



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

(11) BR 122020003032-1 B1



(22) Data do Depósito: 17/10/2011

(45) Data de Concessão: 09/11/2021

(54) Título: MÉTODO DE CODIFICAÇÃO E MÉTODO DE DECODIFICAÇÃO

(51) Int.Cl.: H04N 19/463; H04N 19/105; H04N 19/182; H04N 19/186; H04N 19/44; (...).

(52) CPC: H04N 19/463; H04N 19/105; H04N 19/182; H04N 19/186; H04N 19/44; (...).

(73) Titular(es): KABUSHIKI KAISHA TOSHIBA.

(72) Inventor(es): AKIYUKI TANIZAWA; TAKESHI CHUJOH.

(86) Pedido PCT: PCT JP2011073852 de 17/10/2011

(87) Publicação PCT: WO 2013/057783 de 25/04/2013

(85) Data do Início da Fase Nacional: 13/02/2020

(62) Pedido Original do Dividido: BR112014004805-3 - 17/10/2011

(57) Resumo: A presente invenção refere-se a um método de codificação de acordo com uma concretização que inclui uma etapa de ajustar índice, uma etapa de prever e uma etapa de codificar. A etapa de ajustar índice ajusta um índice que representa informação de uma imagem de referência e de um fator de ponderação. A etapa de prever prevê que um valor de referência do fator de ponderação seja derivado como um valor previsto, quando uma mudança de valor de pixel entre pelo menos uma imagem de referência e uma imagem alvo a serem codificadas for uma referência específica ou menos. A etapa de codificar codifica um valor de diferença entre o fator de ponderação e o valor previsto.

Relatório Descritivo da Patente de Invenção para **"MÉTODO DE CODIFICAÇÃO E MÉTODO DE DECODIFICAÇÃO"**.

Dividido do BR112014004805-3 depositado em 17 de outubro de 2011.

[0001] Uma concretização da presente invenção se refere a um método de codificação e a um método de decodificação.

ANTECEDENTES

[0002] Recentemente, um método de codificar uma imagem com eficiência de codificação notadamente aperfeiçoada foi recomendado como ITU-T REC. H.264 e ISO/IEC 14496-10 (adiante, referido como "H.264") em cooperação com ITU-T (Setor de Normalização das Telecomunicações da União Internacional de Telecomunicações) e ISO (Organização Internacional para Padronização) / IEC (Comissão Eletrotécnica Internacional).

[0003] Em H.264, é descrito um sistema de codificação de interpredição no qual redundância na direção do tempo é eliminada para atingir uma alta eficiência de codificação ao se fazer uma predição de compensação de movimento de precisão fracional usando uma imagem codificada como uma imagem de referência.

[0004] Além disso, é proposto um sistema no qual uma imagem em movimento incluindo um efeito de desvanecimento ou de dissolução é codificada com uma eficiência maior do que aquela de um sistema de codificação de interpredição de acordo com ISO/IEC MPEG (Grupo de Especialistas em Imagens com Movimento)-1, 2, 4. Neste sistema, uma predição de compensação de movimento de precisão fracional é feita para uma imagem em movimento de entrada apresentando luminância e duas diferenças de cor como quadros para prever uma mudança no brilho na direção do tempo. Depois, com o uso de um índice representando uma combinação de uma imagem de referência, um fator de ponderação para cada luminância e duas diferenças de cor, e um deslocamento para cada lumi-

nância e duas diferenças de cor, uma imagem prevista é multiplicada pelo fator de ponderação, e o deslocamento é acrescentado à mesma.

LISTA DE CITAÇÕES

LITERATURA DE PATENTE

[0005] Literatura de Patente 1: Publicação de Patente Japonesa Aberta à Inspeção Pública No. 2004-7377

SUMÁRIO

PROBLEMA TÉCNICO

[0006] Todavia, na tecnologia convencional, conforme descrito acima, uma vez que o índice é codificado como sendo mantido como valores diretos, a eficiência de codificação é reduzida. Um objetivo a ser solucionado pela presente invenção é o de prover um método de codificação e um método de decodificação capazes de aperfeiçoarem a eficiência de codificação.

SOLUÇÃO PARA O PROBLEMA

[0007] Um dispositivo de codificação de acordo com uma concretização inclui uma unidade de ajuste de índice, uma unidade de reconfiguração de índice e uma unidade de codificação entrópica. A unidade de ajuste de índice ajusta um índice que representa informação de uma imagem de referência e de um fator de ponderação. A unidade de reconfiguração de índice prevê um valor de referência do fator de ponderação, em que o valor de referência indica um fator a ser ajustado, se uma diferença de valor de pixel entre uma imagem de referência e uma imagem alvo a serem codificadas for menor ou igual a um valor específico. A unidade de codificação entrópica codifica um valor de diferença entre o fator de ponderação e o valor de referência.

BREVE DESCRIÇÃO DOS DESENHOS

[0008] A Figura 1 é um diagrama de bloco que ilustra um exemplo de um dispositivo de codificação de acordo com uma primeira concretização.

[0009] A Figura 2 é um diagrama explicativo que ilustra um exem-

plo de uma sequência de codificação prevista para um bloco de pixels de acordo com a primeira concretização.

[0010] A Figura 3A é um diagrama que ilustra um exemplo do tamanho de bloco de um bloco de árvore de codificação de acordo com a primeira concretização.

[0011] A Figura 3B é um diagrama que ilustra um exemplo específico do bloco de árvore de codificação de acordo com a primeira concretização.

[0012] A Figura 3C é um diagrama que ilustra um exemplo específico do bloco de árvore de codificação de acordo com a primeira concretização.

[0013] A Figura 3D é um diagrama que ilustra um exemplo específico do bloco de árvore de codificação de acordo com a primeira concretização.

[0014] A Figura 4 é um diagrama de bloco que ilustra um exemplo de uma unidade de geração de imagem prevista de acordo com a primeira concretização.

[0015] A Figura 5 é um diagrama que ilustra um exemplo da relação entre vetores de movimento para uma predição compensada por movimento em uma predição bidirecional de acordo com a primeira concretização.

[0016] A Figura 6 é um diagrama de bloco que ilustra um exemplo de uma unidade de compensação de movimento de múltiplos quadros de acordo com a primeira concretização.

[0017] A Figura 7 é um diagrama explicativo que ilustra um exemplo de precisão de ponto fixo de um fator de ponderação de acordo com a primeira concretização.

[0018] A Figura 8A é um diagrama que ilustra um exemplo de informação de parâmetro WP de acordo com a primeira concretização.

[0019] A Figura 8B é um diagrama que ilustra um exemplo de in-

formação de parâmetro WP de acordo com a primeira concretização.

[0020] A Figura 9 é um diagrama que ilustra um exemplo de sintaxe de acordo com a primeira concretização.

[0021] A Figura 10 é um diagrama que ilustra um exemplo de uma sintaxe de ajuste de parâmetro de imagem de acordo com a primeira concretização.

[0022] A Figura 11 é um diagrama que ilustra um exemplo de uma sintaxe de cabeçalho da fatia de acordo com a primeira concretização.

[0023] A Figura 12 é um digrama que ilustra um exemplo de uma sintaxe de tabela de peso de predição de acordo com a primeira concretização.

[0024] A Figura 13 é um diagrama que ilustra um exemplo de uma configuração de sintaxe que explicitamente representa um método de predição de acordo com a primeira concretização.

[0025] A Figura 14 é um fluxograma que ilustra um exemplo do processo de prever uma precisão de ponto fixo de acordo com a primeira concretização.

[0026] A Figura 15 é um fluxograma que ilustra um exemplo do processo de restaurar a precisão de ponto fixo de acordo com a primeira concretização.

[0027] A Figura 16 é um fluxograma que ilustra um exemplo de um processo de predição de fator de ponderação de acordo com a primeira concretização.

[0028] A Figura 17 é um fluxograma que ilustra um exemplo de um processo de restauração de fator de ponderação de acordo com a primeira concretização.

[0029] A Figura 18 é um fluxograma que ilustra outro exemplo de um processo de predição de fator de ponderação de acordo com a primeira concretização.

[0030] A Figura 19 é um fluxograma que ilustra outro exemplo de

um processo de restauração de fator de ponderação de acordo com a primeira concretização.

[0031] A Figura 20 é um fluxograma que ilustra um exemplo de um processo de predição de um sinal de diferença de cor de acordo com a primeira concretização.

[0032] A Figura 21 é um fluxograma que ilustra um exemplo de um processo de restauração de um sinal de diferença de cor de acordo com a primeira concretização.

[0033] A Figura 22 é um fluxograma que ilustra outro exemplo de um processo de predição de um fator de ponderação de acordo com a primeira concretização.

[0034] A Figura 23 é um fluxograma que ilustra outro exemplo do processo de restauração de um fator de ponderação de acordo com a primeira concretização.

[0035] A Figura 24 é um diagrama de bloco que ilustra um exemplo da configuração de um dispositivo de decodificação de acordo com uma segunda concretização.

DESCRIÇÃO DAS CONCRETIZAÇÕES

[0036] Adiante, concretizações serão descritas em detalhes com referência aos desenhos anexos. Um dispositivo de codificação e um dispositivo de decodificação de acordo com cada concretização apresentada abaixo podem ser implementados por hardware, tal como um chip LSI (Integração em Grande Escala), ou um DSP (Processador de Sinal Digital), ou um FPGA (Arranjo de Portas Programável em Campo). Além disso, um dispositivo de codificação e um dispositivo de decodificação de acordo com cada concretização apresentada abaixo podem ser implementados ao fazer com que um computador execute um programa, em outras palavras, por software. Na descrição apresentada abaixo, o termo "imagem" pode ser apropriadamente substituído por um termo, tal como "vídeo", "pixel", "sinal de imagem", "imagem", ou

"dados de imagem".

Primeira Concretização

[0037] Em uma primeira concretização, será descrito um dispositivo de codificação que codifica uma imagem em movimento.

[0038] A Figura 1 é um diagrama de bloco que ilustra um exemplo da configuração de um dispositivo de codificação 100 de acordo com uma primeira concretização.

[0039] O dispositivo de codificação 100 divide cada quadro ou cada campo configurando uma imagem de entrada em uma pluralidade de blocos de pixels e executa a codificação prevista dos blocos de pixels divididos usando parâmetros de codificação introduzidos de uma unidade de controle de codificação 111, gerando assim uma imagem prevista. Depois, o dispositivo de codificação 100 gera um erro de predição ao subtrair a imagem prevista da imagem de entrada dividida na pluralidade de pixels, gera dados codificados ao executar a transformação ortogonal, e a quantização, e então a codificação entrópica para o erro de predição gerado, e emite os dados codificados gerados.

[0040] O dispositivo de codificação 100 executa a codificação prevista ao seletivamente aplicar uma pluralidade de modos de predição que são diferentes entre si pelo menos no tamanho de bloco do bloco de pixels ou no método de gerar uma imagem prevista. O método de gerar uma imagem prevista pode ser grandemente dividido em dois tipos incluindo uma intrapredição na qual uma predição é feita dentro de um quadro alvo de codificação e uma interpredição na qual uma predição compensada por movimento é feita usando um ou mais quadros de referência de diferentes pontos temporais. A intrapredição é também chamada de predição de tela interna, uma predição de quadro interno, ou semelhante, e a interpredição é também chamada de predição entre telas, uma predição entre quadros, uma predição compensada por movimento, ou semelhante.

[0041] A Figura 2 é um diagrama explicativo que ilustra um exemplo de uma sequência de codificação prevista para um bloco de pixels de acordo com a primeira concretização. No exemplo ilustrado na Figura 2, o dispositivo de codificação 100 executa a codificação prevista do lado esquerdo superior para o lado direito inferior no bloco de pixels. Desse modo, em um quadro alvo de processamento de codificação f , no lado esquerdo e no lado superior do bloco de pixels alvo de codificação c , são localizados blocos de pixels p que foram completados para serem codificados. Adiante, para simplificação da descrição, enquanto é assumido que o dispositivo de codificação 100 executa a codificação prevista na ordem ilustrada na Figura 2, a ordem na codificação prevista não é limitada a esta.

[0042] O bloco de pixels representa uma unidade para processar uma imagem e, por exemplo, um bloco apresentando um tamanho $M \times N$ (aqui, M e N são números naturais), um bloco de árvore de codificação, um macrobloco, um sub-bloco, um pixel, ou semelhante corresponde à mesma. Na descrição apresentada abaixo, basicamente, o bloco de pixels é usado como tendo o significado de um bloco de árvore de codificação, mas pode ser usado como tendo um significado diferente. Por exemplo, na descrição de uma unidade de predição, um bloco de pixels é usado como tendo o significado de um bloco de pixels da unidade de predição. Um bloco pode ser referido como uma unidade ou semelhante. Por exemplo, um bloco de codificação pode ser referido como uma unidade de codificação.

[0043] A Figura 3A é um diagrama que ilustra um exemplo do tamanho de um bloco de árvore de codificação de acordo com a primeira concretização. O bloco de árvore de codificação é tipicamente um bloco de pixels de 64×64 , conforme ilustrado na Figura 3A. Entretanto, o bloco de árvore de codificação não é limitado ao mesmo, mas pode ser um bloco de pixels de 32×32 , um bloco de pixels de 16×16 , um

bloco de pixels de 8 x 8, um bloco de pixels de 4 x 4, ou semelhante. Aqui, o bloco de árvore de codificação pode não ser um quadrado, mas, por exemplo, pode ser um bloco de pixels de um tamanho $M \times N$ (aqui, $M \neq N$).

[0044] As Figuras 3B a 3D são diagramas que representam exemplos específicos do bloco de árvore de codificação de acordo com a primeira concretização. A Figura 3B representa um bloco de árvore de codificação apresentando um tamanho de bloco de 64 x 64 ($N = 32$). Aqui, N representa o tamanho de um bloco de árvore de codificação de referência. O tamanho de um caso em que o bloco de árvore de codificação é dividido é definido como N , e o tamanho de um caso em que o bloco de árvore de codificação não é dividido é definido como $2N$. A Figura 3C representa um bloco de árvore de codificação adquirido com a divisão do bloco de árvore de codificação ilustrado na Figura 3B em uma árvore de quadrantes. O bloco de árvore de codificação, conforme ilustrado na Figura 3C, apresenta uma estrutura de árvores de quadrantes. Em um caso em que o bloco de árvore de codificação é dividido, conforme ilustrado na Figura 3C, os números são ligados a quatro blocos de pixels depois da divisão na ordem de varredura Z .

[0045] Além disso, dentro de cada número da árvore de quadrantes, o bloco de árvore de codificação pode ser adicionalmente dividido em uma árvore de quadrantes. Conseqüentemente, o bloco de árvore de codificação pode ser dividido de maneira hierárquica. Em tal caso, a profundidade da divisão é definida como Profundidade. A Figura 3D representa um dos blocos de árvore de codificação adquirido com a divisão do bloco de árvore de codificação ilustrado na Figura 3B em uma árvore de quadrantes, e o tamanho do bloco do mesmo é 32 x 32 ($N = 16$). A profundidade do bloco de árvore de codificação ilustrado na Figura 3B é "0", e a profundidade do bloco de árvore de codificação ilustrado na Figura 3D é "1". Além disso, um bloco de árvore de codifi-

cação apresentando a maior unidade é chamado de um bloco de árvore de codificação grande, e um sinal de imagem de entrada é codificado em tal unidade na ordem de varredura por rastreo.

[0046] Na descrição apresentada abaixo, o bloco alvo codificado ou o bloco de árvore de codificação de uma imagem de entrada pode ser referido como um bloco alvo de predição ou um bloco de pixels de predição. Além disso, a unidade de codificação não é limitada ao bloco de pixels, mas pelo menos um quadro, um campo, uma fatia, uma linha ou um pixel pode ser usado como a unidade de codificação.

[0047] O dispositivo de codificação 100, conforme ilustrado na Figura 1, inclui uma unidade de subtração 101, uma unidade de transformação ortogonal 102, uma unidade de quantização 103, uma unidade de quantização inversa 104, uma unidade de transformação ortogonal inversa 105, uma unidade de adição 106, uma unidade de geração de imagem prevista 107, uma unidade de ajuste de índice 108, uma unidade de avaliação de movimento 109, e uma unidade de codificação 110. Além disso, a unidade de controle de codificação 11 ilustrada na Figura 1 controla o dispositivo de codificação 100 e, por exemplo, pode ser implementada com o uso de uma CPU (Unidade de Processamento Central) ou semelhante.

[0048] A unidade de subtração 101 adquire um erro de predição com a subtração de uma imagem prevista correspondente de uma imagem de entrada dividida em blocos de pixels. A unidade de subtração 101 emite o erro de predição de modo a ser introduzido na unidade de transformação ortogonal 102.

[0049] A unidade de transformação ortogonal 102 executa uma transformação ortogonal, tal como uma transformada discreta do cosseno (DCT) ou uma transformada discreta do seno (DST) para o erro de predição introduzido da unidade de subtração 101, adquirindo assim um coeficiente de transformação. A unidade de transformação or-

togonal 102 emite o coeficiente de transformação de modo a ser introduzido na unidade de quantização 103.

[0050] A unidade de quantização 103 executa um processo de quantização para o coeficiente de transformação introduzido da unidade de transformação ortogonal 102, adquirindo assim um coeficiente de transformação de quantização. Mais especificamente, a unidade de quantização 103 executa a quantização com base em um parâmetro de quantização designado pela unidade de controle de codificação 111 e informação de quantização, tal como uma matriz de quantização. Descrita em maiores detalhes, a unidade de quantização 103 adquire o coeficiente de transformação de quantização com a divisão do coeficiente de transformação por um tamanho de etapa de quantização derivado com base na informação de quantização. O parâmetro de quantização representa a excelência da quantização. A matriz de quantização é usada para ponderar a excelência da quantização para cada componente do coeficiente de transformação. A unidade de quantização 103 emite o coeficiente de transformação de quantização de modo a ser introduzido na unidade de quantização inversa 104 e na unidade de codificação 110.

[0051] A unidade de quantização inversa 104 executa um processo de quantização inversa para o coeficiente de transformação de quantização introduzido da unidade de quantização 103, adquirindo assim um coeficiente de transformação de restauração. Mais especificamente, a unidade de quantização inversa 104 executa a quantização inversa com base na informação de quantização usada pela unidade de quantização 103. Descrita em detalhes, a unidade de quantização inversa 104 adquire um coeficiente de transformação de restauração com a multiplicação do coeficiente de transformação de quantização pelo tamanho de etapa de quantização derivado com base na informação de quantização. Além disso, a informação de quantização usada

pela unidade de quantização 103 é carregada a partir de uma memória interna, que não é ilustrada na figura, da unidade de controle de codificação 111 e é usada. A unidade de quantização inversa 104 emite o coeficiente de transformação de restauração de modo a ser introduzido na unidade de transformação ortogonal inversa 105.

[0052] A unidade de transformação ortogonal inversa 105 executa uma transformação ortogonal inversa, tal como uma transformada discreta inversa do coseno (IDCT) ou uma transformada discreta inversa do seno (IDST) para o coeficiente de transformação de restauração introduzido da unidade de quantização inversa 104, adquirindo assim um erro de predição de restauração. Aqui, a transformação ortogonal inversa executada pela unidade de transformação ortogonal inversa 105 corresponde a uma transformação ortogonal executada pela unidade de transformação ortogonal 102. A unidade de transformação ortogonal inversa 105 emite o erro de predição de restauração de modo a ser introduzido na unidade de adição 106.

[0053] A unidade de adição 106 acrescenta o erro de predição de restauração introduzido da unidade de transformação ortogonal inversa 105 e uma imagem prevista correspondente, gerando assim uma imagem decodificada local. A unidade de adição 106 emite a imagem decodificada local de modo a ser introduzida na unidade de geração de imagem prevista 107.

[0054] A unidade de geração de imagem prevista 107 armazena a imagem decodificada local introduzida da unidade de adição 106 em uma memória (não ilustrada na Figura 1) como uma imagem de referência e emite a imagem de referência armazenada na memória de modo a ser introduzida na unidade de avaliação de movimento 109. Além disso, a unidade de geração de imagem prevista 107 gera uma imagem prevista com a execução de uma predição ponderada compensada por movimento com base na informação de movimento e na

informação de parâmetro WP introduzida na unidade de avaliação de movimento 109. A unidade de geração de imagem prevista 107 emite a imagem prevista de modo a ser introduzida na unidade de subtração 101 e na unidade de adição 106.

[0055] A Figura 4 é um diagrama de bloco que ilustra um exemplo da configuração da unidade de geração de imagem prevista 107 de acordo com a primeira concretização. A unidade de geração de imagem prevista 107, conforme ilustrada na Figura 4, inclui uma unidade de compensação de movimento de múltiplos quadros 201, uma memória 202, uma unidade de compensação de movimento unidirecional 203, uma unidade controle de parâmetro de predição 204, um seletor de imagem de referência 205, uma memória de quadro 206, e uma unidade de controle de imagem de referência 207.

[0056] A memória de quadro 206 armazena a imagem decodificada local introduzida da unidade de adição 106 como uma imagem de referência sob o controle da unidade de controle de imagem de referência 207. A memória de quadro 206 inclui uma pluralidade de conjuntos de memória FM1 a FMN (aqui, $N \geq 2$) usada para temporariamente armazenar a imagem de referência.

[0057] A unidade de controle de parâmetro de predição 204 prepara uma pluralidade de combinações, cada de um número de imagem de referência e um parâmetro de predição como uma tabela com base na informação de movimento introduzida da unidade de avaliação de movimento 109. Aqui, a informação de movimento representa a informação de um vetor de movimento que representa o desvio de um movimento que é usado para a predição compensada por movimento, o número de imagem de referência, e um modo de predição. O parâmetro de predição representa a informação referente ao vetor de movimento e ao modo de predição. Depois, a unidade de controle de parâmetro de predição 104 seleciona uma combinação de um número de

referência e um parâmetro de predição usados para gerar uma imagem prevista com base na imagem de entrada e emite a combinação selecionada de modo a permitir que o número de imagem de referência seja introduzido no seletor de imagem de referência 205 e a permitir que o parâmetro de predição seja introduzido na unidade de compensação de movimento unidirecional 203.

[0058] O seletor de imagem de referência 205 é um comutador que muda um dos terminais de saída das memórias de quadro FM1 a FMN, que são incluídas na memória de quadro 206, a ser comutado com base em um número de imagem de referência introduzido a partir da unidade de controle de parâmetro de predição 204. Por exemplo, quando o número de imagem de referência for "0", o seletor de imagem de referência 205 conectará o terminal de saída da memória de quadro FM1 ao terminal de saída do seletor de imagem de referência 205, e, quando o número de imagem de referência for N - 1, o seletor de imagem de referência 205 conectará o terminal de saída da memória de quadro FMN ao terminal de saída do seletor de imagem de referência 205. O seletor de imagem de referência 205 emite uma imagem de referência armazenada na memória de quadro à qual o terminal de saída é conectado dentre as memórias de quadro FM1 a FMN incluídas na memória de quadro 206 de modo a ser introduzida na unidade de compensação de movimento unidirecional 203 e na unidade de avaliação de movimento 109.

[0059] A unidade de compensação de movimento previsto unidirecional 203 executa um processo de predição compensado por movimento com base no parâmetro de predição introduzido da unidade de controle de parâmetro de predição 204 e na imagem de referência introduzida do seletor de imagem de referência 205, gerando assim uma imagem prevista unidirecional.

[0060] A Figura 5 é um diagrama que ilustra um exemplo da rela-

ção entre os vetores de movimento para uma predição compensada por movimento em uma predição bidirecional de acordo com a primeira concretização. Na predição compensada por movimento, um processo de interpolação é executado usando a imagem, e uma imagem prevista unidirecional é gerada com base nos desvios de movimentos da imagem interpolada gerada e na imagem introduzida do bloco de pixels localizado na posição alvo de codificação. Aqui, o desvio é um movimento de vetor. Conforme ilustrado na Figura 5, na fatia de predição bidirecional (fatia B), uma imagem prevista é gerada com o uso de dois tipos de imagens de referência e um conjunto de vetores de movimento. Como o processo de interpolação, um processo de interpolação de precisão de 1/2 pixel, um processo de interpolação de precisão de 1/4 de pixel, ou semelhante é usado, e, com a execução de um processo de filtragem para a imagem de referência, é gerado um valor da imagem interpolada. Por exemplo, em H.264, no qual uma interpolação de precisão de até 1/4 de pixel pode ser executada para um sinal de luminância, o desvio é representado como quatro vezes a precisão de pixel de número inteiro.

[0061] A unidade de compensação de movimento previsto unidirecional 203 emite uma imagem prevista unidirecional e temporariamente armazena a imagem prevista unidirecional na memória 202. Aqui, em um caso em que a informação de movimento (parâmetro de predição) representa uma predição bidirecional, a unidade de compensação de movimento de múltiplos quadros 201 faz uma predição ponderada usando dois tipos de imagens previstas unidirecionais. Conseqüentemente, a unidade de compensação de movimento previsto unidirecional 203 armazena uma imagem prevista unidirecional correspondendo ao primeiro tipo na memória 202 e diretamente emite uma imagem prevista unidirecional correspondendo ao segundo tipo para a unidade de compensação de movimento de múltiplos quadros 201. Aqui, a

imagem prevista unidirecional correspondendo ao primeiro tipo será referida como uma primeira imagem prevista, e a imagem prevista unidirecional correspondendo ao segundo tipo será referida como uma segunda imagem prevista.

[0062] Além disso, as unidades de compensação de movimento unidirecional 203 podem ser preparadas e gerar duas imagens previstas unidirecionais. Em tal caso, quando a informação de movimento (parâmetro de predição) representar uma predição unidirecional, a unidade de compensação de movimento unidirecional 203 poderá diretamente emitir a primeira imagem prevista unidirecional para a unidade de compensação de movimento de múltiplos quadros como uma primeira imagem prevista.

[0063] A Figura 6 é um diagrama de bloco que ilustra um exemplo da configuração da unidade de compensação de movimento de múltiplos quadros 201 de acordo com a primeira concretização. Conforme ilustrado na Figura 6, a unidade de compensação de movimento de múltiplos quadros 201 inclui uma unidade de compensação de movimento padrão 301, uma unidade de compensação ponderada de movimento 302, uma unidade de controle de parâmetro WP 303, e seletores WP 304 e 305.

[0064] A unidade de controle de parâmetro WP 303 emite um sinalizador de aplicativo WP e informação de ponderação com base na informação de parâmetro WP introduzida da unidade de avaliação de movimento 109 de modo a introduzir o sinalizador de aplicativo WP nos seletores WP 304 e 305 e introduzir a informação de ponderação na unidade de compensação ponderada de movimento 302.

[0065] Aqui, a informação de parâmetro WP inclui informação da precisão de ponto fixo do fator de ponderação, um primeiro sinalizador de aplicativo WP, um primeiro fator de ponderação, e um primeiro deslocamento correspondendo à primeira imagem prevista, e um segundo

sinalizador de aplicativo WP, um segundo fator de ponderação, e um segundo deslocamento correspondendo à segunda imagem prevista. O sinalizador de aplicativo WP é um parâmetro que pode ser ajustado para cada imagem de referência correspondente e componente de sinal e representa se uma predição de compensação ponderada de movimento é feita ou não. A informação de ponderação inclui informação da precisão de ponto fixo do fator de ponderação, do primeiro fator de ponderação, do primeiro deslocamento, do segundo fator de ponderação, e do segundo deslocamento.

[0066] Descrita em detalhes, quando a informação de parâmetro WP for introduzida da unidade de avaliação de movimento 109, a unidade de controle de parâmetro WP 303 emitirá a informação de parâmetro WP em sendo dividida no primeiro sinalizador de aplicativo WP, no segundo sinalizador de aplicativo WP, e na informação de ponderação, introduzindo assim o primeiro sinalizado de aplicativo WP no seletor WP 304, o segundo sinalizador de aplicativo WP no seletor WP 305, e a informação de ponderação na unidade de compensação ponderada de movimento 302.

[0067] Os seletores WP 304 e 305 mudam as extremidades de conexão das imagens previstas com base nos sinalizadores de aplicativo WP introduzidos da unidade de controle de parâmetro WP 303. Em um caso em que o sinalizador de aplicativo WP correspondente é "0", cada um dos seletores WP 304 e 305 conecta a extremidade de saída do mesmo à unidade compensação de movimento padrão 301. Depois, os seletores WP 304 e 305 emitem a primeira e a segunda imagens previstas de modo a serem introduzidas na unidade de compensação de movimento padrão 301. Por outro lado, em um caso em que o sinalizador de aplicativo WP correspondente é "1", cada um dos seletores WP 304 e 305 conecta a extremidade de saída do mesmo à unidade de compensação ponderada de movimento 302. Depois, os

seletores WP 304 e 305 emitem a primeira e a segunda imagens previstas de modo a serem introduzidas na unidade de compensação ponderada de movimento 302.

[0068] A unidade de compensação de movimento padrão 301 executa o processamento médio com base nas duas imagens previstas unidirecionais (a primeira e a segunda imagens previstas) introduzidas dos seletores WP 304 e 305, gerando assim uma imagem prevista. Mais especificamente, em um caso em que o primeiro e o segundo sinalizadores de aplicativo WP são "0"s, a unidade de compensação de movimento padrão 301 executa o processamento médio com base na Expressão Numérica (1).

$$P[x,y] = \text{Clip1}((\text{PL0}[x,y] + \text{PL1}[x,y] + \text{offset}(2) \gg (\text{shift2})) \quad (1)$$

[0069] Aqui, $P[x,y]$ é uma imagem prevista, $\text{PL0}[x,y]$ é uma primeira imagem prevista, e $\text{PL1}[x,y]$ é uma segunda imagem prevista. Além disso, offset2 e shift2 são parâmetros de um processo de arredondamento no processamento médio e são determinados com base na precisão de cálculo interno da primeira e da segunda imagens previstas. Quando a precisão de bit da imagem prevista for 1, e a precisão de bit da primeira e da segunda imagens previstas for M ($L \leq M$), shift2 será formulado pela Expressão Numérica (2), e offset2 será formulado pela Expressão Numérica (3).

$$\text{shift2} = (M - L + 1) \quad (2)$$

$$\text{offset2} = (1 \ll (\text{shift2} - 1)) \quad (3)$$

[0070] Por exemplo, a precisão de bit da imagem prevista é "8", e a precisão de bit da primeira e da segunda imagens previstas é "14", $\text{shift2} = 7$ com base na Expressão Numérica (2), e $\text{offset2} = (1 \ll 6)$ com base na Expressão Numérica (3).

[0071] Além disso, em um caso em que o modo de predição representado pela informação de movimento (parâmetro de predição) é a predição unidirecional, a unidade de compensação de movimento

padrão 301 calcula uma imagem prevista final usando apenas a primeira imagem prevista com base na Expressão Numérica (4).

$$P[x,y] = \text{Clip1}((\text{PLX}[x,y] + \text{offset1}) \gg (\text{shift1})) \quad (4)$$

[0072] Aqui, $\text{PLX}[x,y]$ representa uma imagem prevista unidirecional (primeira imagem prevista), e X é um identificador que representa ou "0" ou "1" como uma lista de referência. Por exemplo, $\text{PLX}[x,y]$ é $\text{PL0}[x,y]$ em um caso em que a lista de referência é "0" e $\text{PL1}[x,y]$ em um caso em que a lista de referência é "1". Além disso, offset1 e shift1 são parâmetros para um processo de arredondamento e são determinados com base na precisão de cálculo interno da primeira imagem prevista. Quando a precisão de bit da imagem prevista for L, e a precisão de bit da primeira imagem prevista for M, shift1 será formulado pela Expressão Numérica (5), e offset1 será formulado pela Expressão Numérica (6).

$$\text{shift1} = (M - L) \quad (5)$$

$$\text{offset1} = (1 \ll (\text{shift1} - 1)) \quad (6)$$

[0073] Por exemplo, em um caso em que a precisão de bit da imagem prevista é "8", e a precisão de bit da primeira imagem prevista é "14", $\text{shift1} = 6$ com base na Expressão Numérica (5), e $\text{offset1} = (1 \ll 5)$ com base na Expressão Numérica (6).

[0074] A unidade de compensação ponderada de movimento executa a compensação ponderada de movimento com base nas duas imagens previstas unidirecionais (a primeira e a segunda imagens previstas) introduzidas dos seletores WP 304 e 305 e na informação de ponderação introduzida da unidade de controle de parâmetro WP 303. Mais especificamente, a unidade de compensação ponderada de movimento 302 executa o processo de ponderação com base na Expressão Numérica (7) em um caso em que o primeiro e o segundo sinalizadores de aplicativo WP são "1"s.

$$P[x,y] = \text{Clip1}((\text{PL0}[x,y]*w_{0c} + \text{PL1}[x,y]*w_{1c} + (1 \ll \log \text{WD}_c)) \gg$$

$$(\log WD_C + 1)) + ((o_{0C} + o_{1C} + 1) \gg 1)) \quad (7)$$

[0075] Aqui, w_{0C} representa um fator de ponderação correspondendo à primeira imagem prevista, w_{1C} representa um fator de ponderação correspondendo à segunda imagem prevista, o_{0C} representa um deslocamento correspondendo à primeira imagem prevista, e o_{1C} representa um deslocamento correspondendo à segunda imagem prevista. Em seguida, eles serão referidos como um primeiro fator de ponderação, um segundo fator de ponderação, um primeiro deslocamento, e um segundo deslocamento, respectivamente. $\log WD_C$ é um parâmetro que representa a precisão de ponto fixo de cada fator de ponderação. Além disso, uma variável C representa um componente de sinal. Por exemplo, no caso de um sinal espacial YUV, um sinal de luminância será representado por $C = Y$, um sinal de diferença de cor Cr será representado por $C = Cr$, e um componente de diferença de cor Cb será representado por $C = Cb$.

[0076] Além disso, em um caso em que a precisão de cálculo da primeira e da segunda imagens previstas e a precisão de cálculo da imagem prevista são diferentes entre si, a unidade de compensação ponderada de movimento 302 realiza um processo de arredondamento com o controle de $\log WD_C$, que é a precisão de ponto fixo, como na Expressão Numérica (8).

$$\log WD'_C = \log WD_C + \text{offset1} \quad (8)$$

[0077] O processo de arredondamento pode ser realizado com a substituição de $\log WD_C$ representado na Expressão Numérica (7) por $\log WD'_C$ representado na Expressão Numérica (8). Por exemplo, em um caso em que a precisão de bit da imagem prevista é "8", e a precisão de bit da primeira e da segunda imagens é "14", com a redefinição de $\log WD_C$, é possível realizar um processo de arredondamento de lote para a precisão de cálculo similar àquela de shift2 representada na Expressão Numérica (1).

[0078] Além disso, em um caso em que o modo de predição representado pela informação de movimento (parâmetro de predição) é uma predição unidirecional, a unidade de compensação ponderada de movimento 302 calcula uma imagem prevista final usando apenas a primeira imagem prevista com base na Expressão Numérica (9).

$$P[x,y] - \text{Clip1}((\text{PLX}[x,y]*w_{xc} + (1 \ll \log\text{WD}_c - 1)) \gg (\log\text{WD}_c)) \quad (9)$$

[0079] Aqui, $\text{PLX}[x,y]$ representa uma imagem prevista unidirecional (primeira imagem prevista), w_{xc} representa um fator de ponderação correspondendo à predição unidirecional, e X é um identificador que representa ou "0" ou "1" como uma lista de referência. Por exemplo, $\text{PLX}[x,y]$ e w_{xc} são $\text{PL0}[x,y]$ e w_{0c} em um caso em que a lista de referência é "0" e são $\text{PL1}[x,y]$ e w_{1c} em um caso em que a lista de referência é "1".

[0080] Além disso, em um caso em que a precisão de cálculo da primeira e da segunda imagens e a precisão de cálculo da imagem prevista são diferentes entre si, a unidade de compensação ponderada de movimento 302 realiza um processo de arredondamento com o controle de $\log\text{WD}_c$, que é precisão de ponto fixo, como na Expressão Numérica (8), similarmente ao caso da predição bidirecional.

[0081] O processo de arredondamento pode ser realizado com a substituição de $\log\text{WD}_c$ representado na Expressão Numérica (7) por $\log\text{WD}'_c$ representado na Expressão Numérica (8). Por exemplo, em um caso em que a precisão de bit da imagem prevista é "8", e a precisão de bit das primeiras imagens previstas é "14", com a redefinição de $\log\text{WD}_c$, é possível realizar um processo de arredondamento de lote para a precisão de cálculo similar àquela de shift1 representada na Expressão Numérica (4).

[0082] A Figura 7 é um diagrama explicativo que ilustra um exemplo de precisão de ponto fixo de um fator de ponderação de acordo com a primeira concretização e é um diagrama que ilustra um exemplo

de mudanças em uma imagem em movimento apresentando uma mudança de brilho na direção do tempo e um valor de escala de cinza. No exemplo ilustrado na Figura 7, um quadro alvo de codificação é $\text{Frame}(t)$, um quadro que é um quadro antes do quadro alvo de codificação no tempo é $\text{Frame}(t - 1)$, e um quadro que é um quadro depois do quadro alvo de codificação no tempo é $\text{Frame}(t + 1)$. Conforme ilustrado na Figura 7, em uma imagem de desvanecimento que muda do branco para o preto, o brilho (valor de escala de cinza) da imagem diminui de acordo com o tempo decorrido. O fator de ponderação representa o grau de mudança na Figura 7, e, como é evidente das Expressões Numéricas (7) e (9), assume um valor de "1,0" em um caso em que não há nenhuma mudança no brilho. A precisão de ponto fixo é um parâmetro que controla uma largura de intervalo correspondendo a um ponto decimal do fato de ponderação, e o fator de ponderação é $1 \ll \log W D_C$ em um caso em que não há nenhuma mudança no brilho.

[0083] Além disso, no caso de uma predição unidirecional, vários parâmetros (o segundo sinalizador de aplicativo WP, o segundo fator de ponderação, e a segunda informação de deslocamento) correspondendo à segunda imagem prevista não são usados e podem ser ajustados em valores iniciais determinados de antemão.

[0084] Com referência novamente à Figura 1, a unidade de avaliação de movimento 109 executa uma avaliação de movimento entre uma pluralidade de quadros com base em uma imagem de entrada e uma imagem de referência introduzidas da unidade de geração de imagem prevista 107 e emite a informação de movimento e a informação de parâmetro WP, introduzindo assim a informação de movimento na unidade de geração de imagem prevista 107 e na unidade de codificação 110 e introduzindo a informação de parâmetro WP na unidade de geração de imagem prevista 107 e na unidade de ajuste de índice 108.

[0085] A unidade de avaliação de movimento 109 calcula um erro, por exemplo, com o cálculo de diferenças entre uma imagem de entrada de um bloco de pixels alvo de predição e uma pluralidade de imagens de referência correspondendo à mesma posição como um ponto de partida, muda a posição com precisão fracional, e calcula uma ótima informação de movimento usando uma técnica tal como a associação de blocos para encontrar um bloco de um erro mínimo ou semelhante. No caso de uma predição bidirecional, a unidade de avaliação de movimento 109 executa a associação de blocos incluindo uma predição de compensação de movimento padrão, conforme representado nas Expressões Numéricas (1) e (4) usando a informação de movimento derivada da predição unidirecional, calculando assim a informação de movimento da predição bidirecional.

[0086] Neste momento, a unidade de avaliação de movimento 109 pode calcular a informação de parâmetro WP com a execução da associação de blocos incluindo uma predição de compensação ponderada de movimento, conforme representado nas Expressões Numéricas (7) e (9). Além disso, para o cálculo da informação de parâmetro WP, pode ser usado um método de calcular um fator de ponderação ou um deslocamento usando um gradiente de brilho da imagem de entrada, um método de calcular um fator de ponderação ou um deslocamento de acordo com o acúmulo de um erro de predição no momento da codificação, ou semelhante. Adicionalmente, como a informação de parâmetro WP, pode ser usado um valor fixo determinado de antemão para cada dispositivo de codificação.

[0087] Aqui, um método de calcular um fator de ponderação, a precisão de ponto fixo do fator de ponderação, e um deslocamento de uma imagem em movimento apresentando uma mudança de brilho no tempo será descrito com referência à Figura 7. Conforme descrito acima, na imagem de desvanecimento que muda do branco para o preto,

conforme ilustrado na Figura 7, o brilho (valor de escala de cinza) da imagem diminuirá de acordo com o tempo decorrido. A unidade de avaliação de movimento 109 pode calcular o fator de ponderação com o cálculo da inclinação do mesmo.

[0088] A precisão de ponto fixo do fator de ponderação é a informação que representa a precisão da inclinação, e a unidade de avaliação de movimento 109 pode calcular um valor ótimo com base em uma distância para a imagem de referência no tempo e o grau de mudança do brilho da imagem. Por exemplo, na Figura 7, em um caso em que o fator de ponderação entre $\text{Frame}(t - 1)$ e o $\text{Frame}(t + 1)$ é 0,75 com precisão fracional, $3/4$ podem ser representados no caso de $1/4$ de precisão, e, conseqüentemente, a unidade de avaliação de movimento 109 ajusta a precisão de ponto fixo 2^m ($1 \ll 2$). Uma vez que o valor da precisão de ponto fixo tem influência sobre a quantidade de códigos de um caso em que o fator de ponderação é codificado, como o valor da precisão de ponto fixo, um ótimo valor poderá ser selecionado em vista da quantidade de código e da precisão de predição. Além disso, o valor da precisão de ponto fixo pode ser um valor fixo determinado de antemão.

[0089] Além disso, em um caso em que a inclinação não é correspondida, a unidade de avaliação de movimento 109 poderá calcular o valor do deslocamento com a aquisição de um valor de correção (quantidade de desvio) correspondendo à interceptação da função linear. Por exemplo, na Figura 7, em um caso em que um fator de ponderação entre $\text{Frame}(t - 1)$ e $\text{Frame}(t + 1)$ é 0,60 com precisão de ponto decimal, e a precisão de ponto fixo é "1" ($1 \ll 1$), há uma grande possibilidade de que o fator de ponderação seja ajustado em "1" (correspondendo à precisão de ponto decimal de 0,50 do fator de ponderação). Em tal caso, uma vez que a precisão de ponto decimal do fator de ponderação desvia de 0,60, que é um ótimo valor, por 0,10, a uni-

dade de avaliação de movimento 109 calcula um valor de correção correspondendo ao mesmo com base em um valor máximo do pixel e é ajustado como o valor do deslocamento. Em um caso em que o valor máximo do pixel é 255, a unidade de avaliação de movimento 109 poderá ajustar um valor tal como 25 ($255 \times 0,1$).

[0090] Na primeira concretização, embora a unidade de avaliação de movimento 109 seja representada como uma função do dispositivo de codificação 100 como um exemplo, a unidade de avaliação de movimento 109 não é uma configuração essencial do dispositivo de codificação 100, e, por exemplo, a unidade de avaliação de movimento 109 pode ser um dispositivo diferente do dispositivo de codificação 100. Em tal caso, a informação de movimento e a informação de parâmetro WP calculadas pela unidade de avaliação de movimento 109 podem ser carregadas no dispositivo de codificação 100.

[0091] A unidade de ajuste de índice 108 recebe a informação de parâmetro WP introduzida da unidade de avaliação de movimento 109, verifica uma lista de referência (número da lista) e uma imagem de referência (número de referência), e emite a informação de índice de modo a ser introduzida na unidade de codificação 110. A unidade de ajuste de índice 108 gera a informação de índice com o mapeamento da informação de parâmetro WP introduzida da unidade de avaliação de movimento 109 em um elemento de sintaxe a ser descrito posteriormente.

[0092] As Figuras 8A e 8B são diagramas que ilustram exemplos da informação de parâmetro WP de acordo com a primeira concretização. Um exemplo da informação de parâmetro WP no momento da fatia P é conforme ilustrado na Figura 8A, e um exemplo da informação de parâmetro WP no momento da fatia B é como ilustrado nas Figuras 8A e 8B. Um número da lista é um identificador que representa uma direção de predição. O número da lista tem um valor de "0" no

caso de uma predição unidirecional. Por outro lado, no caso de uma predição bidirecional, dois tipos de predição podem ser usados, e, conseqüentemente, o número da lista apresenta dois valores de "0" e "1". Um número de referência é um valor que corresponde a 1 ou N representado na memória de quadro 206. Uma vez que a informação de parâmetro WP é mantida para cada lista de referência e imagem de referência, em um caso em que há imagens de referência N, peças 2N de informação são necessárias no momento da fatia B.

[0093] Com referência novamente à Figura 1, a unidade de codificação 110 executa um processo de codificação de vários parâmetros de codificação, tal como o coeficiente de transformação de quantização introduzido da unidade de quantização 103, a informação de movimento introduzida da unidade de avaliação de movimento 109, a informação de índice introduzida da unidade de ajuste de índice 108, e a informação de quantização designada pela unidade de controle de codificação 111, gerando assim dados codificados. Como o processo de codificação, por exemplo, há uma codificação Huffman ou uma codificação aritmética.

[0094] Aqui, os parâmetros de codificação são parâmetros, tais como a informação de predição que representa um método de predição ou semelhante, a informação referente ao coeficiente de transformação de quantização, e a informação referente à quantização, que são necessários para um processo de decodificação. Por exemplo, pode ser configurado que uma memória interna não ilustrada na figura seja incluída na unidade de controle de codificação 111, os parâmetros de codificação sejam mantidos na memória interna, e os parâmetros de codificação de um bloco de pixels adjacente, que foi completado para ser codificado, sejam usados, quando um bloco de pixels for codificado. Por exemplo, em uma intrapredição de H.264, a informação de predição de um bloco de pixels pode ser derivada da informação de

predição de um bloco adjacente que foi completado como sendo codificado.

[0095] A unidade de codificação 110 emite os dados codificados gerados no tempo de saída apropriado gerenciado pela unidade de controle de codificação 111. Vários tipos de informação, que são dados codificados de saída, por exemplo, são multiplexados por uma unidade de multiplexação não ilustrada na figura ou semelhante, é temporariamente armazenado em um buffer de saída não ilustrado na figura ou semelhante e, então, por exemplo, é emitido para um sistema de armazenamento (meio de armazenamento) ou um sistema de transmissão (linha de comunicação).

[0096] A unidade de codificação 110 inclui uma unidade de codificação entrópica 110A e uma unidade de reconfiguração de índice 110B.

[0097] A unidade de codificação entrópica 110A executa um processo de codificação, tal como a codificação de comprimento variável ou o processo de codificação aritmético para informação que foi introduzida. Por exemplo, em H.264, é usada uma codificação de comprimento variável adaptativa com base em contexto (CAVLC), uma codificação aritmética binária adaptativa com base em contexto (CABAC), ou semelhante.

[0098] A fim de reduzir o comprimento de código de um elemento de sintaxe da informação de índice introduzida da unidade de ajuste de índice 108, a unidade de reconfiguração de índice 110B executa um processo de predição com base nas características dos parâmetros do elemento de sintaxe, calcula uma diferença entre o valor (valor direto) do elemento de sintaxe e um valor previsto, e emite a diferença para a unidade de codificação entrópica 110A. Um exemplo específico do processo de predição será descrito posteriormente.

[0099] A Figura 9 é um diagrama que ilustra um exemplo de sintaxe

xe 500 usado pelo dispositivo de codificação 100 de acordo com a primeira concretização. A sintaxe 500 ilustra a estrutura de dados codificados gerados pela codificação de uma imagem de entrada (dados de imagem em movimento) usando o dispositivo de codificação 100. Quando os dados codificados forem decodificados, um dispositivo de decodificação a ser descrito posteriormente executará uma análise de sintaxe de uma imagem em movimento com referência a uma estrutura de sintaxe que é igual àquela da sintaxe 500.

[00100] A sintaxe 500 inclui três partes incluindo uma sintaxe de alto nível 501, uma sintaxe de nível de fatia 502, e uma sintaxe de nível de árvore de codificação 503. A sintaxe de alto nível 501 inclui a informação de sintaxe de uma camada superior que apresenta um nível mais alto do que a fatia. Aqui, a fatia representa uma área retangular ou uma área contínua incluída em um quadro ou um campo. A sintaxe de nível de fatia 502 inclui informação que é necessária para decodificar cada fatia. A sintaxe de nível de árvore de codificação 503 inclui informação que é necessária para decodificar cada árvore de codificação (em outras palavras, cada bloco de árvore de codificação). Cada destas partes inclui sintaxe mais detalhada.

[00101] A sintaxe de alto nível 501 inclui sintaxes de uma sequência e um nível de imagem, tal como uma sintaxe de ajuste de parâmetro de sequência 504, uma sintaxe de ajuste de parâmetro de imagem 505, e uma sintaxe de ajuste de parâmetro de adaptação 506.

[00102] A sintaxe de nível de fatia 502 inclui uma sintaxe de cabeçalho da fatia 507, uma sintaxe de tabela de peso nivelado 508, uma sintaxe de dados de fatia 509, e semelhante. A sintaxe de tabela de peso nivelado 508 é chamada de sintaxe de cabeçalho da fatia 507.

[00103] A sintaxe de nível de árvore de codificação 503 inclui uma sintaxe de unidade de árvore de codificação 510, uma sintaxe de unidade de transformação 511, uma sintaxe de unidade de predição 512,

e semelhante. A sintaxe de unidade de árvore de codificação 510 pode ter uma estrutura de árvore de quadrantes, Mais especificamente, a sintaxe de unidade de árvore de codificação 510 pode ser recursivamente chamada adicionalmente como um elemento de sintaxe da sintaxe de unidade de árvore de codificação 510. Em outras palavras, o bloco de árvore de codificação pode ser subdividido em árvores de quadrantes. Além disso, a sintaxe de unidade de transformação 511 é incluída na sintaxe de unidade de árvore de codificação 510. A sintaxe de unidade de transformação 511 é chamada de sintaxe de unidade de cada árvore de codificação 510 localizada em uma extremidade traseira da árvore de quadrantes. Na sintaxe de unidade de transformação 511, é descrita a informação referente à transformação ortogonal inversa, à quantização, e semelhante. Nas sintaxes, a informação referente à predição de compensação ponderada de movimento pode ser descrita.

[00104] A Figura 10 é um diagrama que ilustra um exemplo da sintaxe de ajuste de parâmetro de imagem 505 de acordo com a primeira concretização. Aqui, `weighted_pred_flag`, por exemplo, é um elemento de sintaxe que representa a legitimidade ou ilegitimidade de uma predição de compensação ponderada de acordo com a primeira concretização para uma fatia P. Em um caso em que o `weighted_pred_flag` é "0", a predição de compensação ponderada de movimento de acordo com a primeira concretização dentro da fatia P é inválida. Consequentemente, o sinalizador de aplicativo WP incluído na informação de parâmetro WP é continuamente ajustado em "0", e as extremidades de saída dos seletores WP 304 e 305 são conectadas à unidade de compensação de movimento padrão 301. Por outro lado, em um caso em que o `weighted_pred_flag` é "1", a predição de compensação ponderada de movimento de acordo com a primeira concretização dentro da fatia P é válida.

[00105] Como outro exemplo, em um caso em que o `weighted_pred_flag` é "1", a legitimidade ou a ilegitimidade da predição de compensação ponderada de movimento de acordo com a primeira concretização pode ser definida para cada área local dentro da fatia na sintaxe de uma camada inferior (o cabeçalho da fatia, o bloco de árvore de codificação, a unidade de transformação, a unidade de predição, e semelhante).

[00106] Além disso, `weighted_bipred_idc`, por exemplo, é um elemento de sintaxe que representa a legitimidade ou a ilegitimidade de uma predição de compensação ponderada de acordo com a primeira concretização para uma fatia B. Em um caso em que o `weighted_bipred_idc` é "0", a predição de compensação ponderada de movimento de acordo com primeira concretização dentro da fatia B é inválida. Consequentemente, o sinalizador de aplicativo WP incluído na informação de parâmetro WP é constantemente ajustado em "0", e as extremidades de saída dos seletores WP 304 e 305 são conectadas à unidade de compensação de movimento padrão 301. Por outro lado, em um caso em que o `weighted_bipred_idc` é "1", a predição de compensação ponderada de movimento de acordo com a primeira concretização dentro da fatia B é válida.

[00107] Como outro exemplo, em um caso em que o `weighted_bipred_idc` é "1", a legitimidade ou a ilegitimidade da predição de compensação ponderada de movimento de acordo com a primeira concretização pode ser definida para cada área local dentro da fatia na sintaxe de uma camada inferior (o cabeçalho da fatia, o bloco de árvore de codificação, a unidade de transformação, e semelhante).

[00108] A Figura 11 é um diagrama que ilustra um exemplo da sintaxe de cabeçalho da fatia 507 de acordo com a primeira concretização. Aqui, o tipo de fatia representa o tipo de fatia (uma fatia I, uma fatia P, uma fatia B, ou semelhante) de fatia. Além disso, `pic_para-`

meter_set_id é um identificador que apresenta uma sintaxe de ajuste de parâmetro de imagem 505 a ser referida. num_ref_idx_active_override_flag é um sinalizador que representa se atualiza o número de imagens de referência válidas, e, em um caso em que este sinalizador é "1", podem ser usados num_ref_idx_10_active_minus1 e num_ref_idx_11_active_minus1 que definem o número de imagens de referência da lista de referência. Além disso, pred_weight_table() é uma função que representa a sintaxe de tabela de peso de predição usada para uma predição de compensação ponderada de movimento, e esta função é exigida em um caso em que o weighted_pred_flag é "1" no caso de uma fatia P e em um caso em que weighted_bipred_idc é "1" no caso de uma fatia B.

[00109] A Figura 12 é um diagrama que ilustra um exemplo da sintaxe de tabela de peso de predição 508 de acordo com a primeira concretização. Aqui, luma_log2_weight_denom representa a precisão de ponto fixo do fator de ponderação do sinal de luminância em uma fatia e é um valor correspondendo a $\log W_{dC}$ representando na Expressão Numérica (7) ou (9). Além disso, chroma_log2_weight_denom representa a precisão de ponto fixo do fator de ponderação de um sinal de diferença de cor em uma fatia e é um valor que correspondendo a $\log W_{dC}$ representado na Expressão Numérica (7) ou (9). chroma_format_idc é um identificador que representa um espaço de cor, e MONO_IDX é um valor que apresenta um vídeo monocromático. Além disso, num_ref_common_active_minus1 representa um valor que é adquirido com a subtração de uma das inúmeras imagens de referência incluídas em uma lista comum em uma fatia.

[00110] luma_weight_10_flag e luma_weight_11_flag representam sinalizadores de aplicativo WP de sinais de luminância correspondendo às Listas 0 e 1. Em um caso em que este sinalizador é "1", uma predição de compensação ponderada de movimento do sinal de lumi-

nância de acordo com a primeira concretização é válida para todas as áreas dentro da fatia. Além disso, `chroma_weight_10_flag` e `chroma_weight_11_flag` representam sinalizadores de aplicativo WP de sinais de diferença de cor correspondendo às Listas 0 e 1. Em um caso em que este sinalizador é "1", uma predição de compensação ponderada de movimento de um sinal de diferença de cor de acordo com a primeira concretização é válida para toda a área dentro da fatia. `luma_weight_10[i]` e `luma_weight_11[i]` são fatores de ponderação dos $i^{\text{ésimos}}$ sinais de luminância gerenciados pelas Listas 0 e 1. Além disso, `luma_offset_10[i]` e `luma_offset_11[i]` são deslocamentos dos $i^{\text{ésimos}}$ sinais de luminância gerenciados pelas Listas 0 e 1. Estes são valores correspondem a w_{0c} , w_{1c} , o_{0c} , o_{1c} representados na Expressão Numérica (7) ou (9). Aqui, $C = Y$.

[00111] `chroma_weight_10[i][j]` e `chroma_weight_11[i][j]` são fatores de ponderação dos $i^{\text{ésimos}}$ sinais de diferença de cor gerenciados pelas Listas 0 e 1. Além disso, `chroma_offset_10[i][j]` e `chroma_offset_11[i][j]` são deslocamentos dos $i^{\text{ésimos}}$ sinais de diferença de cor gerenciados pelas Listas 0 e 1. Estes são valores correspondem a w_{0c} , w_{1c} , o_{0c} , o_{1c} representados na Expressão Numérica (7) ou (9). Aqui, $C = Cr$ ou Cb . Além disso, j representa um componente da diferença de cor, e, por exemplo, no caso de um sinal de YUV 4 : 2 : 0, $j = 0$ representa um componente Cr , e $j = 1$ representa um componente Cb .

[00112] Aqui, um método de prever cada elemento de sintaxe referente à predição ponderada na configuração de sintaxe será descrito em detalhes. A predição do elemento de sintaxe é executada pela unidade de reconfiguração de índice 110B. A Figura 13 é um diagrama que ilustra um exemplo da configuração de sintaxe que explicitamente representa um método de predição de acordo com a primeira concretização. No exemplo ilustrado na Figura 13, enquanto o elemento de sintaxe do qual foi introduzida a predição é indicado com a conexão de

um prefixo "delta", a configuração de sintaxe basicamente tem os mesmos elementos componentes que a configuração de sintaxe ilustrada na Figura 12.

[00113] Primeiro, será descrito um método de predição entre sinais de `luma_log2_weight_denom` e `chroma_log2_weight_denom` representando a precisão de ponto fixo do fator de ponderação. A unidade de reconfiguração de índice 110B executa o processo de predição entre sinais de `luma_log2_weight_denom` e `chroma_log2_weight_denom` usando a Expressão Numérica (10) e executa um processo de restauração usando a Expressão Numérica (11). Aqui, conforme ilustrado nas Figuras 12 e 13, uma vez que `luma_log2_weight_denom` é definido primeiro, `chroma_log2_weight_denom` é previsto com base no valor de `luma_log2_weight_denom`.

$$\text{delta_chroma_log2_weight_denom} = (\text{chroma_log2_weight_denom} - \text{luma_log2_weight_denom}) \quad (10)$$

$$\text{chroma_log2_weight_denom} = (\text{luma_log2_weight_denom} + \text{delta_chroma_log2_weight_denom}) \quad (11)$$

[00114] A Figura 14 é um fluxograma que ilustra um exemplo do processo de prever `chroma_log2_weight_denom` de acordo com a primeira concretização.

[00115] Primeiro, a unidade de reconfiguração de índice 110B deriva `luma_log2_weight_denom` ajustado na informação de índice como um valor previsto (Etapa S101).

[00116] Subsequentemente, a unidade de reconfiguração de índice 110B subtrai `luma_log2_weight_denom` de `chroma_log2_weight_denom` (Etapa 102) e ajusta um valor de diferença do mesmo como `delta_chroma_log2_weight_denom` na informação de índice (Etapa S103).

[00117] A Figura 15 é um fluxograma que ilustra um exemplo do processo de restaurar `chroma_log2_weight_denom` de acordo com a

primeira concretização.

[00118] Primeiro, a unidade de reconfiguração de índice 110B deriva `luma_log2_weight_denom` que já foi ajustado na informação de índice como um valor previsto (Etapa S201).

[00119] Subsequentemente, a unidade de reconfiguração de índice 110B acrescenta `luma_log2_weight_denom` a `delta_chroma_log2_weight_denom` (Etapa S202) e ajusta um valor acrescentado na informação de índice como `chroma_log2_weight_denom` (Etapa S203).

[00120] Em um efeito de desvanecimento, de modo geral, uma vez que há um pequeno número de casos em que mudanças no tempo são feitas diferentemente para cada espaço de cor, a precisão de ponto fixo para cada componente de sinal tem uma forte correlação com um componente de luminância e um componente de diferença. Consequentemente, com uma predição dentro do espaço de cor, conforme descrito acima, pode ser reduzida a quantidade de informação que representa a precisão de ponto fixo.

[00121] Na Expressão Numérica (10), embora o componente de luminância seja subtraído do componente de diferença de cor, o componente de diferença de cor pode ser subtraído do componente de luminância. Em tal caso, a Expressão Numérica (11) pode ser mudada de acordo com a Expressão Numérica (10).

[00122] A seguir, será descrito um método de prever `luma_weight_1x[i]` e `chroma_weight_1x[i][j]` representando os fatores de ponderação da luminância e do sinal de diferença de cor. Aqui, `x` é um identificador representando "0" ou "1". Os valores de `luma_weight_1x[i]` e `chroma_weight_1x[i][j]` aumentam ou diminuem de acordo com os valores de `luma_log2_weight_denom` e `chroma_log2_weight_denom`. Por exemplo, em um caso em que o valor de `luma_log2_weight_denom` é "3", `luma_weight_1x[i]` é $(1 \ll 3)$ em um caso em que nenhuma mudança no brilho é assumida. Por outro lado, em um caso em que o valor de

$\text{luma_log2_weight_denom}$ é "5", $\text{luma_weight_1x}[i]$ é $(1 \ll 5)$ em um caso em que nenhuma mudança no brilho é assumida.

[00123] Conseqüentemente, a unidade de reconfiguração de índice 110B executa um processo de predição com um fator de ponderação de um caso em que não há nenhuma mudança no brilho que é usado como um coeficiente de referência (valor padrão). Mais especificamente, a unidade de reconfiguração de índice 110B executa um processo de predição de $\text{luma_weight_1x}[i]$ usando as Expressões Numéricas (12) e (13) e executa um processo de restauração usando a Expressão Numérica (14). Similarmente, a unidade de reconfiguração de índice 110B executa um processo de predição de $\text{chroma_weight_1x}[i]$ usando as Expressões Numéricas (15) e (16) e executa um processo de restauração usando a Expressão Numérica (17).

$$\text{delta_luma_weight_1x}[i] = (\text{luma_weight_1x}[i] - \text{default_luma_weight_1x}) \quad (12)$$

$$\text{default_luma_weight_1x} = (1 \ll \text{luma_log2_weight_denom}) \quad (13)$$

$$\text{luma_weight_1x}[i] = (\text{default_luma_weight_1x} + \text{delta_luma_weight_1x}[i]) \quad (14)$$

$$\text{delta_chroma_weight_1x}[i][j] = (\text{chroma_weight_1x}[i][j] - \text{default_chroma_weight_1x}) \quad (15)$$

$$\text{default_chroma_weight_1x} = (1 \ll \text{chroma_log2_weight_denom}) \quad (16)$$

$$\text{chroma_weight_1x}[i][j] = (\text{default_chroma_weight_1x} + \text{delta_chroma_weight_1x}[i][j]) \quad (17)$$

[00124] Aqui, $\text{default_luma_weight_1x}$, $\text{default_chroma_weight_1x}$ são valores padrões de um caso em que não há nenhuma mudança de brilho no componente de luminância e no componente de diferença de cor.

[00125] A Figura 16 é um fluxograma que ilustra um exemplo dos processos de prever $\text{luma_weight_1x}[i]$ de acordo com a primeira con-

cretização.

[00126] Primeiro, a unidade de reconfiguração de índice 110B deriva $\text{luma_log2_weight_denom}$ ajustado na informação de índice (Etapa S301) e calcula $\text{default_luma_weight_1x}$ como um valor previsto (Etapa S302).

[00127] Subsequentemente, a unidade de reconfiguração de índice 110B subtrai $\text{default_luma_weight_1x}$ de $\text{luma_weight_1x}[i]$ (Etapa S303) e ajusta um valor de diferença do mesmo na informação de índice como $\text{delta_luma_weight_1x}[i]$ (Etapa S304).

[00128] Com a repetição deste processo em correspondência com as inúmeras imagens de referência, o processo de predição pode ser aplicado a $\text{luma_weight_1x}[i]$.

[00129] A Figura 17 é um fluxograma que ilustra um exemplo do processo de restaurar $\text{luma_weight_1x}[i]$ de acordo com a primeira concretização.

[00130] Primeiro, a unidade de reconfiguração de índice 110B deriva $\text{delta_luma_weight_1x}[i]$ que já foi ajustado na informação de índice (Etapa S401) e calcula $\text{default_luma_weight_1x}$ como um valor previsto (Etapa S402).

[00131] Subsequentemente, a unidade de reconfiguração de índice 110B acrescenta $\text{delta_luma_weight_1x}[i]$ a $\text{default_luma_weight_1x}$ (etapa S403) e ajusta um valor do mesmo na informação de índice como $\text{luma_weight_1x}[i]$ (Etapa S404).

[00132] Enquanto o fluxograma para o componente de luminância foi ilustrado aqui, um processo de predição e um processo de restauração podem ser similarmente realizados para o componente de diferença de cor ($\text{chroma_weight_1x}[i][j]$).

[00133] Uma imagem incluindo um efeito de desvanecimento se desvanece em um ponto de mudança de desvanecimento específico, e há muitos casos em que as outras imagens são imagens naturais co-

muns ou imagens não apresentando nenhum efeito de desvanecimento. Em tal caso, há muitos casos em que o fator de ponderação assume um caso em que não há nenhuma mudança no brilho. Conseqüentemente, um valor inicial de um caso em que não há nenhuma mudança no brilho é derivado com base na precisão de ponto fixo e é usado como um valor previsto, por meio do que a quantidade de código do fator de ponderação pode ser reduzida.

[00134] Além disso, os valores previstos dos fatores de ponderação ($\text{luma_weight_1x}[i]$ e $\text{chroma_weight_1x}[i][j]$) da luminância e do sinal de diferença de cor podem ser derivados com base em outros números de referência ou outros números POC. Em tal caso, quando um número de referência mais próximo à fatia alvo de codificação for base_idx , a unidade de reconfiguração de índice 110B executará um processo de predição de $\text{luma_weight_1x}[i]$ usando a Expressão Numérica (18) e executará um processo de restauração do mesmo usando a Expressão Numérica (19). Similarmente, a unidade de reconfiguração de índice 110B executa um processo de predição de $\text{chroma_weight_1x}[i][j]$ usando a Expressão Numérica (20) e executa um processo de restauração do mesmo usando a Expressão Numérica (21).

$$\text{deta_luma_weight_1x}[i] = (\text{luma_weight_1x}[i] - \text{luma_weight_1x}[\text{base_idx}]) \quad (18)$$

$$\text{luma_weight_1x}[i] = (\text{deta_luma_weight_1x}[i] + \text{luma_weight_1x}[\text{base_idx}]) \quad (19)$$

$$\text{deta_chroma_weight_1x}[i][j] = (\text{chroma_weight_1x}[i][j] - \text{chroma_weight_1x}[\text{base_idx}][j]) \quad (20)$$

$$\text{chroma_weight_1x}[i][j] = (\text{deta_chroma_weight_1x}[i][j] + \text{chroma_weight_1x}[\text{base_idx}][j]) \quad (21)$$

[00135] Aqui, nas Expressões Numéricas (18) e (20), $i \neq \text{base_idx}$. Para o fator de ponderação do número de referência representado por

base_idx, as Expressões Numéricas (18) e (20) não podem ser usadas, podendo, conseqüentemente, ser usadas as Expressões Numéricas (12), (13), (15) e (16).

[00136] A Figura 18 é um fluxograma que ilustra outro exemplo do processo de predição de $luma_weight_1x[i]$ de acordo com a primeira concretização.

[00137] Primeiro, a unidade de reconfiguração de índice 110B ajusta o baseidx representando um número de referência que é uma referência (Etapa S501). Aqui, o valor de baseidx é provisoriamente assumido como sendo "0".

[00138] Subseqüentemente, a unidade de reconfiguração de índice 110B deriva $luma_weight_1x[baseidx]$ da informação de índice como um valor previsto com base em baseidx (Etapa S502). Além disso, $luma_weight_1x[baseidx]$ da informação de índice representada por baseidx, por exemplo, não é previsto, mas é codificado como um valor direto.

[00139] Subseqüentemente, a unidade de reconfiguração de índice 110B subtrai $luma_weight_1x[baseidx]$ de $luma_weight_1x[i]$ (Etapa S503) e ajusta um valor de diferença do mesmo como $\delta_luma_weight_1x[i]$ na informação de índice (Etapa S504).

[00140] Com a repetição deste processo em correspondência com o número de imagens de referência, o processo de predição pode ser aplicado a $luma_weight_1x[i]$ do que baseidx.

[00141] A Figura 19 é um fluxograma que ilustra outro exemplo do processo de restaurar $luma_weight_1x[i]$ de acordo com a primeira concretização.

[00142] Primeiro, a unidade de reconfiguração de índice 110B ajusta o baseidx representando um número de referência que é uma referência (Etapa S601). Aqui, o valor de baseidx é provisoriamente assumido como sendo "0".

[00143] Subsequentemente, a unidade de reconfiguração de índice 110B deriva $\text{luma_weight_1x}[\text{baseidx}]$ da informação de índice como um valor previsto com base em baseidx (Etapa S602). Além disso, $\text{luma_weight_1x}[\text{baseidx}]$ da informação de índice representada por baseidx , por exemplo, não é previsto, mas é codificado como um valor direto.

[00144] Subsequentemente, a unidade de reconfiguração de índice 110B acrescenta $\text{delta_luma_weight_1x}[i]$ a $\text{luma_weight_1}[\text{baseidx}]$ (Etapa S603) e ajusta um valor acrescentado do mesmo como $\text{luma_weight_1x}[i]$ na informação de índice (Etapa S604).

[00145] Enquanto o fluxograma para o componente de luminância foi ilustrado aqui, um processo de predição e um processo de restauração poderão ser similarmente realizados para o componente de diferença de cor ($\text{chroma_weight_1x}[i][j]$). Além disso, enquanto o método de predição e o método de restauração de $\text{luma_weight_1x}[i]$ foram descritos como um exemplo, $\text{luma_offset_1x}[i]$ pode ser similarmente previsto e restaurado.

[00146] Além disso, os valores previstos dos fatores de ponderação ($\text{luma_weight_1x}[i]$ e $\text{chroma_weight_1x}[i][j]$) da luminância e do sinal de diferença de cor podem ser derivados usando uma distância entre o alvo de codificação e a fatia de referência. Em tal caso, a unidade de reconfiguração 110B executa um processo de predição de $\text{luma_weight_1x}[i]$ usando a Expressão Numérica (22) e executa um processo de restauração do mesmo usando a Expressão Numérica (23). Similarmente, a unidade de reconfiguração de índice 110B executa um processo de predição de $\text{chroma_weight_1x}[i][j]$ usando a Expressão Numérica (24) e executa um processo de restauração do mesmo usando a Expressão Numérica (25).

$$\text{delta_luma_weight_1x}[i] = (\text{luma_weight_1x}[i] - \text{luma_weight_1x}[i - 1])$$

(22)

$$\text{luma_weight_1x}[i] = (\text{delta_luma_weight_1x}[i] + \text{luma_weight_1x}[i - 1]) \quad (23)$$

$$\text{delta_chroma_weight_1x}[i][j] = (\text{chroma_weight_1x}[i][j] - \text{chroma_weight_1x}[i - 1][j]) \quad (24)$$

$$\text{chroma_weight_1x}[i][j] = (\text{delta_chroma_weight_1x}[i][j] + \text{chroma_weight_1x}[i - 1][j]) \quad (25)$$

[00147] Aqui, nas Expressões Numéricas (22) e (24), $i \neq 0$.

[00148] Além disso, uma vez que estes processos de predição e de restauração são iguais àqueles do fluxograma ilustrado nas Figuras 18 e 19 com a introdução do $(i - 1)$ ésimo- valor ($i \neq 0$) em `baseidx`, a descrição dos mesmos não será apresentada. Enquanto o fluxograma para o componente de luminância foi representado aqui, o processo de predição e o processo de restauração podem ser similarmente realizados para o componente de diferença de cor (`chroma_weight_1x[i][j]`). Adicionalmente, enquanto o método de predição e o método de restauração de `luma_weight_1x[i]` foram descritos como um exemplo, `luma_offset_1x[i]` também poderá ser similarmente previsto e restaurado.

[00149] Há muitos casos em que, como uma fatia de referência que pode ser referida como fatia alvo de codificação, uma fatia que está próxima à fatia alvo de codificação em termos de uma distância no tempo ou espaço é ajustada a partir do ponto de vista da eficiência de codificação. Aqui, uma vez que mudanças de luminância de fatias que são contínuas em distância no tempo apresentam uma alta correlação, as correlações referentes a uma distância no tempo entre os fatores de ponderação e os deslocamentos são também elevadas. Desse modo, com o uso do fator de ponderação e do valor de deslocamento da fatia de referência que serve como uma referência, são previstos um fator de ponderação e um valor de deslocamento de uma fatia de referência que é diferente dos mesmos no tempo, por meio do que a quantidade de código pode ser eficientemente reduzida. Além disso, uma

vez que há muitos casos em que fatias de referência que são iguais no espaço assumem fatores de ponderação e valores de deslocamento que são iguais, com a introdução de uma predição pela mesma razão, a quantidade de código poderá ser reduzida.

[00150] A seguir, será descrito um método de predição de $\text{chroma_offset_1x}[i][j]$ que representa um deslocamento do sinal de diferença de cor. No espaço de cor de YUV, um componente de diferença de cor representa uma cor usando a quantidade de um desvio de um valor médio. Consequentemente, a quantidade de mudança com base em uma mudança no brilho com o valor médio considerado pode ser ajustada como um valor previsto usando um fator de ponderação. Mais especificamente, a unidade de reconfiguração de índice 110B executa um processo de predição de $\text{chroma_offset_1x}[i][j]$ usando as Expressões Numéricas (26) e (27) e executa um processo de restauração usando a Expressão Numérica (28).

$$\text{delta_chroma_offset_1x}[i][j] = (\text{chroma_offset_1x}[i][j] + ((\text{MED} * \text{chroma_weight_1x}[i][j]) \gg \text{chroma_log2_weight_denom}) - \text{MED}) \quad (26)$$

$$\text{MED} = (\text{MaxChromaValue} \gg 1) \quad (27)$$

[00151] Aqui, MaxChromaValue representa brilho máximo no qual é obtido um sinal de diferença de cor. Por exemplo, no caso de um sinal de 8 bits, MaxChromaValue é 255, e MED é 128.

$$\text{chroma_offset_1x}[i][j] = (\text{delta_chroma_offset_1x}[i][j] - ((\text{MED} * \text{chroma_weight_1x}[i][j]) \gg \text{chroma_log2_weight_denom}) + \text{MED}) \quad (28)$$

[00152] A Figura 20 é um fluxograma que ilustra um exemplo do processo de predição de $\text{chroma_offset_1x}[i][j]$ de acordo com a primeira concretização.

[00153] Primeiro, a unidade de reconfiguração de índice 110B deriva $\text{chroma_log2_weight_denom}$ ajustado na informação de índice (Etapa S701).

[00154] Subsequentemente, a unidade de reconfiguração de índice

110B deriva $\text{chroma_offset_1x[i][j]}$ ajustado na informação de índice (Etapa S702).

[00155] Subsequentemente, a unidade de reconfiguração de índice 110B deriva um valor médio de valores máximos (sinais máximos) de sinais de diferença de cor (Etapa S703).

[00156] Subsequentemente, a unidade de reconfiguração de índice 110B deriva $\text{delta_chroma_offset_1x[i][j]}$ e ajusta $\text{delta_chroma_offset_1x[i][j]}$ na informação de índice (Etapa S704).

[00157] A Figura 21 é um fluxograma que ilustra um exemplo de um processo de restauração de $\text{chroma_offset_1x[i][j]}$ de acordo com a primeira concretização.

[00158] Primeiro, a unidade de reconfiguração de índice 110B deriva $\text{chroma_log2_weight_denom}$ que já foi ajustado na informação de índice (Etapa S801).

[00159] Subsequentemente, a unidade de reconfiguração de índice 110B deriva $\text{chroma_offset_1x[i][j]}$ ajustado na informação de índice (Etapa S802).

[00160] Subsequentemente, a unidade de reconfiguração de índice 110B deriva um valor médio de valores máximos (sinais máximos) de sinais de diferença de cor (Etapa S803).

[00161] Subsequentemente, a unidade de reconfiguração de índice 110B deriva $\text{chroma_offset_1x[i][j]}$ e ajusta $\text{chroma_offset_1x[i][j]}$ na informação de índice (Etapa S804).

[00162] Com a introdução de um valor previsto adquirido ao se considerar as características de sinal de um desvio do valor médio usando as características de sinal do sinal de diferença de cor, a quantidade de código do valor de deslocamento do sinal de diferença de cor pode ser menor do que aquela de um caso em que o valor de deslocamento é diretamente codificado.

[00163] A seguir, será descrita uma técnica de derivar valores pre-

vistos do fator de ponderação e da precisão de ponto fixo usando o método de derivar parâmetros WP de uma predição ponderada implícita em uma predição ponderada definida em H-264 ou semelhante. Na predição ponderada implícita de H.264, um fator de ponderação é derivado de acordo com uma distância (uma relação de tempo do número POC) no tempo entre fatias de referência (o deslocamento se torna zero). A distância no tempo entre as fatias de referência é adquirida por derivar distâncias entre uma fatia alvo de codificação e fatias de referência com base nos números POC, e o fator de ponderação é determinado com base na relação das distâncias. Neste momento, a precisão de ponto fixo é ajustada como um valor fixo de "5".

[00164] Por exemplo, em H.265, o fator de ponderação é derivado de acordo com um pseudocódigo representado na Expressão Numérica (29).

$$td = \text{Clip3}(-128, 127, POCA - POCB)$$

$$tb = \text{Clip3}(-128, 127, POCT - POCA)$$

$$tx = (td \neq 0) ? ((16384 + \text{abs}(td/2))/td) : (0)$$

$$\text{DistScaleFactor} = \text{Clip3}(-1024, 1023, (tb * tx + 32) \gg 6)$$

$$\text{implicit_luma_weight_10}[i] = 64 (\text{DistScaleFactor} \gg 2)$$

$$\text{implicit_luma_weight_11}[i] = \text{DistScaleFactor} \gg 2 \quad (29)$$

[00165] Aqui, POCA representa um número POC de uma imagem de referência A correspondendo à Lista 1, POCB representa um número POC de uma imagem de referência B correspondendo à Lista 0, e POCT representa um número POC de uma imagem alvo de predição. Além disso, $\text{Clip3}(L, M, N)$ é uma função para executar um processo de recorte de tal modo que um último argumento N não exceda uma faixa de um valor mínimo L e um valor máximo M representados pelos primeiros dois argumentos. Uma função $\text{abs}()$ é uma função para retornar um valor absoluto de um argumento. Além disso, td e tb representam relações de tempo, td representa uma diferença entre um nú-

mero POC de uma imagem de referência correspondendo à Lista 1 e um número POC de uma imagem de referência correspondendo à lista 0, e tb representa uma diferença entre um número POC da imagem alvo de predição e um número POC da imagem de referência correspondendo à Lista 0. Com base em tais valores, é derivada uma variável de escala DistScaleFactor na distância do fator de ponderação. Com base em DistScaleFactor, são derivados os fatores de ponderação (implicit_luma_weight_10[i] e implicit_luma_weight_11[i]) correspondendo às Listas 0 e 1. Além disso, o sinal de diferença de cor é similarmente ajustado. A unidade de reconfiguração de índice 110B prevê a precisão de ponto fixo com base na Expressão Numérica (30) com o uso da precisão de ponto fixo implicit_log2_weight_denom derivado aqui.

$$\text{delta_luma_log2_weight_denom} = (\text{luma_log2_weight_denom} - \text{implicit_log2_weight_denom}) \quad (30)$$

[00166] Além disso, a precisão de ponto fixo do sinal de diferença de cor pode ser prevista usando a Expressão Numérica (30). Este valor é restaurado com o uso da Expressão Numérica (31).

$$\text{luma_log2_weight_denom} = (\text{delta_luma_log2_weight_denom} + \text{implicit_log2_weight_denom}) \quad (31)$$

[00167] Além disso, a precisão de ponto fixo do sinal de diferença de cor pode ser restaurada usando o mesmo método representado na Expressão Numérica (31).

[00168] A seguir, será descrita uma equação para prever o fator de ponderação. Quando um fator de ponderação implícito for implicit_luma_weight_1x[i], a unidade de reconfiguração de índice 110B irá prever um fator de ponderação luma_weight_1x[i] usando a Expressão Numérica (32) e restaurar o fator de ponderação usando a Expressão Numérica (33).

$$\text{se}(\text{luma_log2_weight_denom} > \text{implicit_log2_weight_denom})$$

```

[norm_denom - (luma_log2_weight_denom - implicit_log2_weight_
denom)
delta_luma_weight_1x[i] = (luma_weight_1x[i] - (implic-
it_luma_weight_1x[i] << norm_denom))
)
exceto(
norm_denom_ (implicit_log2_weight_denom -
luma_log2_weight_denom)
delta_luma_weight_1x[i] = (luma_weight_1x[i] -
(implicit_luma_weight_1x[i] >> norm_demon))
)
)
(32)

```

[00169] Aqui, a unidade de reconfiguração de índice 110B corrige o fator de ponderação com base em se a predição ponderada implícita é maior ou menor do que a precisão de ponto fixo e usa o fator de ponderação corrigido para a predição.

```

se(luma)log2_weight_denom >= implicit_log2_weight_
denom)(norm_denom = (luma_log2_weight_denom -
implicit_log2_weight_denom)
luma_weight_1x[i] - (delta_luma_weight_1x[i] +
(implicit_luma_weight_1x[i] << norm_denom))
)
exceto(
norm_denom = (implicit_log2_weight_denom -
luma_log2_weight_denom)
luma_weight_1x[i] = (delta_luma_weight_1x[i] +
(implifict_luma_weight_1x[i] >> norm_denom))
)

```

[00170] Na Expressão Numérica (32), enquanto um exemplo do fator de ponderação do componente de luminância é representado, com o uso do mesmo método para o componente de diferença de cor, um valor previsto pode ser derivado.

[00171] A Figura 22 é um fluxograma que ilustra outro exemplo de um processo de predição de $\text{luma_weight_1x}[i]$ de acordo com a primeira concretização.

[00172] Primeiro, a unidade de reconfiguração de índice 110B deriva $\text{luma_log2_weight_denom}$ ajustado na informação de índice (Etapa S901).

[00173] Subsequentemente, a unidade de reconfiguração de índice 110B deriva $\text{implicit_log2_weight_denom}$ e $\text{implicit_luma_weight_1x}[i]$ de acordo com o método de derivar uma predição ponderada implícita de H.264 (Etapas S902 e S903).

[00174] Subsequentemente, a unidade de reconfiguração de índice 110B determina se $\text{luma_log2_weight_denom}$ é $\text{implicit_log2_weight_denom}$ ou mais (Etapa S904).

[00175] Em um caso em que $\text{luma_log2_weight_denom}$ é $\text{implicit_log2_weight_denom}$ ou mais (Sim, na Etapa S904), a unidade de reconfiguração de índice 110B subtrai $\text{implicit_log2_weight_denom}$ de $\text{luma_log2_weight_denom}$ (Etapa S905) e muda $\text{implicit_luma_weight_1x}[i]$ para o lado esquerdo em uma quantidade correspondendo ao valor subtraído, derivando assim um valor previsto (Etapa S906).

[00176] Por outro lado, em um caso em que $\text{luma_log2_weight_denom}$ não é $\text{implicit_log2_weight_denom}$ ou mais (Não, na Etapa S904), a unidade de reconfiguração 110B subtrai $\text{luma_log2_weight_denom}$ de $\text{implicit_log2_weight_denom}$ (Etapa S907) e muda $\text{implicit_luma_weight_1x}[i]$ para o lado direito em uma quantidade correspondendo ao valor subtraído, derivando assim um valor previsto (Etapa S908).

[00177] Subsequentemente, a unidade de reconfiguração de índice 110B subtrai o valor previsto derivado de $\text{luma_weight_1x}[i]$ (Etapa S909) e ajusta o valor subtraído (valor de diferença) na informação de índice (Etapa S910).

[00178] A Figura 23 é um fluxograma que ilustra outro exemplo do processo de restauração de $\text{luma_weight_1x}[i]$ de acordo com a primeira concretização.

[00179] Primeiro, a unidade de reconfiguração de índice 110B deriva $\text{luma_log2_weight_denom}$ que já foi ajustado na informação de índice (Etapa S1001).

[00180] Subsequentemente, a unidade de reconfiguração de índice 110B deriva $\text{implicit_log2_weight_denom}$ e $\text{implicit_luma_weight_1x}[i]$ de acordo com o método de derivar uma predição ponderada implícita de H.264 (Etapas S1002 e S1003).

[00181] Subsequentemente, a unidade de reconfiguração de índice 110B determina se $\text{luma_log2_weight_denom}$ é $\text{implicit_log2_weight_denom}$ ou mais (Etapa S1004).

[00182] Em um caso em que $\text{luma_log2_weight_denom}$ é $\text{implicit_log2_weight_denom}$ ou mais (Sim, na Etapa S1004), a unidade de reconfiguração de índice 110B subtrairá $\text{implicit_log2_weight_denom}$ de $\text{luma_log2_weight_denom}$ (Etapa S1005) e muda $\text{implicit_luma_weight_1x}[i]$ para o lado esquerdo em uma quantidade correspondendo ao valor subtraído, derivando assim um valor previsto (Etapa S1006).

[00183] Por outro lado, em um caso em que $\text{luma_log2_weight_denom}$ não é $\text{implicit_log2_weight_denom}$ ou mais (Não, na Etapa S1004), a unidade de reconfiguração de índice 110B subtrai $\text{luma_log2_weight_denom}$ de $\text{implicit_log2_weight_denom}$ (Etapa S1007) e muda $\text{implicit_luma_weight_1x}[i]$ para o lado direito em uma quantidade correspondendo ao valor subtraído, derivando assim um valor previsto (Etapa S1008).

[00184] Subsequentemente, a unidade de reconfiguração de índice 110B acrescenta o valor previsto derivado a $\text{delta_luma_weight_1x}[i]$ (Etapa S1009) e ajusta o valor acrescentado do mesmo na informação

de índice (Etapa S1010).

[00185] Uma pluralidade de métodos de predição descritos acima pode não apenas ser usada independentemente, mas também ser usada de maneira combinada. Por exemplo, com a combinação das Expressões Numéricas (10), (12) e (13), (15) e (16), e (26) e (27) ou semelhante, a quantidade de código do elemento de sintaxe da informação de índice pode ser eficientemente reduzida.

[00186] Como acima, de acordo com a primeira concretização, a unidade de ajuste de índice 108 emite a informação de índice na qual a informação de parâmetro WP é mapeada em uma configuração de sintaxe correspondente, e a unidade de reconfiguração de índice 110B prevê uma representação redundante do elemento de sintaxe com base na informação codificada dentro da fatia. Por isso, de acordo com a primeira concretização, a quantidade de código pode ser menor do que aquela de um caso em que o elemento de sintaxe é diretamente codificado (valor direto).

[00187] Aqui, com base na ordem de definição (ordem de codificação) dos elementos de sintaxe usados na fatia alvo de codificação, com a derivação de um valor previsto como uma correlação entre telas de um elemento de sintaxe que foi completado para ser codificado ou com a derivação de um valor previsto de um valor padrão adquirido em assumindo nenhuma mudança no brilho, pode ser feita uma predição que tira vantagem das características dos elementos de sintaxe. Como resultado, é adquirida uma vantagem de reduzir o processamento ou armazenamento em excesso que é necessário para codificar o elemento de sintaxe.

[00188] Além disso, entre fileiras da tabela de sintaxe ilustrada nas Figuras de 10 a 13 de acordo com a primeira concretização como exemplos, um elemento de sintaxe não definido nesta concretização pode ser inserido, ou uma descrição relacionada à outra ramificação

condicional pode ser incluída. Adicionalmente, a tabela de sintaxe pode ser dividida em uma pluralidade de tabelas, ou uma pluralidade de tabelas de sintaxe pode ser integrada. Além disso, o termo de cada elemento de sintaxe representado como um exemplo pode ser arbitrariamente modificado.

[00189] Conforme descrito acima, o dispositivo de codificação 100 de acordo com a primeira concretização soluciona um problema de diminuir a eficiência de codificação com a eliminação da redundância espacial usando as correlações entre parâmetros de informação a serem codificados. O dispositivo de codificação 100 pode reduzir a quantidade de código comparada àquela de uma configuração convencional na qual os elementos de sintaxe usados na predição ponderada compensada por movimento são diretamente codificados (valores diretos).

Segunda Concretização

[00190] Em uma segunda concretização, será descrito um dispositivo de decodificação que decodifica os dados codificados pelo dispositivo de codificação de acordo com a primeira concretização.

[00191] A Figura 24 é um diagrama de bloco que ilustra um exemplo da configuração de um dispositivo de decodificação 800 de acordo com a segunda concretização.

[00192] O dispositivo de decodificação 800 decodifica os dados codificados armazenados em um buffer de entrada não ilustrado na figura ou semelhante em uma imagem decodificada e emite a imagem decodificada para um buffer de saída não ilustrado na figura como uma imagem de saída. Os dados codificados, por exemplo, são emitidos do dispositivo de codificação 100 ilustrado na Figura 1 ou semelhante e são introduzidos no dispositivo de decodificação 800 através de um sistema de armazenamento, um sistema de transmissão, um buffer, ou semelhante não ilustrado na figura.

[00193] O dispositivo de decodificação 800, conforme ilustrado na Figura 24 inclui uma unidade de decodificação 801, uma unidade de quantização inversa 802, uma unidade de transformação ortogonal inversa 803, uma unidade de adição 804, uma unidade de geração de imagem prevista 805, e uma unidade de ajuste de índice 806. A unidade de quantização inversa 802, a unidade de transformação ortogonal inversa 803, a unidade de adição 804, e a unidade de geração de imagem prevista 805 são elementos que são substancialmente iguais ou similares à unidade de quantização inversa 104, à unidade de transformação ortogonal inversa 105, à unidade de adição 106, e à unidade de geração de imagem prevista 107 ilustradas na Figura 1. Além disso, uma unidade de controle de decodificação 807 ilustrada na Figura 24 controla o dispositivo de decodificação 800 e, por exemplo, é realizada por uma CPU ou semelhante.

[00194] A fim de decodificar os dados codificados, a unidade de decodificação 801 executa a decodificação com base na sintaxe para cada quadro ou cada campo. A unidade de decodificação 801 inclui uma unidade de decodificação entrópica 801A e uma unidade de reconfiguração de índice 801B.

[00195] A unidade de decodificação entrópica 810A sequencialmente executa a decodificação entrópica de uma cadeia de códigos de cada sintaxe e regenera a informação de movimento incluindo um modo de predição, um vetor de movimento, e um número de referência, informação de índice usada para prever uma predição ponderada compensada por movimento, e parâmetros de codificação de um bloco alvo de codificação, tal como um coeficiente de transformação de quantização ou semelhante. Aqui, os parâmetros de codificação são todos os parâmetros que são necessários para decodificar a informação referente a um coeficiente de transformação, informação referente à quantização, e semelhante além daqueles descritos acima.

[00196] Mais especificamente, a unidade de decodificação entrópica 801A tem uma função para executar um processo de decodificação, tal como um processo de decodificação de comprimento variável ou um processo de decodificação aritmética para dados codificados de entrada. Por exemplo, em H.264, é usada a codificação de comprimento variável adaptativa com base em contexto (CSVLC), a codificação aritmética binária adaptativa com base em contexto (CABAC), ou semelhante. Tal processo é também denominado de processo de decodificação.

[00197] A unidade de reajuste de índice 810B reconfigura a informação de índice ao restaurar a informação de índice decodificada. Mais especificamente, a fim de reduzir o comprimento de código dos elementos de sintaxe da informação de índice decodificada, a unidade de reconfiguração de índice 801B executa um processo de predição de acordo com as características de parâmetros de elementos de sintaxe, restaura os elementos de sintaxe, e reconfigura a informação de índice. Um exemplo específico do processo de predição será descrito mais adiante.

[00198] A unidade de decodificação 801 emite a informação de movimento, a informação de índice, e o coeficiente de transformação de quantização, de modo a introduzir o coeficiente de transformação de quantização na unidade de quantização inversa 802, a introduzir a informação de índice na unidade de ajuste de índice 806, e a introduzir a informação de movimento na unidade de geração de imagem prevista 805.

[00199] A unidade de quantização inversa 802 executa um processo de quantização inversa para o coeficiente de transformação de quantização introduzido da unidade de decodificação 801 e adquire um coeficiente de transformação de restauração. Mais especificamente, a unidade de quantização inversa 802 executa a quantização inver-

sa com base na informação de quantização usada pela unidade de decodificação 801. Descrita em maiores detalhes, a unidade de quantização inversa 802 multiplica o coeficiente de transformação de quantização por um tamanho de etapa de quantização derivado com base na informação de quantização, adquirindo assim um coeficiente de transformação restaurado. A unidade de quantização inversa 802 emite o coeficiente de transformação restaurado de modo a ser introduzido na unidade de transformação ortogonal inversa 803.

[00200] A unidade de transformação ortogonal inversa 803 executa a transformação ortogonal inversa correspondendo à transformação ortogonal executada no lado de codificação para o coeficiente de transformação restaurado introduzido da unidade de quantização inversa 802, adquirindo assim um erro de predição restaurado. A unidade de transformação ortogonal inversa 803 emite o erro de predição restaurado de modo a ser introduzido na unidade de adição 804.

[00201] A unidade de adição 804 acrescenta o erro de predição restaurado introduzido da unidade de transformação ortogonal inversa 803 e uma imagem prevista correspondente, gerando assim uma imagem decodificada. A unidade de adição 804 emite a imagem decodificada de modo a ser introduzida na unidade de geração de imagem prevista 805. Além disso, a unidade de adição 804 emite a imagem decodificada para o lado de fora como uma imagem de saída. Em seguida, a imagem de saída é temporariamente armazenada em um buffer de saída externo não ilustrado na figura ou semelhante e é emitida para um sistema de dispositivo de visualização, tal como uma tela ou um monitor, não ilustrado na figura ou um sistema de dispositivo de vídeo, por exemplo, no tempo de saída gerenciado pela unidade de controle de decodificação 807.

[00202] A unidade de ajuste de índice 806 recebe a informação de índice introduzida da unidade de decodificação 801, converte a infor-

mação de índice em informação de parâmetro WP, e emite a informação de parâmetro WP de modo a ser introduzida na unidade de geração de imagem prevista 805. Mais especificamente, a unidade de ajuste de índice 806 recebe a informação de índice que foi processada para ser decodificada pela unidade de decodificação entrópica 801A e é reconfigurada pela unidade de reconfiguração de índice 801B. Depois, a unidade de ajuste de índice 806 verifica uma lista de imagens de referência e um número de referência, converte a informação de índice em informação de parâmetro WP, e emite a informação de parâmetro WP para a unidade de geração de imagem prevista 805. A informação de parâmetro WP já foi descrita com referência às Figuras 8A e 8B, e, portanto, a descrição da mesma não será apresentada.

[00203] A unidade de geração de imagem prevista 805 gera uma imagem prevista 815 com o uso da informação de movimento introduzida da unidade de decodificação 801, da informação de parâmetro WP introduzida da unidade de ajuste de índice 806, e da imagem decodificada introduzida da unidade de adição 804.

[00204] Aqui, a unidade de geração de imagem prevista 805 será descrita em detalhes com referência à Figura 4. A unidade de geração de imagem prevista 805, similarmente à unidade de geração de imagem prevista 107, inclui uma unidade de compensação de movimento de múltiplos quadros 201, uma memória 292, uma unidade de compensação de movimento unidirecional 203, uma unidade de controle de parâmetro de predição 204, um seletor de imagem de referência 205, uma memória de quadro 206, e uma unidade de controle de imagem de referência 207.

[00205] A memória de quadro 206 armazena a imagem decodificada introduzida da unidade de adição 106 como uma imagem de referência sob o controle da unidade de controle de imagem de referência 207. A memória de quadro 206 inclui uma pluralidade de conjuntos de

memória FM1 a FMN (aqui, $N \geq 2$) usada para temporariamente armazenar a imagem de referência.

[00206] A unidade de controle de parâmetro de predição 204 prepara uma pluralidade de combinações, cada qual de um número de imagem de referência e um parâmetro de predição como uma tabela com base na informação de movimento introduzida da unidade de decodificação 801. Aqui, a informação de movimento representa a informação de um vetor de movimento que representa o desvio de um movimento que é usado para a predição compensada por movimento, o número de imagem de referência, e um modo de predição, tal como uma predição unidirecional/bidirecional. O parâmetro de predição representa informação referente ao vetor de movimento e ao modo de predição. Depois, a unidade de controle de parâmetro de predição 204 seleciona uma combinação de número de imagem de referência e parâmetro de predição usados para gerar uma imagem prevista com base na informação de movimento e emite a combinação selecionada de modo a permitir que o número de imagem de referência seja introduzido no seletor de imagem de referência 205 e a permitir que o parâmetro de predição seja introduzido na unidade de compensação de movimento unidirecional 203.

[00207] O seletor de imagem de referência 205 é um comutador que muda um dos terminais de saída das memórias de quadro FM1 a FMN, que são incluídas na memória de quadro 206, a ser conectado com base em um número de imagem de referência introduzido da unidade de controle de parâmetro de predição 204. Por exemplo, quando o número de imagem de referência for "0", o seletor de imagem de referência 205 conectará o terminal de saída da memória de quadro FM1 ao terminal de saída do seletor de imagem de referência 205, e, quando o número de imagem de referência for N-1, o seletor de imagem de referência 205 conectará o terminal de saída da memória de quadro

FMN ao terminal de saída do seletor de imagem de referência 205. O seletor de imagem de referência 205 emite uma imagem de referência armazenada na memória de quadro à qual o terminal de saída é conectado dentre as memórias de quadro FM1 a FMN incluída na memória de quadro 206, de modo a ser introduzido na unidade de compensação de movimento unidirecional 203. No dispositivo de decodificação 800, a imagem de referência não é usada por nenhuma unidade que não a unidade de geração de imagem prevista 805, e, consequentemente, a imagem de referência não pode ser emitida para fora da unidade de geração de imagem prevista 805.

[00208] A unidade de compensação de movimento previsto unidirecional executa um processo de predição compensada por movimento com base no parâmetro de predição introduzido da unidade de controle de parâmetro de predição 204 e a imagem de referência introduzida do seletor de imagem de referência 205, gerando assim uma imagem prevista unidirecional. A predição compensada por movimento já foi descrita com referência à Figura 5, e, portanto, a descrição da mesma não será apresentada.

[00209] A unidade de compensação de movimento previsto unidirecional 203 emite uma imagem prevista unidirecional e temporariamente armazena a imagem prevista unidirecional na memória 202. Aqui, em um caso em que a informação de movimento (parâmetro de predição) representa uma predição bidirecional, a unidade de compensação de movimento de múltiplos quadros 201 faz uma predição ponderada usando dois tipos de imagens previstas unidirecionais. Consequentemente, a unidade de compensação de movimento previsto unidirecional 203 armazena uma imagem prevista unidirecional correspondendo ao primeiro tipo na memória 202 e diretamente emite uma imagem prevista unidirecional correspondendo ao segundo tipo para a unidade de compensação de movimento de múltiplos quadros 201. Aqui, a

imagem prevista unidirecional correspondendo ao primeiro tipo será referida como uma primeira imagem prevista, e a imagem prevista unidirecional correspondendo ao segundo tipo será referida como uma segunda imagem prevista.

[00210] Além disso, duas unidades de compensação de movimento unidirecional 203 podem ser preparadas e gerar duas imagens previstas unidirecionais. Em tal caso, quando a informação de movimento (parâmetro de predição) representa uma predição unidirecional, a unidade de compensação de movimento unidirecional 203 pode diretamente emitir a primeira imagem prevista unidirecional para a unidade de compensação de movimento de múltiplos quadros 201 como uma primeira imagem prevista.

[00211] A unidade de compensação de movimento de múltiplos quadros 201 faz uma predição ponderada ao usar a primeira imagem prevista introduzida da memória 202, a segunda imagem prevista introduzida da unidade de compensação de movimento previsto unidirecional 203, e a informação de parâmetro WP introduzida da unidade de avaliação de movimento 109, gerando assim uma imagem prevista. A unidade de compensação de movimento de múltiplos quadros 201 emite a imagem prevista de modo a ser introduzida na unidade de adição 804.

[00212] Aqui, a unidade de compensação de movimento de múltiplos quadros 201 será descrita em detalhes com referência à Figura 6. Similarmente à unidade de geração de imagem prevista 107, a unidade de compensação de movimento de múltiplos quadros 201 inclui uma unidade de compensação de movimento padrão 301, uma unidade de compensação ponderada de movimento 302, uma unidade de controle de parâmetro WP 303, e seletores WP 304 e 305.

[00213] A unidade de controle de parâmetro WP 303 emite um sinalizador de aplicativo WP e informação de ponderação com base na in-

formação de parâmetro WP introduzida da unidade de ajuste de índice 806 de modo a introduzir o sinalizador de aplicativo WP nos seletores WP 304 e 305 e a introduzir a informação de ponderação na unidade de compensação ponderada de movimento 302.

[00214] Aqui, a informação de parâmetro WP inclui informação da precisão de ponto fixo do fator de ponderação, um primeiro sinalizador de aplicativo WP, um primeiro fator de ponderação, e um primeiro deslocamento correspondendo à primeira imagem prevista, e um segundo sinalizador de aplicativo WP, um segundo fator de ponderação, e um segundo deslocamento correspondendo à segunda imagem prevista. O sinalizador de aplicativo WP é um parâmetro que pode ser ajustado para cada imagem de referência correspondente e componente de sinal e representa se uma predição de compensação ponderada de movimento é feita ou não. A informação de ponderação inclui informação da precisão de ponto fixo do fator de ponderação, do primeiro fator de ponderação, do primeiro deslocamento, do segundo fator de ponderação, e do segundo deslocamento. Aqui, a informação de parâmetro WP representa a mesma informação que aquela da primeira concretização.

[00215] Descrita em detalhes, quando a informação de parâmetro WP for introduzida da unidade de ajuste de índice 806, a unidade de controle de parâmetro WP 303 emitirá a informação de parâmetro WP em sendo dividida no primeiro sinalizador de aplicativo WP, no segundo sinalizador de aplicativo WP, e na informação de ponderação, introduzindo assim o primeiro sinalizador de aplicativo WP no seletor WP 304, introduzindo o segundo sinalizador de aplicativo WP no seletor WP 305, e introduzindo a informação de ponderação na unidade de compensação ponderada de movimento 302.

[00216] Os seletores WP 304 e 305 mudam as extremidades de conexão das imagens previstas com base nos sinalizadores de aplica-

tivo WP introduzidos da unidade de controle de parâmetro WP 303. Em um caso em que o sinalizador de aplicativo WP correspondente é "0", cada dos seletores WP 304 e 305 conecta a extremidade de saída do mesmo à unidade de compensação de movimento padrão 301. Depois, os seletores WP 304 e 305 emitem a primeira e a segunda imagens previstas de modo a serem introduzidas na unidade de compensação de movimento padrão 301. Por outro lado, em um caso em que o sinalizador de aplicativo WP correspondente é "1", cada dos seletores WP 304 e 205 conecta a extremidade de saída do mesmo à unidade de compensação ponderada de movimento 302. Depois, os seletores WP 304 e 305 emitem a primeira e a segunda imagens previstas de modo a serem introduzidas na unidade de compensação ponderada de movimento 302.

[00217] A unidade de compensação de movimento padrão 301 executa o processamento médio com base nas imagens previstas unidirecionais (a primeira e a segunda imagens previstas) introduzidas dos seletores WP 304 e 305, gerando assim uma imagem prevista. Mais especificamente, em um caso em que o primeiro e o segundo sinalizadores de aplicativo WP são "0"s, a unidade de compensação de movimento padrão 301 executa o processamento médio com base na Expressão Numérica (1).

[00218] Além disso, em um caso em que o modo de predição representado pela informação de movimento (parâmetro de predição) é a predição unidirecional, a unidade de compensação de movimento padrão 301 calcula uma imagem prevista final usando apenas a primeira imagem prevista com base na Expressão Numérica (4).

[00219] A unidade de compensação ponderada de movimento 302 executa a compensação ponderada de movimento com base nas duas imagens previstas unidirecionais (a primeira e a segunda imagens previstas) introduzidas dos seletores WP 304 e 305 e a informação pon-

derada introduzida da unidade de controle de parâmetro WP 303. Mais especificamente, a unidade de compensação ponderada de movimento 302 executa o processo de ponderação com base na Expressão Numérica (7) em um caso em que o primeiro e o segundo sinalizadores de aplicativo WP são "1"s.

[00220] Além disso, em um caso em que a precisão de cálculo da primeira e da segunda imagens previstas e a precisão de cálculo da imagem prevista são diferentes entre si, a unidade de compensação ponderada de movimento 302 realiza um processo de arredondamento ao controlar $\log W D_c$, que é a precisão de ponto fixo, como na Expressão Numérica (8).

[00221] Além disso, em um caso em que o modo de predição representado pela informação de movimento (parâmetro de predição) é uma predição unidirecional, a unidade de compensação ponderada de movimento 302 calcula uma imagem prevista final usando apenas a primeira imagem prevista com base na Expressão Numérica (9) .

[00222] Além disso, em um caso em que a precisão de cálculo da primeira e da segunda imagens previstas e a precisão de cálculo da imagem prevista são diferentes entre si, a unidade de compensação ponderada de movimento 302 realiza um processo de arredondamento ao controlar $\log W D_c$, que é precisão de ponto fixo, como na Expressão Numérica (8), similarmente ao caso da predição bidirecional.

[00223] A precisão de ponto fixo do fator de ponderação já foi descrita com referência à Figura 7, e, portanto, a descrição da mesma não será apresentada. Além disso, no caso de uma predição unidirecional, vários parâmetros (o segundo sinalizador de aplicativo WP, o segundo fator de ponderação, e a segunda informação de deslocamento) correspondendo à segunda imagem prevista não são usados e podem ser ajustados nos valores iniciais determinados de antemão.

[00224] A unidade de decodificação 801 usa a sintaxe 500 repre-

sentada na Figura 9. A sintaxe 500 representa a estrutura de dados codificados que é um alvo de decodificação da unidade de decodificação 801. A sintaxe 500 já foi descrita com referência à Figura 9, e, portanto, a descrição da mesma não será apresentada. Além disso, a sintaxe de ajuste de parâmetro de imagem 505 foi descrita com referência à Figura 10, exceto que a decodificação é usada em vez da codificação, e, portanto, a descrição da mesma não será apresentada. Adicionalmente, a sintaxe de cabeçalho da fatia 507 já foi descrita com referência à Figura 11, exceto que a decodificação é usada em vez da codificação, e, portanto, a descrição da mesma não será representada. Além disso, a sintaxe da tabela de peso de predição 508 já foi descrita com referência à Figura 12, exceto que a decodificação é usada em vez da codificação, e, portanto, a descrição da mesma não será apresentada.

[00225] Aqui, um método de prever cada elemento de sintaxe referente à predição ponderada na configuração de sintaxe será descrito em detalhes. A predição do elemento de sintaxe é executada pela unidade de reconfiguração de índice 801B. A configuração de sintaxe que explicitamente representa o método de predição de acordo com a segunda concretização é igual àquela da segunda concretização e é conforme ilustrada na Figura 13.

[00226] No método de predição entre sinais de `luma_log2_weight_denom` e `chroma_log2_weight_denom` que representam a precisão de ponto fixo do fator de ponderação, um processo de restauração é executado usando a Expressão Numérica (11). Os detalhes do processo de restauração são como ilustrados na Figura 15.

[00227] No método de predição de `luma_weight_1x[i]` e `chroma_weight_1x[i][j]` representando fatores de ponderação da luminância e o sinal de diferença de cor, um processo de restauração é executado usando as Expressões Numéricas (14) e (17). Detalhes do processo

de restauração são como ilustrados na Figura 17.

[00228] No método de predição no qual os valores previstos dos fatores de ponderação ($\text{luma_weight_1x}[i]$ e $\text{chroma_weight_1x}[i][j]$) da luminância e do sinal de diferença de cor são derivados com outros números de referência ou outros números POC, um processo de restauração é executado usando as Expressões Numéricas (19) e (21). Detalhes do processo de restauração são como ilustrados na Figura 19.

[00229] No método de predição no qual os valores previstos dos fatores de ponderação ($\text{luma_weight_1x}[i]$ e $\text{chroma_weight_1x}[i][j]$) da luminância e do sinal de diferença de cor são derivados usando uma distância entre um alvo de codificação e a fatia de referência, um processo de restauração é executado usando as Expressões Numéricas (23) e (25). Detalhes do processo de restauração são iguais àqueles do fluxograma ilustrado na Figura 19 com a introdução do $(i-1)^{\text{ésimo}}$ valor ($i \neq 0$) em baseidx .

[00230] Na técnica para derivar os valores previstos do fator de ponderação e da precisão de ponto fixo usando o método de derivar um parâmetro WP da predição ponderada implícita definida em H.264 e semelhante, um processo de restauração é executado usando as Expressões Numéricas (31) e (33). Detalhes do processo de restauração são como ilustrados na Figura 23.

[00231] Uma pluralidade de técnicas de predição descritas acima pode não apenas ser usada independentemente, mas também ser usada de maneira combinada. Por exemplo, com a combinação de Expressões Numéricas (11), (14), (17) e (28), é possível eficientemente reduzir a quantidade de código dos elementos de sintaxe da informação de índice.

[00232] Conforme acima, de acordo com a segunda concretização, o dispositivo de decodificação 800 elimina a redundância espacial

usando a correlação entre os parâmetros da informação a ser codificada, por meio do que um problema de diminuir a eficiência de codificação é solucionado. O dispositivo de decodificação 800 pode reduzir a quantidade de código comparada àquela de uma configuração convencional na qual os elementos de sintaxe usados na predição de compensação ponderada de movimento são diretamente codificados (valores diretos).

Modificação

[00233] Na primeira e na segunda concretizações descritas acima, foi descrito um exemplo no qual o quadro é dividido em blocos retangulares, cada qual apresentando um tamanho de 16 x 16 pixels ou semelhante e é codificado/decodificado na ordem de um bloco esquerdo superior da tela para o bloco direito inferior (vide Figura 2A). Contudo, a ordem de codificação e a ordem de decodificação não são limitadas àquelas ilustradas neste exemplo. Por exemplo, a codificação e a decodificação podem ser executadas na ordem do lado direito inferior para o lado esquerdo superior, ou a codificação e a decodificação podem ser executadas de modo a extraírem um vórtice do centro da tela na direção da extremidade da tela. Além disso, a codificação e a decodificação podem ser executadas na ordem do lado direito superior para o lado esquerdo inferior, ou a codificação e a decodificação podem ser executadas de modo a arrastarem um vórtice da extremidade da tela para o centro da tela. Em tal caso, uma vez que a posição de um bloco de pixels adjacente que pode ser referida de acordo com mudanças de ordem de codificação, a posição pode ser mudada para uma posição apropriadamente usável.

[00234] Na primeira e na segunda concretizações descritas acima, enquanto a descrição foi apresentada com o tamanho de um bloco alvo de predição, tal como um bloco de pixels de 4 x 4, um bloco de pixels de 8 x 8, um bloco de pixels de 16 x 16 ou semelhante que é

ilustrado como um exemplo, o bloco alvo de predição não pode ter uma forma de bloco uniforme. Por exemplo, o tamanho do bloco alvo de predição pode ser um bloco de pixels de 16 x 8, um bloco de pixels de 8 x 16, um bloco de pixels de 8 x 4, um bloco de pixels de 4 x 8, ou semelhante. Além disso, não é necessário uniformizar todos os tamanhos de bloco dentro de um bloco de árvore de codificação, e uma pluralidade de tamanhos de bloco diferentes entre si pode ser misturada. Em um caso em que uma pluralidade de tamanhos de bloco diferentes entre si é misturada dentro de um bloco de árvore de codificação, a quantidade de código para informação de divisão de codificação ou decodificação aumenta de acordo com um aumento no número de divisões. Desse modo, é preferível selecionar um tamanho de bloco em vista do equilíbrio entre a quantidade de código da informação de divisão e a qualidade de uma imagem codificada local ou uma imagem decodificada.

[00235] Na primeira e na segunda concretizações descritas acima, para simplificação, uma descrição compreensiva foi apresentada para um componente de sinal de cor sem os processos de predição do sinal de luminância e do sinal de diferença de cor não sendo diferenciados entre si. Contudo, em um caso em que os processos de predição do sinal de luminância e do sinal de diferença de cor são diferentes entre si, pode ser usado o mesmo método de predição ou métodos de predição diferentes entre si. Em um caso em que os métodos de predição diferentes entre si são usados para o sinal de luminância e o sinal de diferença de cor, a codificação e a decodificação podem ser executadas usando o método de predição selecionado para o sinal de diferença de cor similarmente àquelas do sinal de luminância.

[00236] Na primeira e na segunda concretizações descritas acima, para simplificação, uma descrição compreensiva foi apresentada para um componente de sinal de cor sem os processos de predição ponde-

rada compensada por movimento do sinal de luminância e do sinal de diferença de cor não sendo diferenciados entre si. Contudo, em um caso em que os processos de predição ponderada compensada por movimento do sinal de luminância e do sinal de diferença de cor são diferentes entre si, pode ser usado o mesmo método de predição ponderada compensada por movimento ou métodos de predição ponderada compensada por movimento diferentes entre si. Em um caso em que os métodos de predição ponderada compensada por movimento diferentes entre si são usados para o sinal de luminância e o sinal de diferença de cor, a codificação ou a decodificação pode ser executada usando o método de predição ponderada compensada por movimento selecionado para o sinal de diferença de cor similarmente àquelas para o sinal de luminância.

[00237] Na primeira e na segunda concretizações descritas acima, entre as fileiras da tabela representada na configuração de sintaxe, um elemento de sintaxe não definido nesta concretização pode ser inserido, e uma descrição referente a outras ramificações pode ser incluída. Alternativamente, uma tabela de sintaxe pode ser dividida em uma pluralidade de tabelas, ou tabelas de sintaxe podem ser integradas entre si. Além disso, o mesmo termo pode não ser necessariamente usado, mas o tempo pode ser arbitrariamente mudado de acordo com uma forma usada.

[00238] Conforme descrito acima, de acordo com cada concretização, o problema de codificar informação redundante da configuração de sintaxe no momento de executar uma predição de compensação ponderada de movimento é solucionado, sendo realizado o processo de predição ponderada compensada por movimento apresentando alta eficiência. Por isso, de acordo com cada concretização, a eficiência de codificação é aperfeiçoada, e a qualidade de imagem subjetiva é aperfeiçoada.

[00239] Enquanto várias concretizações da presente invenção foram descritas, tais concretizações são apresentadas como exemplos e não para fins de limitar o escopo da invenção. Estas novas concretizações podem ser executadas em outras várias formas, e várias omissões, substituições, e mudanças podem ser feitas nas mesmas em uma faixa que não se afaste do conceito da invenção. Estas concretizações e modificações da mesma pertencem ao escopo ou ao conceito da invenção e pertencem à invenção descrita nas concretizações e a um escopo equivalente à mesma.

[00240] Por exemplo, um programa que realiza o processo de cada concretização descrita acima pode ser provido em sendo armazenado em um meio de armazenamento legível por computador. Como o meio de armazenamento, pode ser usado um meio de armazenamento que pode armazenar um programa e pode ser lido por um computador, tal como um disco magnético, um disco óptico (um CD-ROM, um CD-R, um DVD, ou semelhante), um disco magneto-óptico (um MO ou semelhante), ou uma memória semicondutora, não obstante a forma de armazenamento.

[00241] Além disso, o programa que realiza o processo de cada concretização pode ser armazenado em um computador (servidor) conectado a um computador (cliente) através da rede.

LISTA DE SINAIS DE REFERÊNCIA

- 100 dispositivo de codificação
- 101 unidade de subtração
- 102 unidade de transformação ortogonal
- 103 unidade de quantização
- 104 unidade de quantização inversa
- 105 unidade de transformação ortogonal inversa
- 106 unidade de adição
- 107 unidade de geração de imagem prevista

- 108 unidade de ajuste de índice
- 109 unidade de avaliação de movimento
- 110 unidade de codificação
- 110A unidade de codificação entrópica
- 110B unidade de reconfiguração de índice
- 111 unidade de controle de codificação
- 201 unidade de compensação de movimento de múltiplos quadros
- 202 memória
- 203 unidade de compensação de movimento unidirecional
- 204 unidade de controle de parâmetro de predição
- 205 seletor de imagem de referência
- 206 memória de quadro
- 207 unidade de controle de imagem de referência
- 301 unidade de compensação de movimento padrão
- 302 unidade de compensação de movimento ponderado
- 303 unidade de controle de parâmetro WP
- 304, 305 seletor WP
- 800 dispositivo de decodificação
- 801 unidade de decodificação
- 801A unidade de decodificação entrópica
- 801B unidade de reconfiguração de índice
- 802 unidade de quantização inversa
- 803 unidade de transformação ortogonal inversa
- 804 unidade de adição
- 805 unidade de geração de imagem prevista
- 806 unidade de ajuste de índice
- 807 unidade de controle de decodificação

REIVINDICAÇÕES

1. Método de codificação **caracterizado pelo fato de que** compreende as etapas de:

codificar um primeiro denominador para um fator de ponderação luma de uma imagem alvo;

codificar um primeiro valor de diferença que é igual a um valor de uma diferença entre o primeiro denominador e um segundo denominador para um fator de ponderação de croma da imagem alvo;

codificar um segundo valor de diferença do fator de ponderação luma, em que o segundo valor de diferença é igual a um valor de uma diferença entre o fator de ponderação luma e um primeiro valor de referência, o primeiro valor de referência do fator de ponderação luma estando de acordo com o primeiro denominador, em que o primeiro valor de referência é igual a um valor obtido pelo deslocamento à esquerda de "1" por pelo menos um dígito binário especificado pelo primeiro denominador;

codificar um terceiro valor de diferença do fator de ponderação de croma, em que o terceiro valor de diferença é igual a um valor de uma diferença entre o fator de ponderação de croma e um segundo valor de referência, o segundo valor de referência do fator de ponderação de croma estando de acordo com o segundo denominador, em que o segundo valor de referência é igual a um valor obtido pelo deslocamento à esquerda de "1" por pelo menos um dígito binário especificado pelo segundo denominador;

codificar um quarto valor de diferença do deslocamento de croma, em que o quarto valor de diferença é igual a um valor de uma diferença entre o deslocamento de croma e um terceiro valor de referência, o terceiro valor de referência de um deslocamento de croma da imagem alvo é igual a um valor obtido subtraindo um valor obtido multiplicando um valor mediano de um valor máximo de pixel pelo fator de

ponderação de croma e deslocando para a direita para o pelo menos um dígito binário especificado pelo segundo denominador, a partir do valor mediano.

2. Método de decodificação, **caracterizado pelo fato de que** compreende as etapas de:

inserir dados codificados;

decodificar um primeiro denominador para um fator de ponderação luma de uma imagem alvo a partir dos dados codificados;

decodificar um primeiro valor de diferença igual a um valor de diferença entre o primeiro denominador e um segundo denominador para um fator de ponderação de croma da imagem alvo a partir dos dados codificados;

derivar o segundo denominador adicionando o primeiro denominador e o primeiro valor da diferença;

derivar um primeiro valor de referência do fator de ponderação luma de acordo com o primeiro denominador, em que o primeiro valor de referência é igual a um valor obtido pelo deslocamento à esquerda de "1" por pelo menos um dígito binário especificado pelo primeiro denominador;

decodificar um segundo valor de diferença do fator de ponderação luma a partir dos dados codificados, em que o segundo valor de diferença é igual a um valor de uma diferença entre o fator de ponderação luma e o primeiro valor de referência;

derivar o fator de ponderação luma adicionando o primeiro valor de referência e o segundo valor da diferença;

derivar um segundo valor de referência do fator de ponderação de croma de acordo com o segundo denominador, em que o segundo valor de referência é igual a um valor obtido pelo deslocamento à esquerda de "1" por pelo menos um dígito binário especificado pelo segundo denominador;

decodificar um terceiro valor de diferença do fator de ponderação de croma a partir dos dados codificados, em que o terceiro valor de diferença é igual a um valor de uma diferença entre o fator de ponderação de croma e o segundo valor de referência;

derivar o fator de ponderação de croma adicionando o segundo valor de referência e o terceiro valor da diferença;

derivar um terceiro valor de referência de um deslocamento de croma da imagem alvo subtraindo um valor obtido multiplicando um valor mediano de um valor máximo de pixel pelo fator de ponderação de croma e deslocando para a direita para o pelo menos um dígito binário especificado pelo segundo denominador, a partir do valor mediano;

decodificar um quarto valor de diferença do deslocamento de croma a partir dos dados codificados, em que o quarto valor de diferença é igual a um valor de uma diferença entre o deslocamento de croma e o terceiro valor de referência; e

derivar o deslocamento de croma adicionando o terceiro valor de referência e o quarto valor da diferença.

FIG. 1

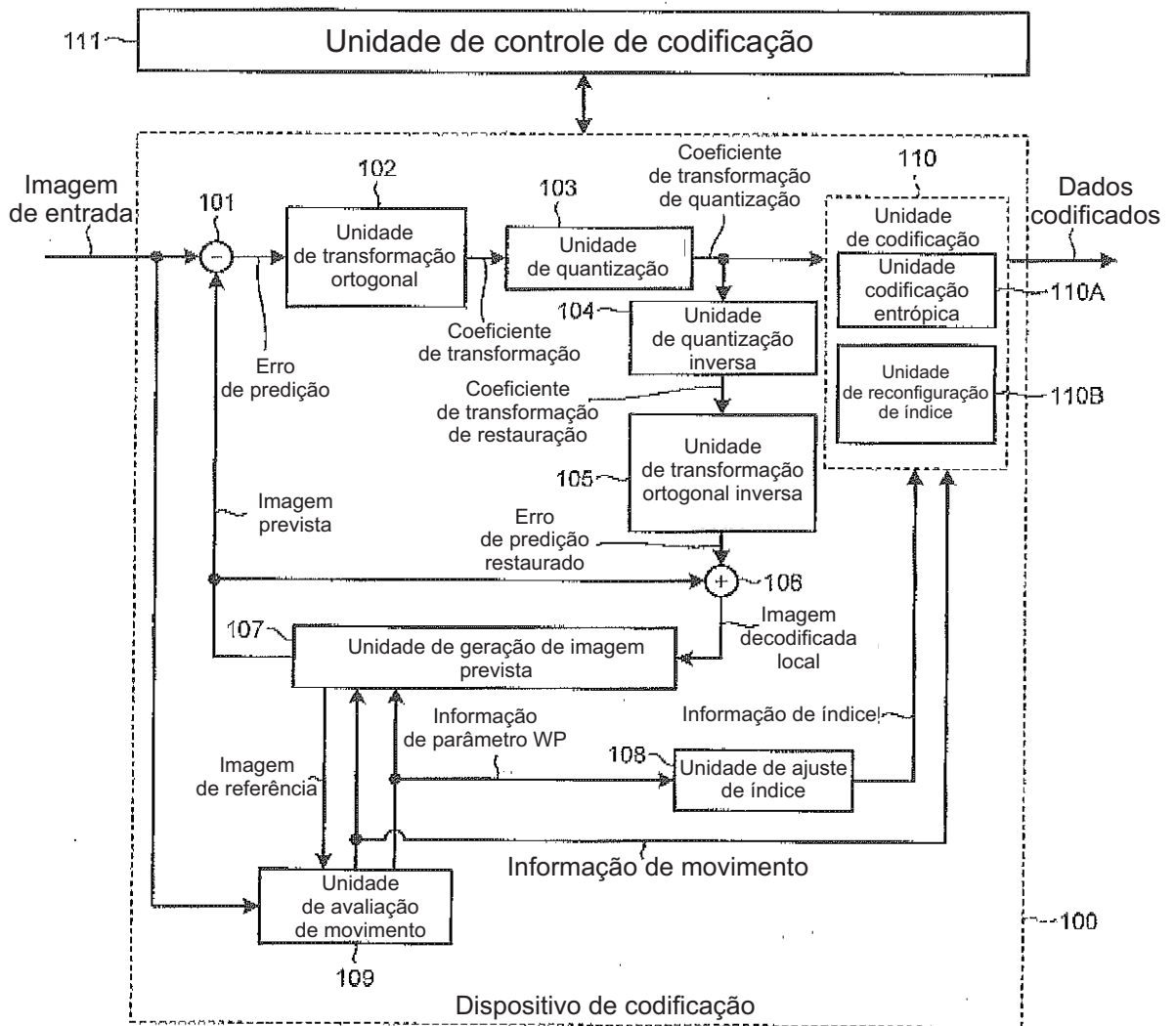


FIG.2

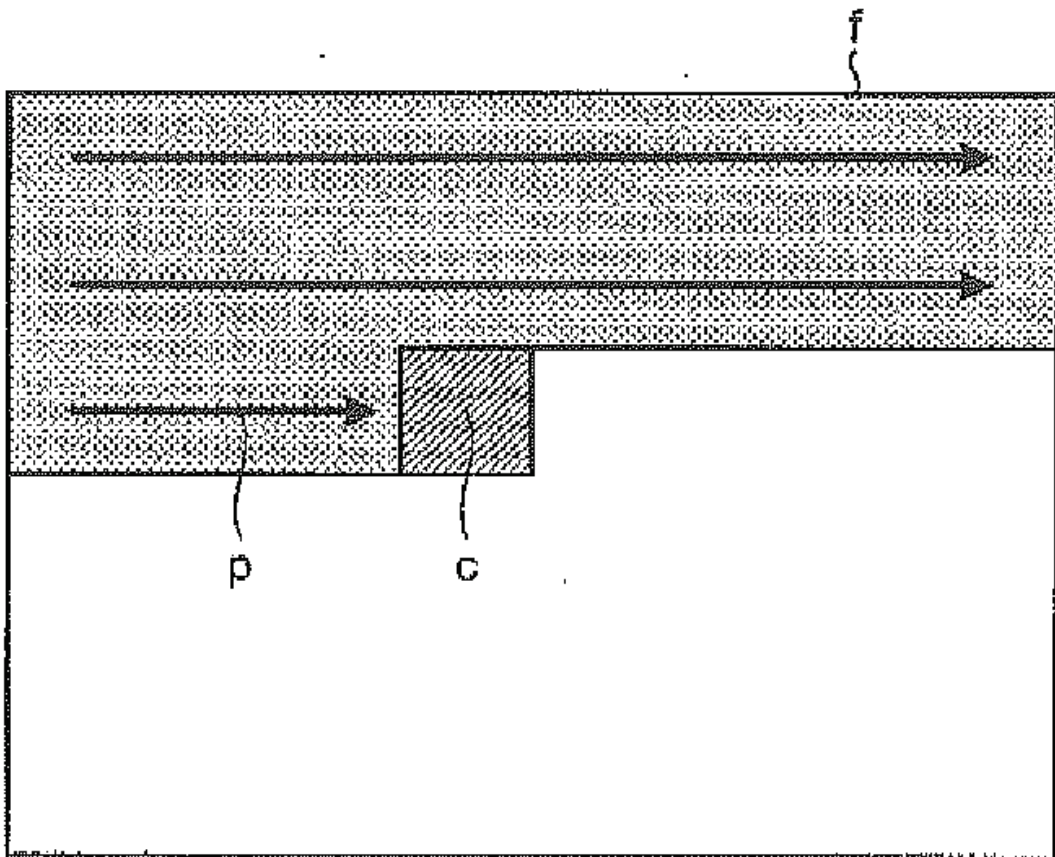


FIG.3A

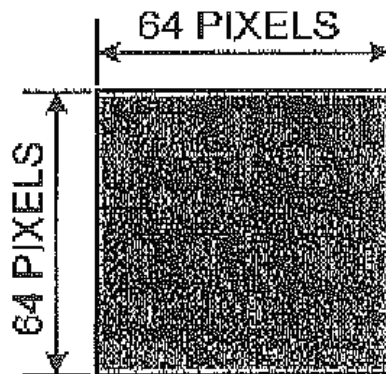


FIG.3B

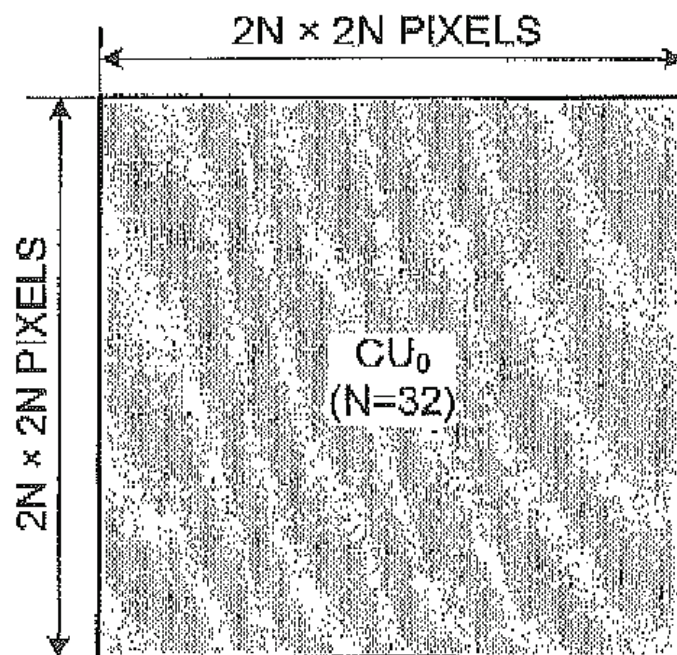


FIG.3C

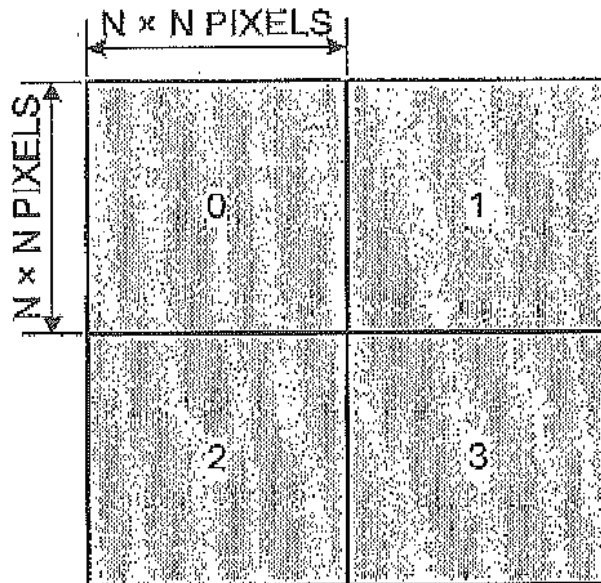


FIG.3D

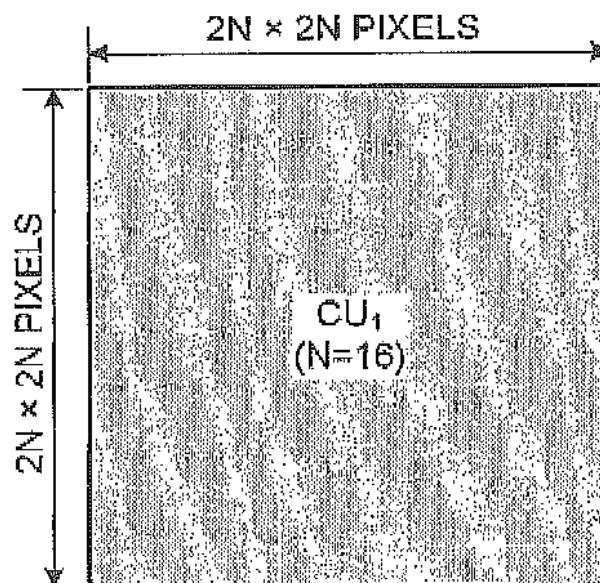


FIG.4

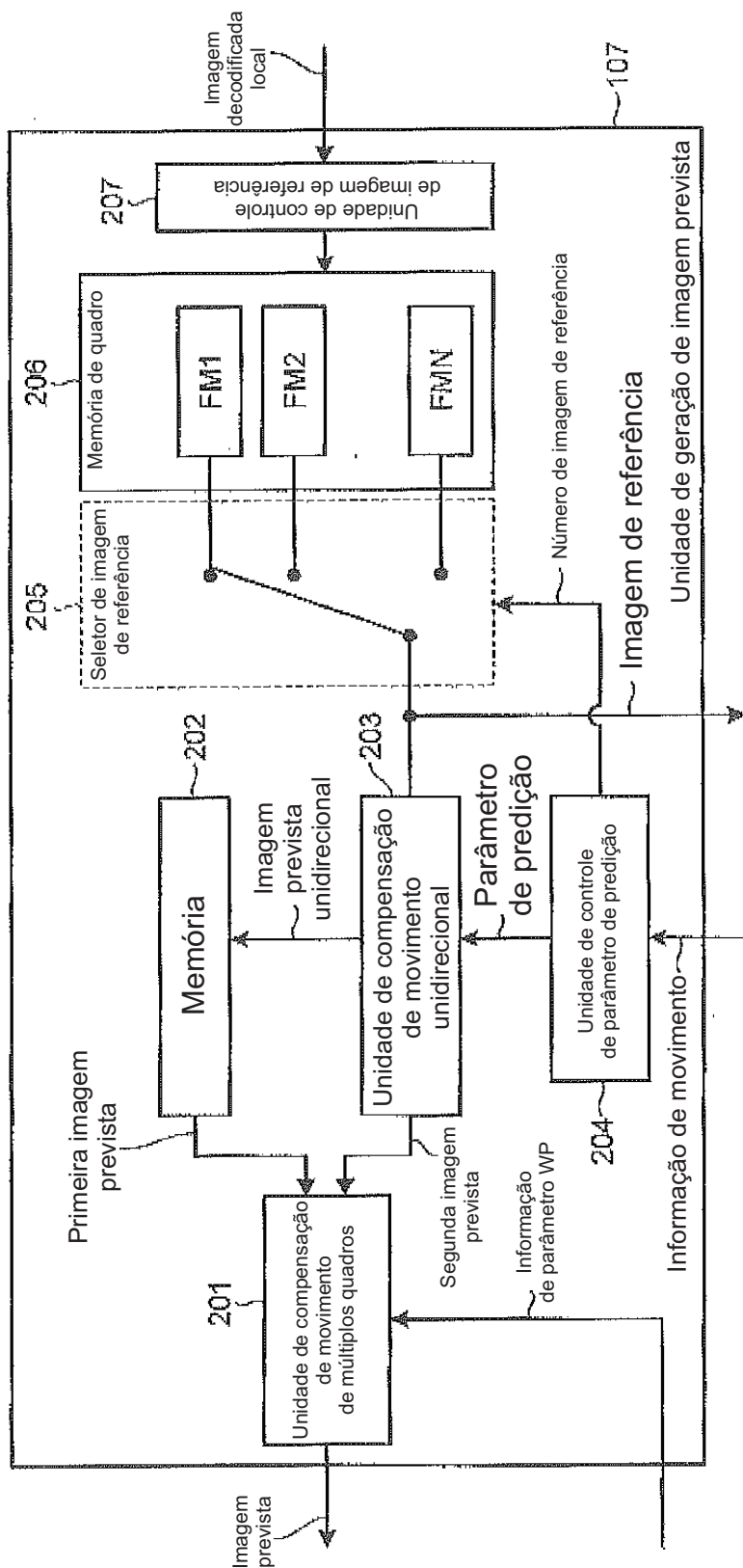


FIG.5

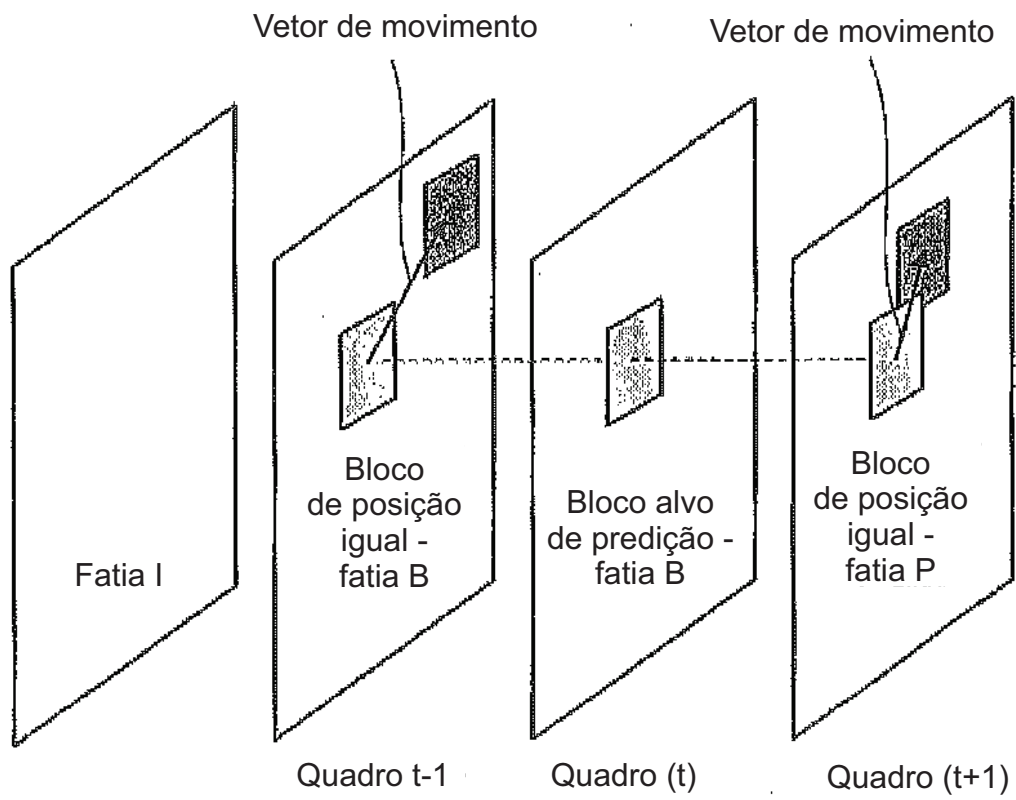


FIG.6

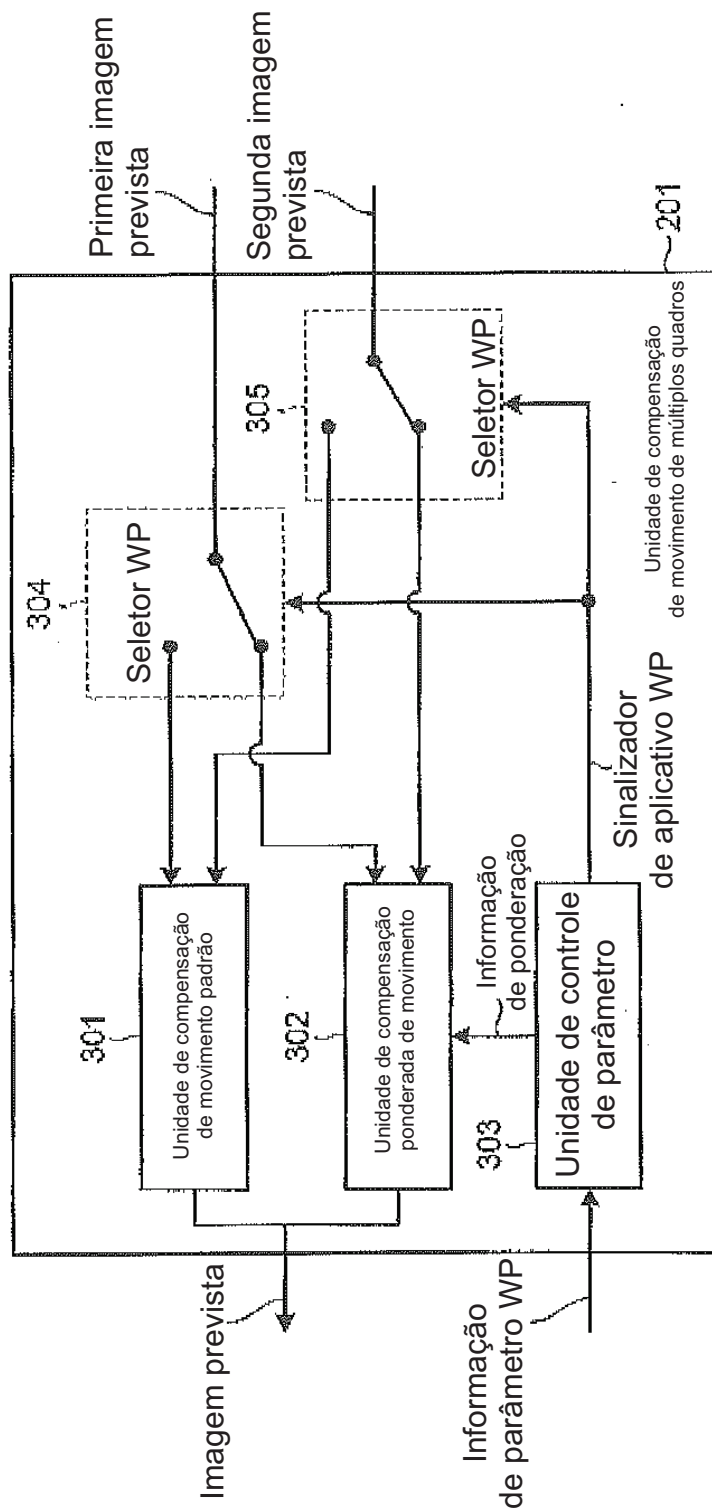


FIG.7

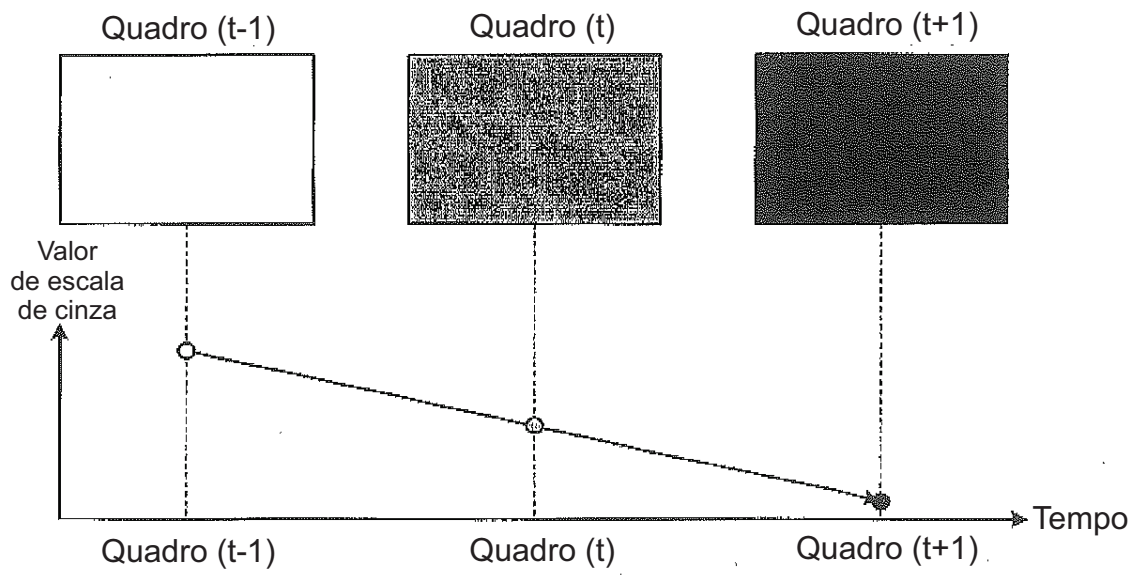


FIG.8A

Número da lista	0						
Número de referência	0	1	2	3	4	...	N
Sinalizador de aplicativo WP	WP_flag[0]	WP_flag[1]	WP_flag[2]	WP_flag[3]	WP_flag[4]	...	WP_flag[N]
Fator de ponderação	Weight[0]	Weight[1]	Weight[2]	Weight[3]	Weight[4]	...	Weight[N]
Deslocamento	Offset[0]	Offset[1]	Offset[2]	Offset[3]	Offset[4]	...	Offset[N]

FIG.8B

Número da lista	1						
Número de referência	0	1	2	3	4	...	N
Sinalizador de aplicativo WP	WP_flag[0]	WP_flag[1]	WP_flag[2]	WP_flag[3]	WP_flag[4]	...	WP_flag[N]
Fator de ponderação	Weight[0]	Weight[1]	Weight[2]	Weight[3]	Weight[4]	...	Weight[N]
Deslocamento	Offset[0]	Offset[1]	Offset[2]	Offset[3]	Offset[4]	...	Offset[N]

FIG.9

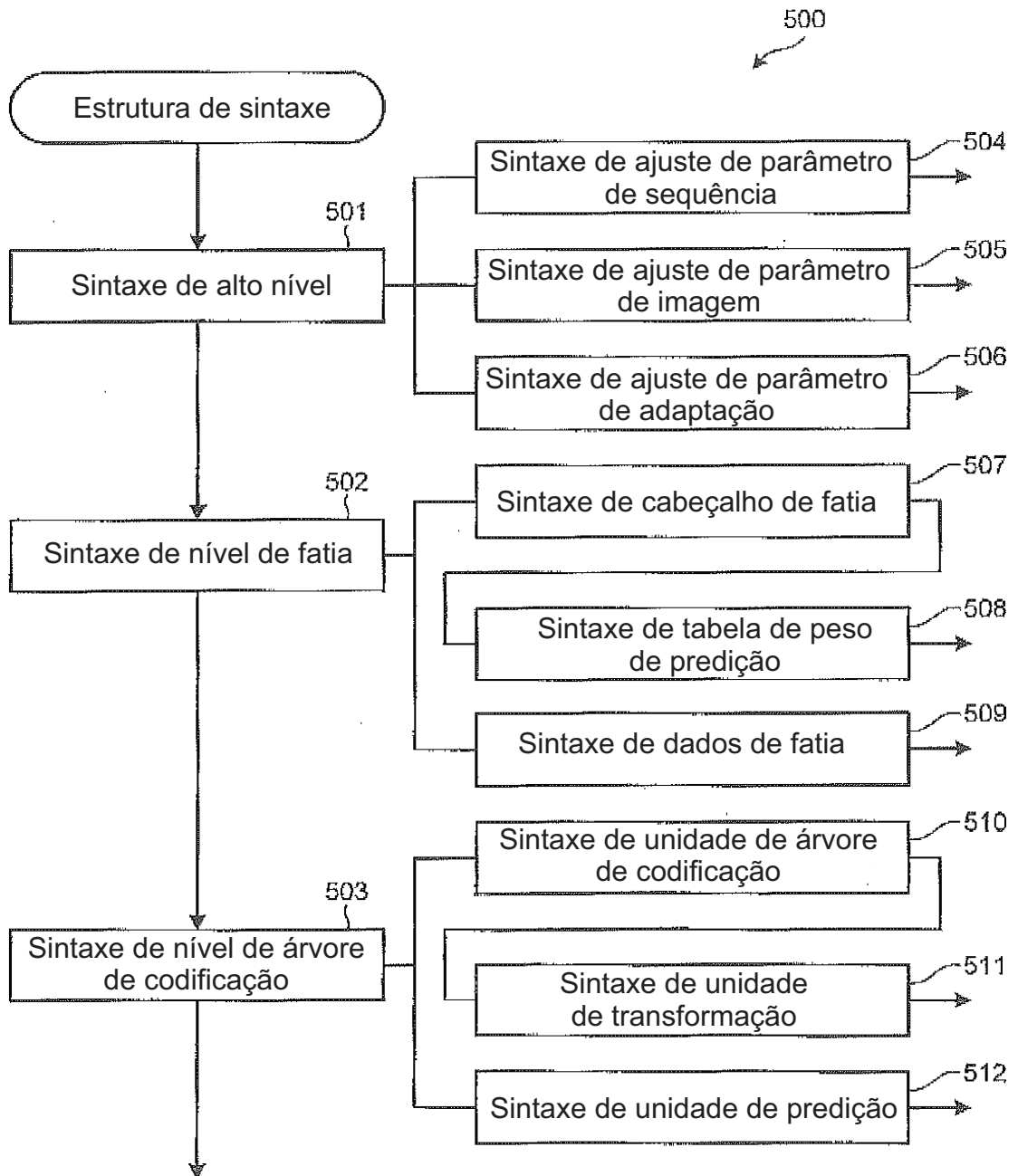


FIG.10

	Descriptor
pic_parameter_set_rbsp() {	
pic_parameter_set_id	u (8)
seq_parameter_set_id	u (8)
...	
num_ref_idx_l0_default_active_minus1	ue (v)
num_ref_idx_l1_default_active_minus1	ue (v)
weighted_pred_flag	u (1)
weighted_bipred_idc	u (2)
...	
rbsp_trailing_bits()	
}	

FIG.11

	Descriptor
slice_header() {	
...	
slice_type	ue (v)
pic_parameter_set_id	ue (v)
...	
if(slice_type == P slice_type == B) {	
num_ref_idx_active_override_flag	u (1)
if(num_ref_idx_active_override_flag) {	
num_ref_idx_l0_active_minus1	ue (v)
if(slice_type == B)	
num_ref_idx_l1_active_minus1	ue (v)
}	
}	
...	
if((weighted_pred_flag && (slice_type == P)) (weighted_bipred_idc == 1 && slice_type == B))	
pred_weight_table()	
...	
}	

FIG.12

	Descriptor
pred_weight_table() {	
luma_log2_weight_denom	ue (v)
if(chroma_format_idc != MONO_IDX)	
chroma_log2_weight_denom	ue (v)
for(i = 0; i <= num_ref_idx_l0_active_minus1; i++) {	
luma_weight_l0_flag	u (1)
if(luma_weight_l0_flag) {	
luma_weight_l0[i]	se (v)
luma_offset_l0[i]	se (v)
}	
if (chroma_format_idc != MONO_IDX) {	
chroma_weight_l0_flag	u (1)
if(chroma_weight_l0_flag)	
for(j = 0; j < COLOR_COMP - 1; j++) {	
chroma_weight_l0[i][j]	se (v)
chroma_offset_l0[i][j]	se (v)
}	
}	
}	
}	
if(slice_type == B_SLICE)	
for(i = 0; i <= num_ref_idx_l1_active_minus1; i++) {	
luma_weight_l1_flag	u (1)
if(luma_weight_l1_flag) {	
luma_weight_l1[i]	se (v)
luma_offset_l1[i]	se (v)
}	
if(chroma_format_idc != MONO_IDX) {	
chroma_weight_l1_flag	u (1)
if(chroma_weight_l1_flag) {	
for(j = 0; j < COLOR_COMP - 1; j++) {	
chroma_weight_l1[i][j]	se (v)
chroma_offset_l1[i][j]	se (v)
}	
}	
}	
}	
}	
}	

FIG.13

	Descriptor
pred_weight_table() {	
luma_log2_weight_denom	ue (v)
if(chroma_format_idc != MONO_IDX)	
delta_chroma_log2_weight_denom	ue (v)
for(i = 0; i <= num_ref_idx_l0_active_minus1; i++) {	
luma_weight_l0_flag	u (1)
if(luma_weight_l0_flag) {	
delta_luma_weight_l0[i]	se (v)
delta_luma_offset_l0[i]	se (v)
}	
if(chroma_format_idc != MONO_IDX) {	
chroma_weight_l0_flag	u (1)
if(chroma_weight_l0_flag)	
for(j = 0; j < COLOR_COMP - 1; j++) {	
delta_chroma_weight_l0[i][j]	se (v)
delta_chroma_offset_l0[i][j]	se (v)
}	
}	
}	
}	
if(slice_type == B_SLICE)	
for(i = 0; i <= num_ref_idx_l1_active_minus1; i++) {	
luma_weight_l1_flag	u (1)
if(luma_weight_l1_flag) {	
delta_luma_weight_l1[i]	se (v)
delta_luma_offset_l1[i]	se (v)
}	
if(chroma_format_idc != MONO_IDX) {	
chroma_weight_l1_flag	u (1)
if(chroma_weight_l1_flag) {	
for(j = 0; j < COLOR_COMP - 1; j++) {	
delta_chroma_weight_l1[i][j]	se (v)
delta_chroma_offset_l1[i][j]	se (v)
}	
}	
}	
}	
}	
}	

FIG.14

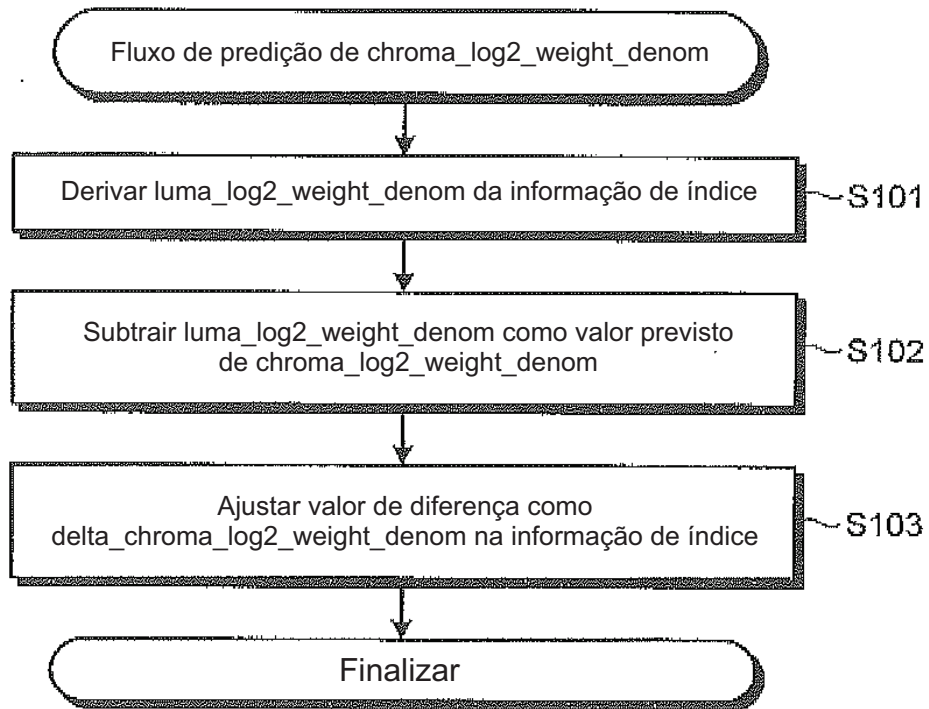


FIG.15

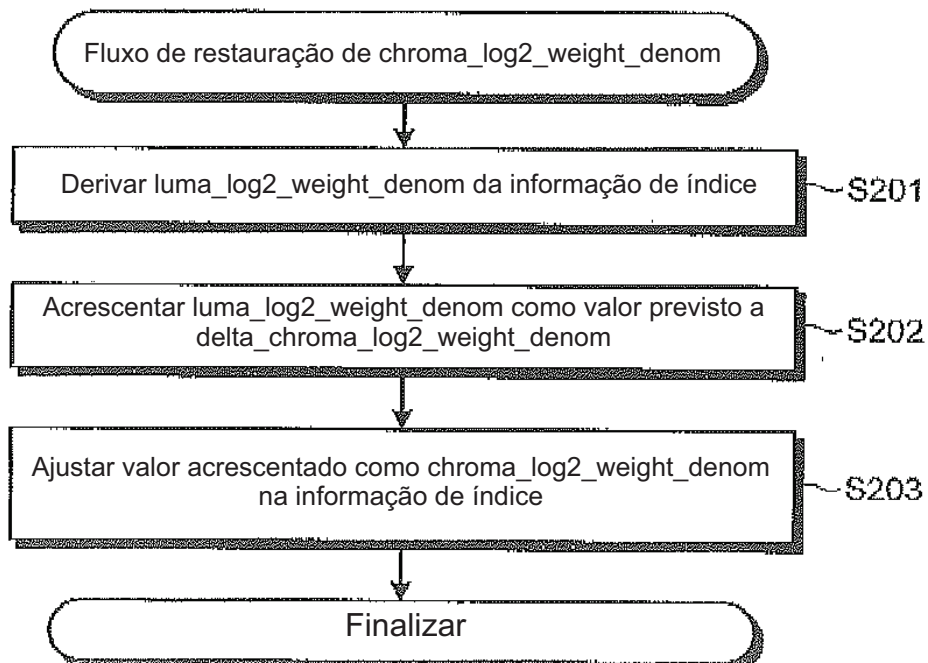


FIG.16

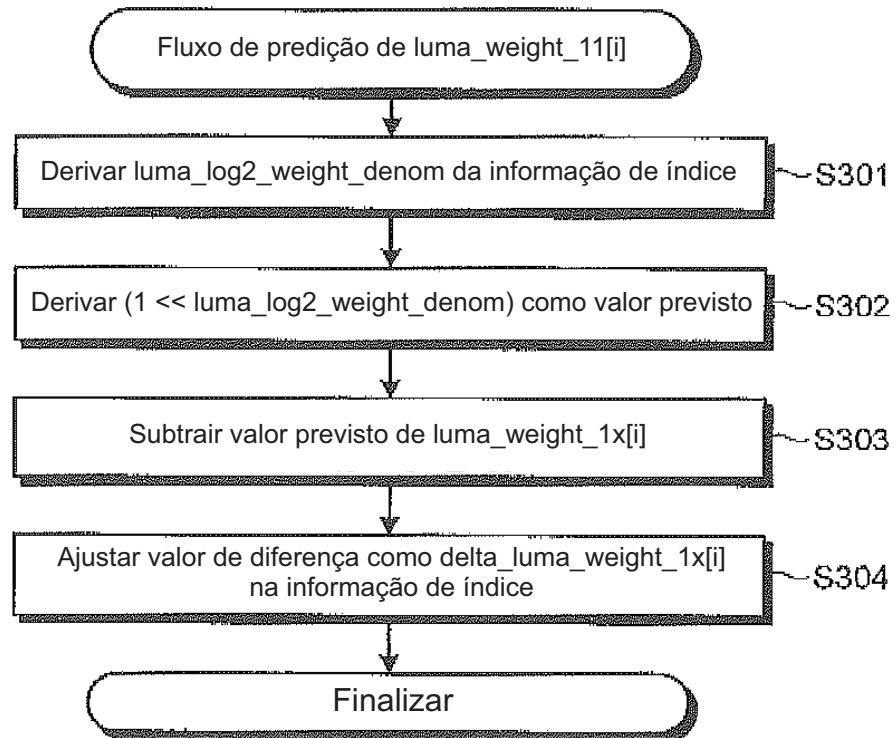


FIG.17

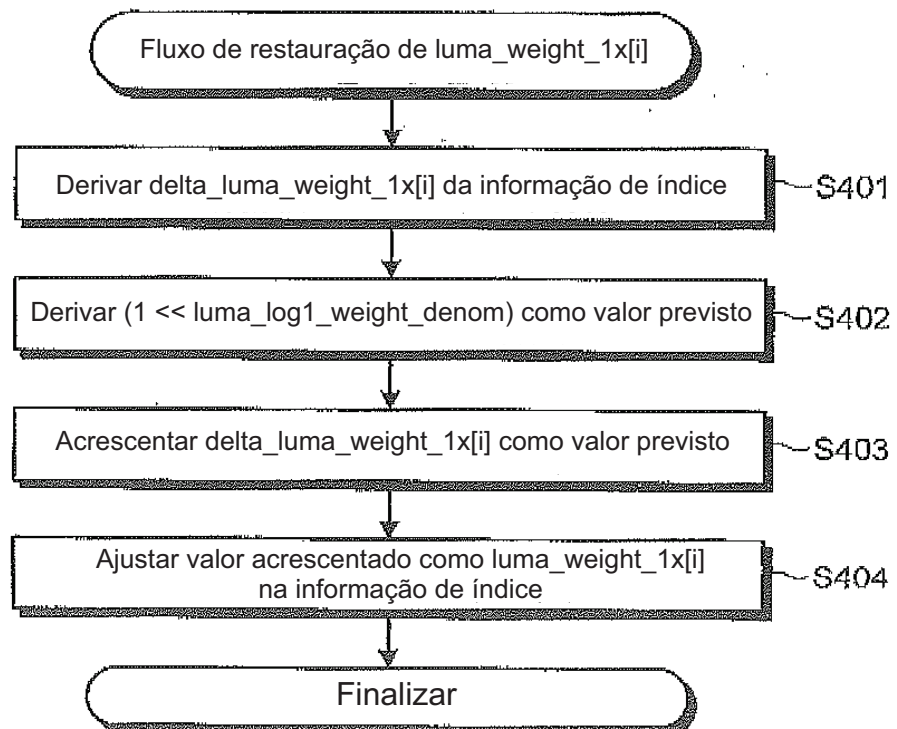


FIG.18

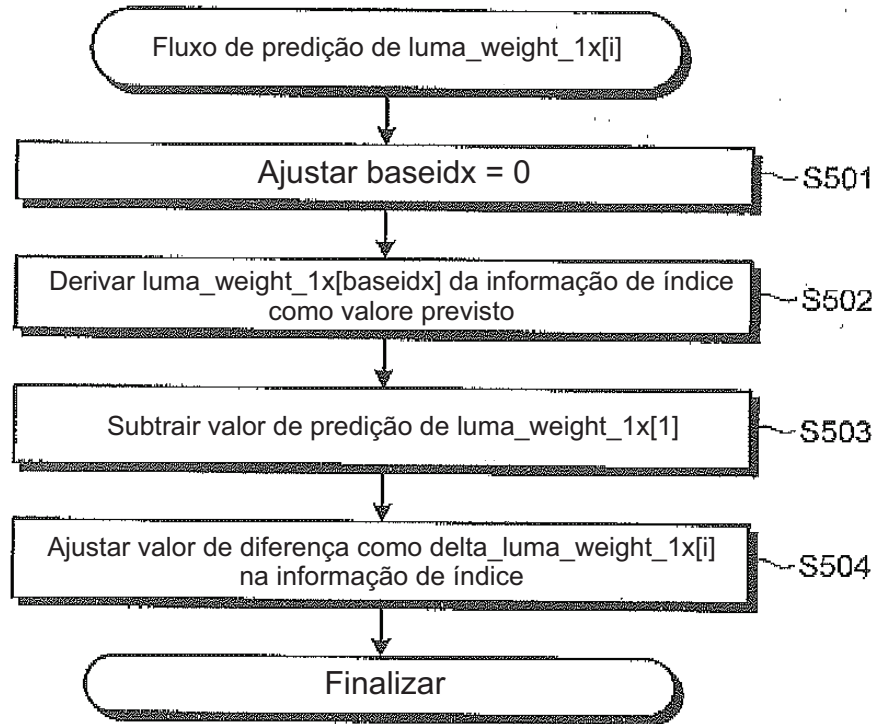


FIG.19

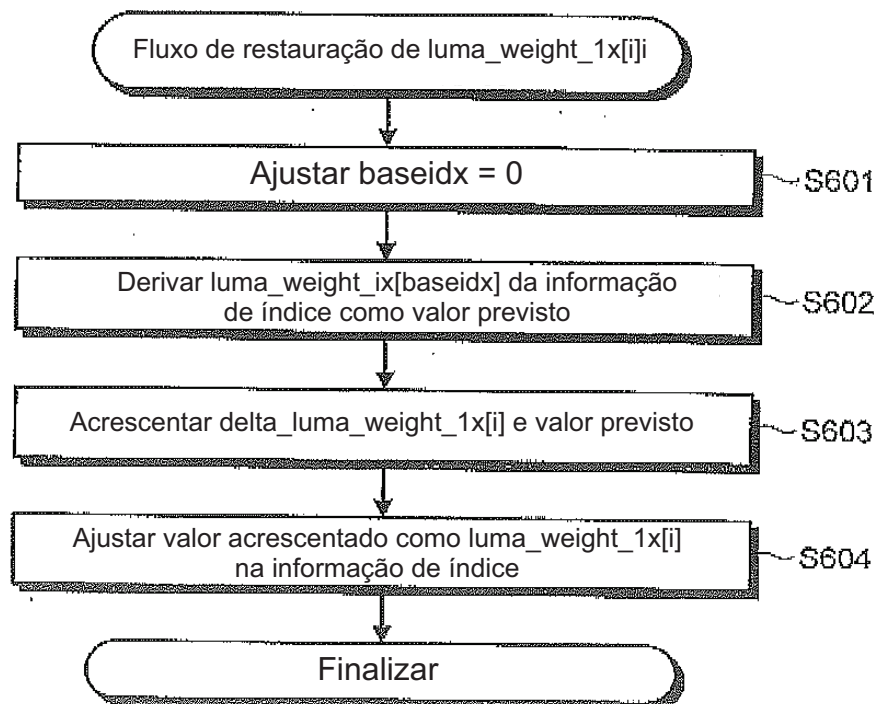


FIG.20

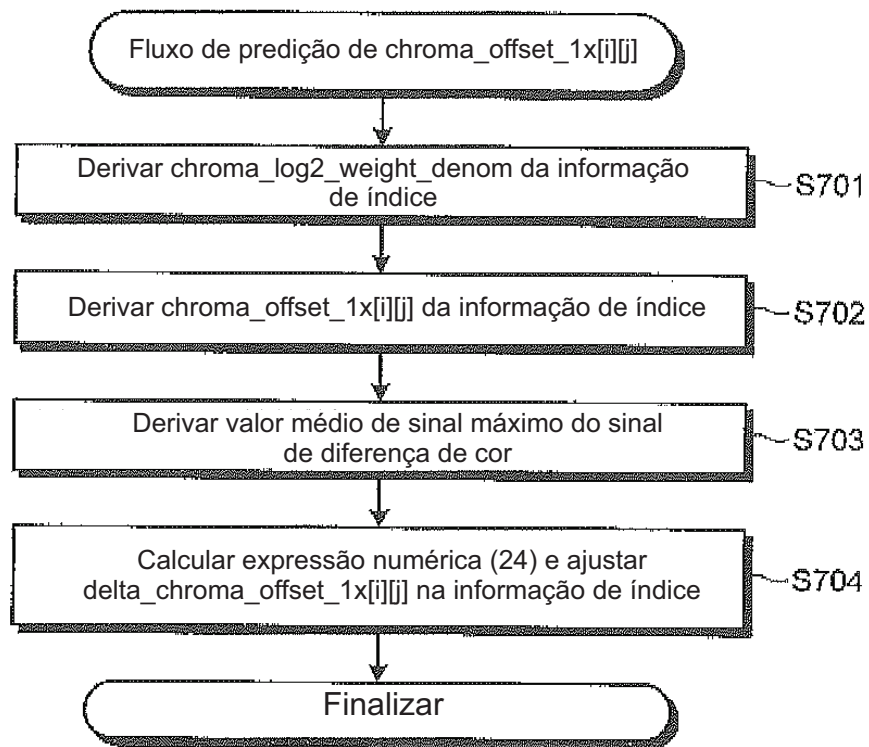


FIG.21

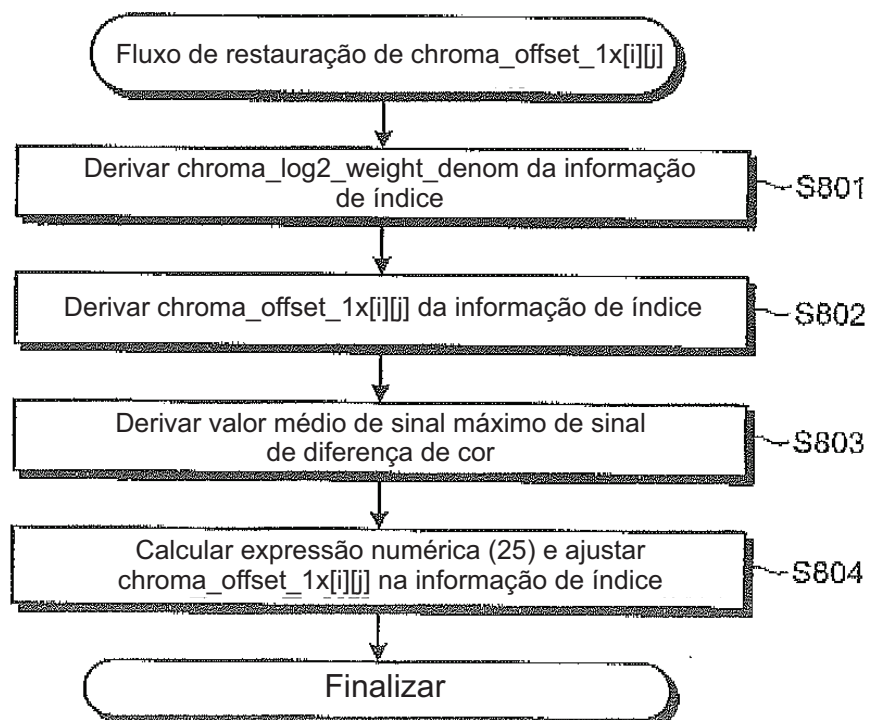


FIG.22

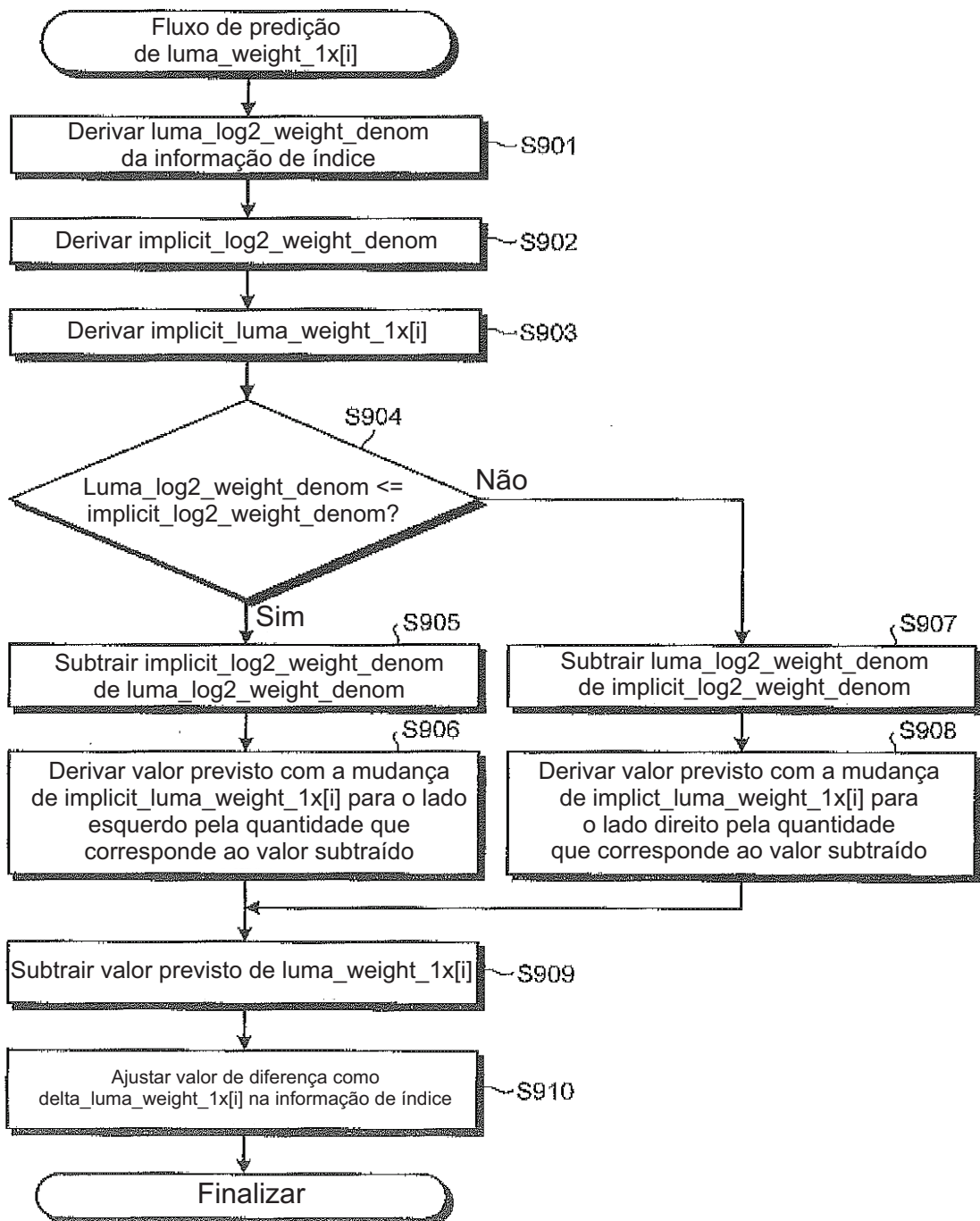


FIG.23

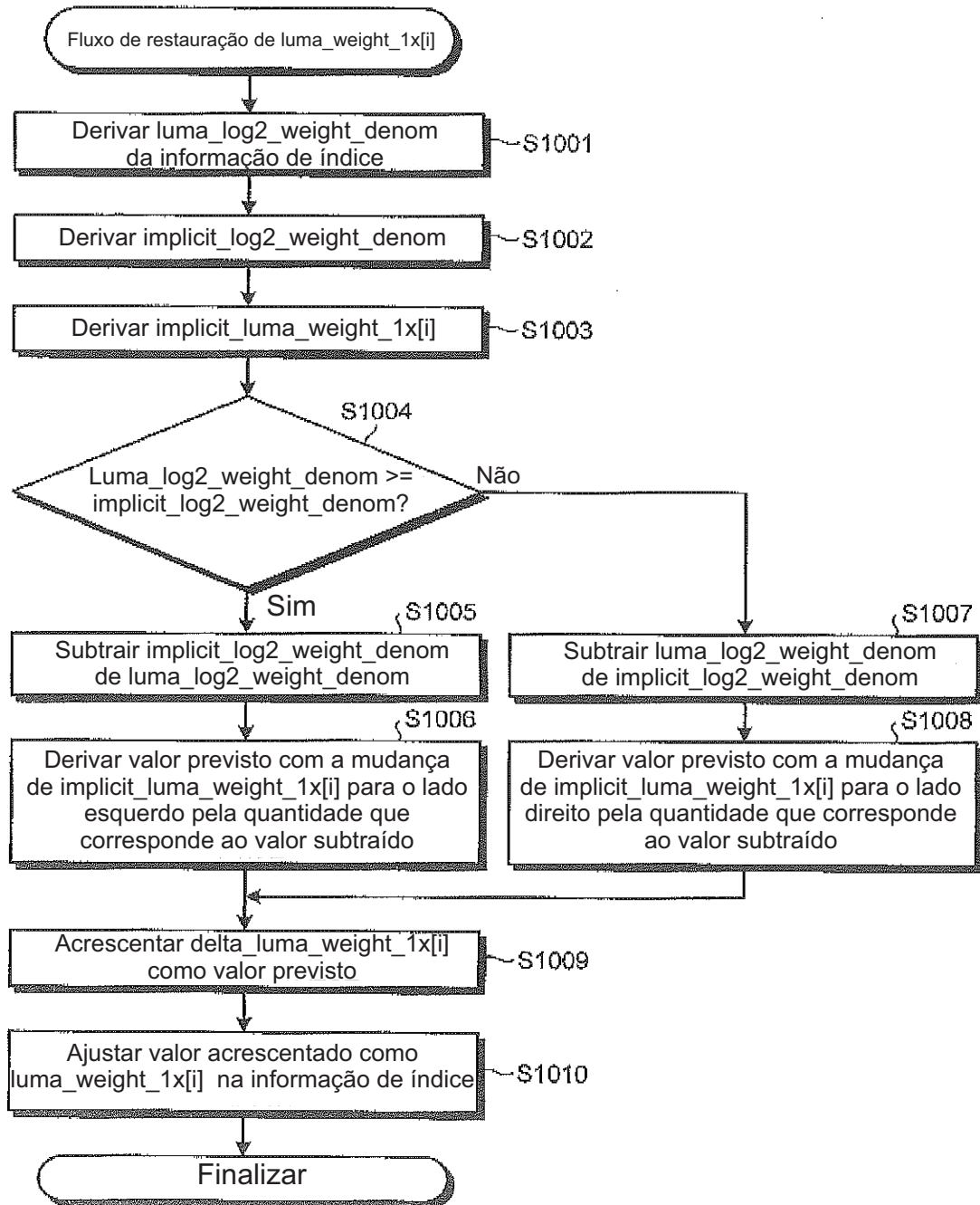


FIG.24

