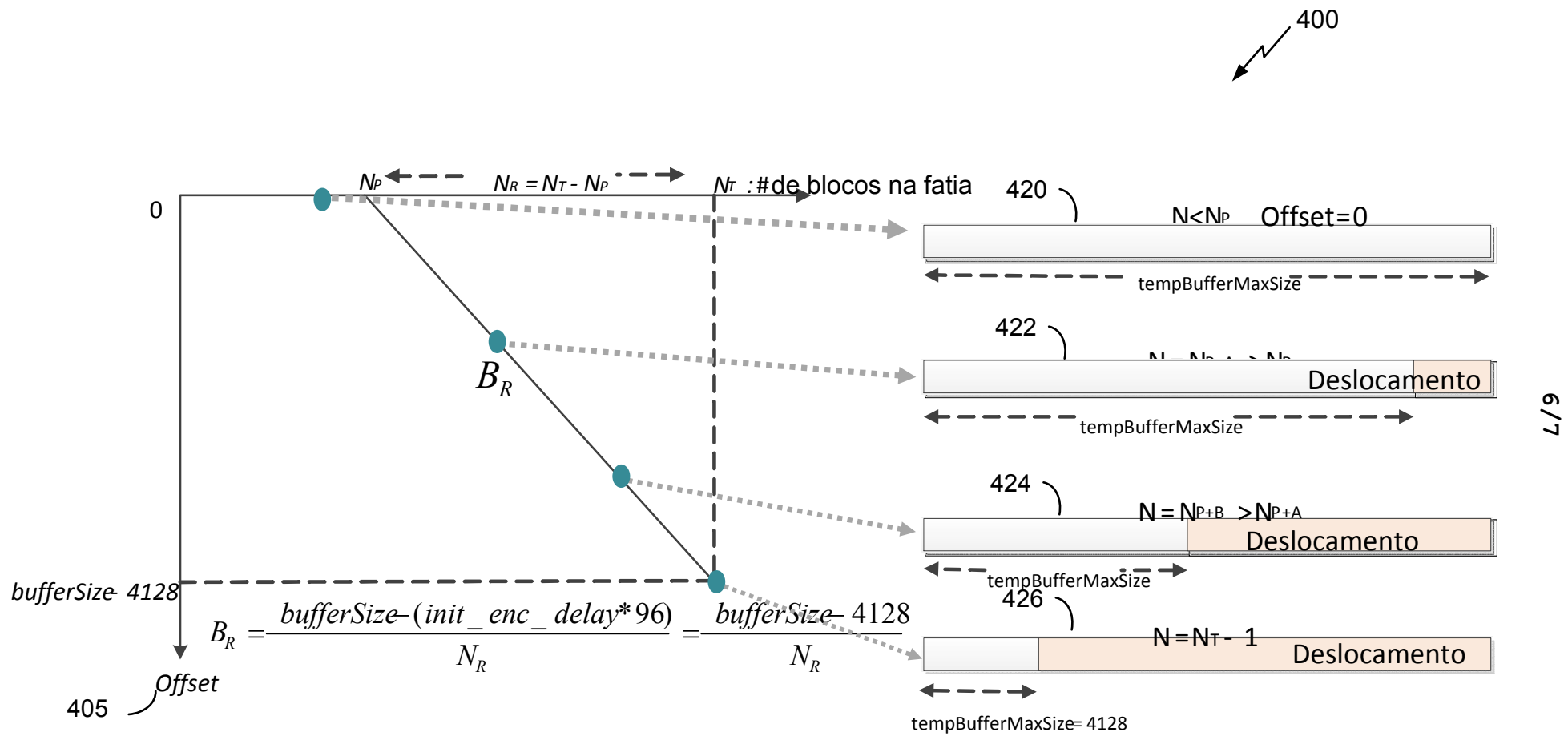
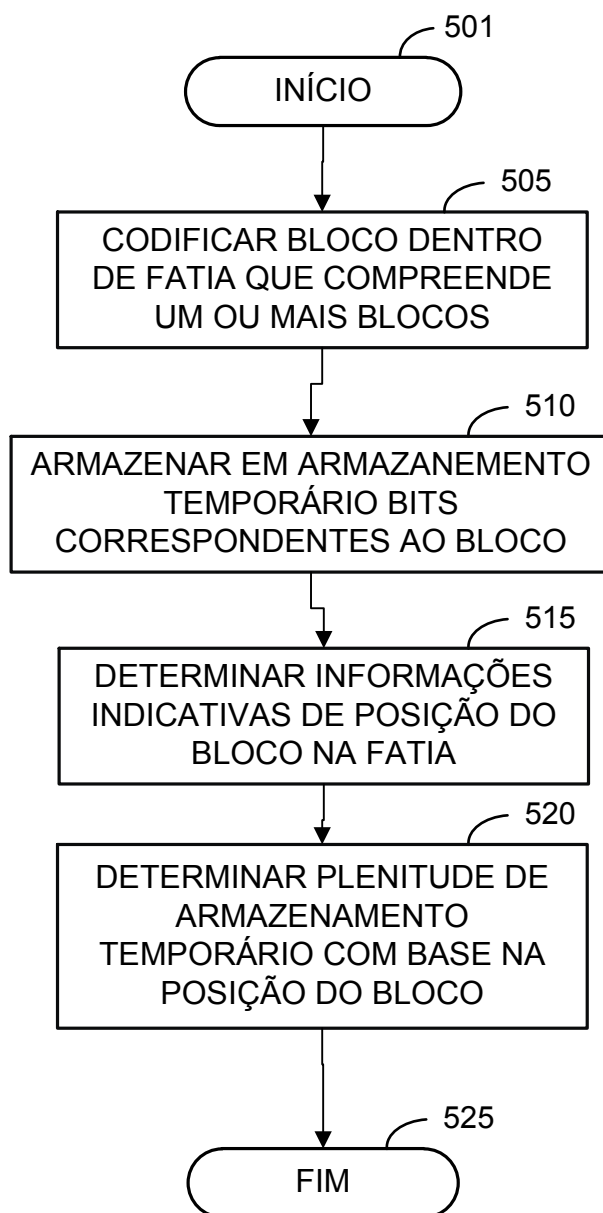


FIG. 4

500

**FIG. 5**