

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

(21) BR 112019019250-6 A2



* B R 1 1 2 0 1 9 0 1 9 2 5 0 A 2 *

(22) Data do Depósito: 21/03/2018

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

(54) Título: SINALIZAÇÃO DE INFORMAÇÕES DE VÍDEO SUPLEMENTARES ESSENCIAIS E NÃO ESSENCIAIS

(51) Int. Cl.: H04N 19/70; H04N 21/854.

(30) Prioridade Unionista: 20/03/2018 US 15/926,833; 21/03/2017 US 62/474,540.

(71) Depositante(es): QUALCOMM INCORPORATED.

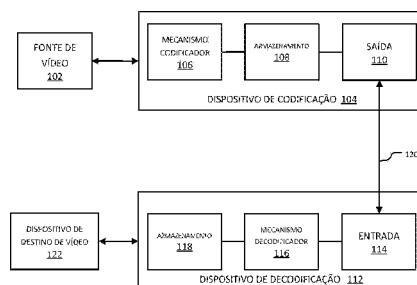
(72) Inventor(es): YE-KUI WANG; THOMAS STOCKHAMMER.

(86) Pedido PCT: PCT US2018023598 de 21/03/2018

(87) Publicação PCT: WO 2018/175609 de 27/09/2018

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

(57) Resumo: Em várias implementações, são providas técnicas, bem como sistemas que implementam essas técnicas, para mensagens SEI que distinguem informações essenciais de informações não essenciais. Em várias implementações, um codificador pode ser configurado para determinar se as informações associadas com dados de vídeo são essenciais ou não são essenciais. Em várias implementações, diferentes tipos de mensagens SEI podem ser definidos, em que um ou mais desses tipos de mensagens SEI podem indicar se as informações associadas com um vídeo particular são essenciais ou não essenciais. Informações essenciais são necessárias para apresentar os dados de vídeo, e não são necessárias para decodificar os dados de vídeo.



**"SINALIZAÇÃO DE INFORMAÇÕES DE VÍDEO SUPLEMENTARES
ESSENCIAIS E NÃO ESSENCIAIS"**

CAMPO

[0001] O presente pedido refere-se, de forma geral, à codificação de vídeo. Por exemplo, sistemas e métodos são descritos para codificação e decodificação de informações suplementares de aprimoramento.

FUNDAMENTO

[0002] 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 ou ISO/IEC MPEG-4 AVC, incluindo suas extensões de Codificação de Vídeo Escalonável (SVC) e Codificação de Vídeo Multivista (MVC), e Codificação de Vídeo de Alta Eficiência (HEVC), também conhecida como ITU-T H.265 e ISO/IEC 23008-2, incluindo sua extensão de codificação escalonável (isto é, Codificação de Vídeo de Alta Eficiência Escalonável, SHVC) e extensão multivista (isto é, Codificação de Vídeo de Alta Eficiência Multivista, MV-HEVC).

BREVE SUMÁRIO

[0003] Em várias implementações, são providas técnicas para sinalização de informações essenciais em mensagens SEI. As informações são essenciais quando um codificador determina que as informações são necessárias para exibição de dados de vídeo. As informações não são necessárias para decodificar os dados de vídeo. As informações relacionadas à exibição ou apresentação de dados de vídeo podem ser providas em mensagens SEI, no entanto, as mensagens SEI podem não ser capazes de indicar

se os dados transportados em cada mensagem são ou não essenciais para a apresentação dos dados de vídeo. Adicionalmente, padrões de codificação que usam mensagens SEI podem não prover uma forma para o dispositivo de decodificação rapidamente identificar e localizar mensagens SEI que incluem informações essenciais.

[0004] Em várias implementações, são providas mensagens SEI que podem indicar que informações essenciais são incluídas com um fluxo de bits de vídeo codificado. Essas mensagens SEI permitem que um dispositivo de decodificação determine que informações essenciais são providas, e localize as informações essenciais.

[0005] De acordo com pelo menos um exemplo, um método de codificação de dados de vídeo é provido, que inclui receber dados de vídeo. O método ainda inclui determinar, por um dispositivo de codificação, se as informações associadas com os dados de vídeo são necessárias para apresentar os dados de vídeo, em que as informações não são necessárias para decodificar os dados de vídeo. O método ainda inclui gerar uma mensagem de informações suplementares de aprimoramento (SEI), em que a mensagem SEI inclui um campo indicando se as informações são necessárias para apresentar os dados de vídeo. O método ainda inclui codificar os dados de vídeo. O método ainda inclui incluir a mensagem SEI com os dados de vídeo codificados.

[0006] Em outro exemplo, um codificador é provido, que inclui uma memória configurada para armazenar dados de vídeo e um processador. O processador é configurado para e pode determinar se as informações

associadas com os dados de vídeo são necessárias para apresentar os dados de vídeo, em que as informações não são necessárias para decodificar os dados de vídeo. O processador é configurado para e pode gerar uma mensagem de informações suplementares de aprimoramento (SEI), em que a mensagem SEI inclui um campo indicando se as informações são necessárias para apresentar os dados de vídeo. O processador é configurado para e pode codificar os dados de vídeo. O processador é configurado para e pode incluir a mensagem SEI com os dados de vídeo codificados.

[0007] Em outro exemplo, um meio legível por computador não transitório é provido compreendendo instruções que, quando executadas por um ou mais processadores, levam os um ou mais processadores a receber dados de vídeo. As instruções ainda fazem com que os um ou mais processadores determinem se as informações associadas com os dados de vídeo são necessárias para apresentar os dados de vídeo, em que as informações não são necessárias para decodificar os dados de vídeo. As instruções ainda fazem com que os um ou mais processadores gerem uma mensagem de informações suplementares de aprimoramento (SEI), em que a mensagem SEI inclui um campo indicando se as informações são necessárias para apresentar os dados de vídeo. As instruções ainda fazem com que os um ou mais processadores codifiquem os dados de vídeo. As instruções ainda fazem com que os um ou mais processadores incluam a mensagem SEI com os dados de vídeo codificados.

[0008] Em outro exemplo, um aparelho é provido que inclui meios para receber dados de vídeo. O aparelho ainda compreende meios para determinar se as informações

associadas com os dados de vídeo são necessárias para apresentar os dados de vídeo, em que as informações não são necessárias para decodificar os dados de vídeo. O aparelho ainda comprehende meios para gerar uma mensagem de informações suplementares de aprimoramento (SEI), em que a mensagem SEI inclui um campo indicando se as informações são necessárias para apresentar os dados de vídeo. O aparelho ainda comprehende meios para codificar os dados de vídeo. O aparelho ainda comprehende meios para incluir a mensagem SEI com os dados de vídeo codificados.

[0009] Em alguns aspectos, o dispositivo de codificação é configurado com um ou mais parâmetros que o dispositivo de codificação pode usar para determinar se as informações são necessárias para apresentar os dados de vídeo.

[0010] Em alguns aspectos, o dispositivo de codificação determina que as informações são necessárias para apresentar os dados de vídeo.

[0011] Em alguns aspectos, o dispositivo de codificação determina que as informações não são necessárias para apresentar os dados de vídeo.

[0012] Em alguns aspectos, um elemento de sintaxe da mensagem SEI indica um tipo da mensagem SEI.

[0013] Em alguns aspectos, as informações são incluídas com os dados de vídeo codificados.

[0014] Em alguns aspectos, os métodos, aparelhos e meio legível por computador descritos acima ainda comprehendem gravar os dados de vídeo codificados e a mensagem SEI em um arquivo, em que os dados de vídeo codificados e a mensagem SEI são gravados no arquivo de

acordo com um formato de arquivo. Os aspectos ainda compreendem usar o formato de arquivo para colocar a mensagem SEI no arquivo em que a SEI pode ser lida antes de os dados de vídeo codificados serem lidos.

[0015] Em alguns aspectos, os métodos, aparelhos e meio legível por computador descritos acima ainda compreendem encapsular os dados de vídeo codificados para streaming, em que os dados de vídeo codificados são encapsulados de acordo com um formato de streaming. Esses aspectos ainda compreendem gerar, de acordo com o formato de streaming, uma descrição de dados de vídeo codificados, em que a descrição inclui a mensagem SEI.

[0016] Em alguns aspectos, os métodos, aparelhos e meio legível por computador descritos acima ainda compreendem encapsular os dados de vídeo codificados para transmissão através de uma rede, em que os dados de vídeo codificados são encapsulados de acordo com um formato do recipiente. Esses aspectos ainda compreendem gerar, de acordo com o formato do recipiente, um primeiro pacote, em que uma porção de carga útil do primeiro pacote inclui a mensagem SEI como um descritor. Esses aspectos ainda compreendem gerar um ou mais pacotes incluindo os dados de vídeo codificados, em que os um ou mais pacotes seguem o primeiro pacote em ordem de transmissão.

[0017] Em alguns aspectos, os métodos, aparelhos e meio legível por computador descritos acima ainda compreendem encapsular os dados de vídeo codificados para transmissão através de uma rede, em que os dados de vídeo codificados são encapsulados de acordo com um protocolo de transferência em tempo real. Esses aspectos ainda

compreendem gerar, de acordo com um protocolo de descrição de sessão, uma descrição de sessão, em que a descrição de sessão inclui a mensagem SEI.

[0018] De acordo com pelo menos um exemplo, um método de decodificação de dados de vídeo é provido que inclui receber, em um dispositivo de decodificação, um fluxo de bits de vídeo codificado, uma mensagem de informações suplementares de aprimoramento (SEI) e as informações associadas com o fluxo de bits de vídeo codificado. O método ainda inclui determinar, usando a mensagem SEI, se as informações são necessárias para apresentar dados de vídeo no fluxo de bits de vídeo codificado, em que as informações não são necessárias para decodificar os dados de vídeo, e em que a mensagem SEI inclui um campo indicando se as informações são necessárias para apresentar os dados de vídeo. O método ainda inclui decodificar o fluxo de bits de vídeo codificado para produzir dados de vídeo, em que o fluxo de bits de vídeo codificado é decodificado de acordo com um resultado da determinação de se as informações são necessárias para apresentar os dados de vídeo.

[0019] Em outro exemplo, um aparelho é provido, que inclui uma memória configurada para armazenar dados de vídeo codificados, uma mensagem de informações suplementares de aprimoramento (SEI), e as informações associadas com o fluxo de bits de vídeo codificado, e um processador. O processador é configurado para e pode determinar, usando a mensagem SEI, se as informações são necessárias para apresentar dados de vídeo no fluxo de bits de vídeo codificado, em que as informações não são

necessárias para decodificar os dados de vídeo, e em que a mensagem SEI inclui um campo indicando se as informações são necessárias para apresentar os dados de vídeo. O processador é configurado para e pode decodificar o fluxo de bits de vídeo codificado para produzir dados de vídeo, em que o fluxo de bits de vídeo codificado é decodificado de acordo com um resultado da determinação de se as informações são necessárias para apresentar os dados de vídeo.

[0020] Em outro exemplo, um meio legível por computador não transitório é provido compreendendo instruções que, quando executadas por um ou mais processadores, levam os um ou mais processadores a receber um fluxo de bits de vídeo codificado, uma mensagem de informações suplementares de aprimoramento (SEI) e as informações associadas com o fluxo de bits de vídeo codificado. As instruções ainda fazem com que os um ou mais processadores determinem, usando a mensagem SEI, se as informações são necessárias para apresentar dados de vídeo no fluxo de bits de vídeo codificado, em que as informações não são necessárias para decodificar os dados de vídeo, e em que a mensagem SEI inclui um campo indicando se as informações são necessárias para apresentar os dados de vídeo. As instruções ainda fazem com que os um ou mais processadores decodifiquem o fluxo de bits de vídeo codificado para produzir dados de vídeo, em que o fluxo de bits de vídeo codificado é decodificado de acordo com um resultado da determinação de se as informações são necessárias para apresentar os dados de vídeo.

[0021] Em outro exemplo, um aparelho é provido, que inclui meios para receber um fluxo de bits de vídeo codificado, uma mensagem de informações suplementares de aprimoramento (SEI) e as informações associadas com o fluxo de bits de vídeo codificado. O aparelho ainda comprehende meios para determinar, usando a mensagem SEI, se as informações são necessárias para apresentar dados de vídeo no fluxo de bits de vídeo codificado, em que as informações não são necessárias para decodificar os dados de vídeo, e em que a mensagem SEI inclui um campo indicando se as informações são necessárias para apresentar os dados de vídeo. O aparelho ainda comprehende meios para decodificar o fluxo de bits de vídeo codificado para produzir dados de vídeo, em que o fluxo de bits de vídeo codificado é decodificado de acordo com um resultado da determinação de se as informações são necessárias para apresentar os dados de vídeo.

[0022] Em alguns aspectos, o dispositivo de decodificação determina que as informações são necessárias para apresentar os dados de vídeo, e em que o dispositivo de decodificação configura os dados de vídeo de acordo com as informações.

[0023] Em alguns aspectos, o dispositivo de decodificação determina que as informações não são necessárias para apresentar os dados de vídeo, e em que o dispositivo de decodificação decodifica os dados de vídeo codificados sem usar as informações.

[0024] Em alguns aspectos, um elemento de sintaxe da mensagem SEI indica um tipo da mensagem SEI.

[0025] Em alguns aspectos, o fluxo de bits de vídeo codificado, a mensagem SEI e as informações associadas com o fluxo de bits de vídeo codificado são recebidos em um arquivo, em que o arquivo é formatado de acordo com um formato de arquivo, e em que, de acordo com o formato de arquivo, a mensagem SEI está em uma parte do arquivo que é lida pelo dispositivo de decodificação antes de uma parte do arquivo que inclui o fluxo de bits de vídeo codificado.

[0026] Em alguns aspectos, o fluxo de bits de vídeo codificado, a mensagem SEI e as informações associadas com o fluxo de bits de vídeo codificado são recebidos em um fluxo de dados, em que o fluxo de dados é formatado de acordo com um formato de streaming. Esses aspectos ainda incluem fazer a leitura, a partir do fluxo de dados, de uma descrição dos dados de vídeo codificados, em que a descrição inclui a mensagem SEI.

[0027] Em alguns aspectos, o fluxo de bits de vídeo codificado, a mensagem SEI e as informações associadas com o fluxo de bits de vídeo codificado são recebidos em uma pluralidade de pacotes de rede. Esses aspectos ainda incluem fazer a leitura da mensagem SEI a partir de uma porção de carga útil de um primeiro pacote da pluralidade de pacotes de rede. Esses aspectos ainda incluem fazer a leitura dos dados de vídeo codificados a partir de um ou mais pacotes da pluralidade de pacotes de rede, em que os um ou mais pacotes seguem o primeiro pacote em ordem de transmissão.

[0028] Em alguns aspectos, o fluxo de bits de vídeo codificado, a mensagem SEI e as informações

associadas com o fluxo de bits de vídeo codificado são recebidos através de uma rede, em que o fluxo de bits de vídeo codificado é encapsulado de acordo com um protocolo de transferência em tempo real, em que a mensagem SEI é incluída em uma descrição de sessão, em que a descrição de sessão é formatada de acordo com um protocolo de descrição de sessão.

BREVE DESCRIÇÃO DOS DESENHOS

[0029] Exemplos ilustrativos do presente pedido são descritos em detalhes abaixo com referência às seguintes figuras de desenhos:

[0030] A Figura 1 é um diagrama de blocos ilustrando um exemplo de um sistema de codificação de vídeo.

[0031] A Figura 2 ilustra um exemplo de uma unidade de acesso.

[0032] A Figura 3 ilustra um exemplo de um arquivo de mídia base ISO.

[0033] A Figura 4 ilustra um exemplo de uma apresentação DASH para conteúdo de vídeo por streaming.

[0034] A Figura 5 ilustra um exemplo de um fluxo de transporte MPEG.

[0035] A Figura 6 ilustra um exemplo do modelo Interconexão de Sistemas Abertos (OSI) 600 para comunicações de rede.

[0036] A Figura 7 ilustra um exemplo de um processo para codificação de dados de vídeo.

[0037] A Figura 8 ilustra um exemplo de um processo para decodificação de dados de vídeo.

[0038] A Figura 9 é um diagrama de blocos ilustrando um dispositivo de codificação exemplificativo.

[0039] A Figura 10 é um diagrama de blocos ilustrando um dispositivo de decodificação exemplificativo.

DESCRIÇÃO DETALHADA

[0040] Determinados aspectos e exemplos da presente invenção são providos abaixo. Alguns desses aspectos e exemplos podem ser aplicados de forma independente e alguns deles podem ser aplicados em combinação, como seria evidente para os versados na técnica. Na descrição a seguir, para fins de explicação, detalhes específicos são apresentados a fim de fornecer um completo entendimento de várias implementações possíveis. No entanto, será evidente que vários exemplos podem ser praticados sem esses detalhes específicos. As figuras e a descrição não pretendem ser restritivas.

[0041] A descrição a seguir fornece apenas implementações exemplificativas, e não se destina a limitar o escopo, a aplicabilidade ou a configuração da invenção. Em vez disso, a descrição a seguir das implementações exemplificativas fornecerá aos versados na técnica uma descrição favorável para a implementação de um exemplo. Deve ser entendido que várias alterações podem ser feitas na função e disposição dos elementos sem se afastar do espírito e escopo do pedido, conforme estabelecido nas reivindicações anexas.

[0042] Detalhes específicos são fornecidos na descrição a seguir para fornecer um entendimento completo dos exemplos. No entanto, será entendido por um versado na técnica que os exemplos podem ser praticados sem esses

detalhes específicos. Por exemplo, circuitos, sistemas, redes, processos e outros componentes podem ser mostrados como componentes na forma de diagrama de blocos para não obscurecer os exemplos em detalhes desnecessários. Em outros casos, circuitos, processos, algoritmos, estruturas e técnicas bem conhecidos podem ser mostrados sem detalhes desnecessários, a fim de evitar obscurecer os exemplos.

[0043] Observa-se também que exemplos individuais podem ser descritos como um processo que é representado como um organograma, um fluxograma, um fluxograma de dados, um diagrama de estrutura ou um diagrama de blocos. Embora um fluxograma possa descrever as operações como um processo sequencial, muitas das operações podem ser executadas em paralelo ou simultaneamente. Além disso, a ordem das operações pode ser rearranjada. Um processo é encerrado quando suas operações são concluídas, mas pode haver etapas adicionais não incluídas em uma figura. Um processo pode corresponder a um método, uma função, um procedimento, uma sub-rotina, um subprograma etc. Quando um processo corresponde a uma função, seu término pode corresponder a um retorno da função à função de chamada ou à função principal.

[0044] O termo "meio legível por computador" inclui, mas sem limitação, dispositivos de armazenamento portáteis ou não portáteis, dispositivos de armazenamento ópticos e vários outros meios capazes de armazenar, conter ou transportar instruções e/ou dados. Um meio legível por computador pode incluir um meio não transitório no qual dados podem ser armazenados e que não incluem ondas portadoras e/ou sinais eletrônicos transitórios que se

propagam sem fio ou através de conexões com fio. Exemplos de um meio não transitório podem incluir, mas sem limitação, uma fita ou disco magnético, meio de armazenamento óptico, tal como disco compacto (CD) ou disco versátil digital (DVD), memória flash, memória ou dispositivos de memória. Um meio legível por computador pode ter nele armazenado código e/ou instruções executáveis por máquina que podem representar um procedimento, uma função, um subprograma, um programa, uma rotina, uma subrotina, um módulo, um pacote de software, uma classe ou qualquer combinação de instruções, estruturas de dados ou instruções de programa. Um segmento de código pode ser acoplado a outro segmento de código ou um circuito de hardware, passando e/ou recebendo informações, dados, argumentos, parâmetros ou conteúdos de memória. Informações, argumentos, parâmetros, dados etc. podem ser passados, encaminhados ou transmitidos por qualquer meio adequado, incluindo compartilhamento de memória, passagem de mensagem, passagem de token, transmissão de rede ou semelhantes.

[0045] Além disso, os exemplos discutidos aqui podem ser implementados por hardware, software, firmware, middleware, microcódigo, linguagens de descrição de hardware ou qualquer combinação dos mesmos. Quando implementado em software, firmware, middleware ou microcódigo, o código de programa ou segmentos de código para executar as tarefas necessárias (por exemplo, um produto de programa de computador) podem ser armazenados em um meio legível por computador ou legível por máquina. Um processador (es) pode executar as tarefas necessárias.

[0046] À medida que mais dispositivos e sistemas fornecem aos consumidores a capacidade de consumir dados de vídeo digitais, torna-se mais importante a necessidade de técnicas eficientes de codificação de vídeo. A codificação de vídeo é necessária para reduzir os requisitos de armazenamento e transmissão necessários para lidar com as grandes quantidades de dados presentes nos dados de vídeo digitais. Várias técnicas de codificação de vídeo podem ser usadas para compactar dados de vídeo em um formato que usa uma taxa de bits mais baixa enquanto mantém alta qualidade do vídeo. Como aqui utilizado, "codificação" refere-se à "codificação" e "decodificação".

[0047] Mensagens de Informações Suplementares de Aprimoramento (SEI) em um fluxo de bits codificado podem ser usadas para auxiliar na decodificação e/ou exibição do fluxo de bits codificado, ou para outra finalidade. As mensagens SEI são "suplementares", em que os dados em mensagens SEI não são necessários para a construção de amostras de luma ou croma durante o processo de decodificação. Além disso, os decodificadores que são compatíveis com um padrão de codificação de vídeo que suporta mensagens SEI não precisam processar mensagens SEI para serem compatíveis. Para alguns padrões de codificação, algumas informações de mensagem SEI podem ser necessárias para verificar a conformidade do fluxo de bits ou para emissão da conformidade do decodificador de tempo.

[0048] As mensagens SEI podem ser usadas para fornecer informações adicionais sobre um fluxo de bits codificado, que pode ser usado para alterar a apresentação do fluxo de bits quando o fluxo de bits é decodificado, ou

para fornecer informações a um decodificador. Por exemplo, as mensagens SEI foram usadas para fornecer informações de compactação de quadro (por exemplo, descrevendo a maneira com que os dados de vídeo são dispostos em um quadro de vídeo), descrições de conteúdo (por exemplo, para indicar que o fluxo de bits codificado é, por exemplo, vídeo 360 graus), e informações sobre cores (por exemplo, gama de cores e/ou faixa de cores), entre outras coisas.

[0049] Embora os padrões de codificação de vídeo forneçam que as mensagens SEI não são necessárias para a decodificação de um fluxo de bits codificado, em alguns casos, as informações transportadas nas mensagens SEI podem ser essenciais para o processamento da decodificação. Por exemplo, as mensagens SEI podem ser usadas para descrever o formato dos dados de vídeo codificados no fluxo de bits, sem os quais um decodificador pode não ser capaz de apresentar o vídeo como pretendido. Por exemplo, uma mensagem SEI pode ser usada para sinalizar para um decodificador que o fluxo de bits codificado inclui vídeo 360 graus. Neste exemplo, o decodificador pode usar essas informações para renderizar os dados de vídeo para uma apresentação 360 graus. Como alternativa, se o decodificador não for capaz de renderizar vídeo 360 graus, o decodificador pode usar essas informações para não renderizar os dados de vídeo, em vez de tentar renderizar os dados de vídeo e apresentar uma sequência de vídeo distorcida. Como outro exemplo, um gerador de conteúdo pode ter especificado que os dados de vídeo codificados sejam renderizados com uma gama de cores específica. Neste exemplo, sem os dados na mensagem SEI, o decodificador pode

renderizar o vídeo com uma gama de cores diferente e, assim, apresentar os dados de vídeo de maneira diferente da pretendida.

[0050] Em várias implementações, são providas técnicas, bem como sistemas que implementam essas técnicas, para mensagens SEI que distinguem informações "essenciais" de informações "não essenciais". Em várias implementações, um codificador pode ser configurado para determinar se as informações associadas com dados de vídeo são essenciais ou não são essenciais. Por exemplo, o codificador pode ser configurado com parâmetros que identificam informações essenciais. Os parâmetros podem indicar, para um vídeo particular, que informações, tais como um arranjo de compactação de quadro ou volume de cor ou outras informações, são essenciais para o vídeo particular.

[0051] Em várias implementações, diferentes tipos de mensagens SEI podem ser definidos, em que um ou mais desses tipos de mensagens SEI podem indicar se as informações associadas com um vídeo particular são essenciais ou não essenciais, por exemplo, conforme determinado por um dispositivo de codificação, para apresentar (por exemplo, a apresentação conforme pretendida pelo dispositivo de codificação) os dados de vídeo. Conforme provido neste documento, os termos "essencial" e "não essencial" são usados para distinguir informações que foram consideradas (por exemplo, por um dispositivo de codificação e/ou provedor/gerador de conteúdo) necessárias para apresentar um vídeo de maneira particular (por exemplo, a apresentação ou renderização pretendida) de informações que não são necessárias para a apresentação do

vídeo da maneira particular. Se as informações são essenciais ou não essenciais pode ser definido (por exemplo, determinado ou indicado) pelo gerador de conteúdo. Por exemplo, para alguns dados de vídeo, informações de compactação de quadro ou informações de conversão de cor podem ser consideradas essenciais pelo gerador de conteúdo, enquanto informações de camada podem ser não essenciais. Como outro exemplo, para alguns dados de vídeo, informações de filtragem podem ser essenciais, enquanto informações de temporização são não essenciais. Tanto informações essenciais quanto não essenciais não são necessárias para decodificação do vídeo de um fluxo de bits.

[0052] Em um exemplo, uma mensagem SEI manifesto e uma mensagem SEI prefixo são providas. Neste exemplo, a mensagem SEI manifesto inclui uma listagem de tipos de mensagens SEI que estão presentes ou ausentes em um fluxo de bits, e se qualquer das mensagens SEI no fluxo de bits inclui informações essenciais. Ainda neste exemplo, uma mensagem SEI prefixo pode incluir uma indicação de prefixo, que pode indicar um tipo da carga útil transportada na mensagem SEI prefixo. A carga útil pode incluir informações que podem ou não ser essenciais.

[0053] Em outro exemplo, uma mensagem SEI é definida para informações essenciais, que é referida como uma mensagem SEI de Informações Suplementares Essenciais (ESI), e uma mensagem SEI é definida para non-informações essenciais, que é referida como uma mensagem SEI de Informações Suplementares Não Essenciais (NSI). Usando a SEI ESI e SEI NSI, um padrão de codificação de vídeo pode suportar dois tipos diferentes de mensagens SEI, que podem

ser priorizadas por um decodificador que suporta esses tipos de mensagem SEI.

[0054] Em vários padrões de codificação de vídeo exemplificativos, mensagens SEI podem ser colocadas em determinadas partes de um fluxo de bits codificado, mas não precisam ser colocadas em nenhum ponto particular. Por exemplo, mensagens SEI não precisam ser colocadas no inicio do fluxo de bits codificado. Nesses exemplos, mensagens SEI que incluem informações essenciais podem não ser vistas por um decodificador antes de o decodificador começar a renderizar os dados de vídeo em um fluxo de bits codificado.

[0055] Em várias implementações, são providas técnicas, e sistemas que implementam essas técnicas, para uso de formatos de encapsulamento, tal que mensagens SEI possam ser vistas no inicio por um decodificador. Formatos de encapsulamento incluem, por exemplo, formato de arquivo de mídia base da Organização para Padronização (ISO) (ISOBMFF), Protocolo de Descrição de Sessão (SDP), Streaming Adaptativo Dinâmico sobre Protocolo de Transferência de Hipertexto (DASH), e Fluxo de Transporte do Grupo de Versados em Imagens em Movimento (MPEG-TS), entre outros. Em várias implementações, os formatos definidos por esses formatos de encapsulamento podem ser usados para colocar mensagens SEI, em que um decodificador pode encontrar as mensagens SEI no inicio do processo de decodificação fazendo a leitura dos dados de vídeo. Dessa forma, os decodificadores que suportam mensagens SEI para informações essenciais podem obter as informações providas

por essas mensagens antes do início da decodificação das imagens.

[0056] O usuário de mensagens SEI que podem indicar informações essenciais ou não essenciais pode melhorar a operação do decodificador. Por exemplo, sem mensagens SEI que podem sinalizar informações essenciais, alguns decodificadores podem tentar decodificar e renderizar um fluxo de bits sem antes determinar se o sistema é capaz de exibir o vídeo. Neste exemplo, o resultado pode ser que o vídeo, quando exibido, é muito distorcido ou de outra forma não é apresentado da maneira pretendida pelo gerador de conteúdo.

[0057] Em alguns exemplos, o decodificador pode, após ter iniciado a decodificação do vídeo, encontrar as informações essenciais, por exemplo, em uma mensagem SEI incluída no fluxo de bits. Nesses exemplos, alguns decodificadores podem ser capazes de corrigir a saída, e iniciar a exibição do vídeo conforme pretendido. Alguns decodificadores, no entanto, podem determinar, a partir da mensagem SEI, que o sistema não suporta vídeo, e não é capaz de exibir o vídeo. Nesses casos, o sistema pode ter que decidir se deve parar de decodificar e renderizar o vídeo ou encontrar uma versão do vídeo que o sistema é capaz de exibir. Se o sistema puder encontrar uma versão do vídeo que o sistema é capaz de exibir, então, o sistema pode ainda ter que decidir se deve iniciar a decodificação do vídeo do início, ou realizar operações de busca para iniciar próximo ao último ponto no tempo que o espectador assistiu. Em qualquer caso, o decodificador gasta tempo de processamento para iniciar. Adicionalmente, o tempo de

processamento gasto na decodificação do vídeo que poderia não ser adequadamente exibido é desperdiçado.

[0058] Uma solução para o problema de obtenção das informações essenciais é o decodificador pesquisar o fluxo de bits codificado para as informações essenciais. Esta solução, no entanto, pode exigir uma quantidade significativa de memória para que o decodificador armazene o fluxo de bits codificado, e pode causar um atraso significativo antes que o sistema possa começar a reproduzir o vídeo.

[0059] Em cada um dos exemplos acima, fornecendo informações essenciais de maneira a permitir que um decodificador rapidamente identifique e encontre informações essenciais, o decodificador pode evitar ter que procurar as informações essenciais, pode evitar decodificar um fluxo de bits que sistema não é capaz de exibir, e pode evitar que o sistema exiba o fluxo de bits decodificado de maneira diferente da pretendida. O decodificador pode ainda ser operado com mais eficiência, com menos tempo de processamento desperdiçado.

[0060] A Figura 1 é um diagrama de blocos ilustrando um exemplo de um sistema de codificação de vídeo 100 incluindo um dispositivo de codificação 104 e um dispositivo de decodificação 112. Em alguns exemplos, o dispositivo de codificação 100 do sistema de codificação de vídeo 100 pode gerar mensagens SEI ESI e/ou mensagens SEI NSI, e o dispositivo de decodificação 112 pode decodificar mensagens SEI ESI e/ou mensagens SEI NSI, e fazer uso das informações incluídas nessas mensagens. O dispositivo de codificação 104 pode ser parte de um dispositivo fonte, e o

dispositivo de decodificação 112 pode ser parte de um dispositivo receptor. O dispositivo fonte e/ou o dispositivo receptor pode incluir um dispositivo eletrônico, tal como um telefone móvel ou aparelho telefônico estacionário (por exemplo, smartphone, telefone celular ou semelhantes), um desktop, um laptop ou notebook, um tablet, uma set-top box, uma televisão, uma câmera, um dispositivo de exibição, um reproduutor de mídia digital, um console de videogame, um dispositivo de streaming de vídeo, uma câmera de Protocolo da Internet (IP), ou qualquer outro dispositivo eletrônico adequado. Em alguns exemplos, o dispositivo fonte e o dispositivo receptor podem incluir um ou mais transceptores sem fio para comunicações sem fio. As técnicas de codificação descritas neste documento são aplicáveis à codificação de vídeo em várias aplicações multimídia, incluindo transmissões de vídeo por streaming (por exemplo, através da Internet), broadcasts ou transmissões televisivas, codificação de vídeo digital para armazenamento em um meio de armazenamento de dados, decodificação de vídeo digital armazenado em um meio de armazenamento de dados, ou outras aplicações. Em alguns exemplos, o sistema 100 pode suportar transmissão de vídeo unidirecional ou bidirecional para suportar aplicações, tais como videoconferência, streaming de vídeo, reprodução de vídeo, broadcasting de vídeo, jogos e/ou videotelefonia.

[0061] O dispositivo de codificação 104 (ou codificador) pode ser usado para codificar dados de vídeo usando um padrão de codificação de vídeo ou protocolo para gerar um fluxo de bits de vídeo codificado. Exemplos de 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), incluindo suas extensões de Codificação de Vídeo Escalonável (SVC) e Codificação de Vídeo Multivista (MVC), e Codificação de Vídeo de Alta Eficiência (HEVC) ou ITU-T H.265. Várias extensões para HEVC lidar com codificação de vídeo multicamada existem, incluindo as extensões de codificação de conteúdo de faixa e tela, codificação de vídeo 3D (3D-HEVC) e extensões multivista (MV-HEVC) e extensão escalonável (SHVC). A HEVC e suas extensões foram desenvolvidas pela Equipe de Colaboração Conjunta de Codificação de Vídeo (JCT-VC), bem como Equipe de Colaboração Conjunta de Desenvolvimento de Extensão de Codificação de Vídeo 3D (JCT-3V) do Grupo de Versados em Codificação de Vídeo (VCEG) ITU-T e Grupo de Versados em Imagens em Movimento ISO/IEC (MPEG). VECG ITU-T e MPEG também formaram uma equipe de vídeo de exploração conjunta (JVET) para explorar novas ferramentas de codificação para a nova geração de padrão de codificação de vídeo. O software de referência é denominado JEM (modelo de exploração conjunta).

[0062] Muitos exemplos descritos neste documento fornecem exemplos usando o modelo JEM, o padrão HEVC e/ou suas extensões. No entanto, as técnicas e os sistemas descritos neste documento também podem ser aplicáveis a outros padrões de codificação, tais como AVC, MPEG, suas extensões ou outros padrões de codificação adequados atualmente existentes ou padrões de codificação futuros. Por conseguinte, embora as técnicas e sistemas descritos

neste documento possam ser descritos com referência a um padrão de codificação de vídeo particular, um versado na técnica apreciará que a descrição não deve ser interpretada para aplicação apenas a esse padrão particular.

[0063] Referindo-se à Figura 1, uma fonte de vídeo 102 pode fornecer dados de vídeo ao dispositivo de codificação 104. A fonte de vídeo 102 pode ser parte do dispositivo fonte ou pode ser parte de outro dispositivo que não o dispositivo fonte. A fonte de vídeo 102 pode incluir um dispositivo de captura de vídeo (por exemplo, uma câmera de vídeo, um telefone com câmera, um videofone, ou semelhantes), um arquivo de vídeo contendo vídeo armazenado, um servidor de vídeo ou provedor de conteúdo que fornece dados de vídeo, uma interface de alimentação de vídeo recebendo vídeo de um servidor de vídeo ou provedor de conteúdo, um sistema de computação gráfica para gerar dados de vídeo de computação gráfica, uma combinação dessas fontes ou qualquer outra fonte de vídeo adequada.

[0064] Os dados de vídeo da fonte de vídeo 102 podem incluir uma ou mais imagens ou quadros de entrada. Uma imagem ou quadro de um vídeo é uma imagem estática de uma cena. O mecanismo codificador 106 (ou codificador) do dispositivo de codificação 104 codifica os dados de vídeo para gerar um fluxo de bits de vídeo codificado. Em alguns exemplos, um fluxo de bits de vídeo codificado (ou "fluxo de bits de vídeo" ou "fluxo de bits") é uma série de uma ou mais sequências de vídeo codificadas. Uma sequência de vídeo codificada (CVS) inclui uma série de unidades de acesso (AUs) iniciando com uma AU que possui uma imagem de ponto de acesso aleatório na camada base e com determinadas

propriedades até e não incluindo uma próxima AU que possui uma imagem de ponto de acesso aleatório na camada base e com determinadas propriedades. Por exemplo, as determinadas propriedades de uma imagem de ponto de acesso aleatório que inicia uma CVS podem incluir um indicador RASL (por exemplo, NoRaslOutputFlag) igual a 1. Caso contrário, uma imagem de ponto de acesso aleatório (com indicador RASL igual a 0) não inicia uma CVS. Uma unidade de acesso (AU) inclui uma ou mais imagens codificadas e informações de controle correspondentes às imagens codificadas que compartilham o mesmo tempo de saída. As fatias codificadas de imagens são encapsuladas no nível de fluxo de bits em unidades de dados denominadas unidades de camada de abstração de rede (NAL). Por exemplo, um fluxo de bits de vídeo HEVC pode incluir uma ou mais CVSS, incluindo unidades NAL. Cada unidade NAL possui um cabeçalho de unidade NAL. Por exemplo, o cabeçalho é de um byte para H.264/AVC (exceto para extensões multicamada) e de dois bytes para HEVC. Os elementos de sintaxe no cabeçalho da unidade NAL levam os bits designados e, portanto, são visíveis para todos os tipos de sistemas e camadas de transporte, tais como Fluxo de Transporte, Protocolo de Transporte em Tempo Real (RTP), Formato de Arquivo, entre outros.

[0065] Existem duas classes de unidades NAL no padrão HEVC, incluindo unidades NAL de camada de codificação de vídeo (VCL) e unidades NAL não VCL. Uma unidade NAL VCL inclui uma fatia ou segmento de fatia (descrito abaixo) de dados de imagem codificados, e uma unidade NAL não VCL inclui informações de controle que se

referem a uma ou mais imagens codificadas. Em alguns casos, uma unidade NAL pode ser referida como pacote. Uma AU HEVC inclui unidades NAL VCL contendo dados de imagem codificados e unidades NAL não VCL (se houver) correspondentes aos dados de imagem codificados.

[0066] As unidades NAL podem conter uma sequência de bits formando uma representação codificada dos dados de vídeo (por exemplo, um fluxo de bits de vídeo codificado, uma CVS de um fluxo de bits ou semelhantes), tais como representações codificadas de imagens em um vídeo. O mecanismo codificador 106 gera representações codificadas de imagens particionando cada imagem em várias fatias. Uma fatia é independente de outras fatias, de modo que as informações na fatia sejam codificadas sem dependência dos dados de outras fatias na mesma imagem. Uma fatia inclui um ou mais segmentos de fatia, incluindo um segmento de fatia independente e, se presente, um ou mais segmentos de fatia dependentes que dependem dos segmentos de fatia anteriores. As fatias são, então, particionadas em blocos de árvore de codificação (CTBs) de amostras de luma e amostras de croma. Um CTB de amostras de luma e um ou mais CTBs de amostras de croma, juntamente com sintaxe para as amostras, são referidos como uma unidade de árvore de codificação (CTU). Uma CTU é a unidade básica de processamento para codificação HEVC. Uma CTU pode ser particionada em várias unidades de codificação (CUs) de tamanhos variados. Uma CU contém matrizes de amostra de luma e croma que são referidas como blocos de codificação (CBs).

[0067] Os CBs de luma e croma podem ser ainda particionados em blocos de previsão (PBs). Um PB é um bloco

de amostras do componente de luma ou de um componente de croma que utiliza os mesmos parâmetros de movimento para interpretação ou previsão de cópia intrabloco (quando disponível ou ativada para uso). O PB de luma e um ou mais PBs de croma, juntamente com sintaxe associada, formam uma unidade de previsão (PU). Para interpretação, um conjunto de parâmetros de movimento (por exemplo, um ou mais vetores de movimento, índices de referência ou semelhantes) é sinalizado no fluxo de bits para cada PU e é usado para interpretação do PB de luma e dos um mais PBs de croma. Os parâmetros de movimento também podem ser referidos como informações de movimento. Um CB também pode ser particionado em um ou mais blocos de transformada (TBs). Um TB representa um bloco quadrado de amostras de um componente de cor no qual a mesma transformada bidimensional é aplicada para codificação de um sinal residual de previsão. Uma unidade de transformada (TU) representa os TBs de amostras de luma e croma e os elementos de sintaxe correspondentes.

[0068] Um tamanho de uma CU corresponde a um tamanho do modo de codificação e pode ter formato quadrado. Por exemplo, um tamanho de uma CU podem ser amostras de 8 x 8, amostras 16 x 16, amostras 32 x 32, amostras 64 x 64 ou qualquer outro tamanho apropriado até o tamanho da CTU correspondente. A frase "N x N" é usada aqui para se referir às dimensões em pixels de um bloco de vídeo em termos de dimensões verticais e horizontais (por exemplo, 8 pixels x 8 pixels). Os pixels em um bloco podem ser dispostos em linhas e colunas. Em alguns exemplos, os blocos podem não ter o mesmo número de pixels na direção

horizontal e na vertical. Os dados de sintaxe associados a uma CU podem descrever, por exemplo, o particionamento da CU em uma ou mais PUs. Os modos de particionamento podem diferir entre se a CU é codificada pelo modo de intraprevisão ou codificada pelo modo de interprevisão. As PUs podem ser particionadas para ter um formato não quadrado. Os dados de sintaxe associados a uma CU também podem descrever, por exemplo, o particionamento da CU em uma ou mais TUs de acordo com uma CTU. Uma TU pode ter formato quadrado ou não quadrado.

[0069] De acordo com o padrão HEVC, as transformações podem ser executadas usando unidades de transformada (TUs). As TUs podem variar para diferentes CUs. As TUs podem ser dimensionadas com base no tamanho das PUs dentro de uma determinada CU. As TUs podem ser do mesmo tamanho ou menores que as PUs. Em alguns exemplos, amostras residuais correspondentes a uma CU podem ser subdivididas em unidades menores usando uma estrutura de quadtree conhecida como quadtree residual (RQT). Os nós das folhas da RQT podem corresponder às TUs. Os valores de diferença de pixel associados às TUs podem ser transformados para produzir coeficientes de transformada. Os coeficientes de transformada podem, então, ser quantizados pelo mecanismo codificador 106.

[0070] Após as imagens dos dados de vídeo serem particionadas em CUs, o mecanismo codificador 106 prevê que cada PU utilize um modo de previsão. A unidade de previsão ou o bloco de previsão é, então, subtraído dos dados de vídeo originais para obter resíduos (descritos abaixo). Para cada CU, um modo de previsão pode ser sinalizado

dentro do fluxo de bits usando dados de sintaxe. Um modo de previsão pode incluir intraprevisão (ou previsão intraimagem) ou interprevisão (ou previsão interimagem). A intraprevisão utiliza a correlação entre amostras espacialmente adjacentes dentro de uma imagem. Por exemplo, usando a intraprevisão, cada PU é prevista a partir de dados de imagem adjacentes na mesma imagem usando, por exemplo, previsão DC encontrar um valor médio para a PU, previsão planar para ajustar uma superfície plana à PU, previsão de direção para extrapolação a partir de dados adjacentes, ou quaisquer outros tipos adequados de previsão. A interprevisão utiliza a correlação temporal entre imagens para derivar uma previsão compensada por movimento para um bloco de amostras de imagem. Por exemplo, usando a interprevisão, cada PU é prevista usando previsão de compensação de movimento a partir de dados de imagem em uma ou mais imagens de referência (antes ou após a imagem atual na ordem de saída). A decisão de codificar uma área de imagem usando previsão interimagem ou intraimagem pode ser feita, por exemplo, no nível da CU.

[0071] Em alguns exemplos, as uma ou mais fatias de uma imagem são atribuídas a um tipo de fatia. Os tipos de fatia incluem uma fatia I, uma fatia P e uma fatia B. Uma fatia I (intraquadros, decodificável de forma independente) é uma fatia de uma imagem que é codificada apenas por intraprevisão e, portanto, é decodificável de forma independente, uma vez que a fatia I requer apenas os dados dentro do quadro para prever qualquer unidade de previsão ou bloco de previsão da fatia. Uma fatia P (quadros previstos unidirecionais) é uma fatia de uma

imagem que pode ser codificada com intraprevisão e com interprevisão unidirecional. Cada unidade de previsão ou bloco de previsão dentro de uma fatia P é codificada com intraprevisão ou interprevisão. Quando a interprevisão se aplica, a unidade de previsão ou o bloco de previsão é previsto apenas por uma imagem de referência e, portanto, as amostras de referência são apenas de uma região de referência de um quadro. Uma fatia B (quadros preditivos bidirecionais) é uma fatia de uma imagem que pode ser codificada com intraprevisão e com interprevisão (por exemplo, biprevisão ou uniprevisão). Uma unidade de previsão ou bloco de previsão de uma fatia B pode ser previsto bidirecionalmente a partir de duas imagens de referência, em que cada imagem contribui com uma região de referência e conjuntos de amostras das duas regiões de referência são ponderados (por exemplo, com pesos iguais ou com pesos diferentes) para produzir o sinal de previsão do bloco previsto bidirecional. Como explicado acima, as fatias de uma imagem são codificadas de forma independente. Em alguns casos, uma imagem pode ser codificada como apenas uma fatia.

[0072] Uma PU pode incluir os dados (por exemplo, parâmetros de movimento ou outros dados adequados) relacionados ao processo de previsão. Por exemplo, quando a PU é codificada usando intraprevisão, a PU pode incluir dados que descrevem um modo de intraprevisão para a PU. Como outro exemplo, quando a PU é codificada usando interprevisão, a PU pode incluir dados que definem um vetor de movimento para a PU. Os dados que definem o vetor de movimento para uma PU podem descrever, por exemplo, um

componente horizontal do vetor de movimento (Δx), um componente vertical do vetor de movimento (Δy), uma resolução para o vetor de movimento (por exemplo, precisão inteira, precisão de um quarto de pixel ou precisão de um oitavo de pixel), uma imagem de referência para a qual o vetor de movimento aponta, um índice de referência, uma lista de imagens de referência (por exemplo, Lista 0, Lista 1 ou Lista C) para o vetor de movimento, ou qualquer combinação dos mesmos.

[0073] O dispositivo de codificação 104 pode, então, executar transformada e quantização. Por exemplo, após a previsão, o mecanismo codificador 106 pode calcular valores residuais correspondentes à PU. Os valores residuais podem compreender valores de diferença de pixel entre o bloco atual de pixels que está sendo codificado (a PU) e o bloco de previsão usado para prever o bloco atual (por exemplo, a versão prevista do bloco atual). Por exemplo, após a geração de um bloco de previsão (por exemplo, emitindo interprevisão ou intraprevisão), o mecanismo codificador 106 pode gerar um bloco residual subtraindo o bloco de previsão produzido por uma unidade de previsão do bloco atual. O bloco residual inclui um conjunto de valores de diferença de pixel que quantificam as diferenças entre os valores de pixel do bloco atual e os valores de pixel do bloco de previsão. Em alguns exemplos, o bloco residual pode ser representado em um formato de bloco bidimensional (por exemplo, uma matriz bidimensional ou matriz de valores de pixels). Nesses exemplos, o bloco residual é uma representação bidimensional dos valores de pixel.

[0074] Quaisquer dados residuais que possam ser remanescentes após a realização da previsão são transformados usando uma transformada de bloco, que pode ser com base em transformada discreta de cosseno, transformada discreta de seno, uma transformada inteira, uma transformada wavelet, outra função de transformada adequada ou qualquer combinação das mesmas. Em alguns casos, uma ou mais transformadas de bloco (por exemplo, tamanhos 32 x 32, 16 x 16, 8 x 8, 4 x 4, ou semelhantes) podem ser aplicadas a dados residuais em cada CU. Em alguns exemplos, uma TU pode ser usada para os processos de transformada e quantização implementados pelo mecanismo codificador 106. Uma determinada CU com uma ou mais PUs também pode incluir uma ou mais TUs. Conforme descrito em mais detalhes abaixo, os valores residuais podem ser transformados em coeficientes de transformada usando as transformadas de bloco e, em seguida, podem ser quantizados e verificados usando TUs para produzir coeficientes de transformada serializados para codificação por entropia.

[0075] Em alguns exemplos após a codificação intrapreditiva ou interpreditiva usando PUs de uma CU, o mecanismo codificador 106 pode calcular dados residuais para as TUs da CU. As PUs podem compreender dados de pixel no domínio espacial (ou domínio de pixel). As TUs podem compreender coeficientes no domínio de transformada após a aplicação de um bloco de transformada. Como observado anteriormente, os dados residuais podem corresponder aos valores de diferença de pixel entre os pixels da imagem não codificada e os valores de previsão correspondentes às PUs. O mecanismo codificador 106 pode formar as TUs, incluindo

os dados residuais para a CU, e pode transformar as TUs para produzir coeficientes de transformada para a CU.

[0076] O mecanismo codificador 106 pode realizar a quantização dos coeficientes de transformada. A quantização fornece compressão adicional pela quantização dos coeficientes de transformada para reduzir a quantidade de dados usados para representar os coeficientes. Por exemplo, a quantização pode reduzir a profundidade de bits associada a alguns ou a todos os coeficientes. Por exemplo, um coeficiente com um valor de n -bits pode ser arredondado para baixo para um valor de m -bits durante a quantização, sendo n maior que m .

[0077] Uma vez realizada a quantização, o fluxo de bits de vídeo codificado inclui coeficientes de transformada quantizados, informações de previsão (por exemplo, modos de previsão, vetores de movimento, vetores de bloco ou semelhantes), informações de particionamento e outros dados adequados, tais como outros dados de sintaxe. Os diferentes elementos do fluxo de bits de vídeo codificado podem, então, ser codificados por entropia pelo mecanismo codificador 106. Em alguns exemplos, o mecanismo codificador 106 pode utilizar uma ordem de varredura predefinida para verificar os coeficientes de transformada quantizados para produzir um vetor serializado que pode ser codificado por entropia. Em alguns exemplos, o mecanismo codificador 106 pode executar uma varredura adaptativa. Após a varredura dos coeficientes de transformada quantizados para formar um vetor (por exemplo, um vetor unidimensional), o mecanismo codificador 106 pode codificar por entropia o vetor. Por exemplo, o mecanismo codificador

106 pode usar codificação de comprimento variável adaptativo ao contexto, codificação aritmética binária adaptativa ao contexto, codificação aritmética binária adaptativa ao contexto com base em sintaxe, codificação por entropia com particionamento de intervalo de probabilidade ou outra técnica de codificação por entropia adequada.

[0078] Como descrito anteriormente, um fluxo de bits HEVC inclui um grupo de unidades NAL, incluindo unidades NAL VCL e unidades NAL não VCL. As unidades NAL VCL incluem dados de imagem codificados, formando um fluxo de bits de vídeo codificado. Por exemplo, uma sequência de bits que forma o fluxo de bits de vídeo codificado é reenviada em unidades NAL VCL. As unidades NAL não VCL podem conter conjuntos de parâmetro com informações de alto nível relacionadas ao fluxo de bits de vídeo codificado, além de outras informações. Por exemplo, um conjunto de parâmetros pode incluir um conjunto de parâmetros de vídeo (VPS), um conjunto de parâmetros de sequência (SPS) e um conjunto de parâmetros de imagem (PPS). Exemplos de objetivos dos conjuntos de parâmetros incluem eficiência de taxa de bits, resiliência a erros e fornecimento de interfaces de camada de sistemas. Cada fatia faz referência a um único PPS, SPS e VPS ativo para acessar informações que o dispositivo de decodificação 112 pode usar para decodificar a fatia. Um identificador (ID) pode ser codificado para cada conjunto de parâmetros, incluindo um VPS ID, um SPS ID e um PPS ID. Um SPS inclui um SPS ID e um VPS ID. Um PPS inclui um PPS ID e um SPS ID. Cada cabeçalho de fatia inclui um PPS ID. Usando os IDs, conjuntos de

parâmetros ativos podem ser identificados para uma determinada fatia.

[0079] Um PPS inclui informações que se aplicam a todas as fatias em uma determinada imagem. Por isso, todas as fatias de uma imagem se referem ao mesmo PPS. Fatias em imagens diferentes também podem se referir ao mesmo PPS. Um SPS inclui informações que se aplicam a todas as imagens em um mesmo fluxo de bits ou sequência de vídeo codificada (CVS). Como descrito anteriormente, uma sequência de vídeo codificada é uma série de unidades de acesso (AUs) que inicia com uma imagem de ponto de acesso aleatório (por exemplo, uma imagem de referência de decodificação instantânea (IDR) ou imagem de acesso de link quebrado (BLA) ou outra imagem de ponto de acesso aleatório apropriada) na camada base e com determinadas propriedades (descritas acima) até e não incluindo uma próxima AU que tenha uma imagem de ponto de acesso aleatório na camada base e com determinadas propriedades (ou no final do fluxo de bits). As informações em um SPS não podem mudar de imagem para imagem dentro de uma sequência de vídeo codificada. As imagens em uma sequência de vídeo codificada podem usar o mesmo SPS. O VPS inclui informações que se aplicam a todas as camadas dentro de um fluxo de bits ou sequência de vídeo codificada. O VPS inclui uma estrutura de sintaxe com elementos de sintaxe que se aplicam a sequências de vídeo codificadas inteiras. Em alguns exemplos, o VPS, SPS ou PPS pode ser transmitido em banda com o fluxo de bits codificado. Em alguns exemplos, o VPS, SPS ou PPS pode ser transmitido fora de banda em uma

transmissão separada das unidades NAL que contêm dados de vídeo codificados.

[0080] Um fluxo de bits de vídeo também pode incluir mensagens de Informações Suplementares de Aprimoramento (SEI). Por exemplo, uma unidade NAL SEI pode ser parte do fluxo de bits de vídeo. Em alguns casos, uma mensagem SEI pode conter informações que não são necessárias para processo de decodificação. Por exemplo, as informações em uma mensagem SEI podem não ser essenciais para o decodificador decodificar imagens de vídeo do fluxo de bits, mas o decodificador pode usar as informações para melhorar a exibição ou o processamento das imagens (por exemplo, a saída decodificada). As informações em uma mensagem SEI podem ser metadados incorporados. Em um exemplo ilustrativo, as informações em uma mensagem SEI podem ser usadas por entidades do lado do decodificador para melhorar a visibilidade do conteúdo. Em alguns casos, determinados padrões de aplicação podem exigir a presença de tais mensagens SEI no fluxo de bits, para que a melhoria na qualidade possa ser levada a todos os dispositivos compatíveis com o padrão de aplicação (por exemplo, o transporte da mensagem SEI de compactação de quadros para o formato de vídeo 3DTV de plano estereoscópico compatível com quadro, em que a mensagem SEI é transportada por cada quadro do vídeo, a manipulação de uma mensagem SEI de ponto de recuperação, uso do retângulo pan-scan da mensagem SEI em DVB, além de muitos outros exemplos).

[0081] Uma saída 110 do dispositivo de codificação 104 pode enviar as unidades NAL que compõem os dados de vídeo codificados através do link de comunicação

120 para o dispositivo de decodificação 112 do dispositivo receptor. A entrada 114 do dispositivo de decodificação 112 pode receber as unidades NAL. O link de comunicação 120 pode incluir um canal fornecido por uma rede sem fio, uma rede com fio ou uma combinação de uma rede com fio e uma rede sem fio. Uma rede sem fio pode incluir qualquer interface sem fio ou combinação de interfaces sem fio e pode incluir qualquer rede sem fio adequada (por exemplo, a Internet ou outra rede de área ampla, uma rede baseada em pacotes, WiFi™, radiofrequência (RF), UWB, WiFi-Direto, celular, Evolução de Longo Prazo (LTE), WiMax™, ou semelhantes). Uma rede com fio pode incluir qualquer interface com fio (por exemplo, fibra, ethernet, ethernet powerline, ethernet sobre cabo coaxial, linha de sinal digital (DSL) ou semelhantes). As redes com e/ou sem fio podem ser implementadas usando vários equipamentos, tais como estações de base, roteadores, pontos de acesso, pontes, gateways, comutadores ou semelhantes. Os dados de vídeo codificados podem ser modulados de acordo com um padrão de comunicação, tal como um protocolo de comunicação sem fio, e transmitidos ao dispositivo receptor.

[0082] Em alguns exemplos, o dispositivo de codificação 104 pode armazenar dados de vídeo codificados no armazenamento 108. A saída 110 pode recuperar os dados de vídeo codificados do mecanismo codificador 106 ou do armazenamento 108. O armazenamento 108 pode incluir qualquer de uma variedade de meios de armazenamento de dados distribuídos ou acessados localmente. Por exemplo, o armazenamento 108 pode incluir um disco rígido, um disco de armazenamento, memória flash, memória volátil ou não

volátil ou qualquer outra mídia de armazenamento digital adequada para armazenar dados de vídeo codificados.

[0083] A entrada 114 do dispositivo de decodificação 112 recebe os dados de fluxo de bits de vídeo codificados e pode fornecer os dados de fluxo de bits de vídeo ao mecanismo decodificador 116 ou ao armazenamento 118 para uso posterior pelo mecanismo decodificador 116. O mecanismo decodificador 116 pode decodificar os dados de fluxo de bits de vídeo codificados por decodificação por entropia (por exemplo, usando um decodificador de entropia) e extrair os elementos de uma ou mais sequências de vídeo codificadas que compõem os dados de vídeo codificados. O mecanismo decodificador 116 pode, então, redimensionar e executar uma transformada inversa nos dados de fluxo de bits de vídeo codificados. Os dados residuais são, então, passados para um estágio de previsão do mecanismo decodificador 116. O mecanismo decodificador 116, então, prediz um bloco de pixels (por exemplo, uma PU). Em alguns exemplos, a previsão é adicionada à saída da transformada inversa (os dados residuais).

[0084] O dispositivo de decodificação 112 pode enviar o vídeo decodificado para um dispositivo de destino de vídeo, que pode incluir um visor ou outro dispositivo de saída para exibição dos dados de vídeo decodificados a um consumidor do conteúdo. Em alguns aspectos, o dispositivo de destino de vídeo 122 pode ser parte do dispositivo de recepção que inclui o dispositivo de decodificação 112. Em alguns aspectos, o dispositivo de destino de vídeo 122 pode ser parte de um dispositivo separado que não o dispositivo de recepção.

[0085] Em alguns exemplos, o dispositivo de codificação de vídeo 104 e/ou o dispositivo de decodificação de vídeo 112 podem ser integrados a um dispositivo de codificação de áudio e um dispositivo de decodificação de áudio, respectivamente. O dispositivo de codificação de vídeo 104 e/ou o dispositivo de decodificação de vídeo 112 também podem incluir outro hardware ou software necessário para implementar as técnicas de codificação descritas acima, tal como um ou mais microprocessadores, processadores de sinal digital (DSPs), circuitos integrados de aplicação específica (ASICs), matrizes de portas programáveis em campo (FPGAs), lógica discreta, software, hardware, firmware ou qualquer combinação dos mesmos. O dispositivo de codificação de vídeo 104 e o dispositivo de decodificação de vídeo 112 podem ser integrados como parte de um codificador/decodificador combinado (codec) em um respectivo dispositivo. Um exemplo de detalhes específicos do dispositivo de codificação 104 é descrito abaixo com referência à Figura 9. Um exemplo de detalhes específicos do dispositivo de decodificação 112 é descrito abaixo com referência à Figura 10.

[0086] As extensões ao padrão HEVC incluem a extensão de Codificação de Vídeo Multivista, denominada MV-HEVC, e a extensão de Codificação de Vídeo Escalonável, denominada SHVC. As extensões MV-HEVC e SHVC compartilham o conceito de codificação em camadas, com diferentes camadas sendo incluídas no fluxo de bits de vídeo codificado. Cada camada em uma sequência de vídeo codificada é endereçada por um identificador de camada exclusivo (ID). Um ID de

camada pode estar presente no cabeçalho de uma unidade NAL para identificar uma camada com a qual a unidade NAL é associada. No MV-HEVC, diferentes camadas geralmente representam diferentes visualizações da mesma cena no fluxo de bits de vídeo. Em SHVC, diferentes camadas escalonáveis são providas, que representam o fluxo de bits de vídeo em diferentes resoluções espaciais (ou resolução de imagem) ou em diferentes fidelidades de reconstrução. As camadas escalonáveis podem incluir uma camada base (com ID de camada = 0) e uma ou mais camadas de aprimoramento (com IDs de camada = 1, 2, ... n). A camada base pode se conformar a um perfil da primeira versão de HEVC, e representa a camada disponível mais inferior em um fluxo de bits. As camadas de aprimoramento têm elevada resolução espacial, resolução temporal ou taxa de quadros e/ou fidelidade de reconstrução (ou qualidade) em comparação com a camada base. As camadas de aprimoramento são hierarquicamente organizadas e podem (ou não) depender das camadas inferiores. Em alguns exemplos, as diferentes camadas podem ser codificadas usando um único codec padrão (por exemplo, todas as camadas são codificadas usando HEVC, SHVC ou outro padrão de codificação). Em alguns exemplos, diferentes camadas podem ser codificadas usando um codec multipadrão. Por exemplo, uma camada base pode ser codificada usando AVC, enquanto uma ou mais camadas de aprimoramento podem ser codificadas usando extensões SHVC e/ou MV-HEVC para o padrão HEVC.

[0087] Como observado acima, as mensagens SEI podem fornecer informações que não são necessárias para gerar valores de luma e croma de um fluxo de bits de vídeo codificado. As mensagens SEI, no entanto, podem transportar

informações que podem ser consideradas essenciais para renderizar o fluxo de bits conforme planejado pelo gerador de conteúdo.

[0088] A Figura 2 ilustra um exemplo de uma unidade de acesso 200 (AU) que pode ser parte de um fluxo de bits. Como o fluxo de bits pode incluir uma série de unidades de acesso, cada um fornece uma imagem codificada e possivelmente algumas informações sobre o fluxo de bits. Decodificar cada unidade de acesso resulta em uma imagem decodificada. Unidades de acesso, tais como a unidade de acesso exemplificativa 200 ilustrada aqui, estão incluídas nos padrões H.264/AVC e H.265/HEVC.

[0089] Uma unidade de acesso 200, como ilustrado no exemplo da Figura 2, pode incluir um conjunto de unidades NAL, que juntas compõem uma imagem codificada primária. Uma unidade de acesso 200 pode incluir unidades NAL opcionais, que são ilustradas aqui em linhas pontilhadas. Uma unidade de acesso 200 pode, opcionalmente, iniciar com um delimitador de unidade de acesso 202. O delimitador de unidade de acesso pode ajudar a localizar o início da unidade de acesso. A unidade de acesso 200 também pode, opcionalmente, incluir informações suplementares de aprimoramento 204 (por exemplo, uma mensagem SEI) que precede a imagem codificada primária 206. A SEI pode conter dados, tais como informações de temporização de imagem. A imagem codificada primária 206 pode incluir unidades NAL de camada de codificação de vídeo (VCL). As unidades NAL VCL incluem fatias ou partições de dados de fatia que representam as amostras da imagem de vídeo. Após a imagem codificada principal 206, pode haver algumas unidades NAL

VCL adicionais que contêm uma imagem codificada redundante 208. A imagem codificada redundante 208 pode incluir representações redundantes de áreas da mesma imagem de vídeo, e estão disponíveis para uso por um decodificador na recuperação de perdas ou corrupção dos dados na imagem codificada primária 206. Decodificadores não são necessários para decodificar imagens codificadas redundantes, se houver imagens codificadas redundantes.

[0090] Quando a imagem codificada primária 206 é a última imagem de uma sequência de vídeo codificada, a unidade de acesso 200 pode incluir uma unidade NAL de final de sequência 210, para indicar o final da sequência. Quando a imagem codificada primária 206 é também a última imagem codificada no fluxo de bits, a unidade de acesso 200 também pode incluir uma unidade NAL de final de fluxo 212 (também chamada de unidade NAL de final de fluxo de bits).

[0091] Uma unidade NAL SEI pode ser usada para fornecer várias informações a um decodificador. A Tabela 1 abaixo fornece um exemplo da sintaxe geral de mensagens SEI, conforme fornecida pela International Telecommunications Union (ITU-T) H.265 "Series H: Audiovisual and Multimidia Systems Infrastructure of Audiovisual Services - Coding of Moving Video" (doravante denominada Recomendação ITU-T H.265). Na Tabela 1, diferentes tipos de carga útil foram definidos para diferentes informações que podem ser incluídas em uma mensagem SEI. A sintaxe fornecida abaixo inclui apenas um subconjunto de tipos de mensagem SEI que foram definidos e/ou incluídos no padrão HEVC.

Tabela 1

sei_payload(payloadType, payloadSize)	Descriptor
if(nal_unit_type == PREFIX_SEI_NUT)	
if(payloadType == 0)	
buffering_period(payloadSize)	
else if(payloadType == 1)	
pic_timing(payloadSize)	
else if(payloadType == 2)	
pan_scan_rect(payloadSize)	
else if(payloadType == 3)	
filler_payload(payloadSize)	
else if(payloadType == 4)	
user_data_registered_itu_t_t35(payloadSize)	
else if(payloadType == 5)	
user_data_unregistered(payloadSize)	
else if(payloadType == 6)	
recovery_point(payloadSize)	
else if(payloadType == 9)	
scene_info(payloadSize)	
else if(payloadType == 15)	
picture_snapshot(payloadSize)	
else if(payloadType == 16)	
progressive_refinement_segment_start(payloadSize)	
else if(payloadType == 17)	
progressive_refinement_segment_end(payloadSize)	
else if(payloadType == 19)	
film_grain_characteristics(payloadSize)	
else if(payloadType == 22)	
post_filter_him(payloadSize)	
else if(payloadType == 23)	
tone_mapping_info(payloadSize)	
else if(payloadType == 45)	

frame_packing_arrangement(payloadSize)	
else if(payloadType == 47)	
display_orientation(payloadSize)	
else if(payloadType == 128)	
structure_of_pictures_info(payloadSize)	
else if(payloadType == 129)	
active_parameter_sets(payloadSize)	
else if(payloadType == 130)	
decoding_unit_info(payloadSize)	
else if(payloadType == 131)	
temporal_sub_layer_zero_index(payloadSize)	
else if(payloadType == 133)	
scalable_nesting(payloadSize)	
else if(payloadType == 134)	
region_refresh_info(payloadSize)	
else if(payloadType == 135)	
no_display(payloadSize)	
else if(payloadType == 136)	
time_code(payloadSize)	
else if(payloadType == 137)	
mastering_display_colour_volume(payloadSize)	
else if(payloadType == 138)	
segmented_rect_frame_packing_arrangement(
payloadSize)	
else if(payloadType == 139)	
temporal_motion_constrained_tile_sets(payloadSize)	
else if(payloadType == 140)	
chroma_resampling_filter_hint(payloadSize)	
else if(payloadType == 141)	
knee_function_info(payloadSize)	
else if(payloadType == 142)	
colour_remapping_info(payloadSize)	

else if(payloadType == 143)	
deinterlaced_field_identification(payloadSize)	
else if(payloadType == 160)	
layers_not_present(payloadSize)	
else if(payloadType == 161)	
inter_layer_constrained_tile_sets(payloadSize)	
else if(payloadType == 162)	
bsp_nesting(payloadSize)	
else if(payloadType == 163)	
bsp_initial_arrival_time(payloadSize)	
else if(payloadType == 164)	
sub_bitstream_property(payloadSize)	
else if(payloadType == 165)	
alpha_channel_info(payloadSize)	
else if(payloadType == 166)	
overlay_info(payloadSize)	
else if(payloadType == 167)	
temporal_mv_prediction_constraints(payloadSize)	
else if(payloadType == 168)	
frame_field_info(payloadSize)	
else if(payloadType == 176)	
three_dimensional_reference_displays_info(payloadSize)	
}	
else if(payloadType == 177)	
depth_representation_info(payloadSize)	
else if(payloadType == 178)	
multiview_scene_info(payloadSize)	
else if(payloadType == 179)	
multiview_acquisition_info(payloadSize)	
else if(payloadType == 180)	
multiview_view_position(payloadSize)	
else if(payloadType == 181)	

alternative_depth_info(payloadSize)	
else	
reserved_sei_message(payloadSize)	
else /* nal_unit_type == SUFFIX_SEI_NUT */	
if(payloadType == 3)	
filler_payload(payloadSize)	
else if(payloadType == 4)	
user_data_registered_itu_t_t35(payloadSize)	
else if(payloadType == 5)	
user_data_unregistered(payloadSize)	
else if(payloadType == 17)	
progressive_refinement_segment_end(payloadSize)	
else if(payloadType == 22)	
post_filter_hint(payloadSize)	
else if(payloadType == 132)	
decoded_picture_hash(payloadSize)	
else	
reserved_sei_message(payloadSize)	
if(more_data_in_payload()) {	
if(payload_extension_present())	
reserved_payload_extension_data	u(v)
payload_bit_equal_to_one	f(1)
while(!byte_aligned())	
payload_bit_equal_to_zero	f(1)
}	
}	

[0092] Em várias especificações de codificação de vídeo, contanto que uma parte das informações não afete a decodificação de amostras de imagens, tais informações podem ser incluídas em uma mensagem SEI. Algumas informações transportadas em mensagens SEI podem ser para

fins de otimização e podem ser ignoradas por um decodificador sem que a renderização do vídeo seja apresentada de forma inaceitável. Algumas outras informações transportadas em mensagens SEI, no entanto, tais como informações de compactação de quadro, informações de indicação de projeção omnidirecional, e informações de legendas, podem ser essenciais para garantir o desempenho do sistema de vídeo aceitável ou para a experiência de visualização. Por exemplo, determinadas operações pós-decodificação, tais como descompactação ou reversão de distorção (de-warping), devem ser aplicadas antes que um vídeo projetado omnidirecional 360 graus ou com quadros compactados possa ser exibido. Da mesma forma, legendas podem ser necessárias quando o áudio que acompanha um vídeo não está em um idioma que o espectador entenda.

[0093] Para alguns aplicativos e sistemas de codificação, as informações, tais como informações de compactação de quadro, informações de indicação de projeção omnidirecional e informações de legendas, são essenciais para o sistema de aplicação de vídeo funcionar de forma eficiente e/ou correta (por exemplo, apresentar um vídeo como o vídeo foi feito para ser visto). Para esses sistemas e aplicativos, tais informações devem, portanto, ser tratadas de maneira similar às informações que são importantes para a interoperabilidade de decodificação de vídeo (por exemplo, como codec, perfil e nível em operações de sistemas). Por exemplo, essas informações essenciais devem ser consideradas ao encapsular um fluxo de bits de vídeo codificado, por exemplo, em um arquivo de mídia de acordo com o formato de arquivo de mídia base ISO (ISOBMFF).

ou uma apresentação de mídia de acordo com o streaming adaptativo dinâmico sobre HTTP (DASH).

[0094] Ao encapsular um fluxo de bits de vídeo codificado em um arquivo ISOBMFF ou uma apresentação de mídia DASH, as informações essenciais, tal como mencionado acima, podem ser expostas em alto nível, como, por exemplo, na entrada de amostra do formato de arquivo, na descrição da apresentação de mídia DASH (MPD) e/ou em parâmetros do tipo MIME. Colocando as informações essenciais em um nível alto na formação de encapsulamento, o decodificador pode evitar ter que realizar a varredura em todo um fluxo de bits para determinar quais informações podem ser incluídas nas mensagens SEI no fluxo de bits. Adicionalmente, o decodificador pode evitar ter que categorizar as mensagens SEI com base nas condições de essencialidade. Em vez disso, ter tais informações facilmente acessíveis no início do fluxo de bits (por exemplo, onde informações de perfil e nível são sinalizadas) ou, para os padrões existentes, coletivamente em uma mensagem SEI, poderia simplificar a implementação do decodificador. Ter acesso antecipado às informações essenciais pode permitir que o decodificador tome decisões preventivas sobre a melhor forma de lidar com o vídeo decodificado. Além disso, os padrões de codificação de vídeo não precisam ser modificados sempre que uma mensagem SEI que contenha informações essenciais for definida.

[0095] Em várias implementações, uma mensagem SEI é definida para informações essenciais, que é referida como uma mensagem SEI de Informações Suplementares Essenciais (ESI), e uma mensagem SEI é definida para informações não

essenciais, que é referida como uma mensagem SEI de Informações Suplementares Não Essenciais (NSI). Como observado acima, os termos “essencial” e “não essencial” estão sendo usados para classificar dois diferentes tipos de mensagens SEI. As informações sendo transportadas em mensagens SEI ESI e SEI NSI são determinadas por um gerador de conteúdo, por exemplo, na configuração de um codificador que está codificando uma apresentação de vídeo. SEI ESI e SEI NSI não precisam transmitir quaisquer informações específicas, e podem ser usadas por um decodificador na identificação das informações que o decodificador deve processar antes de processar outros dados.

[0096] Em várias implementações, um codificador pode determinar informações que são essenciais. Por exemplo, o codificador pode ser configurado para identificar determinadas informações como essenciais. Nesses exemplos, o codificador pode fazer uso da SEI ESI para as informações essenciais. O codificador pode, ainda, opcionalmente utilizar a ESI NSI para informações não essenciais.

[0097] Em várias implementações, um decodificador pode ser configurado para identificar mensagens SEI ESI e SEI NSI, e para fazer uso das informações incluídas nessas mensagens. Quando o decodificador não encontra nenhuma dessas mensagens, o decodificador pode proceder com a decodificação do fluxo de bits de acordo com o padrão aplicável. O decodificador que não está configurado para processar mensagens SEI ESI ou SEI NSI pode tratar essas mensagens como mensagens SEI não identificáveis. Em alguns exemplos, quando um decodificador encontra uma mensagem SEI

não identificável, o decodificador pode ser configurado para interromper o processamento do fluxo de bits. Alternativamente, em alguns exemplos, o decodificador pode proceder com a decodificação do fluxo de bits.

[0098] Nos exemplos a seguir, as mensagens SEI ESI e SEI NSI são descritas no contexto de HEVC, e entende-se que esses exemplos também podem ser aplicados a outros padrões de codificação de vídeo que suportam mensagens SEI, tais como AVC.

[0099] Mensagem SEI ESI

[0100] A seguir, é provido um exemplo de sintaxe e semântica para o uso de mensagem SEI ESI. Sintaxe e semântica similares podem ser usadas com vários padrões de codificação de vídeo.

[0101] Sintaxe de Mensagem SEI ESI

[0102] Um exemplo de sintaxe da mensagem SEI ESI é como segue:

essential_supplemental_information(payloadSize) {	Descriptor
num_essential_sei_msg_types	u(16)
for(i = 0; i < num_essential_sei_msg_types; i++) {	
essential_sei_payload_type[i]	u(16)
num_esei_init_payload_databits[i]	u(16)
for(j = 0; j < num_esei_init_pidbits; j++) {	
esei_payload_data_bit[i][j]	u(1)
while(!byte_aligned())	
esei_alignment_bit_equal_to_zero /* equal to 0 */	f(1)
}	
}	

[0103] Alternativamente, a codificação dos três elementos de sintaxe codificados por u(16) é substituída

por codificação ue(v) (isto é, substituir todas as instâncias de "u(16)" acima por "ue(v)").

[0104] Semântica de Mensagem SEI ESI

[0105] A mensagem SEI de Informações Suplementares Essenciais (ESI) transmite informações sobre todas as mensagens SEI essenciais presentes no fluxo de bits. Uma mensagem SEI essencial é uma mensagem SEI para a qual as informações transportadas são consideradas pelo codificador (isto é, o produtor de conteúdo) como essenciais para o lado do decodificador processar adequadamente para permitir uma experiência de usuário desejável. Fica a critério do codificador (isto é, o produtor de conteúdo) determinar quais tipos de mensagens SEI são essenciais em um determinado fluxo de bits.

[0106] Quando uma mensagem SEI ESI está presente em qualquer unidade de acesso (AU) de uma CVS, uma mensagem SEI ESI deve estar presente na primeira unidade de acesso da CVS. A mensagem SEI ESI persiste na ordem de decodificação da unidade de acesso atual até o final da CVS.

[0107] Uma unidade NAL SEI contendo uma mensagem SEI ESI não deve conter nenhuma outra mensagem SEI que não seja uma mensagem SEI de informações suplementares não essenciais. Quando houver múltiplas mensagens ESI presentes em uma CVS, elas devem ter o mesmo conteúdo.

[0108] `num_essential_sei_msg_types` especifica o número de tipos de mensagens SEI presentes na CVS que são consideradas mensagens SEI essenciais.

[0109] `essential_sei_payload_type[i]` indica o valor de payloadType do i-ésimo tipo de mensagens SEI

essenciais. Para quaisquer dois valores diferentes m e n , os valores de `essential_sei_payload_type[m]` e `essential_sei_payload_type[n]` não devem ser idênticos.

[0110] `num_esei_init_payload_databits[i]` especifica o número dos bits seguintes que fornecem informações adicionais sobre o i -ésimo tipo de mensagens SEI essenciais presentes na CVS.

[0111] `esei_payload_data_bit[i][j]` especifica o j -ésimo bit dos bits que fornecem informações adicionais sobre o i -ésimo tipo de mensagens SEI essenciais presentes na CVS.

[0112] Os bits `num_esei_init_payload_databits[i]` seguem a sintaxe de carga útil SEI para o `payloadType` do i -ésimo tipo de mensagens SEI essenciais, e contêm um número de elementos de sintaxe a partir do primeiro elemento de sintaxe na sintaxe de carga útil SEI, mas pode ou não conter todos os elementos de sintaxe na sintaxe de carga útil SEI. O último bit deve ser o último bit de um elemento de sintaxe na sintaxe de carga útil SEI. O elemento de sintaxe contidos nesses bits têm a mesma semântica como se estivessem em uma mensagem SEI com o `payloadType`, com a exceção de que as informações aqui se aplicam a toda a CVS.

[0113] Esses bits devem fornecer informações suficientes para indicar quais tipos de processamento são necessários no lado do decodificador a partir do ponto de vista de capacidade de processamento, com base nas quais as entidades de camada do sistema podem determinar se o lado do decodificador pode processar corretamente o fluxo de bits para permitir uma experiência de usuário desejável. Por exemplo, se o `payloadType` for 45 (isto é, a mensagem

SEI de arranjo de compactação de quadro), esses bits devem incluir pelo menos o elemento de sintaxe `frame_packing_arrangement_type`. Como outro exemplo, se o `payloadType` for 150 (isto é, a mensagem SEI de indicação de projeção omnidirecional como especificado em JCTVC-Z1005), estes bits devem incluir o `projection_type` do elemento de sintaxe.

[0114] `esei_alignment_bit-equal_to_zero` deve ser igual a 0.

[0115] Mensagem SEI NSI

[0116] A seguir, é provida um exemplo de sintaxe e semântica para a mensagem SEI NSI. Sintaxe e semântica similares podem ser usadas com vários padrões de codificação de vídeo.

[0117] Sintaxe de mensagem SEI NSI

[0118] Um exemplo da sintaxe de mensagem SEI NSI é como segue:

nonessential_supplemental_information(payloadSize) {	Descriptor
num_nonessential_sei_msg_types	u(16)
for(i = 0; i < num_nonessential_sei_msg_types; i++) {	
nonessential_sei_payload_type[i]	u(16)
num_nesei_init_payload_databits[i]	u(16)
for(j = 0; j < num_nesei_init_pldbits; j++) {	
nesei_payload_data_bit[i][j]	u(1)
while(!byte_aligned())	
nesei_alignment_bit_equal_to_zero /* equal to 0 */	f(1)
}	
}	

[0119] Alternativamente, a codificação dos três elementos de sintaxe codificados por u(16) é substituída

por codificação ue(v), isto é, substituir todas as instâncias de "u(16)" acima por "ue(v)".

[0120] Semântica de Mensagem SEI NSI

[0121] A mensagem SEI de Informações Suplementares Não Essenciais (NSI) transmite informações sobre todas as mensagens SEI não essenciais presentes no fluxo de bits. Uma mensagem SEI não essencial é uma mensagem SEI que não é considerada pelo codificador (isto é, o produtor de conteúdo) uma mensagem SEI essencial. Fica à critério do codificador (isto é, o produtor de conteúdo) determinar quais tipos de mensagens SEI são mensagens SEI não essenciais em um determinado fluxo de bits.

[0122] Quando uma mensagem SEI NSI está presente em qualquer unidade de acesso de uma CVS, uma mensagem SEI NSI deve estar presente na primeira unidade de acesso da CVS. A mensagem SEI NSI persiste na ordem de decodificação da unidade de acesso atual até o final da CVS.

[0123] Uma unidade NAL SEI contendo uma mensagem SEI NSI não deve conter nenhuma outra mensagem SEI que não seja uma mensagem SEI ESI. Quando há múltiplas mensagens NSI presentes em uma CVS, elas devem ter o mesmo conteúdo.

[0124] Quando um determinado tipo de mensagem SEI é identificado por uma mensagem SEI ESI em uma CVS como mensagem SEI essencial, o mesmo tipo de mensagem SEI não deve ser identificado por uma mensagem SEI NSI na CVS como mensagens SEI não essenciais.

[0125] `num_nonessential_sei_msg_types` especifica o número de tipos de mensagem SEI presentes na CVS que são considerados mensagens SEI não essenciais.

[0126] `nonessential_sei_payload_type[i]` indica o valor de payloadType do i-ésimo tipo de mensagens SEI não essenciais. Para quaisquer dois valores m e n diferentes, os valores de `nonessential_sei_payload_type[m]` e `nonessential_sei_payload_type[n]` não devem ser idênticos.

[0127] `num_nesei_init_payload_databits[i]` especifica o número dos seguintes bits que fornecem informações adicionais sobre o i-ésimo tipo de mensagens SEI não essenciais presentes na CVS.

[0128] `nesei_payload_data_bit[i][j]` especifica o j-ésimo bit dos bits que fornecem informações adicionais sobre o i-ésimo tipo de mensagens SEI não essenciais presentes na CVS.

[0129] Os bits `num_nesei_init_payload_databits[i]` seguem a sintaxe de carga útil SEI para o payloadType do i-ésimo tipo de mensagens SEI não essenciais, e contêm um número de elementos de sintaxe a partir do primeiro elemento de sintaxe na sintaxe de carga útil SEI, mas podem ou não conter todos os elementos de sintaxe na sintaxe de carga útil SEI. O último bit deve ser o último bit de um elemento de sintaxe na sintaxe de carga útil SEI. Os elementos de sintaxe contidos nesses bits têm a mesma semântica como se estivessem em uma mensagem SEI com o payloadType, com exceção de que as informações aqui se aplicam a toda a CVS.

[0130] `nesei_alignment_bit_equal_to_zero` deve ser igual a 0.

[0131] Em várias implementações, sintaxe e semântica são providas para indicar as presenças, ausência e/ou essencialidade de mensagens SEI. Em várias

implementações, um manifesto SEI é definido, bem como uma mensagem SEI que inclui indicações de prefixo SEI. O manifesto SEI pode ser usado para transmitir informações sobre as mensagens SEI que estão incluídas em um fluxo de bits. Indicações de prefixo SEI podem ser usadas para identificar a carga útil em uma mensagem SEI, e podem ser usadas por um decodificador para determinar o tipo de processo que é necessário para o conteúdo em um fluxo de bits.

[0132] Em várias implementações, um codificador pode determinar informações que são essenciais. Por exemplo, o codificador pode ser configurado para identificar determinadas informações que são essenciais para a exibição de um vídeo. Nesses exemplos, o codificador pode fazer uso da mensagem SEI de indicação de prefixo SEI para indicar as informações essenciais, e a mensagem SEI de manifesto SEI para descrever as mensagens SEI incluídas no fluxo de bits

[0133] Em várias implementações, um decodificador pode ser configurado para identificar as mensagens SEI de manifesto SEI e mensagens SEI de prefixo SEI, e para fazer uso das informações incluídas nessas mensagens. Quando o decodificador não encontra nenhuma dessas mensagens, o decodificador pode proceder com a decodificação do fluxo de bits de acordo com o padrão aplicável. Os decodificadores que não são configurados para processar as mensagens de prefixo SEI e manifesto SEI podem tratar essas mensagens como mensagens SEI não identificáveis. Em alguns exemplos, quando um decodificador encontra uma mensagem SEI não identificável, o decodificador pode ser configurado para

interromper o processamento do fluxo de bits. Alternativamente, em alguns exemplos, o decodificador pode proceder com a decodificação do fluxo de bits.

[0134] Um primeiro exemplo fornece uma mensagem SEI de manifesto SEI e uma mensagem SEI de indicação de prefixo SEI. Sintaxe e semântica para cada uma dessas mensagens SEI são providas abaixo.

[0135] A mensagem SEI de manifesto SEI

[0136] Sintaxe de mensagem SEI de manifesto SEI

fes(f payloadSize) {	Descriptor
manifest num sei msg types	u(16)
for(i = 0; i < manifest num sei msg types; i++) {	
manifest_sei_payload_type[i]	u(16)
manifest_sei_description[i]	u(8)
}	
}	

[0137] Semântica de mensagem SEI de manifesto SEI

[0138] Uma mensagem SEI de manifesto SEI transmite informações em mensagens SEI que podem ser presumidas por um receptor do fluxo de bits como presentes ou não presentes. Essas informações podem incluir:

[0139] 1) Um receptor do fluxo de bits pode presumir que determinados tipos de mensagens SEI estão presentes na CVS.

[0140] NOTA 1 - Com o uso de tal expressão "presumir", não seria um erro quando as "promessas" fossem quebradas e, independentemente de as "promessas" serem quebradas, as premissas do cliente ainda podem ser consideradas corretas.

[0141] 2) Para cada tipo de mensagem SEI que possa ser presumida como presente na CVS, a essencialidade das mensagens SEI deste tipo.

[0142] A essencialidade pode ser indicada como essencial, não essencial ou indeterminada.

[0143] Uma mensagem SEI é considerada essencial pelo codificador (isto é, o produtor do conteúdo) quando as informações transportadas são consideradas essenciais do lado do decodificador para processar adequadamente o conteúdo e permitir uma experiência de usuário desejável. Fica a critério do codificador determinar quais mensagens SEI são consideradas essenciais em um fluxo de bits particular. No entanto, algumas mensagens SEI, tais como as mensagens SEI de arranjo de compactação de quadro, de arranjo de compactação de quadro retangular segmentado, de orientação de exibição e de indicação de projeção omnidirecional, devem sempre ser consideradas essenciais.

[0144] 3) Um receptor do fluxo de bits pode presumir que determinados tipos de mensagens SEI não estão presentes na CVS.

[0145] NOTA 2 - Por exemplo, com base na mensagem SEI de manifesto SEI sinalizada, um receptor do fluxo de bits poderia assumir que não há uma mensagem SEI de arranjo de compactação de quadro, mensagem SEI de arranjo de compactação de quadros retangular segmentado, mensagem SEI de orientação de exibição, ou mensagem SEI de indicação de projeção omnidirecional no fluxo de bits. Portanto, as imagens de vídeo decodificadas provavelmente poderiam ser renderizadas para visualização sem a necessidade de qualquer pós-processamento adicional que é comumente necessário quando tais mensagens SEI estão presentes.

[0146] Quando uma mensagem SEI de manifesto SEI está presente em qualquer unidade de acesso de uma CVS, uma

mensagem SEI de manifesto SEI deve estar presente na primeira unidade de acesso da CVS. A mensagem SEI de manifesto SEI persiste na ordem de decodificação da unidade de acesso atual até o final da CVS. Quando há múltiplas mensagens SEI de manifesto SEI presentes em uma CVS, elas devem ter o mesmo conteúdo.

[0147] Uma unidade NAL SEI contendo uma mensagem SEI de manifesto SEI não deve conter qualquer outra mensagem SEI além de mensagens SEI de indicação de prefixo SEI. Quando presente em uma unidade NAL SEI, uma mensagem SEI de manifesto SEI deve ser a primeira mensagem SEI na unidade NAL SEI.

[0148] `manifest_num_sei_msg_types` especifica o número de tipos de mensagens SEI para as quais as informações são providas na mensagem SEI de manifesto SEI.

[0149] `manifest_sei_payload_type[i]` indica o valor payloadType do i-ésimo tipo de mensagem SEI para o qual as informações são providas na mensagem SEI de manifesto SEI. Os valores de `manifest_sei_payload_type[m]` e `manifest_sei_payload_type[n]` não devem ser idênticos quando m não é igual a n.

[0150] `manifest_sei_description[i]` fornece informações sobre mensagens SEI com payloadType igual a `manifest_sei_payload_type[i]` como especificado na Tabela 2.

Tabela 2: Valores de `manifest_sei_description[i]`

Valor	Descrição
0	Indica que pode-se presumir que não existe mensagem SEI com payloadType igual a <code>manifest_sei_payload_type[i]</code> presente na

	CVS.
1	Indica que 1) pode-se presumir que existem mensagens SEI com payloadType igual a manifest_sei_payload_type[i] presentes na CVS, e 2) essas mensagens SEI são consideradas essenciais.
2	Indica que 1) pode-se presumir que existem mensagens SEI com payloadType igual a manifest_sei_payload_type[i] presentes na CVS, e 2) essas mensagens SEI são consideradas não essenciais.
3	Indica que 1) pode-se presumir que existem mensagens SEI com payloadType igual a manifest_sei_payload_type[i] presentes na CVS, e 2) a essencialidade dessas mensagens SEI é indeterminada.
4-255	Reservado

[0151] O valor de manifest_sei_description[i] deve estar na faixa de 0 a 3, inclusive, nos fluxos de bits compatíveis com a presente versão desta Especificação. Outros valores para manifest_sei_description[i] são reservados para uso futuro por ITU-T|ISO/IEC. Os decodificadores devem permitir que o valor de manifest_sei_description[i] maior ou igual 4 apareça na sintaxe e deve ignorar todas as informações para payloadType igual a manifest_sei_payload_type[i] sinalizadas na mensagem SEI de manifesto SEI e deve ignorar todas as mensagens SEI de indicação de prefixo SEI com prefix_sei_payload_type igual a manifest_sei_payload_type[i].

[0152] A mensagem SEI de indicação de prefixo SEI

[0153] Sintaxe de mensagem SEI de indicação de prefixo SEI

sei_prefix_indication(payloadSize) {	Descriptor
prefix_sei_payload_type	u(16)
num_sei_prefix_indications_minus1	u(8)
for(i = 0; i <= num_sei_prefix_indications_minus1; i++) {	
num_bits_in_prefix_indication_minus1[i]	u(16)
for(j = 0; j <= num_bits_in_prefix_indication_minus1[i]; j++)	
sei_prefix_databit[i][j]	u(1)
while(!byte_aligned())	
byte_alignment_bit_equal_to_zero /* equal to 0 */	f(1)
}	
}	

[0154] Semântica de mensagem SEI de indicação de prefixo SEI

[0155] A mensagem SEI de indicação de prefixo SEI transporta uma ou mais indicações de prefixo SEI para mensagens SEI de um payloadType particular. Cada indicação de prefixo SEI é uma cadeia de bits que segue a sintaxe de carga útil SEI daquele payloadType e contém um número de elementos de sintaxe completos iniciando a partir do primeiro elemento de sintaxe na carga útil SEI.

[0156] O receptor pode presumir que algumas ou todas as mensagens SEI deste payloadType na CVS iniciam com essas cadeias de bits. Uma cadeia de bits inicial normalmente contém apenas um subconjunto verdadeiro de uma carga útil SEI do tipo de mensagens SEI, pode conter uma carga útil SEI completa, mas nunca contém mais do que uma carga útil SEI completa.

[0157] NOTA - É permitido que algumas mensagens SEI deste payloadType não iniciem com nenhuma das cadeias de bits indicadas.

[0158] Essas indicações de prefixo SEI devem fornecer informações suficientes para indicar qual tipo de processamento é necessário ou qual tipo de conteúdo está incluído. O anterior (tipo de processamento) indica a capacidade de processamento do lado do decodificador, por exemplo, se algum tipo de descompactação de quadros é necessário. O último (tipo de conteúdo) indica, por exemplo, se o fluxo de bits contém legendas em um idioma específico. Com base na camada de sistemas de informação, as entidades podem determinar se o lado do decodificador pode processar adequadamente o fluxo de bits para permitir uma experiência de usuário desejável, ou se o fluxo de bits satisfaz as necessidades de aplicação.

[0159] Em um exemplo, quando o payloadType indica mensagem SEI de arranjo de compactação de quadros, uma indicação de prefixo SEI deve incluir até pelo menos o elemento de sintaxe frame_packing_arrangement_type; quando o payloadType indica a mensagem SEI de indicação de projeção omnidirecional, uma indicação de prefixo SEI deve incluir até pelo menos o elemento de sintaxe projection_type.

[0160] Em outro exemplo, para as mensagens SEI de dados de usuários registrados que são usadas para transportar informações de legendas, uma indicação de prefixo SEI deve incluir até pelo menos o código de idioma; e para as mensagens SEI de dados de usuários não registrados estendidas para uso privado, uma indicação de prefixo SEI deve incluir até pelo menos o UUID.

[0161] Quando uma mensagem SEI de indicação de prefixo SEI está presente em qualquer unidade de acesso de

uma CVS, uma mensagem SEI de indicação de prefixo SEI deve estar presente na primeira unidade de acesso da CVS. A mensagem SEI de indicação de prefixo SEI persiste na ordem de decodificação da unidade de acesso atual até o final da CVS. Quando há várias mensagens SEI de indicação de prefixo SEI presentes em uma CVS, elas devem ter o mesmo conteúdo.

[0162] Uma unidade NAL SEI contendo uma mensagem SEI de indicação de prefixo SEI não deve conter qualquer outra mensagem SEI além de uma mensagem SEI de manifesto SEI.

[0163] `prefix_sei_payload_type` indica o valor de `payloadType` das mensagens SEI para as quais uma ou mais indicações de prefixo SEI são fornecidas na mensagem SEI de indicação de prefixo SEI. O valor de `prefix_sei_payload_type` deve ser igual a um dos valores de `manifest_sei_payload_type[m]` para os quais `manifest_sei_description[m]` é igual a 1 a 3, inclusive, conforme indicado por uma mensagem SEI de manifesto SEI que se aplica à CVS.

[0164] `num_sei_prefix_indications_minus1` mais 1 especifica o número de indicações de prefixo SEI.

[0165] `num_bits_in_prefix_indication_minus1[i]` mais 1 especifica o número de bits na i-ésima indicação de prefixo SEI.

[0166] `sei_prefix_databit[i][j]` especifica o j-ésimo bit da i-ésima indicação de prefixo SEI.

[0167] Os bits `sei_prefix_databit[i][j]` para j variando de 0 a `num_bits_in_prefix_indication_minus1[i]`, inclusive, seguem a sintaxe da carga útil SEI com `payloadType` igual a `prefix_sei_payload_type`, e contêm um

número de elementos de sintaxe completos a partir do primeiro elemento de sintaxe na sintaxe de carga útil SEI, e podem ou não conter todos os elementos de sintaxe na sintaxe de carga útil SEI. O último bit deve ser o último bit de um elemento de sintaxe na sintaxe de carga útil SEI.

[0168] byte_alignment_bit_equal_to_zero deve ser igual a 0.

[0169] No exemplo anterior, as descrições de manifesto SEI, que podem ser pequenas, podem estar em uma mensagem SEI separada. O tamanho pequeno dessa mensagem SEI permite que a mensagem SEI seja transportada em um nível de sistemas, por exemplo, como um parâmetro tipo MIME. O exemplo precedente também permite a adição de indicações de prefixo para tipos SEI para os quais nenhuma indicação de prefixo tenha sido ainda definida. Por exemplo, uma nova mensagem SEI pode ser adicionada sem modificar a mensagem SEI existente. Adicionalmente, várias mensagens SEI não são necessárias para as indicações de prefixo SEI para um tipo de SEI particular. Ao adicionar indicações de prefixo para um tipo de SEI para o qual indicações de prefixo já tenham sido definidas, pode ser necessário alterar a mensagem SEI existente.

[0170] Em vários exemplos, sintaxe alternativa pode ser utilizada.

[0171] Em um primeiro exemplo, em vez de definir duas mensagens SEI separadas, a descrição de manifesto SEI e os prefixos SEI podem ser transportados em uma mensagem SEI. Exemplo de sintaxe para tal mensagem SEI segue abaixo:

sei_manifest(payloadSize) {	Descriptor
manifest_num_sei_msg_types	u(16)
for(i = 0; i < manifest_num_sei_msg_types; i++) {	
manifest_sei_payload_type[i]	u(16)
manifest_sei_description[i]	u(8)
}	
for(i = 0; i < manifest_num_sei_msg_types; i++) {	
num_sei_prefix_indications[i]	u(8)
for(j = 0; j < num_sei_prefix_indications[i]; j++) {	
num_bits_in_prefix_indication[i][j]	u(16)
for(k = 0; k < num_bits_in_prefix_indication[i][j]; k++)	
sei_prefix_database[i][j][k]	u(1)
while(!byte_aligned())	
byte_alignment_bit_equal_to_zero /* equal to 0 */	f(1)
}	
}	
}	

[0172] No exemplo acima, várias mensagens SEI (por exemplo, para as descrições de manifesto SEI e as indicações de prefixo SEI para todos os tipos SEI) não precisam ser definidas. Quando informações adicionais são necessárias, no entanto, uma mensagem SEI pode precisar ser alterada.

[0173] Em um segundo exemplo, em vez de permitir múltiplas indicações de prefixo SEI em uma mensagem SEI de indicações de prefixo SEI, uma mensagem de prefixo SEI pode incluir apenas uma indicação de prefixo SEI. Um exemplo de sintaxe para tal mensagem SEI segue abaixo:

sei_prefix_indication(payloadSize) {	Descriptor
prefix_sei_payload_type	u(16)
num_bits_in_prefix_indication_minus1	u(16)
for(i = 0; i <= num_bits_in_prefix_indication_minus1; i++)	
sei_prefix_database[i]	u(1)

[0174] No exemplo acima, a mensagem SEI existente não precisa ser alterada ao adicionar indicações de prefixo

para um tipo SEI para o qual indicações de prefixo já tenham sido definidas. Múltiplas mensagens SEI para sinalização das indicações de prefixo SEI para um tipo SEI particular podem ser necessárias, no entanto.

[0175] Em vários exemplos, técnicas alternativas podem ser utilizadas para definir o escopo de persistência de mensagens SEI.

[0176] Em um primeiro exemplo, o escopo de persistência de mensagens SEI pode ser o mesmo que o SPS ativo. Neste exemplo, quando o SPS ativo se aplica a várias CVS, o escopo de persistência da mensagem SEI também serão as várias CVS.

[0177] Em um segundo exemplo, o escopo de persistência pode ser especificado no fluxo de bits. Neste exemplo, a mensagem SEI pode precisar ser alterada quando vários fluxos de bits são fatiados em um fluxo de bits.

[0178] Em vários exemplos, redação alternativa para a semântica pode ser utilizada.

[0179] Em um primeiro exemplo, redação assertiva, tal como "indica se determinados tipos de mensagens SEI estão presentes ou ausentes", pode ser utilizada, por exemplo, na semântica do manifesto SEI.

[0180] Como um segundo exemplo, redação não assertiva (conforme observado do ponto de vista de um decodificador), tal como "indica se determinados tipos de mensagens SEI podem ser presumidos como presentes ou ausentes", pode ser utilizada.

[0181] Como um terceiro exemplo, redação não assertiva (conforme observado do ponto de vista de um decodificador), tal como "indica se determinados tipos de

mensagens SEI são indicados pelo codificador como presentes ou ausentes" pode ser usada. Neste exemplo, pode-se afirmar ainda que um decodificador pode tomar decisões sobre como processar essas informações, embora como processar as informações podem ser deixadas até o decodificador.

[0182] No caso do primeiro exemplo acima, pode haver instâncias em que um manifesto SEI pode indicar que um determinado tipo de mensagem SEI está presente em um fluxo de bits, mas as mensagens SEI daquele tipo podem ter sido removidas. Nesses exemplos, a semântica do fluxo de bits pode estar incorreta, e o fluxo de bits pode ser não conforme.

[0183] Com o segundo e terceiro exemplos acima, a remoção de mensagens SEI no fluxo de bits não faz com que o fluxo de bits seja não conforme. Quando, por exemplo, o manifesto SEI indica que um determinado tipo de mensagem SEI está presente no fluxo de bits, mas a mensagem SEI foi removida (por exemplo, após o fluxo de bits ser codificado), a redação não assertiva do segundo e terceiro exemplos não se tornaria incorreta. No terceiro exemplo, a redação não assertiva é menos rigorosa do que a redação não assertiva do segundo exemplo.

[0184] Mensagens SEI Essenciais e Não Essenciais

[0185] Várias mensagens SEI foram definidas, por exemplo, no Anexo D da especificação HEVC, em propostas de alteração à especificação HEVC (tal como o texto da proposta de alteração em JCTVC-Z1005), e no Anexo F e anexos subsequentes da especificação.

[0186] São providos abaixo exemplos de mensagens SEI que podem ser consideradas como incluindo informações

essenciais. Essas mensagens SEI, quando presentes (por exemplo, presentes nativamente ou contidas em uma mensagem SEI de nidificação) podem ter um impacto direto sobre a renderização de imagens decodificadas. Embora essas mensagens SEI possam ser consideradas essenciais, rotular como essenciais as mensagens SEI que incluem essas informações é deixado para o codificador.

[0187] - A mensagem SEI de arranjo de compactação de quadros

[0188] - A mensagem SEI de orientação de exibição

[0189] - A mensagem SEI de arranjo de compactação de quadros retangulares segmentados

[0190] - A mensagem SEI de indicação de projeção omnidirecional

[0191] As seguintes mensagens SEI, quando presentes (por exemplo, presentes nativamente ou contidas em uma mensagem SEI de nidificação), podem afetar a renderização de imagens decodificadas. Essas mensagens SEI também são exemplos de mensagens SEI que poderiam ser consideradas essenciais.

[0192] - A mensagem SEI de características de grãos do filme

[0193] - A mensagem SEI de dica pós-filtro

[0194] - A mensagem SEI de informações de mapeamento de tom

[0195] - A mensagem SEI de não exibição

[0196] - A mensagem SEI de não exibição

[0197] - A mensagem SEI de masterização do volume de cor do visor

[0198] - O mensagem SEI de dica de filtro de reamostragem de croma

[0199] - A mensagem SEI de informações da função "knee"

[0200] - A mensagem SEI de informações de remapeamento de cor

[0201] - A mensagem SEI de informações de imagem desentrelaçada

[0202] - A mensagem SEI de informações do nível de luz do conteúdo

[0203] - A mensagem SEI de características de transferência alternativas

[0204] - A mensagem SEI do contexto de visualização de ambiente

[0205] - A mensagem SEI de volume de cor do conteúdo

[0206] Outras mensagens SEI que são menos propensas a afetar a renderização da imagem decodificada podem ou não ser incluídas na lista de mensagens SEI essenciais, a critério do codificador. Por exemplo, o codificador pode determinar que um dado de usuário registrado pela mensagem SEI de Recomendação ITU-T T.35 ou uma mensagem SEI de dados de usuários não registrados transporta informações essenciais e consequentemente incluir esse tipo de carga útil SEI na lista de mensagens SEI essenciais e fornecer algumas adicionais informações para indicar suficientemente a capacidade de processamento necessária.

[0207] Exposição de Nível de Sistemas de Informações Suplementares Essenciais

[0208] Em várias implementações, as mensagens SEI, tais como aquelas descritas acima (por exemplo, SEI ESI, SEI NSI, SEI de manifesto SEI e SEI de prefixo SEI, que serão referidas aqui coletivamente como SEI transportando informações essenciais) podem ser colocadas em formatos de encapsulamento, tal que um decodificador possa ter acesso a essas SEI no inicio do processo de decodificação de um segmento de vídeo. Formatos de encapsulamento incluem, por exemplo, formatos de arquivos, formatos de streaming, formatos de rede, e outros formatos.

[0209] Padrões de formato de arquivo podem definir o formato para compactar e descompactar dados de vídeo (e possivelmente também de áudio) em um ou mais arquivos. Padrões de formato de arquivo incluem o formato de arquivo de mídia base da Organização Internacional para Padronização (ISO) (ISOBMFF, definido em ISO/IEC 14496-12) e outros formatos de arquivo derivados do ISOBMFF, incluindo o formato de arquivo do Grupo de Especialistas em Imagens em Movimento (MPEG) MPEG-4 (definido no ISO/IEC 14496-15), formato de arquivo da 3th Generation Partnership Project (3GPP) (definido em 3 GPP TS 26.244) e formato de Codificação de Video Avançada (AVC) e formato de arquivo de Codificação de Vídeo de Alta Eficiência (HEVC) (ambos definidos em ISO/IEC 14496-15). Os textos preliminares de novas edições recentes da ISO/IEC 14496-12 e 14496-15 estão disponíveis em http://phenix.int-evry.fr/mpeg/doc_end_user/documents/111_Geneva/wg11/wl5177-v6-wl5177.zip e em http://phenix.int-evry.fr/mpeg/doc_end_user/documents/112_Warsaw/wg11/wl5479-v2-wl5479.zip, respectivamente.

[0210] O ISOBMFF é usado como base para muitos formatos de encapsulamento de codec (por exemplo, o formato de arquivo AVC ou qualquer outro formato de encapsulamento de codec adequado), bem como para muitos formatos de recipiente multimídia (por exemplo, o formato de arquivo MPEG-4, o formato de arquivo 3GPP (3GPP), o formato de arquivo Digital Video Broadcasting (DVB), ou qualquer outro formato de recipiente multimídia adequado). Os formatos de arquivo baseados em ISOBMFF podem ser usados para mídia contínua, que também é conhecida como mídia em streaming.

[0211] Além de mídia contínua (por exemplo, áudio e vídeo), mídia estática (por exemplo, imagens) e metadados podem ser armazenados em um arquivo em conformidade com ISOBMFF. Arquivos estruturados de acordo com o ISOBMFF podem ser usados para muitas finalidades, incluindo a reprodução local de arquivos de mídia, download progressivo de um arquivo remoto, como segmentos para Streaming Adaptativo Dinâmico sobre HTTP (DASH), como recipientes para conteúdo a ser transmitido em streaming (nesse caso, os recipientes incluem instruções de compactação), para registrar fluxos de mídia em tempo real recebidos, ou outros usos.

[0212] O ISOBMFF e seus formatos de arquivo derivados (por exemplo, o formato de arquivo AVC ou outros formatos de arquivo derivados) são amplamente usados para armazenamento e encapsulamento de conteúdo de mídia (por exemplo, incluindo vídeo, áudio e texto temporizado) em muitas aplicações multimídia. Em várias implementações, o formato de arquivo ISOBMFF pode ser usado para fornecer mensagens SEI transportando informações essenciais à frente

dos dados de vídeo aos quais as informações se aplicam. Por exemplo, a SEI transportando informações essenciais podem ser incluídas em uma entrada de amostra, conforme discutido mais adiante.

[0213] A Figura 3 ilustra um exemplo de um arquivo de mídia base ISO 300 que contém dados e metadados para uma apresentação de vídeo, formatado de acordo com o ISOBMFF. O ISOBMFF é projetado para conter informações de mídia temporizada em um formato flexível e extensível que facilita o intercâmbio, gerenciamento, edição e apresentação da mídia. A apresentação da mídia pode ser "local" ao sistema contendo a apresentação, ou a apresentação pode ser através de uma rede ou outro mecanismo de distribuição de fluxo.

[0214] Uma "apresentação", conforme definida pela especificação ISOBMFF, é uma sequência de imagens, muitas vezes relacionadas por terem sido capturadas sequencialmente por um dispositivo de captura de vídeo, ou relacionadas por algum outro motivo. Aqui, uma apresentação também pode ser referida como um filme ou uma apresentação de vídeo. Uma apresentação pode incluir áudio. Uma única apresentação pode estar contida em um ou mais arquivos, com um arquivo contendo os metadados para toda a apresentação. Os metadados incluem informações sobre tempos e dados de quadro, descritores, ponteiros, parâmetros e outras informações que descrevem a apresentação. Os metadados não incluem dados de vídeo e/ou áudio em si. Arquivos que não o arquivo que contém os metadados não precisam ser formatados de acordo com o ISOBMFF, e só precisam ser formatados de

forma que esses arquivos possam ser referidos pelos metadados.

[0215] A estrutura de arquivo de um arquivo de mídia base ISO é orientada a objeto, e a estrutura de um objeto individual no arquivo pode ser inferida diretamente a partir do tipo do objeto. Os objetos em um arquivo de mídia base ISO são referidos como "caixas" pela especificação ISOBMFF. Um arquivo de mídia base ISO é estruturado como uma sequência de caixas, que podem conter outras caixas. As caixas geralmente incluem um cabeçalho que fornece o tamanho e o tipo da caixa. O tamanho descreve o tamanho total da caixa, incluindo o cabeçalho, campos e todas as caixas contidas na caixa. As caixas com um tipo que não é reconhecido por um dispositivo participante são normalmente ignoradas e omitidas.

[0216] Como ilustrado pelo exemplo da Figura 3, no nível superior do arquivo, um arquivo de mídia base ISO 300 pode incluir uma caixa do tipo arquivo 310, uma caixa de filme 320, e um ou mais fragmentos de filme 330a, 330n. Outras caixas que podem ser incluídas neste nível, mas que não são representadas neste exemplo, incluem caixas de espaço livre, caixas de metadados, caixas de dados de mídia, entre outras.

[0217] Um arquivo de mídia base ISO pode incluir uma caixa do tipo arquivo 310, identificada pelo tipo de caixa "ftyp". A caixa do tipo arquivo 310 identifica uma especificação ISOBMFF que é a mais adequada para analisar o arquivo. "A maioria" neste exemplo significa que o arquivo de mídia base ISO 300 pode ter sido formatado de acordo com uma determinada especificação ISOBMFF, mas é provavelmente

compatível com outras iterações da especificação. Esta especificação mais adequada é referida como a marca principal. Um dispositivo participante pode usar a marca principal para determinar se o dispositivo é capaz de decodificar e exibir os conteúdos do arquivo. A caixa do tipo arquivo 310 também pode incluir um número de versão, que pode ser usado para indicar uma versão da especificação ISOBMFF. A caixa do tipo arquivo 310 também pode incluir uma lista de marcas compatíveis, que inclui uma lista de outras marcas com as quais o arquivo é compatível. Um arquivo de mídia base ISO pode ser compatível com mais de uma marca principal.

[0218] Quando um arquivo de mídia base ISO 300 inclui uma caixa do tipo arquivo 310, há apenas uma caixa do tipo arquivo. Um arquivo de mídia base ISO 300 pode omitir a caixa do tipo arquivo 310, a fim de ser compatível com dispositivos participantes mais antigos. Quando um arquivo de mídia base ISO 300 não inclui um caixa do tipo arquivo 310, um aparelho participante pode assumir uma marca principal padrão (por exemplo "mp41"), versão menor (por exemplo, "0"), e marca compatível (por exemplo, "Mp41"). A caixa do tipo arquivo 310 é normalmente colocada o mais cedo possível no arquivo de mídia base ISO 300.

[0219] Um arquivo de mídia base ISO pode ainda incluir uma caixa de filme 320, que contém os metadados para a apresentação. A caixa de filme 320 é identificada pelo tipo de caixa "moov". A ISO/IEC 14496-12 provê que uma apresentação, se contida em um arquivo ou vários arquivos, pode incluir apenas uma caixa de filme 320. Frequentemente, a caixa de filme 320 está próxima ao começo de um arquivo

de mídia base ISO. A caixa de filme 320 inclui uma caixa de cabeçalho de filme 322, e pode incluir uma ou mais caixas de faixa 324, bem como as outras caixas.

[0220] A caixa de cabeçalho de filme 322, identificada pelo tipo de caixa "mvhd", pode incluir informações que são independentes de mídia e relevantes para a apresentação como um todo. Por exemplo, a caixa de cabeçalho de filme 322 pode incluir informações, tais como um horário de criação, um horário de modificação, uma escala de tempo e/ou uma duração da apresentação, entre outras coisas. A caixa de cabeçalho de filme 322 também pode incluir um identificador que identifica a próxima faixa na apresentação. Por exemplo, o identificador pode apontar para a caixa de faixa 324 contida na caixa de filme 320 no exemplo ilustrado.

[0221] A caixa de faixa 324, identificada pelo tipo de caixa "trak", pode conter as informações de uma faixa para uma apresentação. Uma apresentação pode incluir uma ou mais faixas, em que cada faixa é independente de outras faixas na apresentação. Cada faixa pode incluir as informações espaciais e temporais que são específicas para o conteúdo na faixa, e cada faixa pode ser associada a uma caixa de mídia. Os dados em uma faixa podem ser dados de mídia, em cujo caso a faixa é uma faixa de mídia, ou os dados podem ser informações de compactação para os protocolos de streaming, caso em que a faixa é uma faixa de dicas. Dados de mídia incluem, por exemplo, dados áudio e vídeo. No exemplo ilustrado, a caixa de faixa exemplificativa 324 inclui uma caixa de cabeçalho de faixa 324a e uma caixa de mídia 324b. Uma caixa de faixa pode

incluir outras caixas, tais como uma caixa de referência de faixa, uma caixa de grupo de faixas, uma caixa de edição, uma caixa de dados de usuário, uma caixa meta e outras.

[0222] A caixa de cabeçalho de faixa 324a, identificada pelo tipo de caixa "tkhd", pode especificar as características de uma faixa contida na caixa de faixa 324. Por exemplo, a caixa de cabeçalho de faixa 324a pode incluir um horário de criação, horário de modificação, duração, identificador de faixa, identificador de camada, identificador de grupo, volume, largura e/ou altura da faixa, entre outras coisas. Para uma faixa de mídia, a caixa de cabeçalho de faixa 324a pode ainda identificar se a faixa está habilitada, se a faixa deve ser reproduzida como parte da apresentação, ou se a faixa pode ser usada para pré-visualizar a apresentação, entre outras coisas. Presume-se que a apresentação de uma faixa esteja geralmente no início de uma apresentação. A caixa de faixa 324 pode incluir uma caixa de listagem de edição, não ilustrada aqui, que pode incluir um mapa de cronograma explícito. O mapa de cronograma pode especificar, entre outras coisas, um tempo de deslocamento para a faixa, em que o deslocamento indica um tempo de inicialização, após o inicio da apresentação, para a faixa.

[0223] No exemplo ilustrado, a caixa de faixa 324 também inclui uma caixa de mídia 324b, identificada pelo tipo de caixa "mdia". A caixa de mídia 324b pode conter os objetos e informações sobre os dados de mídia na faixa. Por exemplo, a caixa de mídia 324b pode conter uma caixa de referência de manipulador, que pode identificar o tipo de mídia da faixa e o processo pelo qual a mídia na faixa é

apresentada. Como outro exemplo, a caixa de mídia 324b pode conter uma caixa de informações de mídia, que pode especificar as características da mídia na faixa. A caixa de informações de mídia pode incluir uma tabela de amostras, em que cada amostra descreve um conjunto de dados de mídia (por exemplo, dados de áudio ou vídeo) incluindo, por exemplo, a localização dos dados para a amostra. Os dados de uma amostra são armazenados em uma caixa de dados de mídia, discutida mais abaixo. Como com a maioria das outras caixas, a caixa de mídia 324b também pode incluir uma caixa de cabeçalho de mídia.

[0224] No exemplo ilustrado, o arquivo de mídia base ISO exemplificativo 300 também inclui múltiplos fragmentos 330a, 330b, 330c, 330n da apresentação. Os fragmentos 330a, 330b, 330c, 330n não são caixas ISOBMFF, mas descrevem uma caixa de fragmentos de filme 332 e a caixa de dados de mídia 338 que é referenciada pela caixa de fragmentos de filme 332. A caixa de fragmentos de filme 332 e as caixas de dados de mídia 338 são caixas de nível superior, mas são agrupadas aqui para indicar a relação entre uma caixa de fragmento de filme 332 e uma caixa de dados de mídia 338.

[0225] Uma caixa de fragmentos de filme 332, identificada pelo tipo de caixa "moof", pode estender uma apresentação incluindo informações adicionais que seriam armazenadas na caixa de filme 320. Usando as caixas de fragmentos de filme 332, uma apresentação pode ser construída de forma incremental. Uma caixa de fragmentos de filme 332 pode incluir uma caixa de cabeçalho de fragmentos

de filme 334 e uma caixa de fragmentos de faixa 336, bem como outras caixas não ilustradas aqui.

[0226] A caixa de cabeçalho de fragmentos de filme 334, identificada pelo tipo de caixa "mfhd", pode incluir um número de sequência. Um dispositivo participante pode usar o número de sequência para verificar se o fragmento 330a inclui a próxima peça de dados para a apresentação. Em alguns casos, os conteúdos de um arquivo, ou os arquivos para uma apresentação, podem ser fornecidos a um dispositivo participante fora de ordem. Por exemplo, os pacotes de rede podem frequentemente chegar em uma ordem diferente da ordem na qual os pacotes foram originalmente transmitidos. Nesses casos, o número de sequência pode auxiliar um dispositivo participante na determinação da ordem correta dos fragmentos.

[0227] A caixa de fragmentos de filme 332 também pode incluir uma ou mais caixas de fragmentos de faixas 336, identificadas pelo tipo de caixa "traf". Uma caixa de fragmentos de filmes 332 pode incluir um conjunto de fragmentos de faixa, zero ou mais por faixa. Os fragmentos de faixa podem conter zero ou mais execuções de faixa, cada qual descreve uma execução contígua de amostras para uma faixa. Os fragmentos de faixa podem ser usados para adicionar tempo vazio a uma faixa, além de adicionar amostras à faixa.

[0228] A caixa de dados de mídia 338, identificada pelo tipo de caixa "mdat", contém dados de mídia. Nas faixas de vídeo, a caixa de dados de mídia 338 conteria quadros de vídeo. Uma caixa de dados de mídia pode alternativamente ou adicionalmente incluir dados de áudio.

Uma apresentação pode incluir zero ou mais caixas de dados de mídia, contidas em um ou mais arquivos individuais. Os dados de mídia são descritos por metadados. No exemplo ilustrado, os dados de mídia na caixa de dados de mídia podem ser descritos por metadados incluídos na caixa de fragmentos de faixa 336. Em outros exemplos, os dados de mídia em uma caixa de dados de mídia podem ser descritos por metadados na caixa de filme 320. Os metadados podem se referir a determinados dados de mídia por um deslocamento absoluto dentro do arquivo 300, tal que um cabeçalho de dados de mídia e/ou espaço livre dentro da caixa de dados de mídia 338 pode ser ignorado.

[0229] Outros fragmentos 330b, 330c, 330n no arquivo de mídia base ISO 300 podem conter caixas semelhantes às ilustradas para o primeiro fragmento 330a e/ou podem conter outras caixas.

[0230] O ISOBMFF inclui suporte para streaming de dados de mídia através de uma rede, além de suportar a reprodução local da mídia. O arquivo ou arquivos que incluem uma apresentação de filme podem incluir faixas adicionais, chamadas de faixas de dicas, que contêm instruções que podem auxiliar um servidor de streaming na formação e transmissão do arquivo ou arquivos como pacotes. Essas instruções podem incluir, por exemplo, dados para o servidor enviar (por exemplo, informações de cabeçalho) ou referências a segmentos dos dados de mídia. Um arquivo pode incluir faixas de dicas separadas para diferentes protocolos de streaming. As faixas de dicas também podem ser adicionadas a um arquivo sem a necessidade de reformatar o arquivo.

[0231] No arquivo de mídia exemplificativo 300 ilustrado na Figura 3, a SEI transportando informações essenciais pode ser incluída, por exemplo, em uma entrada de amostra. Por exemplo, uma unidade NAL SEI contendo mensagem SEI transportando informações essenciais pode ser diretamente incluída no conjunto de unidades NAL SEI no registro de configuração do decodificador de uma entrada de amostra HEVC. Em alguns casos, a unidade NAL SEI contendo uma mensagem SEI transportando informações essenciais pode ser incluída como a primeira unidade NAL SEI da matriz de unidades NAL SEI no registro de configuração do decodificador. Em vários exemplos, mensagens SEI transportando informações essenciais podem ser, alternativamente ou adicionalmente, transportadas em caixas de nível superior, tais como a caixa de cabeçalho de filme 322 ou a caixa de cabeçalho de faixa 324a, entre outras.

[0232] Formatos de encapsulamento de streaming, tais como DASH, também podem ser usados para apresentar mensagens SEI transportando informações essenciais a um decodificador. DASH, que é especificado na ISO/IEC 23009-1, é um padrão para aplicativos de streaming HTTP (adaptativos). DASH especifica um formato conhecido como a descrição de apresentação de mídia (MPD), também conhecido como manifesto, e um formato de segmento de mídia. A MPD descreve a mídia disponível em um servidor e permite que um cliente DASH baixe de forma autônoma uma determinada versão de mídia em um determinado momento na mídia.

[0233] O streaming HTTP com base em DASH pode incluir as seguintes etapas:

[0234] 1) Um cliente obtém a MPD de um conteúdo de streaming (por exemplo, um filme). A MPD inclui informações sobre diferentes representações alternativas, tais como, por exemplo, taxa de bits, resolução de vídeo, taxa de quadros, idioma de áudio, do conteúdo de streaming, bem como as URLs dos recursos HTTP (por exemplo, o segmento de inicialização e os segmentos de mídia).

[0235] 2) Com base nas informações da MPD e nas informações locais do cliente, (por exemplo, largura de banda de rede, recursos de decodificação/exibição e/ou preferência do usuário), o cliente solicita a(s) representação(ões) desejada(s), um segmento (ou uma parte dele) de cada vez.

[0236] 3) Quando o cliente detecta uma alteração de largura de banda de rede, solicita segmentos de uma representação diferente com uma taxa de bits de melhor correspondência, idealmente partindo de um segmento que começa com um ponto de acesso aleatório.

[0237] Durante uma sessão de streaming HTTP, para responder à solicitação do usuário para retornar a uma posição passada ou adiantar para uma posição futura, o cliente solicita segmentos passados ou futuros iniciando de um segmento que está próximo da posição desejada. A partir daquele ponto, o cliente pode iniciar em um ponto de acesso aleatório. O usuário também pode solicitar o avanço rápido do conteúdo, o que pode ser realizado solicitando dados suficientes para decodificação apenas das imagens de vídeo intracodificadas ou apenas um subconjunto temporal do fluxo de vídeo.

[0238] A Figura 4 ilustra um exemplo de uma apresentação DASH 400 para streaming de conteúdo de vídeo. O conteúdo de vídeo está contido em uma apresentação de mídia 402, que pode descrever um fluxo de vídeo contíguo. A apresentação de mídia 402 pode incluir uma descrição de apresentação de mídia (MPD), que também pode ser chamada de manifesto. A MPD é um documento, formatado usando, por exemplo, Linguagem de Marcação Extensível (XML), que contém informações sobre segmentos de mídia na apresentação de mídia 402. Essas informações podem incluir, por exemplo, as relações entre segmentos e informações que podem ser utilizadas para escolher entre os segmentos. A MPD também pode incluir outros dados que os dispositivos clientes (por exemplo, dispositivos que recebem o conteúdo) podem usar.

[0239] A apresentação da mídia 402 pode ser dividida em períodos 404a-404c. Um período, como definido por DASH, é um intervalo de tempo dentro da apresentação de mídia 402. A apresentação consiste, portanto, em uma sequência contígua de períodos 404a-404c. Um período 404a-404c pode ser descrito, por exemplo, por um tempo de inicialização, em que o tempo de inicialização indica uma hora em que o intervalo de tempo no período começa. Dito de outra forma, o tempo de inicialização é um deslocamento do tempo zero.

[0240] Dentro de um período 404b (por exemplo, período 2 no exemplo ilustrado), o conteúdo de mídia pode ter um conjunto consistente de codificações, incluindo, por exemplo, uma taxa de bits média, um idioma, uma configuração de legenda etc. O período 404b também pode fornecer uma fonte (por exemplo, um Localizador de Recursos

Uniforme (URL) base) de onde o conteúdo pode ser transmitido por streaming. O conteúdo no período 404b pode ser organizado em conjuntos de adaptação 424, 426. Um conjunto de adaptação representa um conjunto de versões codificadas permutáveis de um ou mais componentes de conteúdo de mídia. Por exemplo, um período pode incluir um conjunto de adaptação 424 para o componente de vídeo principal e um conjunto de adaptação separado 426 para o componente de áudio principal. Quando há outro conteúdo disponível, tais como legendas ou descrições de áudio, cada um deles pode ter um conjunto de adaptação separado.

[0241] Um conjunto de adaptação (por exemplo, o conjunto de adaptação de vídeo 424) pode conter várias representações alternativas 432. Uma representação descreve uma versão codificada de entrega de um ou vários componentes de conteúdo de mídia. Qualquer representação única dentro de um conjunto de adaptação pode ser usada para renderizar os componentes de conteúdo de mídia no período. Diferentes representações em uma adaptação enviada podem ser consideradas perceptivelmente equivalentes, o que significa que um dispositivo cliente pode mudar dinamicamente de uma representação para outra representação dentro do conjunto de adaptação, a fim de se adaptar às condições de rede ou a outros fatores. Por exemplo, cada representação pode ter uma largura de banda específica, e altura e largura de quadro, bem como outras informações, tais como uma taxa de quadro ou tipo de codificação. Uma representação 432 pode ainda incluir informações de segmento 442 que descrevem os segmentos na apresentação de mídia 402.

[0242] As informações de segmento 442 podem descrever um segmento de inicialização 444 e um ou mais segmentos de mídia 446a-446c. O segmento de inicialização 444 pode incluir algum conteúdo que precede a própria mídia. Cada segmento de mídia 446a-446c inclui uma porção do período total 404b. Os segmentos de mídia 446a-446c podem ser descritos por um tempo de inicialização relativo ao início do período 404b, e um local de origem (por exemplo, uma URL).

[0243] ISOBMFF especifica os subparâmetros para o parâmetro 'codecs' do tipo MIME para AVC e HEVC e suas extensões. Ao transmitir conteúdo por streaming usando DASH, o parâmetro do tipo MIME pode ser provido como um atributo (por exemplo, @mimeType) na MPD. Uma unidade NAL SEI contendo uma mensagem SEI transportando informações essenciais pode ser incluída como parte do parâmetro 'codecs' ou um parâmetro do tipo MIME diferente, permitindo assim que essas mensagens SEI sejam extraídas da MPD através do atributo @mimeType.

[0244] O fluxo de transporte MPEG (MPEG-TS) é um formato de encapsulamento pelo qual um fluxo de bits codificado pode ser transmitido através de uma rede. A Figura 5 ilustra um exemplo de fluxo de transporte MPEG 500. MPEG-TS é um formato de recipiente usado para transmitir e armazenar dados de áudio, vídeo e programação. O MPEG-TS especifica um formato de recipiente encapsulando fluxos elementares de compactação, com correção de erros e recursos de sincronização de fluxo para manter a integridade da transmissão quando o sinal é degradado. O MPEG-TS destina-se à otimização através de meios de

transmissão não fiáveis, tais como broadcast terrestre e por satélite.

[0245] Um fluxo de transporte MPEG 500 inclui vários pacotes de transporte 502. Um pacote é uma unidade básica de dados em um fluxo de transporte. O fluxo de transporte em si é uma sequência de pacotes, sem nenhum cabeçalho global. Cada pacote 502 inclui um cabeçalho 504 e uma carga útil 506. O cabeçalho 504 fornece informações sobre o pacote 502, incluindo, por exemplo, descrever os conteúdos da carga útil, a sequência de pacotes com relação a outros pacotes, informações de erro, e assim por diante.

[0246] Dentre outros campos, o cabeçalho 504 pode incluir um identificador de pacote 508 (PID). O identificador de pacote 508 pode descrever os dados incluídos na carga útil 506. Por exemplo, PID = 2 indica que a carga útil 506 inclui alguns ou todos de uma tabela de descritores de fluxo de transporte (TSDT). A tabela de descritores de fluxo de transporte pode conter descritores 510 relativos ao fluxo de transporte 500.

[0247] Em alguns exemplos, mensagens SEI transportando informações essenciais podem ser incluídas como parte de um descritor que contém informações de nível e perfil de codec de vídeo, como um descritor similar. Por ser incluído entre esses descritores, um decodificador pode analisar as mensagens SEI transportando informações essenciais junto com as informações de decodificação.

[0248] Algumas aplicações exemplificativas podem usar Protocolo de Transferência em Tempo Real (RTP) para o transporte de mídia, e Protocolo de Descricao de Sessão (SDP) para sinalizar as informações sobre a mídia. O SDP

pode ser usado para descrever sessões de comunicação multimídia. Tais descrições podem ser utilizadas, por exemplo, para anúncio de sessão, convite para sessão e negociação de parâmetros. O SDP não é usado para entrega da mídia em si, mas pode ser usado entre os terminais para negociação do tipo de mídia, formato e propriedades associadas. Um conjunto de propriedades e parâmetros é frequentemente referido como um perfil de sessão. O SDP era originalmente um componente do Protocolo de Anúncio de Sessão (SAP), mas encontrou outros usos em conjunto com o Protocolo de Transferência em Tempo Real (RTP), Protocolo de Streaming em Tempo Real (STSR), Protocolo de Inicialização de Sessão (SIP) e como um formato autônomo para descrever sessões multicast. O SDP é descrito em RFC 4566.

[0249] A Figura 6 ilustra um exemplo do modelo Interconexão de Sistemas Abertos (OSI) 600 para comunicações de rede. Dentro deste modelo, o SDP é implementado na camada de sessão 610, em que o gerenciamento de conexão, recuperação de erro, segurança, operação remota, e outras funções ocorrem. Os atributos de sessão podem ser trocados na camada de sessão 610.

[0250] O modelo OSI 600 padroniza as funções de comunicação, e a conformidade permite a interoperabilidade entre diversos sistemas de comunicação. Cada camada no modelo 600 serve a camada acima e é servida pela camada abaixo. A camada física 602 define a transmissão e recepção de fluxos de bits brutos através de um meio físico. A camada de link de dados 604 define a transmissão confiável de quadros de dados entre dois nós. As operações, tais como

endereçamento físico, são tratadas na camada de link de dados 604. A camada de rede 606 define a estrutura e gerenciamento de uma rede de múltiplos nós, incluindo endereçamento, roteamento e controle de tráfego. Operações como fragmentação de pacotes e endereçamento lógico ocorrem na camada de rede 606. A camada de transporte 608 define a transmissão confiável de segmentos de dados entre pontos em uma rede, incluindo segmentação, confirmação e multiplexação. Operações como gerenciamento de conexões ponta-a-ponta, segmentação de mensagens, sequenciamento de mensagens, confiabilidade e controle de fluxo podem ocorrer na camada de transporte 608. A camada de sessão 610 define o gerenciamento de sessões, que são trocas contínuas de informações na forma de múltiplas transmissões consecutivas entre dois nós. Como observado anteriormente, operações, tais como o gerenciamento de conexão, recuperação de erros, segurança e operação remota, podem ocorrer na camada de sessão 610. A camada de apresentação 612 define a tradução de dados entre um serviço de rede e um aplicativo, incluindo codificação de caracteres, compressão de dados e criptografia e/ou descriptografia. A camada de aplicação 614 inclui Interfaces de Programação de Aplicativo (APIs) de alto nível, incluindo compartilhamento de recursos, acesso remoto a arquivos e outras operações.

[0251] Em alguns exemplos, mensagens SEI transportando informações essenciais podem ser incluídas em um arquivo SDP como um atributo SDP. Abaixo está um exemplo dos campos em uma descrição de sessão. Os campos opcionais são especificados com “=*” e, no exemplo abaixo, as mensagens SEI transportando informações essenciais podem

ser especificadas usando as linhas de atributo de sessão e/ou as linhas de atributo de mídia.

[0252] Descrição de Sessão

v = (número da versão do protocolo, atualmente apenas 0)

o = (identificador de originador e sessão: nome de usuário, id, número da versão, endereço de rede)

s = (nome da sessão: obrigatório com pelo menos um caractere codificado em UTF-8)

i =* (título da sessão ou informações curtas)

u =* (URI da descrição)

e =* (zero ou mais endereços de e-mail com nome opcional de contatos)

p =* (zero ou mais números de telefone com nome opcional de contatos)

c =* (informações de conexão - não necessárias se incluídas em todas as mídias)

b =* (zero ou mais linhas de informações de largura de banda)

Uma ou mais descrições de tempo (linhas "t=" e "r="; veja abaixo)

z =* (ajustes de fuso horário)

k =* (chave de criptografia)

a =* (zero ou mais linhas de atributo de sessão)

Zero ou mais descrições de mídia (cada uma iniciando com uma linha "m ="; veja abaixo)

[0253] Descrição de Tempo

t = (tempo em que a sessão está ativa)

r =* (zero ou mais vezes de repetição)

[0254] Descrição de Mídia

m = (nome da mídia e endereço de transporte)
i =* (título da mídia ou campo de informações)
c =* (informações de conexão - opcional se incluídas no nível de sessão)

b =* (zero ou mais linhas de informações sobre largura de banda)

k =* (chave de criptografia)

a =* (zero ou mais linhas de atributo de mídia - substituindo as linhas de atributo de Sessão)

[0255] A Figura 7 ilustra um exemplo de um processo 700 para codificação de dados de vídeo. O método pode ser implementado por um sistema de codificação de vídeo, tal como descrito neste documento.

[0256] Em 702, o processo 700 inclui receber dados de vídeo. Os dados de vídeo podem ser recebidos, por exemplo, por um dispositivo de captura de vídeo de um sistema de codificação de vídeo. Como outro exemplo, os dados de vídeo podem ser recebidos de um dispositivo de armazenamento, tal como um disco rígido ou dispositivo de memória flash, e/ou através de uma rede.

[0257] Em 704, o processo 700 inclui determinar, por um dispositivo de codificação, se as informações associadas com os dados de vídeo são necessárias para apresentar os dados de vídeo, em que as informações não são necessárias para decodificar os dados de vídeo. O dispositivo de codificação pode ser parte do sistema de codificação de vídeo. As informações podem ser recebidas ao mesmo tempo que os dados de vídeo são recebidos. Alternativamente, as informações podem ser providas ao dispositivo de codificação antes de o dispositivo de

codificação receber os dados de vídeo. Por exemplo, o dispositivo de codificação pode ser configurado previamente com as informações. Alternativamente, as informações podem ser um componente dos dados de vídeo.

[0258] Em vários exemplos, as informações, quando usadas por um dispositivo de decodificação, podem permitir que o dispositivo de decodificação processe e/ou exiba os dados de vídeo de determinada maneira. Por exemplo, as informações podem incluir um arranjo de compactação de quadro, que indica a maneira em que uma imagem foi compactada em um quadro antes de o quadro ser codificado. Como outro exemplo, as informações podem incluir uma orientação de exibição, que indica uma orientação desejada (por exemplo, uma rotação) do visor no qual os dados de vídeo devem ser apresentados. Como outro exemplo, as informações incluem um arranjo de compactação de quadro retangular segmentado, que pode indicar que a imagem foi segmentada antes de ser compactada em um quadro, e a maneira em que os segmentos foram compactados no quadro, antes de o quadro ser codificado. Como outro exemplo, as informações podem incluir uma indicação de projeção omnidirecional, que pode indicar a um dispositivo de decodificação que os dados são dados de vídeo 360. Nesses e em outros exemplos, em alguns casos, o dispositivo de decodificação pode usar as informações para determinar se o dispositivo de decodificação é capaz de exibir os dados de vídeo.

[0259] Em vários exemplos, o dispositivo de codificação é configurado com um ou mais parâmetros que o dispositivo de codificação pode usar para determinar se as

informações são necessárias para apresentar os dados de vídeo. Esses parâmetros podem incluir, por exemplo, características de grão de filme ou mapeamento de tom para aplicar aos dados de vídeo, um volume de cor ou mapeamento de cor usado para produzir os dados de vídeo, e/ou uma configuração de compactação de quadro, entre outras coisas.

[0260] Em 706, o processo 700 inclui gerar uma mensagem de informações suplementares de aprimoramento (SEI), em que a mensagem SEI inclui um campo indicando se as informações são necessárias para apresentar os dados de vídeo. Por exemplo, a mensagem SEI pode ser uma mensagem SEI manifesto, que pode incluir um campo identificando tipos de mensagens SEI incluídas com os dados de vídeo, e um campo indicando a essência de cada uma dessas mensagens SEI. Como outro exemplo, a mensagem SEI pode ser uma mensagem SEI prefixo que pode incluir informações essenciais. Como outro exemplo, a mensagem SEI pode ser uma mensagem SEI ESI. Como outro exemplo, a mensagem SEI pode ser uma mensagem SEI NSI.

[0261] Em alguns exemplos, um elemento de sintaxe da mensagem SEI indica um tipo da mensagem SEI. Nesses exemplos, um dispositivo de decodificação pode usar o elemento de sintaxe para identificar a mensagem SEI como diferente de uma mensagem SEI, tal como é definida na Tabela 1.

[0262] Em 708, o processo 700 inclui codificar os dados de vídeo. Os dados de vídeo podem ser codificados usando, por exemplo, os padrões HEVC ou AVC, ou outro padrão de codificação de vídeo. Em vários exemplos, codificar os dados de vídeo pode produzir um fluxo de bits

de vídeo codificado, em que o fluxo de bits de vídeo codificado inclui uma versão comprimida dos dados de vídeo.

[0263] Em 710, o processo 700 inclui incluir a mensagem SEI com os dados de vídeo codificados. Em alguns exemplos, a mensagem SEI pode ser incluída nos dados de vídeo codificados. Em alguns exemplos, a mensagem SEI pode ser incluída juntamente com os dados de vídeo codificados em um formato encapsulado.

[0264] Por exemplo, o processo 700 pode incluir gravar os dados de vídeo codificados e a mensagem SEI em um arquivo, em que os dados de vídeo codificados e a mensagem SEI são gravados no arquivo de acordo com um formato de arquivo. Nesses exemplos, o processo 700 pode ainda incluir usar o formato de arquivo para colocar a mensagem SEI no arquivo em que a SEI pode ser lida antes de os dados de vídeo codificados serem lidos.

[0265] Como outro exemplo, o processo 700 pode incluir encapsular os dados de vídeo codificados para streaming, em que os dados de vídeo codificados são encapsulados de acordo com um formato de streaming. Neste exemplo, o processo 700 pode ainda incluir gerar, de acordo com o formato de streaming, uma descrição de dados de vídeo codificados, em que a descrição inclui a mensagem SEI.

[0266] Como outro exemplo, o processo 700 pode incluir encapsular os dados de vídeo codificados para transmissão através de uma rede, em que os dados de vídeo codificados são encapsulados de acordo com um formato do recipiente. Neste exemplo, o processo 700 pode ainda incluir gerar, de acordo com o formato do recipiente, um primeiro pacote, em que uma porção de carga útil do

primeiro pacote inclui a mensagem SEI como um descritor. O processo 700 pode ainda incluir gerar um ou mais pacotes incluindo os dados de vídeo codificados, em que os um ou mais pacotes seguem o primeiro pacote em ordem de transmissão.

[0267] Como outro exemplo, o processo 700 pode ainda incluir encapsular os dados de vídeo codificados para transmissão através de uma rede, em que os dados de vídeo codificados são encapsulados de acordo com um protocolo de transferência em tempo real. Neste exemplo, o processo 700 pode ainda incluir gerar, de acordo com um protocolo de descrição de sessão, uma descrição de sessão, em que a descrição de sessão inclui a mensagem SEI.

[0268] Em alguns exemplos, o dispositivo codificado determina que as informações são necessárias para apresentar os dados de vídeo. Nesses exemplos, o campo na mensagem SEI pode indicar que informações essenciais são incluídas nos dados de vídeo codificados ou são providas com os dados de vídeo codificados. Nesses exemplos, um dispositivo de decodificação pode usar a mensagem SEI para identificar e localizar as informações essenciais.

[0269] Em alguns exemplos, o dispositivo de codificação determina que as informações não são necessárias para apresentar os dados de vídeo. Nesses exemplos, o campo na mensagem SEI pode indicar que nenhuma informação essencial está incluída em ou com os dados de vídeo codificados.

[0270] Em alguns exemplos, as informações são incluídas com os dados de vídeo codificados. Por exemplo, as informações podem ser incluídas na mensagem SEI.

Alternativamente ou adicionalmente, as informações podem ser incluídas em uma ou mais mensagens SEI adicionais. Em alguns exemplos, as mensagens SEI adicionais podem identificar o tipo de informação que cada mensagem SEI transporta. Em alguns exemplos, cada uma das mensagens SEI adicionais pode indicar a essência das informações transportadas por cada mensagem SEI. Em alguns exemplos, a essência de cada mensagem SEI é indicada na mensagem SEI gerada conforme discutido acima.

[0271] A Figura 8 ilustra um exemplo de um processo 800 para decodificação de dados de vídeo. O processo 800 pode ser implementado, por exemplo, por um dispositivo de codificação de vídeo, tal como é descrito neste documento.

[0272] Em 802, o processo 800 inclui receber, em um dispositivo de decodificação, um fluxo de bits de vídeo codificado, uma mensagem de informações suplementares de aprimoramento (SEI) e as informações associadas com o fluxo de bits de vídeo codificado. Em alguns exemplos, o dispositivo de decodificação é parte do dispositivo de codificação de vídeo. Em alguns exemplos, o dispositivo de decodificação recebe o fluxo de bits de vídeo codificado de um dispositivo de armazenamento e/ou através de uma rede. Em alguns exemplos, o dispositivo de decodificação recebe o fluxo de bits de vídeo codificado de um dispositivo de codificação.

[0273] Em alguns exemplos, a mensagem SEI e/ou as informações associadas com o fluxo de bits de vídeo codificado são incluídas no fluxo de bits de vídeo codificado. Em alguns exemplos, a mensagem SEI e/ou as

informações são providas com o fluxo de bits de vídeo codificado, por exemplo, em um formato encapsulado que inclui o fluxo de bits de vídeo codificado.

[0274] Por exemplo, o fluxo de bits de vídeo codificado, a mensagem SEI e as informações associadas com o fluxo de bits de vídeo codificado podem ser recebidos em um arquivo, em que o arquivo é formatado de acordo com um formato de arquivo. Neste exemplo, o formato de arquivo pode prover que a mensagem SEI esteja em uma parte do arquivo que é lida pelo dispositivo de decodificação antes de uma parte do arquivo que inclui o fluxo de bits de vídeo codificado.

[0275] Como outro exemplo, o fluxo de bits de vídeo codificado, a mensagem SEI e as informações associadas com o fluxo de bits de vídeo codificado podem ser recebidos em um fluxo de dados, em que o fluxo de dados é formatado de acordo com um formato de streaming. Neste exemplo, o dispositivo de decodificação pode ler, a partir do fluxo de dados, uma descrição dos dados de vídeo codificados, em que a descrição inclui a mensagem SEI.

[0276] Como outro exemplo, o fluxo de bits de vídeo codificado, a mensagem SEI e as informações associadas com o fluxo de bits de vídeo codificado podem ser recebidos em uma pluralidade de pacotes de rede. Neste exemplo, o dispositivo de decodificação pode ler a mensagem SEI de uma porção de carga útil de um primeiro pacote da pluralidade de pacotes de rede. A mensagem SEI pode, por exemplo, ser provida como um descritor na carga útil. Neste exemplo, o dispositivo de decodificação pode ainda ler os dados de vídeo codificados a partir de um ou mais pacotes

da pluralidade de pacotes de rede, em que os um ou mais pacotes seguem o primeiro pacote em ordem de transmissão.

[0277] Como outro exemplo, o fluxo de bits de vídeo codificado, a mensagem SEI e as informações associadas com o fluxo de bits de vídeo codificado podem ser recebidos através de uma rede. Neste exemplo, o fluxo de bits de vídeo codificado pode ser encapsulado de acordo com um protocolo de transferência em tempo real, e a mensagem SEI pode ser incluída em uma descrição de sessão, em que a descrição de sessão é formatada de acordo com um protocolo de descrição de sessão.

[0278] Em 804, o processo 800 inclui determinar, usando a mensagem SEI, se as informações são necessárias para apresentar dados de vídeo no fluxo de bits de vídeo codificado, em que as informações não são necessárias para decodificar os dados de vídeo, e em que a mensagem SEI inclui um campo indicando se as informações são necessárias para apresentar os dados de vídeo.

[0279] Em alguns exemplos, um elemento de sintaxe da mensagem SEI indica um tipo da mensagem SEI. Nesses exemplos, o dispositivo de decodificação pode usar o elemento de sintaxe para determinar que a mensagem SEI pode incluir uma indicação de se as informações são necessárias para apresentar os dados de vídeo.

[0280] Em alguns exemplos, as informações são incluídas na mensagem SEI e/ou em uma ou mais mensagens SEI adicionais incluídas com os dados de vídeo codificados. Por exemplo, a mensagem SEI pode incluir uma listagem dos tipos de mensagens SEI incluídos com os dados de vídeo codificados. Neste exemplo, a listagem pode ainda indicar a

essência dos dados incluídos em cada uma das mensagens SEI. Como outro exemplo, o tipo de mensagem SEI pode indicar se a mensagem SEI inclui ou não inclui informações essenciais.

[0281] Em 806, o processo 800 inclui decodificar o fluxo de bits de vídeo codificado para produzir dados de vídeo, em que o fluxo de bits de vídeo codificado é decodificado de acordo com um resultado da determinação de se as informações são necessárias para apresentar os dados de vídeo.

[0282] Em alguns exemplos, o dispositivo de decodificação determina que as informações são necessárias para apresentar os dados de vídeo. Nesses exemplos, o dispositivo de decodificação configura os dados de vídeo de acordo com as informações. Por exemplo, ao decodificar os dados de vídeo codificados, o dispositivo de decodificação pode rearranjar os conteúdos de um quadro de vídeo de acordo com um formato de compactação de quadro. Como outro exemplo, o dispositivo de decodificação pode dispor um quadro de vídeo para exibição usando um visor de 360 graus. Como outro exemplo, o dispositivo de decodificação pode prover informações de cor para um dispositivo de exibição, em que os dados de vídeo podem ser exibidos usando as informações de cor.

[0283] Em alguns exemplos, o dispositivo de decodificação determina que as informações não são necessárias para apresentar os dados de vídeo. Nesses exemplos, o dispositivo de decodificação pode decodificar os dados de vídeo codificados sem usar as informações.

[0284] Em alguns exemplos, os processos 700, 800 podem ser executados por um aparelho ou um dispositivo de

computação, tal como o sistema de codificação de vídeo 100. Por exemplo, os processos 700, 800 podem ser executados pelo sistema de codificação de vídeo 100 mostrado na Figura 1. Em alguns casos, o aparelho ou dispositivo de computação pode incluir um processador, microprocessador, microcomputador ou outro componente de um dispositivo que é configurado para realizar as etapas dos processos 700, 800. Em alguns exemplos, o aparelho ou dispositivo de computação pode incluir uma câmera configurada para capturar dados de vídeo (por exemplo, uma sequência de vídeo) incluindo quadros de vídeo. Por exemplo, o dispositivo de computação pode incluir um dispositivo de câmera (por exemplo, uma câmera IP ou outro tipo de dispositivo de câmera) que pode incluir um codec de vídeo. Em alguns exemplos, uma câmera ou outro dispositivo de captura que captura os dados de vídeo é separado do dispositivo de computação, in que case o dispositivo de computação recebe os dados de vídeo capturados. O dispositivo de computação pode ainda incluir uma interface de rede configurada para comunicar os dados de vídeo. A interface de rede pode ser configurada para comunicar dados baseados em Protocolo de Internet (IP).

[0285] Os processos 700, 800 são ilustrados como fluxogramas lógicos, cuja operação representa uma sequência de operações que podem ser implementadas em hardware, instruções de computador ou uma combinação destes. No contexto das instruções de computador, as operações representam instruções executáveis por computador armazenadas em um ou mais meios de armazenamento legíveis por computador que, quando executadas por um ou mais processadores, executam as operações citadas. Geralmente,

as instruções executáveis por computador incluem rotinas, programas, objetos, componentes, estruturas de dados e similares que executam funções específicas ou implementam tipos de dados específicos. A ordem em que as operações são descritas não pretende ser considerada uma limitação, e qualquer número das operações descritas pode ser combinado em qualquer ordem e/ou em paralelo para implementar os processos.

[0286] Adicionalmente, os processos 700, 800 podem ser executados sob o controle de um ou mais sistemas de computador configurados com instruções executáveis e podem ser implementados como código (por exemplo, instruções executáveis, um ou mais programas de computador ou uma ou mais aplicações) sendo executados coletivamente em um ou mais processadores, por hardware, ou combinações dos mesmos. Como observado acima, o código pode ser armazenado em um meio de armazenamento legível por computador ou legível por máquina, por exemplo, na forma de um programa de computador que compreende uma pluralidade de instruções executáveis por um ou mais processadores. O meio de armazenamento legível por computador ou legível por máquina pode ser não transitório.

[0287] Os métodos e operações discutidos aqui podem ser implementados usando vídeo compactado, e podem ser implementados em um sistema de codificação e decodificação de vídeo exemplificativo (por exemplo, sistema 100). Em alguns exemplos, um sistema inclui um dispositivo fonte que fornece dados de vídeo codificados a serem decodificados posteriormente por um dispositivo de destino. Em particular, o dispositivo fonte fornece os

dados de vídeo ao dispositivo de destino através de um meio legível por computador. O dispositivo fonte e o dispositivo de destino podem compreender uma ampla variedade de dispositivos, incluindo computadores de mesa, notebooks (isto é, laptops), tablets, set-top boxes, aparelhos de telefone, tais como os chamados "smartphones", os chamados "smart" pads, televisões, câmeras, dispositivos de exibição, reprodutores de mídia digital, consoles de videogame, dispositivo de streaming de vídeo etc. Em alguns casos, o dispositivo fonte e o dispositivo de destino podem ser equipados para comunicação sem fio.

[0288] O dispositivo de destino pode receber os dados de vídeo codificados a serem decodificados através do meio legível por computador. O meio legível por computador pode compreender qualquer tipo de meio ou dispositivo capaz de mover os dados de vídeo codificados do dispositivo fonte para o dispositivo de destino. Em um exemplo, um meio legível por computador pode compreender um meio de comunicação para habilitar o dispositivo fonte para transmitir dados de vídeo codificados diretamente ao dispositivo de destino em tempo real. Os dados de vídeo codificados podem ser modulados de acordo com um padrão de comunicação, tal como um protocolo de comunicação sem fio, e transmitidos ao dispositivo de destino. O meio de comunicação pode compreender qualquer meio de comunicação sem fio ou com fio, tal como um espectro de radiofrequência (RF) ou uma ou mais linhas de transmissão físicas. O meio de comunicação pode fazer parte de uma rede baseada em pacotes, tal como uma rede de área local, uma rede de área ampla ou uma rede global, tal como a Internet. O meio de

comunicação pode incluir roteadores, comutadores, estações de base ou qualquer outro equipamento que possa ser útil para facilitar a comunicação do dispositivo fonte com o dispositivo de destino.

[0289] Em alguns exemplos, os dados codificados podem ser enviados da interface de saída para um dispositivo de armazenamento. Da mesma forma, os dados codificados podem ser acessados a partir do dispositivo de armazenamento pela interface de entrada. O dispositivo de armazenamento pode incluir qualquer um de uma variedade de meios de armazenamento de dados distribuídos ou acessados localmente, tais como um disco rígido, discos Blu-ray, DVDs, CD-ROMs, memória flash, memória volátil ou não volátil ou qualquer outro meio de armazenamento digital adequado para armazenamento de dados de vídeo codificados. Em outro exemplo, o dispositivo de armazenamento pode corresponder a um servidor de arquivos ou outro dispositivo de armazenamento intermediário que possa armazenar o vídeo codificado gerado pelo dispositivo fonte. O dispositivo de destino pode acessar os dados de vídeo armazenados no dispositivo de armazenamento através de streaming ou download. O servidor de arquivos pode ser qualquer tipo de servidor capaz de armazenar dados de vídeo codificados e transmitir tais dados de vídeo codificados ao dispositivo de destino. Servidores de arquivos exemplificativos incluem um servidor Web (por exemplo, para um site), um servidor FTP, dispositivos de armazenamento ligado em rede (NAS) ou uma unidade de disco local. O dispositivo de destino pode acessar os dados de vídeo codificados através de qualquer conexão de dados padrão, incluindo uma conexão com a

Internet. Isso pode incluir um canal sem fio (por exemplo, uma conexão Wi-Fi), uma conexão com fio (por exemplo, DSL, modem a cabo etc.) ou uma combinação de ambas que seja adequada para acessar dados de vídeo codificados armazenados em um servidor de arquivos. A transmissão de dados de vídeo codificados do dispositivo de armazenamento pode ser uma transmissão de streaming, uma transmissão de download ou uma combinação destas

[0290] As técnicas da presente invenção não são necessariamente limitadas a aplicações ou configurações sem fio. As técnicas podem ser aplicadas à codificação de vídeo em suporte a diversas aplicações multimídia, tais como broadcasts de televisão pelo ar, transmissões de televisão a cabo, transmissões de televisão por satélite, transmissões de vídeo por streaming pela Internet, tais como streaming adaptativo dinâmico sobre HTTP (DASH), vídeo digital que é codificado em um meio de armazenamento de dados, decodificação de vídeo digital armazenado em um meio de armazenamento de dados, ou outras aplicações. Em alguns exemplos, o sistema pode ser configurado para suportar transmissão de vídeo unidirecional ou bidirecional para suportar aplicações como streaming de vídeo, reprodução de vídeo, broadcast de vídeo e/ou videotelefonia.

[0291] Em um exemplo, o dispositivo fonte inclui uma fonte de vídeo, um codificador de vídeo e uma interface de saída. O dispositivo de destino pode incluir uma interface de entrada, um decodificador de vídeo e um dispositivo de exibição. O codificador de vídeo do dispositivo fonte pode ser configurado para aplicar as técnicas aqui divulgadas. Em outros exemplos, um

dispositivo fonte e um dispositivo de destino podem incluir outros componentes ou disposições. Por exemplo, o dispositivo fonte pode receber dados de vídeo de uma fonte de vídeo externa, tal como uma câmera externa. Da mesma forma, o dispositivo de destino pode fazer interface com um dispositivo de exibição externo, em vez de incluir um dispositivo de exibição integrado.

[0292] O sistema exemplificativo acima é apenas um exemplo. As técnicas para processamento de dados de vídeo em paralelo podem ser executadas por qualquer dispositivo de codificação e/ou decodificação de vídeo digital. Embora geralmente as técnicas da presente invenção sejam executadas por um dispositivo de codificação de vídeo, as técnicas também podem ser executadas por um codificador/decodificador de vídeo, normalmente chamado de "CODEC". Além disso, as técnicas da presente invenção também podem ser executadas por um pré-processador de vídeo. O dispositivo fonte e o dispositivo de destino são apenas exemplos desses dispositivos de codificação nos quais o dispositivo fonte gera dados de vídeo codificados para transmissão ao dispositivo de destino. Em alguns exemplos, os dispositivos fonte e de destino podem operar de maneira substancialmente simétrica, de modo que cada dispositivo inclua componentes de codificação e decodificação de vídeo. Portanto, sistemas exemplificativos podem suportar transmissão de vídeo unidirecional ou bidirecional entre dispositivos de vídeo, por exemplo, para streaming de vídeo, reprodução de vídeo, broadcast de vídeo ou videotelefonia.

[0293] A fonte de vídeo pode incluir um dispositivo de captura de vídeo, tal como uma câmera de vídeo, um arquivo de vídeo contendo vídeo capturado anteriormente e/ou uma interface de alimentação de vídeo para receber vídeo de um provedor de conteúdo de vídeo. Como outra alternativa, a fonte de vídeo pode gerar dados baseados em gráficos de computador como o vídeo fonte ou uma combinação de vídeo ao vivo, vídeo arquivado e vídeo gerado por computador. Em alguns casos, se a fonte de vídeo for uma câmera de vídeo, o dispositivo fonte e o dispositivo de destino podem formar os chamados telefones com câmera ou videofones. Conforme mencionado acima, no entanto, as técnicas descritas neste relatório podem ser aplicáveis à codificação de vídeo em geral, e podem ser aplicadas a aplicações com e/ou sem fio. Em cada caso, o vídeo capturado, pré-capturado ou gerado por computador pode ser codificado pelo codificador de vídeo. As informações de vídeo codificado podem, então, ser enviadas pela interface de saída para o meio legível por computador.

[0294] Como observado, o meio legível por computador pode incluir meio transitório, tal como um broadcast sem fio ou transmissão de rede com fio, ou meio de armazenamento (ou seja, meio de armazenamento não transitório), tal como um disco rígido, unidade flash, disco compacto, disco de vídeo digital, disco Blu-ray ou outro meio legível por computador. Em alguns exemplos, um servidor de rede (não mostrado) pode receber dados de vídeo codificados do dispositivo fonte e fornecer os dados de vídeo codificados ao dispositivo de destino, por exemplo, através de transmissão de rede. Da mesma forma, um

dispositivo de computação de uma facilidade de produção de meio, tal como uma facilidade de estampagem de disco, pode receber dados de vídeo codificados do dispositivo fonte e produzir um disco contendo os dados de vídeo codificados. Portanto, pode-entender que o meio legível por computador inclui um ou mais meios legíveis por computador de várias formas, em vários exemplos.

[0295] A interface de entrada do dispositivo de destino recebe informações do meio legível por computador. As informações do meio legível por computador podem incluir informações de sintaxe definidas pelo codificador de vídeo, que também são usadas pelo decodificador de vídeo, que incluem elementos de sintaxe que descrevem as características e/ou o processamento de blocos e outras unidades codificadas, por exemplo, grupo de imagens (GOP). Um dispositivo de exibição exibe os dados de vídeo decodificados a um usuário e pode incluir qualquer um de uma variedade de dispositivos de exibição, tais como um tubo de raios catódicos (CRT), uma tela de cristal líquido (LCD), uma tela de plasma, uma tela de diodo orgânico emissor de luz (OLED), ou qualquer outro tipo de dispositivo de exibição. Vários exemplos foram descritos.

[0296] Detalhes específicos do dispositivo de codificação 104 e do dispositivo de decodificação 112 são mostrados na Figura 9 e na Figura 10, respectivamente. A Figura 9 é um diagrama de blocos ilustrando um dispositivo de codificação exemplificativo 104 que pode implementar uma ou mais das técnicas descritas neste relatório. O dispositivo de codificação 104 pode, por exemplo, gerar as estruturas de sintaxe aqui descritas (por exemplo, as

estruturas de sintaxe de um VPS, SPS, PPS ou outros elementos de sintaxe). O dispositivo de codificação 104 pode executar codificação intraprevisão e interprevisão de blocos de vídeo dentro de fatias de vídeo. Conforme descrito anteriormente, a intracodificação depende, pelo menos em parte, de previsão espacial para reduzir ou remover redundância espacial dentro de um determinado quadro de vídeo ou imagem. A intercodificação depende, pelo menos em parte, de previsão temporal para reduzir ou remover redundância temporal dentro de quadros próximos ou adjacentes de uma sequência de vídeo. O intramodo (modo I) pode se referir a vários modos de compressão com base em espaço. Os intermodos, tais como previsão unidirecional (modo P) ou biprevisão (modo B), podem se referir a vários modos de compressão com base em tempo.

[0297] O dispositivo de codificação 104 inclui uma unidade de particionamento 35, unidade de processamento de previsão 41, unidade de filtro 63, memória de imagem 64, adicionador (summer) 50, unidade de processamento de transformada 52, unidade de quantização 54 e unidade de codificação por entropia 56. A unidade de processamento de previsão 41 inclui unidade de estimativa de movimento 42, unidade de compensação de movimento 44 e unidade de processamento intraprevisão 46. Para a reconstrução de bloco de vídeo, o dispositivo de codificação 104 também inclui unidade de quantização inversa 58, unidade de processamento de transformada inversa 60 e adicionador 62. A unidade de filtro 63 destina-se a representar um ou mais filtros de loop, tais como um filtro de desbloqueio, um filtro de loop adaptativo (ALF) e um filtro de deslocamento

adaptativo de amostra (SAO). Embora a unidade de filtro 63 seja mostrada na Figura 9 como sendo um filtro de loop, em outras configurações, a unidade de filtro 63 pode ser implementada como um filtro pós-loop. Um dispositivo de pós-processamento 57 pode executar um processamento adicional em dados de vídeo codificados gerados pelo dispositivo de codificação 104. As técnicas da presente invenção podem, em alguns casos, ser implementadas pelo dispositivo de codificação 104. Em outros casos, no entanto, uma ou mais das técnicas da presente invenção podem ser implementadas pelo dispositivo de pós-processamento 57.

[0298] Como mostrado na Figura 9, o dispositivo de codificação 104 recebe dados de vídeo, e a unidade de particionamento 35 particiona os dados em blocos de vídeo. O particionamento também pode incluir particionamento em fatias, segmentos de fatia, mosaicos ou outras unidades maiores, bem como particionamento de bloco de vídeo, por exemplo, de acordo com uma estrutura de quadtree de LCUs e CUs. O dispositivo de codificação 104 geralmente ilustra os componentes que codificam os blocos de vídeo dentro de uma fatia de vídeo a ser codificada. A fatia pode ser dividida em vários blocos de vídeo (e possivelmente em conjuntos de blocos de vídeo chamados de mosaicos). A unidade de processamento de previsão 41 pode selecionar um de uma pluralidade de modos de codificação possíveis, tais como um de uma pluralidade de modos de codificação intraprevisão ou um de uma pluralidade de modos de codificação interprevisão, para o bloco de vídeo atual com base nos resultados de erro (por exemplo, taxa de codificação e

nível de distorção). A unidade de processamento de previsão 41 pode fornecer o bloco intra- ou intercodificado resultante para o adicionador 50 para gerar dados de bloco residuais e para o adicionador 62 para reconstruir o bloco codificado para uso como uma imagem de referência.

[0299] A unidade de processamento intraprevisão 46 dentro da unidade de processamento de previsão 41 pode executar codificação intraprevisão do bloco de vídeo atual em relação a um ou mais blocos adjacentes no mesmo quadro ou fatia que o bloco atual a ser codificado para fornecer compressão espacial. A unidade de estimativa de movimento 42 e a unidade de compensação de movimento 44 dentro da unidade de processamento de previsão 41 executam a codificação interpreditiva do bloco de vídeo atual em relação a um ou mais blocos preditivos em uma ou mais imagens de referência para fornecer compressão temporal.

[0300] A unidade de estimativa de movimento 42 pode ser configurada para determinar o modo de interprevisão para uma fatia de vídeo de acordo com um padrão predeterminado para uma sequência de vídeo. O padrão predeterminado pode designar fatias de vídeo na sequência como fatias P, fatias B ou fatias GPB. A unidade de estimativa de movimento 42 e a unidade de compensação de movimento 44 podem ser altamente integradas, mas são ilustradas separadamente para fins conceituais. A estimativa de movimento, realizada pela unidade de estimativa de movimento 42, é o processo de geração de vetores de movimento, que estima o movimento para blocos de vídeo. Um vetor de movimento, por exemplo, pode indicar o deslocamento de uma unidade de previsão (PU) de um bloco de

vídeo dentro de um quadro de vídeo ou imagem atual em relação a um bloco preditivo dentro de uma imagem de referência.

[0301] Um bloco preditivo é um bloco que corresponde à PU do bloco de vídeo a ser codificado em termos de diferença de pixels, que pode ser determinada pela soma da diferença absoluta (SAD), soma da diferença quadrada (SSD) ou outras métricas de diferença. Em alguns exemplos, o dispositivo de codificação 104 pode calcular valores para posições de pixels subinteiros de imagens de referência armazenadas na memória de imagem 64. Por exemplo, o dispositivo de codificação 104 pode interpolar valores de posições de um quarto de pixel, posições de um oitavo de pixel ou outras posições de pixel fracionárias da imagem de referência. Portanto, a unidade de estimativa de movimento 42 pode executar uma pesquisa de movimento em relação às posições completas de pixel e posições fracionárias de pixel e gerar um vetor de movimento com precisão de pixel fracionária.

[0302] A unidade de estimativa de movimento 42 calcula um vetor de movimento para uma PU de um bloco de vídeo em uma fatia intercodificada comparando a posição da PU com a posição de um bloco preditivo de uma imagem de referência. A imagem de referência pode ser selecionada de uma primeira lista de imagens de referência (Lista 0) ou de uma segunda lista de imagens de referência (Lista 1), cada uma das quais identifica uma ou mais imagens de referência armazenadas na memória de imagem 64. A unidade de estimativa de movimento 42 envia o vetor de movimento

calculado para a unidade de codificação por entropia 56 e a unidade de compensação de movimento 44.

[0303] A compensação de movimento, realizada pela unidade de compensação de movimento 44, pode envolver a pesquisa e carregamento (fetching) ou geração do bloco preditivo com base no vetor de movimento determinado pela estimativa de movimento, possivelmente executando interpolações com precisão de subpixel. Ao receber o vetor de movimento para a PU do bloco de vídeo atual, a unidade de compensação de movimento 44 pode localizar o bloco preditivo para o qual o vetor de movimento aponta em uma lista de imagens de referência. O dispositivo de codificação 104 forma um bloco de vídeo residual subtraindo os valores de pixel do bloco preditivo dos valores de pixel do bloco de vídeo atual sendo codificado, formando valores de diferença de pixel. Os valores de diferença de pixel formam dados residuais para o bloco e podem incluir os componentes de diferença de luma e croma. O adicionador 50 representa o componente ou componentes que executam essa operação de subtração. A unidade de compensação de movimento 44 também pode gerar elementos de sintaxe associados aos blocos de vídeo e à fatia de vídeo para uso pelo dispositivo de decodificação 112 na decodificação dos blocos de vídeo da fatia de vídeo.

[0304] A unidade de processamento intraprevisão 46 pode intraprever um bloco atual, como uma alternativa à interprevação realizada pela unidade de estimativa de movimento 42 e pela unidade de compensação de movimento 44, como descrito acima. Em particular, a unidade de processamento intraprevisão 46 pode determinar um modo de

intraprevisão a ser usado para codificar um bloco atual. Em alguns exemplos, a unidade de processamento intraprevisão 46 pode codificar um bloco atual usando vários modos de intraprevisão, por exemplo, durante passagens de codificação separadas, e a unidade de processamento intraprevisão 46 pode selecionar um modo de intraprevisão apropriado para usar a partir dos modos testados. Por exemplo, a unidade de processamento intraprevisão 46 pode calcular valores de distorção de taxa usando uma análise de distorção de taxa para os vários modos de intraprevisão testados, e pode selecionar o modo de intraprevisão tendo as melhores características de distorção de taxa entre os modos testados. A análise de distorção de taxa geralmente determina uma quantidade de distorção (ou erro) entre um bloco codificado e um bloco não codificado original que foi codificado para produzir o bloco codificado, bem como uma taxa de bits (ou seja, um número de bits) usada para produzir o bloco codificado. A unidade de processamento intraprevisão 46 pode calcular as razões a partir das distorções e taxas para os vários blocos codificados para determinar qual modo de intraprevisão exibe o melhor valor de distorção da taxa para o bloco.

[0305] Em qualquer caso, após selecionar um modo de intraprevisão para um bloco, a unidade de processamento intraprevisão 46 pode fornecer informações indicativas do modo de intraprevisão selecionado para o bloco para a unidade de codificação por entropia 56. A unidade de codificação por entropia 56 pode codificar as informações indicando o modo de intraprevisão selecionado. O dispositivo de codificação 104 pode incluir, na

configuração de fluxo de bits transmitido, as definições de dados de contextos de codificação para vários blocos, bem como indicações de um modo de intraprevisão mais provável, uma tabela de índice de modo de intraprevisão e uma tabela de índice de modo de intraprevisão modificada para uso para cada um dos contextos. Os dados de configuração do fluxo de bits podem incluir uma pluralidade de tabelas de índice de modo de intraprevisão e uma pluralidade de tabelas de índice de modo de intraprevisão modificadas (também conhecidas como tabelas de mapeamento de palavras-código).

[0306] Após a unidade de processamento de previsão 41 gerar o bloco preditivo para o bloco de vídeo atual por meio de interprevisão ou intraprevisão, o dispositivo de codificação 104 forma um bloco de vídeo residual subtraindo o bloco preditivo do bloco de vídeo atual. Os dados de vídeo residuais no bloco residual podem ser incluídos em uma ou mais TUs e aplicados à unidade de processamento de transformada 52. A unidade de processamento de transformada 52 transforma os dados de vídeo residuais em coeficientes de transformada residuais usando uma transformada, tal como uma transformada discreta de cosseno (DCT) ou uma transformada conceitualmente semelhante. A unidade de processamento de transformada 52 pode converter os dados de vídeo residuais de um domínio de pixel em um domínio de transformada, tal como um domínio de frequência.

[0307] A unidade de processamento de transformada 52 pode enviar os coeficientes de transformada resultantes para a unidade de quantização 54. A unidade de quantização 54 quantiza os coeficientes de transformada para reduzir

ainda mais a taxa de bits. O processo de quantização pode reduzir a profundidade de bits associada a alguns ou a todos os coeficientes. O grau de quantização pode ser modificado ajustando um parâmetro de quantização. Em alguns exemplos, a unidade de quantização 54 pode, então, executar uma varredura da matriz incluindo os coeficientes de transformada quantizados. Alternativamente, a unidade de codificação por entropia 56 pode executar a varredura.

[0308] Após a quantização, a unidade de codificação por entropia 56 codifica por entropia os coeficientes de transformada quantizados. Por exemplo, a unidade de codificação por entropia 56 pode executar codificação de comprimento variável adaptativa ao contexto (CAVLC), codificação aritmética binária adaptativa ao contexto (CABAC), codificação aritmética binária adaptativa ao contexto com base em sintaxe (SBAC), codificação por entropia de particionamento de intervalo de probabilidade (PIPE) ou outra técnica de codificação por entropia. Após a codificação por entropia pela unidade de codificação por entropia 56, o fluxo de bits codificado pode ser transmitido ao dispositivo de decodificação 112, ou arquivado para transmissão ou recuperação posterior pelo dispositivo de decodificação 112. A unidade de codificação por entropia 56 também pode codificar por entropia os vetores de movimento e os outros elementos de sintaxe para a fatia de vídeo atual sendo codificada.

[0309] A unidade de quantização inversa 58 e a unidade de processamento de transformada inversa 60 aplicam quantização inversa e transformação inversa, respectivamente, para reconstruir o bloco residual no

domínio de pixel para uso posterior como um bloco de referência de uma imagem de referência. A unidade de compensação de movimento 44 pode calcular um bloco de referência adicionando o bloco residual a um bloco preditivo de uma das imagens de referência dentro de uma lista de imagens de referência. A unidade de compensação de movimento 44 também pode aplicar um ou mais filtros de interpolação ao bloco residual reconstruído para calcular valores de pixels subinteiros para uso na estimativa de movimento. O adicionador 62 adiciona o bloco residual reconstruído ao bloco de previsão de compensação de movimento produzido pela unidade de compensação de movimento 44 para produzir um bloco de referência para armazenamento na memória de imagem 64. O bloco de referência pode ser usado pela unidade de estimativa de movimento 42 e pela unidade de compensação de movimento 44 como um bloco referência para interpretação de um bloco em uma imagem ou quadro de vídeo subsequente.

[0310] Dessa forma, o dispositivo de codificação 104 da Figura 9 representa um exemplo de um codificador de vídeo configurado para gerar sintaxe para um fluxo de bits de vídeo codificado. O dispositivo de codificação 104 pode, por exemplo, gerar conjuntos de parâmetros VPS, SPS e PPS, conforme descrito acima. O dispositivo de codificação 104 pode executar qualquer uma das técnicas descritas aqui, incluindo os processos descritos acima em relação às Figuras 7 e 8. As técnicas da presente invenção foram geralmente descritas com relação ao dispositivo de codificação 104, mas, como mencionado acima, algumas das

técnicas da presente invenção também podem ser implementadas pelo dispositivo de pós-processamento 57.

[0311] A Figura 10 é um diagrama de blocos ilustrando um dispositivo de decodificação exemplificativo 112. O dispositivo de decodificação 112 inclui uma unidade de decodificação por entropia 80, unidade de processamento de previsão 81, unidade de quantização inversa 86, unidade de processamento de transformada inversa 88, adicionador 90, unidade de filtro 91 e memória de imagem 92. A unidade de processamento de previsão 81 inclui a unidade de compensação de movimento 82 e a unidade de processamento de intraprevisão 84. O dispositivo de decodificação 112 pode, em alguns exemplos, executar uma passagem de decodificação geralmente recíproca à passagem de codificação descrita em relação ao dispositivo de codificação 104 da Figura 9.

[0312] Durante o processo de decodificação, o dispositivo de decodificação 112 recebe um fluxo de bits de vídeo codificado que representa os blocos de vídeo de uma fatia de vídeo codificada e os elementos de sintaxe associados enviados pelo dispositivo de codificação 104. Em alguns exemplos, o dispositivo de decodificação 112 pode receber o fluxo de bits de vídeo codificado do dispositivo de codificação 104. Em alguns exemplos, o dispositivo de decodificação 112 pode receber o fluxo de bits de vídeo codificado de uma entidade de rede 79, tal como um servidor, um elemento de rede com reconhecimento de mídia (MANE), um editor/fatiador de vídeo, ou outro dispositivo configurado para implementar uma ou mais das técnicas descritas acima. A entidade de rede 79 pode ou não incluir o dispositivo de codificação 104. Algumas das técnicas

descritas na presente invenção podem ser implementadas pela entidade de rede 79 antes de a entidade de rede 79 transmitir o fluxo de bits de vídeo codificado ao dispositivo de decodificação 112. Em alguns sistemas de decodificação de vídeo, a entidade de rede 79 e o dispositivo de decodificação 112 podem ser partes de dispositivos separados, enquanto, em outros casos, a funcionalidade descrita em relação à entidade de rede 79 pode ser executada pelo mesmo dispositivo que compreende o dispositivo de decodificação 112.

[0313] A unidade de decodificação por entropia 80 do dispositivo de decodificação 112 decodifica por entropia o fluxo de bits para gerar coeficientes quantizados, vetores de movimento e outros elementos de sintaxe. A unidade de decodificação por entropia 80 encaminha os vetores de movimento e outros elementos da sintaxe para a unidade de processamento de previsão 81. O dispositivo de decodificação 112 pode receber os elementos de sintaxe no nível da fatia de vídeo e/ou no nível do bloco de vídeo. A unidade de decodificação por entropia 80 pode processar e analisar elementos de sintaxe de comprimento fixo e elementos de sintaxe de comprimento variável em ou mais conjuntos de parâmetros, tais como VPS, SPS e PPS.

[0314] Quando a fatia de vídeo é codificada como uma fatia intracodificada (I), a unidade de processamento intraprevisão 84 da unidade de processamento de previsão 81 pode gerar dados de previsão para um bloco de vídeo da fatia de vídeo atual com base em um modo de intraprevisão sinalizado e dados dos blocos previamente decodificados da imagem ou quadro atual. Quando o quadro de vídeo é

codificado como uma fatia intercodificada (isto é, B, P ou GPB), a unidade de compensação de movimento 82 da unidade de processamento de previsão 81 produz blocos preditivos para um bloco de vídeo da fatia de vídeo atual com base nos vetores de movimento e outros elementos de sintaxe recebidos da unidade de decodificação por entropia 80. Os blocos preditivos podem ser produzidos a partir de uma das imagens de referência dentro de uma lista de imagens de referência. O dispositivo de decodificação 112 pode construir as listas de quadros de referência, Lista 0 e Lista 1, usando técnicas de construção padrão com base nas imagens de referência armazenadas na memória de imagem 92.

[0315] A unidade de compensação de movimento 82 determina as informações de previsão para um bloco de vídeo da fatia de vídeo atual analisando os vetores de movimento e outros elementos da sintaxe, e usa as informações de previsão para produzir os blocos preditivos para o bloco de vídeo atual que está sendo decodificado. Por exemplo, a unidade de compensação de movimento 82 pode usar um ou mais elementos de sintaxe em um conjunto de parâmetros para determinar um modo de previsão (por exemplo, intra- ou interprevisão) usado para codificar os blocos de vídeo da fatia de vídeo, um tipo de fatia interprevisão (por exemplo, fatia B, fatia P ou fatia GPB), informações de construção para uma ou mais listas de imagens de referência para a fatia, vetores de movimento para cada bloco de vídeo codificado da fatia, status de interprevisão para cada bloco de vídeo codificado da fatia, e outras informações para decodificar os blocos de vídeo na fatia de vídeo atual.

[0316] A unidade de compensação de movimento 82 também pode executar interpolação com base em filtros de interpolação. A unidade de compensação de movimento 82 pode usar filtros de interpolação como usados pelo dispositivo de codificação 104 durante a codificação dos blocos de vídeo para calcular valores interpolados para pixels subinteiros de blocos de referência. Nesse caso, a unidade de compensação de movimento 82 pode determinar os filtros de interpolação usados pelo dispositivo de codificação 104 a partir dos elementos de sintaxe recebidos, e pode usar os filtros de interpolação para produzir blocos preditivos.

[0317] A unidade de quantização inversa 86 quantiza inversamente, ou desquantiza, os coeficientes de transformada quantizados providos no fluxo de bits e decodificados pela unidade de decodificação por entropia 80. O processo de quantização inversa pode incluir a utilização de um parâmetro de quantização calculado pelo dispositivo de codificação 104 para cada o bloco de vídeo na fatia de vídeo para determinar um grau de quantização e, da mesma forma, um grau de quantização inversa que deve ser aplicado. A unidade de processamento de transformada inversa 88 aplica uma transformada inversa (por exemplo, uma DCT inversa ou outra transformada inversa adequada), uma transformada inteira inversa ou um processo de transformada inversa conceitualmente semelhante aos coeficientes de transformada, a fim de produzir blocos residuais no domínio do pixel.

[0318] Após a unidade de compensação de movimento 82 gerar o bloco preditivo para o bloco de vídeo atual com base nos vetores de movimento e outros elementos da

sintaxe, o dispositivo de decodificação 112 forma um bloco de vídeo decodificado pela soma dos blocos residuais da unidade de processamento de transformada inversa 88 com os blocos preditivos correspondentes gerados pela unidade de compensação de movimento 82. O adicionador 90 representa o componente ou componentes que executam esta operação de soma. Se desejado, os filtros de loop (no loop de codificação ou após o loop de codificação) também podem ser usados para suavizar as transições de pixel ou para de outra forma melhorar a qualidade do vídeo. A unidade de filtro 91 destina-se a representar um ou mais filtros de loop, tais como um filtro de desbloqueio, um filtro de loop adaptativo (ALF) e um filtro de deslocamento adaptativo de amostra (SAO). Embora a unidade de filtro 91 seja mostrada na Figura 10 como sendo um filtro de loop, em outras configurações, a unidade de filtro 91 pode ser implementada como um filtro pós-loop. Os blocos de vídeo decodificados em um determinado quadro ou imagem são, então, armazenados na memória de imagem 92, que armazena as imagens de referência usadas para a compensação de movimento subsequente. A memória de imagem 92 também armazena o vídeo decodificado para apresentação posterior em um dispositivo de exibição, tal como o dispositivo de destino de vídeo 122 mostrado na Figura 1.

[0319] Na descrição acima, os aspectos do pedido são descritos com referência a exemplos específicos deste, mas os versados na técnica reconhecerão que o pedido não é limitado a tais exemplos. Assim, embora exemplos do pedido tenham sido descritos em detalhes neste documento, deve-se entender que os conceitos da invenção podem ser de outra

forma incorporados e empregados, e que as reivindicações anexas pretendem ser interpretadas para incluir tais variações, exceto quando limitado pela técnica anterior. Vários recursos e aspectos dos exemplos descritos acima podem ser usados individualmente ou em conjunto. Além disso, os exemplos descritos podem ser utilizados em qualquer número de ambientes e aplicações além daqueles aqui descritos, sem se afastar do espírito e escopo mais amplo da especificação. A especificação e os desenhos devem, portanto, ser considerados ilustrativos e não restritivos. Para fins de ilustração, os métodos foram descritos em uma ordem específica. Deve-se apreciar que, em exemplos alternativos, os métodos podem ser executados em uma ordem diferente da descrita.

[0320] Quando os componentes são descritos como "configurados para" executar determinadas operações, essa configuração pode ser realizada, por exemplo, projetando circuitos eletrônicos ou outro hardware para executar a operação, programando circuitos eletrônicos programáveis (por exemplo, microprocessadores ou outros circuitos eletrônicos adequados) para executar a operação, ou qualquer combinação dos mesmos.

[0321] Os vários blocos lógicos, módulos, circuitos e etapas de algoritmo ilustrativos descritos em conexão com os exemplos aqui divulgados podem ser implementados como hardware eletrônico, software de computador, firmware ou combinações dos mesmos. Para ilustrar claramente essa permutabilidade de hardware e software, vários componentes, blocos, módulos, circuitos e etapas ilustrativos foram descritos acima geralmente em

termos de sua funcionalidade. A implementação dessa funcionalidade como hardware ou software depende da aplicação específica e restrições de projeto impostas ao sistema geral. Os versados na técnica podem implementar a funcionalidade descrita de várias maneiras para cada aplicação específica, mas essas decisões de implementação não devem ser interpretadas como causando um afastamento do escopo do presente pedido.

[0322] As técnicas descritas neste documento também podem ser implementadas em hardware eletrônico, software de computador, firmware ou qualquer combinação dos mesmos. Tais técnicas podem ser implementadas em vários dispositivos, tais como computadores de uso geral, aparelhos de dispositivos de comunicação sem fio ou dispositivos de circuito integrado com vários usos, incluindo aplicação em aparelhos de dispositivos de comunicação sem fio e outros dispositivos. Quaisquer recursos descritos como módulos ou componentes podem ser implementados em conjunto em um dispositivo lógico integrado ou separadamente como dispositivos lógicos discretos, mas interoperáveis. Se implementadas em software, as técnicas podem ser realizadas pelo menos em parte por um meio de armazenamento de dados legível por computador, que compreende o código de programa incluindo instruções que, quando executadas, realizam um ou mais dos métodos descritos acima. O meio de armazenamento de dados legível por computador pode fazer parte de um produto de programa de computador, que pode incluir materiais de embalagem. O meio legível por computador pode compreender memória ou meio de armazenamento de dados, tal memória de

acesso aleatório (RAM), memória de acesso aleatório dinâmico síncrono (SDRAM), memória somente de leitura (ROM), memória de acesso aleatório não volátil (NVRAM), memória somente de leitura programável apagável eletricamente (EEPROM), memória FLASH, meio de armazenamento de dados magnético ou óptico, e semelhantes. As técnicas adicionais, ou alternativamente, podem ser realizadas, pelo menos em parte, por um meio de comunicação legível por computador que transporta ou comunica código de programa na forma de instruções ou estruturas de dados e que pode ser acessado, lido e/ou executado por um computador, tal como ondas ou sinais propagados.

[0323] 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 de aplicação específica (ASICs), matrizes lógicas programáveis em campo (FPGAs), ou outro circuito lógico integrado ou discreto equivalente. Esse processador pode ser configurado para executar qualquer uma das técnicas descritas nesta divulgação. Um processador de propósito geral pode ser um microprocessador; mas, em alternativa, o processador pode ser qualquer processador, controlador, microcontrolador ou máquina de estado convencional. Um processador também pode ser implementado 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 conjunto com um núcleo DSP, ou qualquer outra configuração. Por conseguinte, o termo

"processador", como aqui utilizado, pode se referir a qualquer estrutura anterior, qualquer combinação da estrutura anterior, ou qualquer outra estrutura ou aparelho adequado para a implementação das técnicas aqui descritas. Além disso, em alguns aspectos, a funcionalidade aqui descrita pode ser fornecida em módulos de software dedicados ou módulos de hardware configurados para codificação e decodificação, ou incorporados em um codificador-decodificador de vídeo combinado (CODEC).

REIVINDICAÇÕES

1. Método de codificação de dados de vídeo, compreendendo:

receber dados de vídeo;

determinar, por um dispositivo de codificação, se as informações associadas com os dados de vídeo são necessárias para apresentar os dados de vídeo, em que as informações não são necessárias para decodificar os dados de vídeo;

gerar uma mensagem de informações suplementares de aprimoramento (SEI), em que a mensagem SEI inclui um campo indicando se as informações são necessárias para apresentar os dados de vídeo;

codificar os dados de vídeo; e

incluir a mensagem SEI com os dados de vídeo codificados.

2. Método, de acordo com a reivindicação 1, em que o dispositivo de codificação é configurado com um ou mais parâmetros que o dispositivo de codificação pode usar para determinar se as informações são necessárias para apresentar os dados de vídeo.

3. Método, de acordo com a reivindicação 1, em que o dispositivo de codificação determina que as informações são necessárias para apresentar os dados de vídeo.

4. Método, de acordo com a reivindicação 1, em que o dispositivo de codificação determina que as informações não são necessárias para apresentar os dados de vídeo.

5. Método, de acordo com a reivindicação 1, em que um elemento de sintaxe da mensagem SEI indica um tipo da mensagem SEI.

6. Método, de acordo com a reivindicação 1, em que as informações são incluídas com os dados de vídeo codificados.

7. Método, de acordo com a reivindicação 1, compreendendo ainda:

gravar os dados de vídeo codificados e a mensagem SEI em um arquivo, em que os dados de vídeo codificados e a mensagem SEI são gravados no arquivo de acordo com um formato de arquivo; e

usar o formato de arquivo para colocar a mensagem SEI no arquivo em que a SEI pode ser lida antes de os dados de vídeo codificados serem lidos.

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

encapsular os dados de vídeo codificados para streaming, em que os dados de vídeo codificados são encapsulados de acordo com um formato de streaming; e

gerar, de acordo com o formato de streaming, uma descrição de dados de vídeo codificados, em que a descrição inclui a mensagem SEI.

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

encapsular os dados de vídeo codificados para transmissão através de uma rede, em que os dados de vídeo codificados são encapsulados de acordo com um formato do recipiente;

gerar, de acordo com o formato do recipiente, um primeiro pacote, em que uma porção de carga útil do primeiro pacote inclui a mensagem SEI como um descritor; e

gerar um ou mais pacotes incluindo os dados de vídeo codificados, em que os um ou mais pacotes seguem o primeiro pacote em ordem de transmissão.

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

encapsular os dados de vídeo codificados para transmissão através de uma rede, em que os dados de vídeo codificados são encapsulados de acordo com um protocolo de transferência em tempo real; e

gerar, de acordo com um protocolo de descrição de sessão, uma descrição de sessão, em que a descrição de sessão inclui a mensagem SEI.

11. Dispositivo de codificação, compreendendo:

uma memória configurada para armazenar dados de vídeo; e

um processador configurado para:

determinar se as informações associadas com os dados de vídeo são necessárias para apresentar os dados de vídeo, em que as informações não são necessárias para decodificar os dados de vídeo;

gerar uma mensagem de informações suplementares de aprimoramento (SEI), em que a mensagem SEI inclui um campo indicando se as informações são necessárias para apresentar os dados de vídeo;

codificar os dados de vídeo; e

incluir a mensagem SEI com os dados de vídeo codificados.

12. Dispositivo de codificação, de acordo com a reivindicação 11, em que o dispositivo de codificação é configurado com um ou mais parâmetros que o dispositivo de codificação pode usar para determinar se as informações são necessárias para apresentar os dados de vídeo.

13. Dispositivo de codificação, de acordo com a reivindicação 11, em que o processador é ainda configurado para:

determinar que as informações são necessárias para apresentar os dados de vídeo.

14. Dispositivo de codificação, de acordo com a reivindicação 11, em que o processador é ainda configurado para:

determinar que as informações não são necessárias para apresentar os dados de vídeo.

15. Dispositivo de codificação, de acordo com a reivindicação 11, em que um elemento de sintaxe da mensagem SEI indica um tipo da mensagem SEI.

16. Dispositivo de codificação, de acordo com a reivindicação 11, em que as informações são incluídas com os dados de vídeo codificados.

17. Meio legível por computador não transitório compreendendo instruções que, quando executadas por um ou mais processadores, levam os um ou mais processadores a:

receber dados de vídeo;

determinar se as informações associadas com os dados de vídeo são necessárias para apresentar os dados de vídeo, em que as informações não são necessárias para decodificar os dados de vídeo;

gerar uma mensagem de informações suplementares de aprimoramento (SEI), em que a mensagem SEI inclui um campo indicando se as informações são necessárias para apresentar os dados de vídeo;

codificar os dados de vídeo; e

incluir a mensagem SEI com os dados de vídeo codificados.

18. Aparelho, compreendendo:

meios para receber dados de vídeo;

meios para determinar se as informações associadas com os dados de vídeo são necessárias para apresentar os dados de vídeo, em que as informações não são necessárias para decodificar os dados de vídeo;

meios para gerar uma mensagem de informações suplementares de aprimoramento (SEI), em que a mensagem SEI inclui um campo indicando se as informações são necessárias para apresentar os dados de vídeo;

meios para codificar os dados de vídeo; e

meios para incluir a mensagem SEI com os dados de vídeo codificados.

19. Método de decodificação de dados de vídeo, compreendendo:

receber, em um dispositivo de decodificação, um fluxo de bits de vídeo codificado, uma mensagem de informações suplementares de aprimoramento (SEI) e as informações associadas com o fluxo de bits de vídeo codificado;

determinar, usando a mensagem SEI, se as informações são necessárias para apresentar dados de vídeo no fluxo de bits de vídeo codificado, em que as informações

não são necessárias para decodificar os dados de vídeo, e em que a mensagem SEI inclui um campo indicando se as informações são necessárias para apresentar os dados de vídeo; e

decodificar o fluxo de bits de vídeo codificado para produzir dados de vídeo, em que o fluxo de bits de vídeo codificado é decodificado de acordo com um resultado da determinação de se as informações são necessárias para apresentar os dados de vídeo.

20. Método, de acordo com a reivindicação 19, em que o dispositivo de decodificação determina que as informações são necessárias para apresentar os dados de vídeo, e em que o dispositivo de decodificação configura os dados de vídeo de acordo com as informações.

21. Método, de acordo com a reivindicação 19, em que o dispositivo de decodificação determina que as informações não são necessárias para apresentar os dados de vídeo, e em que o dispositivo de decodificação decodifica os dados de vídeo codificados sem usar as informações.

22. Método, de acordo com a reivindicação 19, em que um elemento de sintaxe da mensagem SEI indica um tipo da mensagem SEI.

23. Método, de acordo com a reivindicação 19, em que o fluxo de bits de vídeo codificado, a mensagem SEI e as informações associadas com o fluxo de bits de vídeo codificado são recebidos em um arquivo, em que o arquivo é formatado de acordo com um formato de arquivo, e em que, de acordo com o formato de arquivo, a mensagem SEI está em uma parte do arquivo que é lida pelo dispositivo de

decodificação antes de uma parte do arquivo que inclui o fluxo de bits de vídeo codificado.

24. Método, de acordo com a reivindicação 19, em que o fluxo de bits de vídeo codificado, a mensagem SEI e as informações associadas com o fluxo de bits de vídeo codificado são recebidos em um fluxo de dados, em que o fluxo de dados é formatado de acordo com um formato de streaming; e

fazendo a leitura, a partir do fluxo de dados, de uma descrição dos dados de vídeo codificados, em que a descrição inclui a mensagem SEI.

25. Método, de acordo com a reivindicação 19, em que o fluxo de bits de vídeo codificado, a mensagem SEI e as informações associadas com o fluxo de bits de vídeo codificado são recebidos em uma pluralidade de pacotes de rede;

fazendo a leitura da mensagem SEI a partir de uma porção de carga útil de um primeiro pacote da pluralidade de pacotes de rede; e

fazendo a leitura dos dados de vídeo codificados a partir de um ou mais pacotes da pluralidade de pacotes de rede, em que os um ou mais pacotes seguem o primeiro pacote em ordem de transmissão.

26. Método, de acordo com a reivindicação 19, em que o fluxo de bits de vídeo codificado, a mensagem SEI e as informações associadas com o fluxo de bits de vídeo codificado são recebidos através de uma rede, em que o fluxo de bits de vídeo codificado é encapsulado de acordo com um protocolo de transferência em tempo real, em que a mensagem SEI é incluída em uma descrição de sessão, em que

a descrição de sessão é formatada de acordo com um protocolo de descrição de sessão.

27. Dispositivo de decodificação, compreendendo:
uma memória configurada para armazenar um fluxo de bits de vídeo codificado, uma mensagem de informações suplementares de aprimoramento (SEI) e as informações associadas com o fluxo de bits de vídeo codificado; e
um processador configurado para:

determinar, usando a mensagem SEI, se as informações são necessárias para apresentar dados de vídeo no fluxo de bits de vídeo codificado, em que as informações não são necessárias para decodificar os dados de vídeo, e em que a mensagem SEI inclui um campo indicando se as informações são necessárias para apresentar os dados de vídeo; e

decodificar o fluxo de bits de vídeo codificado para produzir dados de vídeo, em que o fluxo de bits de vídeo codificado é decodificado de acordo com um resultado da determinação de se as informações são necessárias para apresentar os dados de vídeo.

28. Dispositivo de decodificação, de acordo com a reivindicação 27, em que o dispositivo de decodificação determina que as informações são necessárias para apresentar os dados de vídeo, e em que o dispositivo de decodificação configura os dados de vídeo de acordo com as informações.

29. Dispositivo de decodificação, de acordo com a reivindicação 27, em que o dispositivo de decodificação determina que as informações não são necessárias para apresentar os dados de vídeo, e em que o dispositivo de

decodificação decodifica os dados de vídeo codificados sem usar as informações.

30. Dispositivo de decodificação, de acordo com a reivindicação 27, em que um elemento de sintaxe da mensagem SEI indica um tipo da mensagem SEI.

31. Meio legível por computador não transitório compreendendo instruções que, quando executadas por um ou mais processadores, levam os um ou mais processadores a:

receber um fluxo de bits de vídeo codificado, uma mensagem de informações suplementares de aprimoramento (SEI) e as informações associadas com o fluxo de bits de vídeo codificado;

determinar, usando a mensagem SEI, se as informações são necessárias para apresentar dados de vídeo no fluxo de bits de vídeo codificado, em que as informações não são necessárias para decodificar os dados de vídeo, e em que a mensagem SEI inclui um campo indicando se as informações são necessárias para apresentar os dados de vídeo; e

decodificar o fluxo de bits de vídeo codificado para produzir dados de vídeo, em que o fluxo de bits de vídeo codificado é decodificado de acordo com um resultado da determinação de se as informações são necessárias para apresentar os dados de vídeo.

32. Aparelho, compreendendo:

meios para receber um fluxo de bits de vídeo codificado, uma mensagem de informações suplementares de aprimoramento (SEI) e as informações associadas com o fluxo de bits de vídeo codificado;

meios para determinar, usando a mensagem SEI, se as informações são necessárias para apresentar dados de vídeo no fluxo de bits de vídeo codificado, em que as informações não são necessárias para decodificar os dados de vídeo, e em que a mensagem SEI inclui um campo indicando se as informações são necessárias para apresentar os dados de vídeo; e

meios para decodificar o fluxo de bits de vídeo codificado para produzir dados de vídeo, em que o fluxo de bits de vídeo codificado é decodificado de acordo com um resultado da determinação de se as informações são necessárias para apresentar os dados de vídeo.

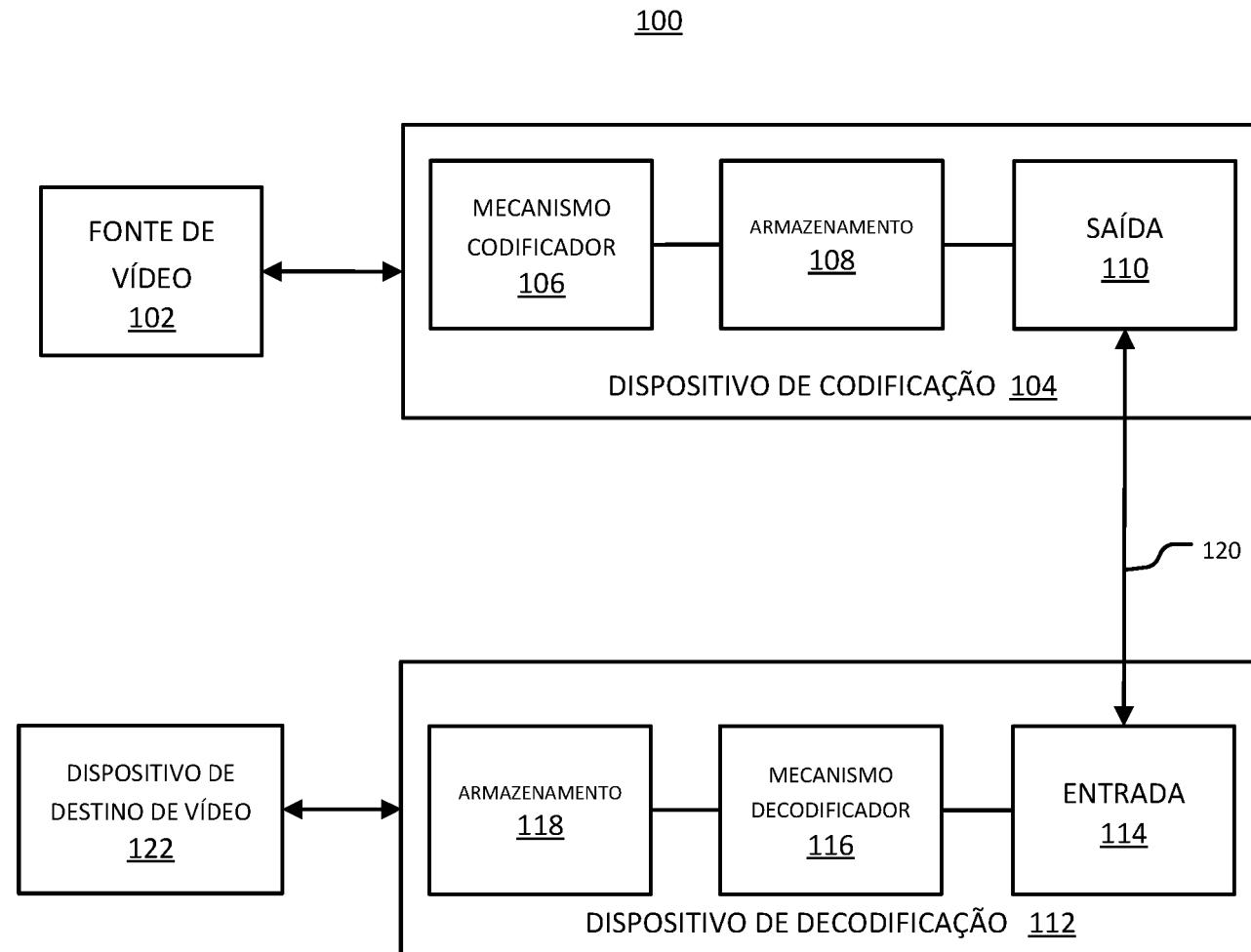
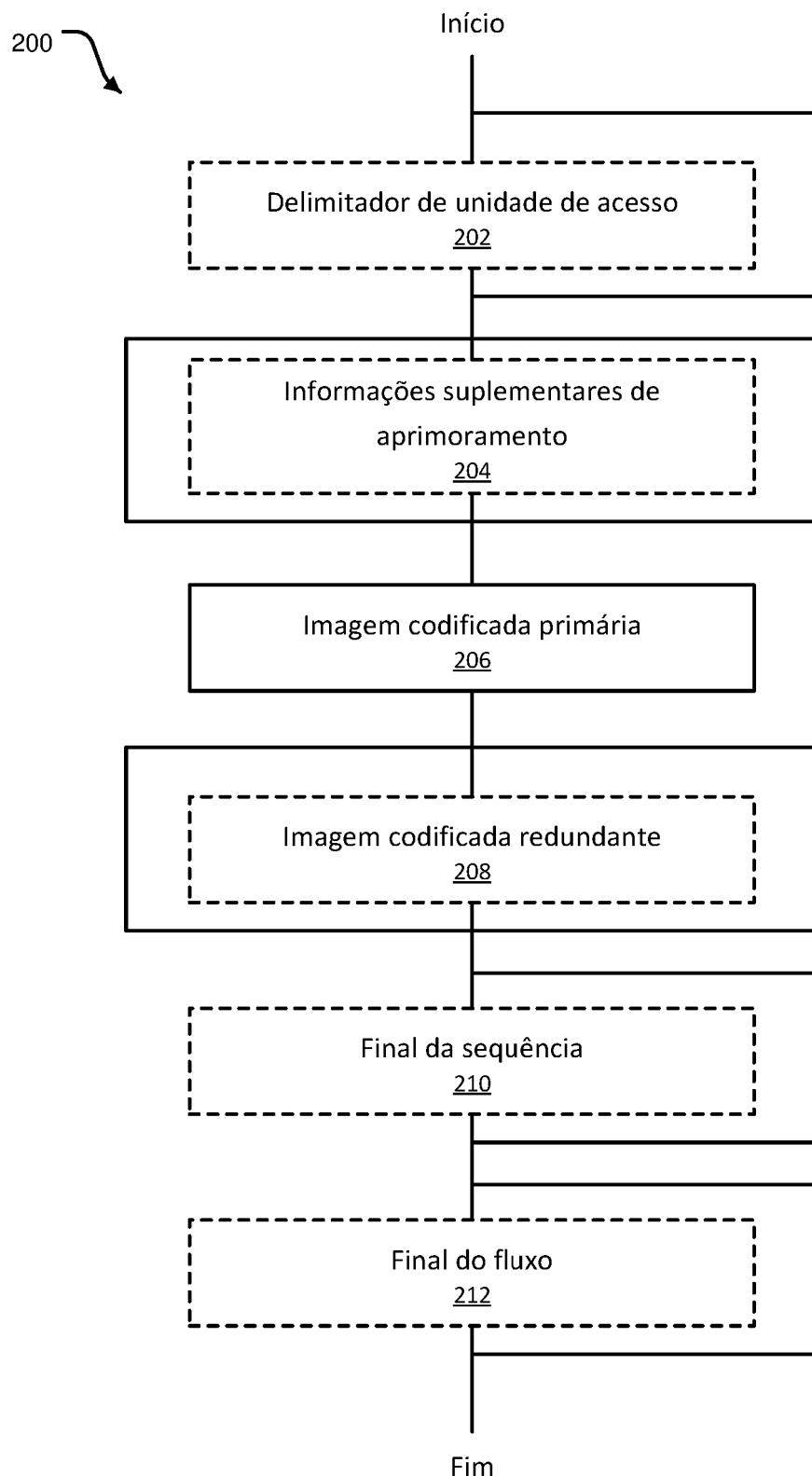
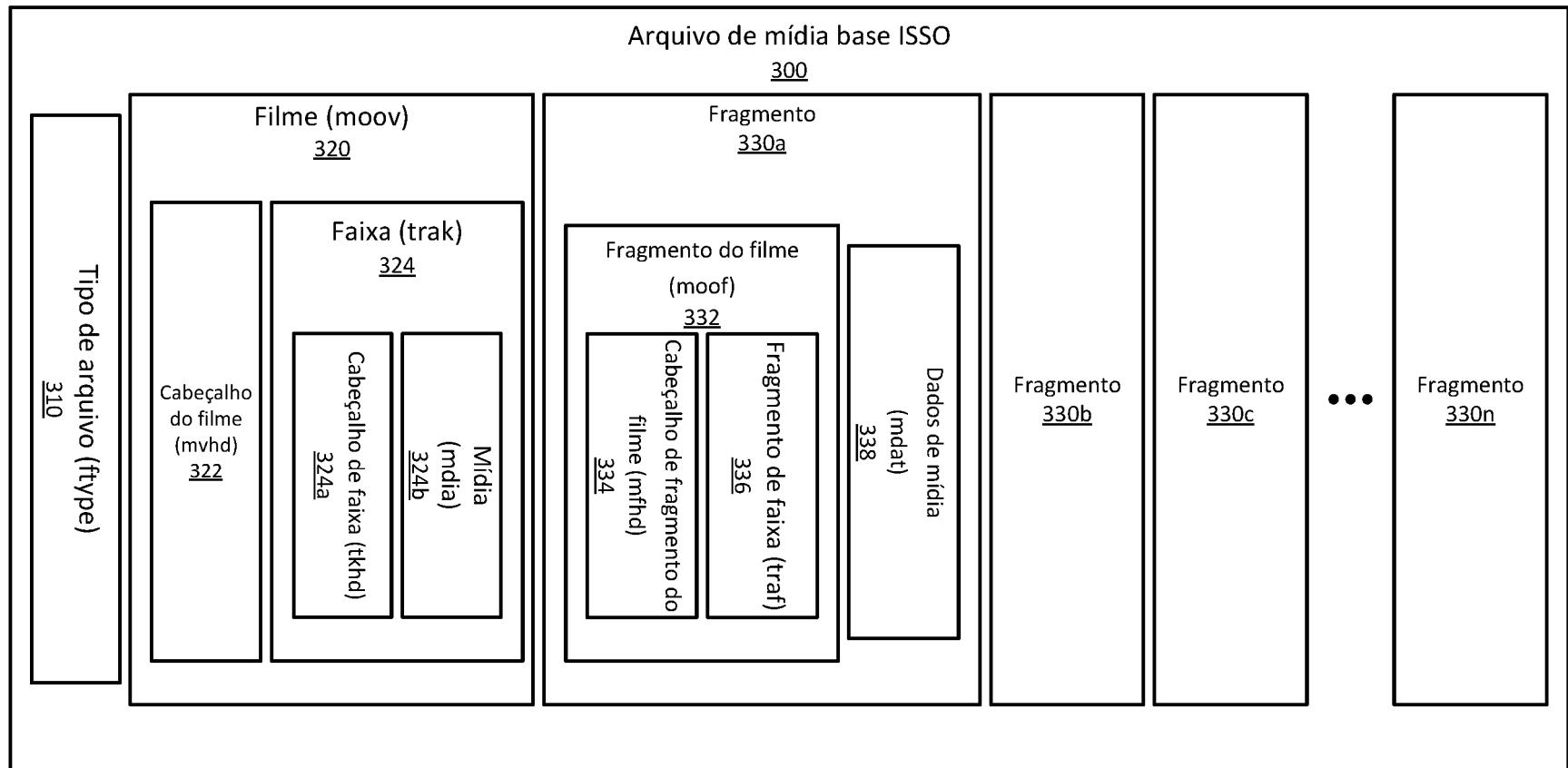


FIG. 1

**FIG. 2**

**FIG. 3**

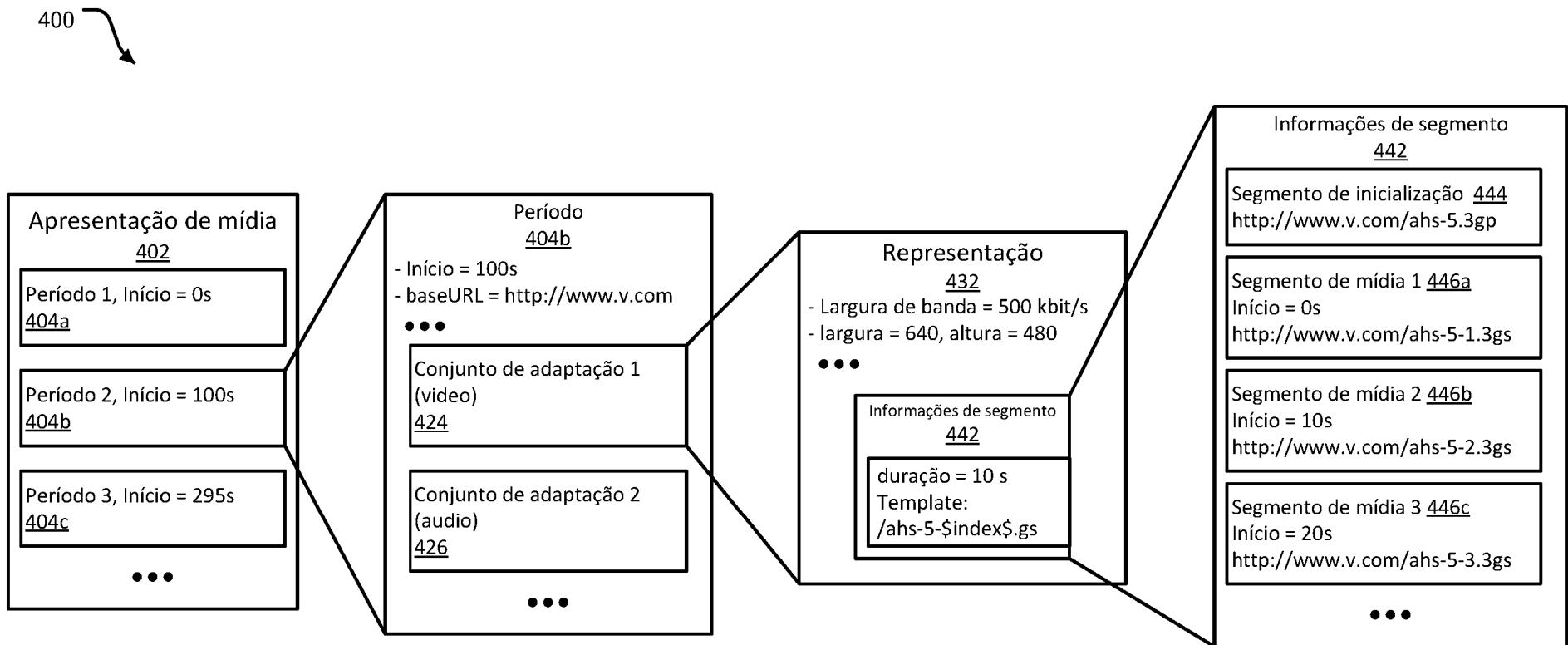


FIG. 4

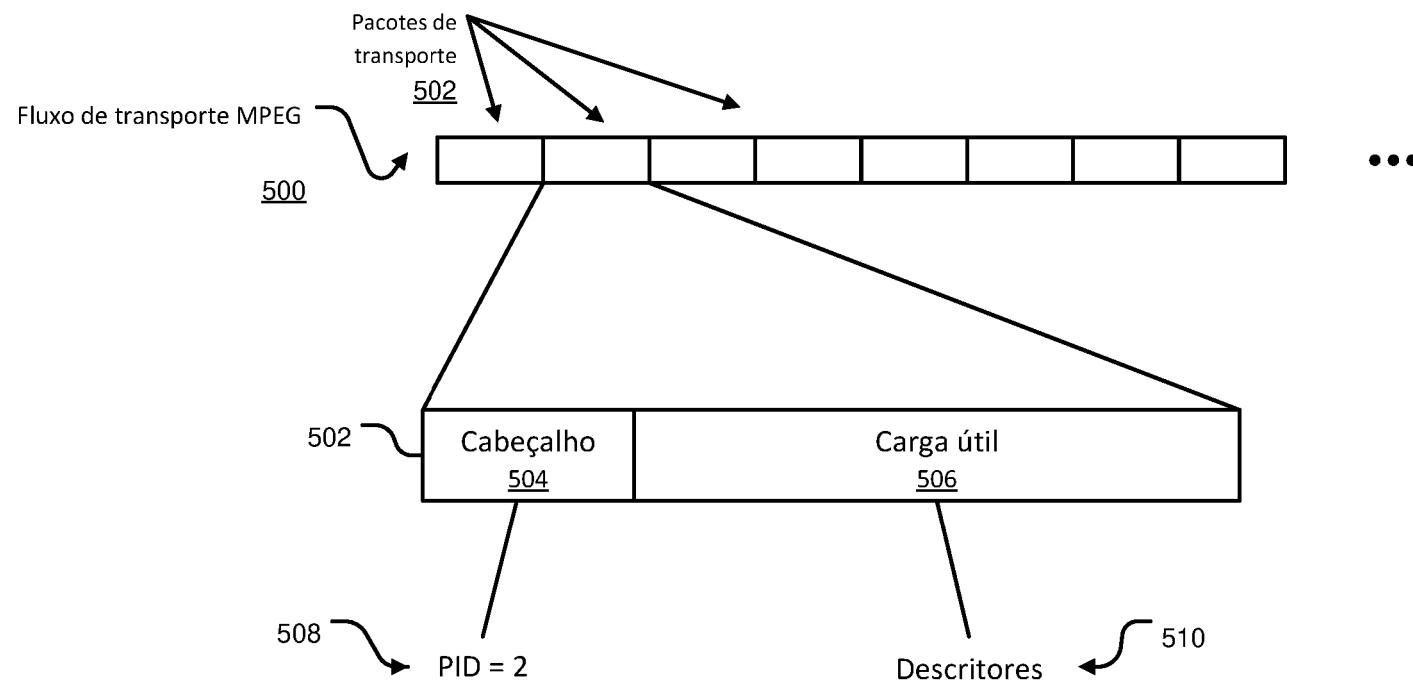
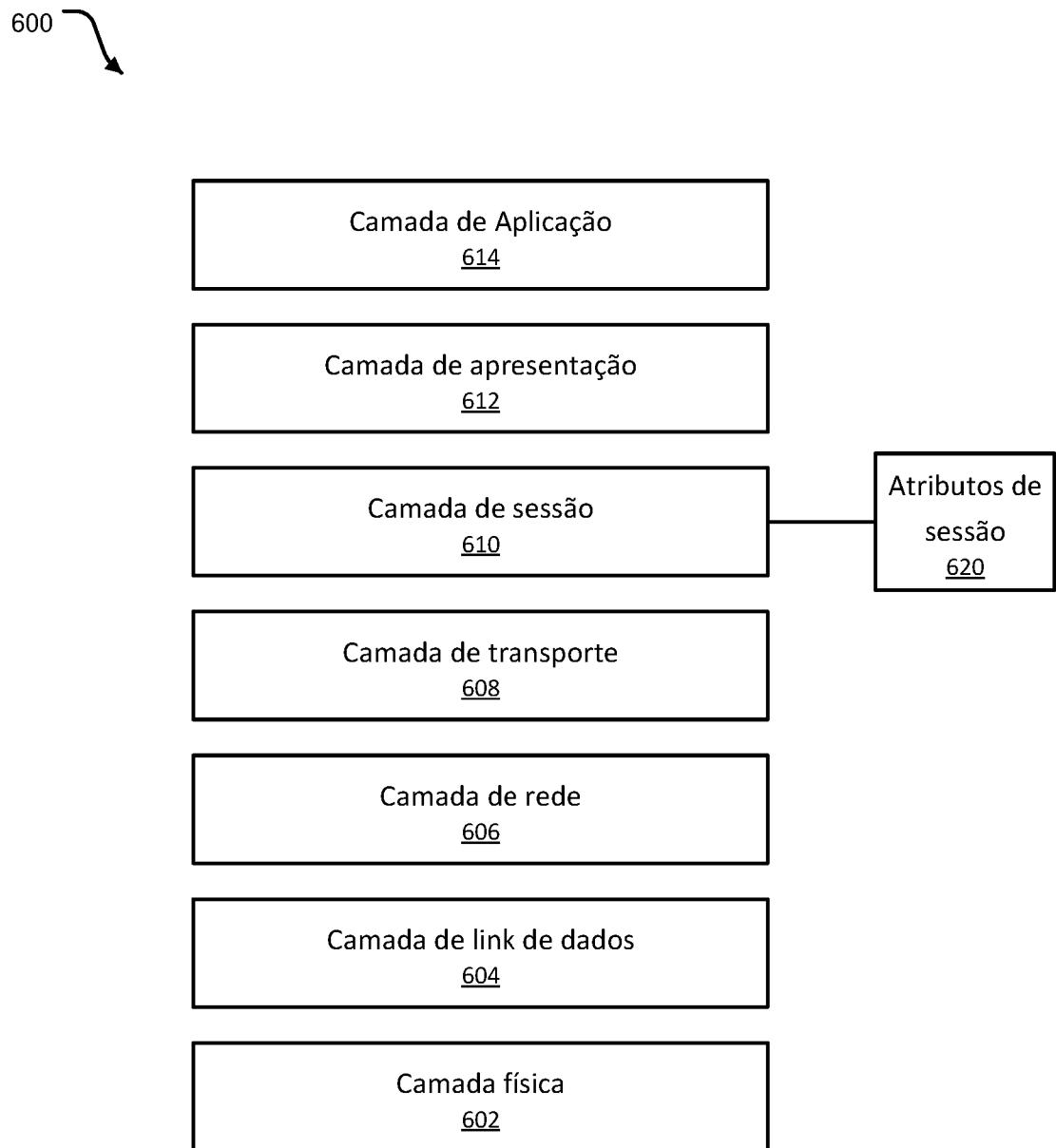
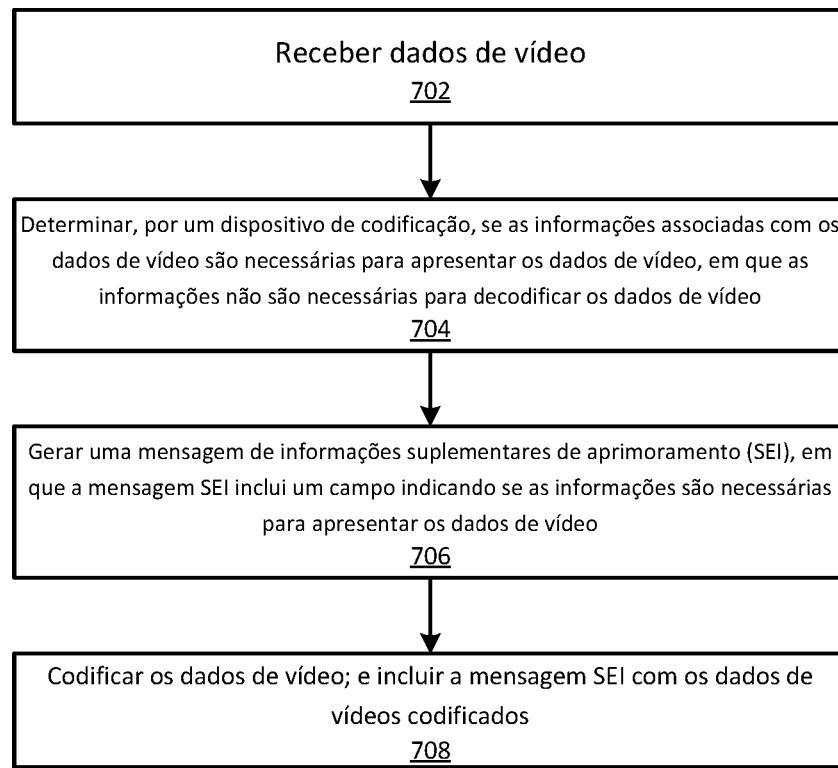


FIG. 5

**FIG. 6**

700**FIG. 7**

800

Receber, em um dispositivo de decodificação, um fluxo de bits de vídeo codificado, uma mensagem de informações suplementares de aprimoramento (SEI) e as informações associadas com o fluxo de bits de vídeo codificado

802

Determinar, usando a mensagem SEI, se as informações são necessárias para apresentar dados de vídeo no fluxo de bits de vídeo codificado, em que as informações não são necessárias para decodificar os dados de vídeo, e em que a mensagem SEI inclui um campo indicando se as informações são necessárias para apresentar os dados de vídeo

804

Decodificar o fluxo de bits de vídeo codificado para produzir dados de vídeo, em que o fluxo de bits de vídeo codificado é decodificado de acordo com um resultado da determinação de se as informações são necessárias para apresentar os dados de vídeo

806

FIG. 8

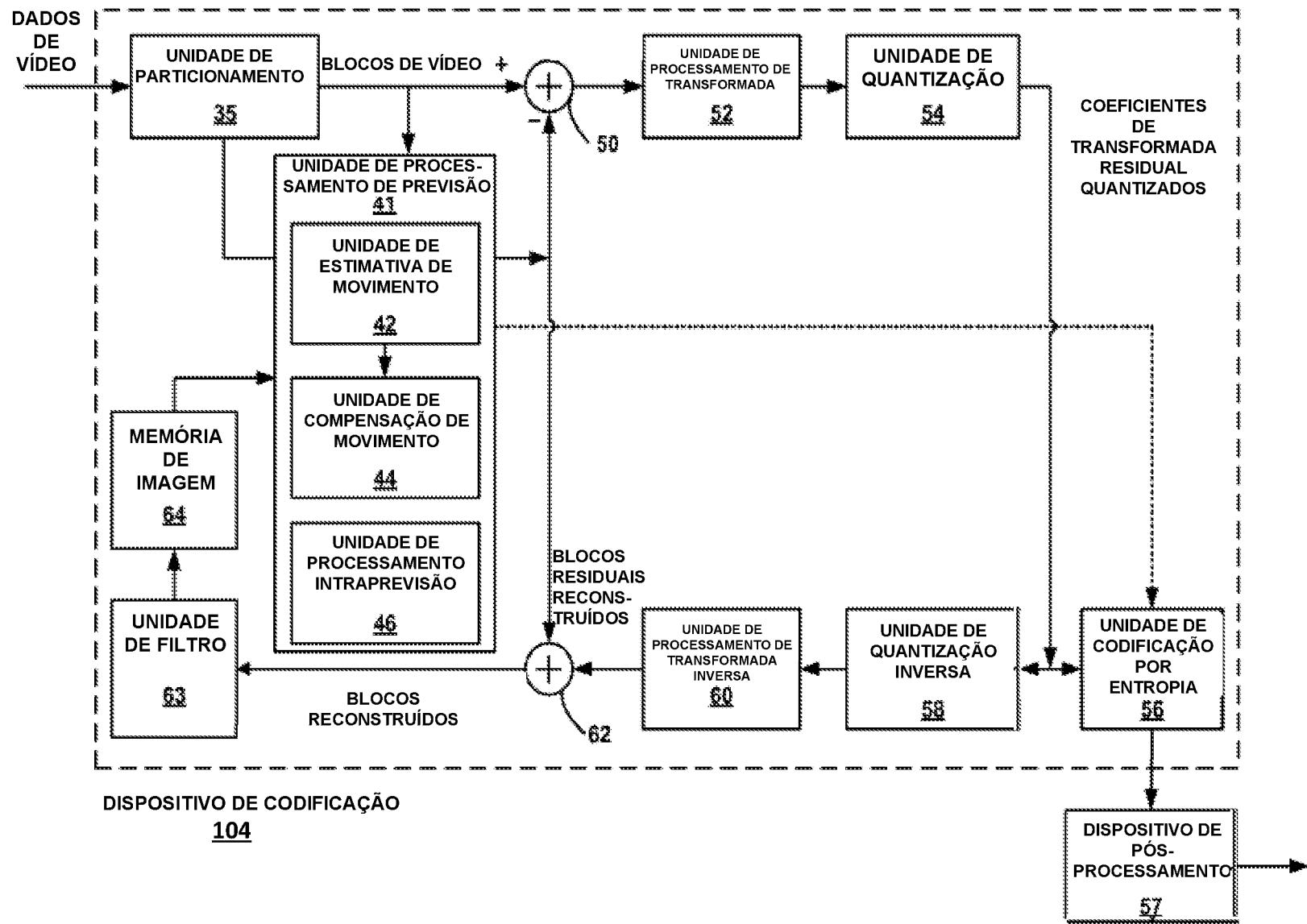
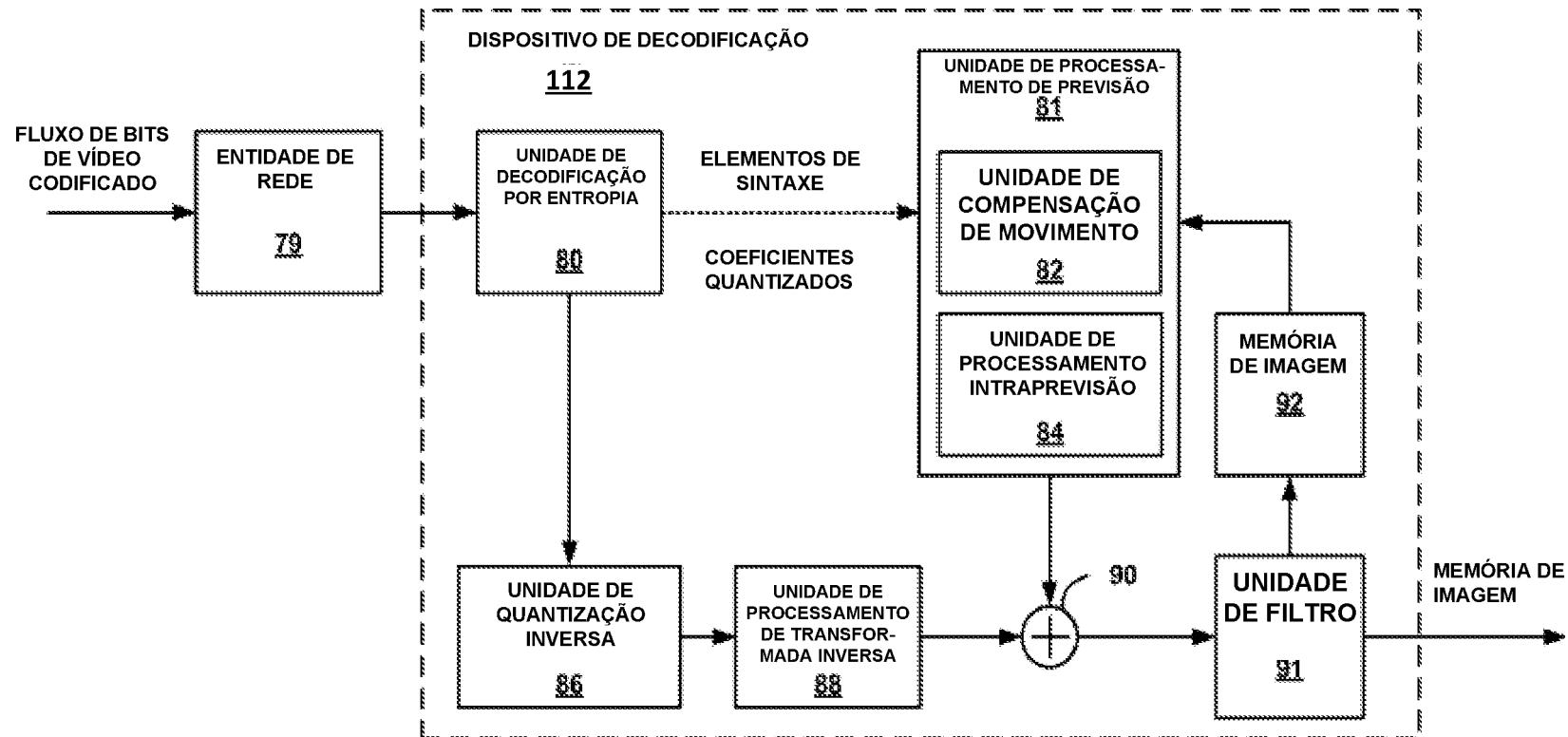


FIG. 9

**FIG. 10**

RESUMO

**"SINALIZAÇÃO DE INFORMAÇÕES DE VÍDEO SUPLEMENTARES
ESSENCIAIS E NÃO ESSENCIAIS"**

Em várias implementações, são providas técnicas, bem como sistemas que implementam essas técnicas, para mensagens SEI que distinguem informações essenciais de informações não essenciais. Em várias implementações, um codificador pode ser configurado para determinar se as informações associadas com dados de vídeo são essenciais ou não são essenciais. Em várias implementações, diferentes tipos de mensagens SEI podem ser definidos, em que um ou mais desses tipos de mensagens SEI podem indicar se as informações associadas com um vídeo particular são essenciais ou não essenciais. Informações essenciais são necessárias para apresentar os dados de vídeo, e não são necessárias para decodificar os dados de vídeo.