



República Federativa do Brasil
Ministério da Indústria, Comércio Exterior
e Serviços
Instituto Nacional da Propriedade Industrial

(11) PI 0412166-0 B1

(22) Data do Depósito: 05/07/2004

(45) Data de Concessão: 06/06/2017



(54) Título: APARATO E MÉTODO PARA A CONVERSÃO EM UMA REPRESENTAÇÃO TRANSFORMADA OU PARA A CONVERSÃO INVERSA DA REPRESENTAÇÃO TRANSFORMADA

(51) Int.Cl.: G06F 17/14; G10L 19/02; G06T 9/00; H04N 7/30

(30) Prioridade Unionista: 14/07/2003 DE 103 31 803.8

(73) Titular(es): FRAUNHOFER-GESELLSCHAFT ZUR FOERDERUNG DER ANGEWANDTEN FORSCHUNG E. V.

(72) Inventor(es): RALF GEIGER; GERALD SCHULLER; THOMAS SPORER

"APARATO E MÉTODO PARA A CONVERSÃO EM UMA REPRESENTAÇÃO TRANSFORMADA OU PARA A CONVERSÃO INVERSA DA REPRESENTAÇÃO TRANSFORMADA"

Descrição

[0001] A presente invenção refere-se a algoritmos de compressão para valores discretos tendo informações de áudio e/ou imagem e, especialmente, a algoritmos de transformação que são especiais para uso em codificadores baseados em transformação, o que significa que realizam quantização/codificação não dos sinais originais de áudio e/ou imagem, mas compreendem a transformação em uma faixa espectral antes da quantização/codificação.

[0002] Os métodos modernos de codificação de áudio, tais como o MPEG Camada 3 (MP3) ou MPEG AAC [Advanced Audio Coding - Codificação Avançada de Áudio], usam transformações, tais como a assim chamada Transformação Discreta de Co-Seno Modificada (MDCT - Modified Discrete Cosine Transformation) para obter uma representação de frequência em blocos de um sinal de áudio. Normalmente, um codificador de áudio desses recebe um fluxo de amostras de áudio discretas no tempo. O fluxo de amostras de áudio é janelado para se obter um bloco de janela de, por exemplo, 1024 ou 2048 amostras de áudio janeladas. Para o processo de janelamento, são usadas diferentes funções de janelas, tais como a janela de seno, etc.

[0003] As amostras de áudio discretas no tempo janeladas são em seguida convertidas em uma representação espectral através de um banco de filtros. Em princípio, uma

transformação de Fourier ou, por razões específicas, uma variação da transformação de Fourier, tais como a FFT ou, como explicado acima, a MDCT, podem ser usadas. O bloco de valores espectrais de áudio na saída do banco de filtros pode em seguida ser ainda processado, caso necessário. Nos codificadores de áudio acima mencionados, a quantização dos valores espectrais do áudio se segue, os níveis de quantização sendo tipicamente escolhidos tal que o ruído de quantização introduzido por ela fique abaixo do limiar de mascaramento psicoacústico, isto é, seja "removido por mascaramento". A quantização é uma codificação com perdas. Para se obter uma redução adicional da quantidade de dados, os valores espectrais quantizados são em seguida codificados por entropia, por exemplo, através da codificação de Huffman. Através do acréscimo de informações laterais, tais como fatores de escala, etc., um multiplexador de fluxo de bits forma um fluxo de bits a partir dos valores espectrais quantizados codificados por entropia, os quais podem ser armazenados ou transmitidos.

[0004] No decodificador de áudio, o fluxo de bits é dividido em valores espectrais quantizados codificados e informações laterais através de um demultiplexador de fluxo de bits. Os valores espectrais quantizados codificados por entropia são primeiro decodificados por entropia para se obter os valores espectrais quantizados. Os valores espectrais quantizados são em seguida inversamente quantizados para se obter os valores espectrais decodificados, os quais possuem ruído de quantização, o qual

está abaixo do limiar de mascaramento psicoacústico e, assim, será inaudível. Esses valores espectrais serão em seguida convertidos em uma representação de tempo por meio de um banco de filtros de síntese para se obter amostras de áudio decodificado discretas no tempo. Um algoritmo de transformação inverso ao algoritmo de transformação tem de ser usado no banco de filtros de síntese. Além disso, o janelamento tem de ser cancelado após a transformação inversa de tempo da frequência.

[0005] Para se obter uma boa seletividade da frequência, os codificadores de áudio modernos tipicamente usam a sobreposição de blocos. Um caso desses é ilustrado na Fig. 12a. Primeiro, por exemplo, 2048 amostras de áudio discretas no tempo são tomadas e janeladas através de um meio 402. O meio de representação em janelas 402 possui um comprimento de janela de $2N$ amostras e fornece um bloco de $2N$ amostras janeladas no lado de saída. De modo a se obter a superposição de janelas, um segundo bloco de $2N$ amostras é formado através de um meio 404, o qual é ilustrado na Fig. 12a, meramente por razões de clareza, separadamente do meio 402. No entanto, as 2048 amostras fornecidas ao meio 404 são não as amostras de áudio discretas no tempo imediatamente adjacentes à primeira janela, mas sim compreendem a segunda metade das amostras janeladas pelo meio 402 e compreendem adicionalmente meramente 1024 “novas” amostras. A sobreposição é simbolicamente ilustrada pelo meio 406 na Fig. 12a, o qual efetua um grau de sobreposição de 50%. Tanto a saída das $2N$ amostras janeladas através do meio 402 como a

saída das $2N$ amostras janeladas pelo meio 404 são então sujeitas ao algoritmo MDCT através dos meios 408 e 410, respectivamente. De acordo com o algoritmo MDCT conhecido, o meio 408 fornece N valores espectrais para a primeira janela, enquanto o meio 410 também fornece N valores espectrais, mas para a segunda janela, onde existe uma sobreposição de 50% entre a primeira e a segunda janelas.

[0006] Conforme ilustrado na Fig. 12b, no decodificador os N valores espectrais da primeira janela são fornecidos ao meio 412, o qual realiza uma transformação do co-seno discreta modificada inversa. O mesmo se aplica aos N valores espectrais da segunda janela. Esses são fornecidos ao meio 414, o qual também realiza uma transformação do co-seno discreta modificada inversa. Tanto o meio 412 como o meio 414 fornecem $2N$ amostras para a primeira janela e $2N$ amostras para a segunda janela, respectivamente.

[0007] No meio 416, que é designado como TDAC (Time Domain Aliasing Cancelation - Cancelamento por Suavização no Domínio do Tempo) na Fig. 12b, o fato de que as duas janelas estão sobrepostas é considerado. Particularmente, uma amostra y_1 da segunda metade da primeira janela, o que quer dizer que tem um índice $N+k$, é somada com uma amostra y_2 da primeira metade da segunda janela, o que quer dizer que tem um índice k , de modo que N amostras decodificadas no tempo são produzidas no lado de saída, o que quer dizer no decodificador.

[0008] Deve-se notar que através da função do meio 416, que também é chamada de função de adição, o janelamento

realizado no codificador ilustrado esquematicamente na Fig. 12a é levado em consideração automaticamente, de modo que nenhum "janelamento inverso" tem de ser realizado no decodificador ilustrado na Fig. 12b.

[0009] Quando a função de janela implementada pelos meios 402 ou 404 é chamada de $w(k)$, onde o índice k representa o índice do tempo, a condição tem de ser cumprida tal que o peso da janela $w(k)$ ao quadrado somado ao peso da janela $w(N+k)$ ao quadrado resulte em 1, onde k varia de 0 a $N-1$. Caso seja usada uma janela de seno, os pesos da janela da qual se segue a primeira meia-onda da função seno, essa condição é sempre preenchida, visto que o quadrado do seno e o quadrado do co-seno somados, para todos os ângulos, resulta no valor 1.

[00010] Uma desvantagem do método de janelamento com função MDCT subsequente, que é descrita na Fig. 12a, é que o janelamento é obtido pela multiplicação de um valor de amostra discreto no tempo, ao se considerar uma janela de seno, com ponto flutuante, visto que o seno de um ângulo entre 0 e 180 graus não resulta em um inteiro, exceto no ângulo de 90 graus. Mesmo quando amostras discretas inteiras no tempo são janeladas, resultam números em ponto flutuante após o janelamento.

[00011] Dessa forma, mesmo quando nenhum codificador psicoacústico é usado, o que significa que uma codificação sem perdas precisa ser obtida, é necessária quantização na saída dos meios 408 e 410, respectivamente, de modo a ser

possível a realização de uma codificação por entropia razoavelmente gerenciável.

[00012] Geralmente, as transformações inteiras atualmente conhecidas para codificação sem perda de áudio e/ou vídeo são obtidas através da separação das transformações ali usadas em rotações de Givens e pela aplicação do esquema de elevação a cada rotação de Givens. Dessa forma, um erro de arredondamento é introduzido em cada etapa. Para estágios subseqüentes das rotações de Givens, o erro de arredondamento continua a se acumular. O erro de aproximação resultante torna-se especialmente problemático para abordagens de codificador de áudio sem perdas, especialmente quando são usadas transformações longas, as quais fornecem, por exemplo, 1.024 valores espectrais, tal como ocorre, por exemplo, no caso da MDCT (Modified Discrete Cosine Transformation - Transformação do Co-Seno Discreta Modificada) conhecida com sobreposição e adição. Particularmente na faixa de alta freqüência, onde de qualquer forma o sinal de áudio tipicamente tem uma pequena quantidade de energia, o erro de aproximação pode rapidamente se tornar maior do que o próprio sinal, de modo que essa abordagem é problemática com relação à codificação sem perdas e especialmente com relação à eficiência da codificação obtida através dela.

[00013] Com relação à codificação de áudio, transformações inteiras, o que quer dizer algoritmos de transformação que geram valores de saída inteiros, são particularmente baseados no conhecido DCT-IV, que não leva em

consideração nenhum componente constante, enquanto transformações inteiras para aplicações de imagem são baseadas, em vez disso, no DCT-II, o qual particularmente contém as disposições para o componente constante. Essas transformações inteiras são descritas, por exemplo, em Y. Zeng, G. Bi e Z. Lin, "Integer sinusoidal transforms based on lifting factorization", em Proc. ICASSP'01, maio de 2001, pp. 1.181 - 1.184, K. Komatsu e K. Sezaki, "Reversible Discrete Cosine Transform", em Proc. ICASSP, 1998. Vol. 3, pp. 1.769 - 1.772, P. Hao e Q. Shi, "Matrix factorizations for reversible integer mapping", IEEE Trans. Signal Processing, Signal Processing, Vol. 49, pp. 2.314 - 2.324, e J. Wang, J. Sun e S. Yu, "1-d and 2-d transforms from integers to integers", em Proc. ICASSP'03, Hong Kong, abril de 2003.

[00014] Como foi explicado acima, as transformações inteiras lá descritas se baseiam na separação da transformação em rotações de Givens e na aplicação do esquema de elevação conhecido às rotações de Givens, o que envolve o problema da acumulação de erros de arredondamento. Isso se deve particularmente ao fato de que o arredondamento tem de ser realizado várias vezes dentro de uma transformação, o que significa após cada etapa de elevação, de modo que especialmente em transformações longas, as quais envolvem por consequência muitas etapas de elevação, o arredondamento tem de ser realizado muito freqüentemente. Como foi explicado, isso resulta em um erro acumulado e particularmente em um processamento relativamente dispendioso, visto que o

arredondamento é realizado após cada uma das etapas de elevação para realizar a próxima etapa de elevação.

[00015] O objeto da presente invenção é prover um conceito mais eficiente e exato para a conversão de valores discretos em uma representação transformada e para uma conversão inversa correspondente da representação transformada.

[00016] Esse objeto é alcançado através de um aparato de conversão de acordo com a reivindicação 1, um método de conversão de acordo com a reivindicação 17, um aparato para a conversão inversa de acordo com a reivindicação 19, um método para a conversão inversa de acordo com a reivindicação 21 e um programa de computador de acordo com a reivindicação 24.

[00017] Um objeto adicional da presente invenção é prover um aparato de transformação para frente assim como um aparato de transformação para trás, os quais compreendam o conceito inventivo para conversão e o conceito inventivo para conversão inversa, respectivamente.

[00018] Esse objeto é alcançado através do aparato de transformação para frente de acordo com a reivindicação 16, um método para transformação para frente de acordo com a reivindicação 18, um aparato de transformação para trás de acordo com a reivindicação 22 ou um método para transformação para trás de acordo com a reivindicação 23 ou um programa de computador de acordo com a reivindicação 24.

[00019] A presente invenção se baseia no conhecimento de que através da extensão do esquema de elevação unidimensional para um esquema de elevação multidimensional,

assim como através da execução em cascata de pelo menos duas dessas etapas de elevação, por um lado, o erro de arredondamento é reduzido e, por outro lado, a eficiência da computação é aumentada. Por conseguinte, de acordo com a invenção, pelo menos dois blocos de valores compreendendo informações de áudio e/ou imagem são necessários, os quais são, cada um deles, submetidos a um algoritmo de transformação arbitrário. De acordo com a invenção, o arredondamento é realizado apenas após a realização do algoritmo completo de transformação. Isso significa que primeiro, por exemplo, os valores de resultado de saída, tais como os valores espectrais, da 1ª transformação são arredondados e os resultados de saída da 2ª transformação são arredondados, respectivamente. Assim, não é necessário um arredondamento dentro da transformação. Assim, qualquer regra de transformação arbitrária existente e particularmente qualquer já testada, na forma de, por exemplo, um código de programa, o qual existe e esteja livremente disponível, pode ser usado como a 1ª e a 2ª regras de transformação, sem ter de separar a própria transformação em rotações individuais como na técnica anterior, através da interferência com a transformação efetiva.

[00020] De acordo com a invenção, o segundo bloco é usado como portador para a representação transformada do primeiro bloco, através da soma do segundo bloco com a representação transformada do primeiro bloco. Além disso, de acordo com a invenção, o primeiro bloco é usado como portador para uma representação transformada do segundo bloco através

da subtração da representação transformada do segundo bloco (a qual já contém os valores espectrais do primeiro bloco) a partir do primeiro bloco.

[00021] Visto que, de acordo com a invenção, o arredondamento é realizado meramente após a 1ª e a 2ª transformações, respectivamente, e visto que os valores arredondados são aditivamente e subtrativamente sobrepostos, respectivamente, aos valores de portador correspondentes, pode-se obter um cancelamento das etapas realizadas na conversão sem perda de dados na conversão inversa, de modo a gerar um algoritmo de transformação inteiro, o qual pode, por um lado, ser implementado de maneira eficiente do ponto de vista computacional e no qual, por outro lado, não ocorra nenhuma acumulação de erros de arredondamento. Isso se deve ao fato de que o arredondamento é realizado apenas após uma 1ª e 2ª transformações completas, o que, por um lado, elimina a acumulação de erros de arredondamento e, por outro lado, reduz significativamente o número de etapas de arredondamento, em comparação com o caso em que a própria transformação é separada em rotações, em que o arredondamento é realizado após cada uma das próprias etapas de elevação, dentro do próprio algoritmo de transformação.

[00022] Uma vantagem da presente invenção é que eventuais regras de transformação não inteiras convencionais podem ser usadas, visto que o arredondamento tem de ser realização apenas após a transformação.

[00023] Além disso, a presente invenção é vantajosa, pois podem ser usados algoritmos rápidos, cujo esforço 0 não

suba de acordo com N^2 , mas meramente suba a $N \log N$. Isso é especialmente significativo para sinais de áudio, visto que o valor N , que significa o comprimento da transformação, é relativamente grande em sinais de áudio, e é cerca de 1.024 valores nos métodos de compressão de áudio conhecidos descritos acima.

[00024] Em uma configuração preferida da presente invenção, na qual é usada uma versão inteira da transformação MDCT de ponto flutuante conhecida, a transformação DCT-IV conhecida é a regra de transformação, a qual compreende uma matriz de transformação idêntica como regra de transformação para a 1ª e 2ª transformações. Além disso, de modo a obter uma transformação completamente inteira, a qual corresponde à MDCT com sobreposição e adição, é preferível combinar o janelamento da MDCT comum com a funcionalidade de cancelamento de suavização no domínio do tempo (funcionalidade TDAC) e expressá-la através de rotações de Givens, as quais podem novamente ser calculadas de uma maneira inteira por um esquema de elevação, para se obter uma versão totalmente inteira da MDCT.

[00025] Os dois blocos de valores discretos fornecidos ao aparato de conversão inventivo correspondem aos valores rotacionados de Givens dos blocos janelados com sobreposição das amostras de áudio discretas no tempo ou amostras de imagem discretas ou valores residuais de imagem após uma previsão em um algoritmo de compressão de vídeo moderno, onde o algoritmo DCT-IV é preferido como algoritmo de transformação no caso de dados de áudio. No lado do

decodificador, novamente, o algoritmo DCT-IV é preferido como o aparato de transformação frequência-tempo, o qual é novamente seguido de um estágio de rotação, o qual realiza um esquema de elevação inverso para recuperar os N arredondamentos introduzidos no lado do codificador, o que significa os arredondamentos na transformação de tempo-frequência, de uma maneira sem perdas.

[00026] As configurações preferidas da presente invenção serão discutidas abaixo com referência aos desenhos de acompanhamento. Eles mostram:

[00027] Fig. 1: um diagrama em blocos de um aparato inventivo para conversão;

[00028] Fig. 2: um diagrama em blocos de um aparato inventivo para conversão inversa;

[00029] Fig. 3: um diagrama em blocos de um aparato para conversão de acordo com uma configuração preferida da presente invenção;

[00030] Fig. 4: um aparato para conversão inversa de acordo com uma configuração preferida da presente invenção;

[00031] Fig. 5: uma representação da transformação de dois blocos subseqüentes de valores, conforme aplicável para a presente invenção;

[00032] Fig. 6: uma representação detalhada de uma etapa de elevação multidimensional com uma matriz de transformação para frente;

[00033] Fig. 7: uma representação de uma etapa de elevação inversa multidimensional com uma matriz de transformação para trás;

[00034] Fig. 8: uma representação da presente invenção para separar um DCT-IV de comprimento N em dois DCT-IV do comprimento $N/2$;

[00035] Fig. 9: um diagrama em blocos de um meio preferido para o processamento de amostras de áudio discretas no tempo para a obtenção de valores inteiros a partir dos quais valores espectrais inteiros possam ser determinados;

[00036] Fig. 10: uma representação esquemática da separação de uma MDCT e uma MDCT inversa em rotações de Givens e duas operações de DCT-IV;

[00037] Fig. 11: uma representação para ilustração da separação da MDCT com 50 por cento de sobreposição em rotações e operações de DCT-IV;

[00038] Fig. 12a: um diagrama em blocos esquemático de um codificador conhecido com MDCT e 50 por cento de sobreposição; e

[00039] Fig. 12b: um diagrama em blocos de um decodificador conhecido para a decodificação dos valores gerados pela Fig. 10a.

[00040] A Fig. 1 mostra um aparato inventivo para a conversão de valores discretos em uma representação transformada com valores inteiros. Os valores discretos são fornecidos a um aparato inventivo através de uma primeira entrada 100a, assim como de uma segunda entrada 100b. O primeiro bloco de valores discretos é inserido através da entrada 100a, enquanto um segundo bloco de valores discretos é inserido através da entrada 100b. Os valores discretos representam dados de áudio ou dados de imagem e dados de

vídeo, respectivamente. Como será explicado abaixo, o primeiro bloco de valores discretos e o segundo bloco de valores discretos podem de fato compreender dois blocos de amostra de áudio sucessivos no tempo. O primeiro e o segundo blocos de valores discretos também podem compreender duas imagens e valores residuais, respectivamente, representados pelos valores discretos após uma previsão ou valores de diferença em uma codificação de diferença, etc. Como alternativa, os dois blocos de valores discretos podem ter sido sujeitos a pré-processamento, tal como na implementação inteira da MDCT, onde o primeiro bloco e o segundo bloco de valores discretos foram gerados por rotações de Givens de amostras efetivamente janeladas. O primeiro e o segundo blocos de valores discretos podem, assim, ser derivados dos dados originais de áudio ou de imagem através de quaisquer processamentos, tais como rotações, permutações, borboletas para mais/menos, escalonamentos, etc. Todavia, o primeiro e o segundo blocos de valores discretos contêm informações de áudio e informações de imagem, respectivamente, embora não sejam diretamente amostras de áudio ou valores de imagem discretos.

[00041] O primeiro bloco de valores discretos é fornecido ao meio 102 para processamento do primeiro bloco de valores discretos através do uso de uma 1ª regra de transformação através da entrada 100a, para obter um primeiro bloco de valores transformados na saída do meio 102, como é mostrado na Fig. 1. Esse primeiro bloco de valores transformados tipicamente não será inteiro, mas compreenderá

valores de ponto flutuante, visto que eles são tipicamente obtidos por alguma regra de transformação, tal como uma transformação de Fourier, uma transformação de Laplace, uma FFT, uma DCT, uma DST, uma MDCT, uma MDST ou qualquer outra transformação, tal como uma transformação do tipo *wavelet* com funções de base arbitrárias. O primeiro bloco de valores transformados é fornecido ao meio 104 para arredondamento do primeiro bloco de valores transformados no lado da saída. O meio 104 para arredondamento é conformado para realizar qualquer função de arredondamento, tal como o arredondamento por truncamento ou arredondamento para cima ou arredondamento simples, respectivamente, para ser realizado dependendo do valor de ponto flutuante.

[00042] A regra de arredondamento, que é implementada pelo meio 104, é assim responsável pelo fato de que o primeiro bloco de valores transformados arredondados novamente meramente possui valores inteiros, cuja precisão é determinada pela regra de arredondamento usada pelo meio 104. O primeiro bloco de valores transformados arredondados é fornecido ao meio 106 para soma, assim como o segundo bloco de valores discretos aplicado à segunda entrada 100b, para obter um segundo bloco de valores somados. Caso se considere o exemplo do sinal de áudio, pode-se ver que os valores espectrais do primeiro bloco são somados aos valores transformados arredondados do primeiro bloco de valores transformados arredondados que são somados aos valores de tempo do segundo bloco de valores discretos pelo meio 106. Caso os valores discretos do segundo bloco estejam, por

exemplo, presentes como valores de tensão, recomenda-se que o primeiro bloco de valores transformados arredondados esteja também presente como amplitudes de tensão, o que significa valores com a unidade V. Nesse caso, não haverá problemas na soma. No entanto, os técnicos no assunto perceberão que qualquer padronização de unidade pode ser feita com o primeiro bloco de valores transformados arredondados e o segundo bloco de valores discretos, respectivamente, em que tanto o primeiro bloco de valores transformados arredondados e o segundo bloco de valores discretos são, por exemplo, isentos de unidades.

[00043] O segundo bloco de valores somados é fornecido a um meio 108 para processamento do segundo bloco de valores somados através do uso de uma 2ª regra de transformação para se obter um segundo bloco de valores transformados. Caso a regra de transformação usada no meio 102 seja, por exemplo, uma regra de transformação de tempo-frequência, então a 2ª regra de transformação, que é usada no bloco 108, é, por exemplo, uma regra de transformação de tempo-frequência. Essas relações também podem ser invertidas, de modo que o primeiro e segundo blocos de valores discretos sejam, por exemplo, valores espectrais, de modo que os valores de tempo sejam obtidos pelo meio 102 para processamento sob a regra de transformação, enquanto novamente os valores espectrais sejam obtidos pelo meio para processamento através da regra de transformação inversa, a qual é o meio 108. Assim, a 1ª e 2ª regras de transformação podem ser uma regra de transformação para frente ou uma regra

de transformação para trás, onde a regra de transformação inversa é a regra de transformação para trás e a regra de transformação para frente, respectivamente.

[00044] O segundo bloco de valores transformados é fornecido a um meio 110 para arredondamento, conforme mostrado na Fig. 1, para se obter um segundo bloco de valores transformados arredondados, o qual será em seguida fornecido a um meio 112 para subtração, para subtrair o segundo bloco de valores transformados arredondados a partir do primeiro bloco de valores discretos, o qual é introduzido através da primeira entrada 108a, para se obter um bloco de valores inteiros de saída da representação transformada, o qual pode ser emitido por uma saída 114. Através do processamento do bloco de valores de saída inteiros da representação transformada através do uso de uma terceira regra de transformação arbitrária, a qual também tenha sido usada no meio 102 ou difira dele, e através do subsequente arredondamento do bloco de valores de saída transformados para se obter um bloco de valores de saída transformados arredondados e através da subsequente soma do bloco de valores de saída transformados arredondados com o segundo bloco de valores somados, pode-se obter um bloco adicional de valores de saída inteiros da representação transformada, o que fornece uma representação transformada completa do primeiro e segundo blocos de valores discretos com o bloco de valores de saída inteiros aplicado à saída 114.

[00045] Mas mesmo sem as últimas três etapas de processamento, arredondamento e soma, em que o bloco de

valores de saída inteiros da representação transformada é aplicado à saída 114, parte de toda a representação transformada pode já ser obtida, a saber, por exemplo, a primeira metade, a qual permite um recálculo do primeiro e do segundo blocos de valores discretos, quando sujeito ao processamento inverso.

[00046] Aqui se deve observar que, dependendo da regra de transformação, a 1ª, a 2ª e, se necessário, a 3ª regras de transformação podem ser idênticas. Esse é, por exemplo, o caso com a DCT-IV. Caso uma FFT tenha sido usada como 1ª regra de transformação, a IFFT, que não é idêntica à FFT, poderia ser usada como uma segunda regra de transformação (inversa).

[00047] Por motivos computacionais é preferível fornecer a regra de transformação na forma de uma matriz, a qual é então uma matriz quadrada de $N \times N$ quando o número de valores discretos do primeiro bloco é igual ao número de valores discretos dos segundo bloco, quando o número de valores discretos do primeiro bloco e o número de valores discretos do segundo bloco são ambos iguais a N .

[00048] Em uma configuração preferida da presente invenção, os meios 104 e 110 para o arredondamento são formados para arredondar, de acordo com uma função de arredondamento que fornece resultados arredondados, a precisão que é menos do que uma precisão de máquina inerente ao computador que executa as funcionalidades mostradas na Fig. 1. De acordo com a função de arredondamento, deve-se observar que ela mapeia um número não inteiro para o próximo

inteiro mais alto ou mais baixo, meramente em uma configuração preferida. A função de arredondamento também pode mapear para outros inteiros, tal como o número 17,7 para o número 10 ou o número 20, desde que a função de arredondamento efetue uma diminuição da precisão do número a ser arredondado. No exemplo acima, o número não arredondado é um número com um dígito após a vírgula decimal, enquanto o número arredondado é um número sem dígito após a vírgula decimal.

[00049] Embora o meio 102 para processamento através do uso da 1ª regra de transformação e do meio 108 para processamento através do uso da 2ª regra de transformação sejam mostrados como meios separados na Fig. 1, deve-se observar que em uma implementação específica, meramente uma unidade de função de transformação pode estar presente, a qual, controlada por uma seqüência específica de controle, primeiro transforme o primeiro bloco de valores discretos e em seguida transforme inversamente o segundo bloco de valores somados no tempo respectivo do algoritmo. Então, a primeira e a segunda regras de transformação seriam idênticas. O mesmo se aplica aos dois meios 104, 110 para arredondamento. Eles também não têm de ser fornecidos como meios separados, mas podem ser implementados por uma unidade de função de arredondamento, a qual, novamente controlada pelo controle de seqüência dependendo dos requisitos do algoritmo, primeiro arredonda o primeiro bloco de valores transformados e em seguida o segundo bloco de valores transformados.

[00050] Em uma configuração preferida da presente invenção, o primeiro bloco de valores discretos e o segundo bloco de valores discretos são as amostras janeladas inteiras, visto que elas são obtidas na saída do bloco 28 na Fig. 9, a qual será discutida abaixo. A DCT inteira no bloco 14 da Fig. 9 é então implementada pelo algoritmo inteiro mostrado na Fig. 1, tal que a representação transformada represente o valor espectral inteiro na saída 30 do aparato mostrado na Fig. 9, em um exemplo do sinal de áudio relativo à Fig. 9.

[00051] A seguir, com relação à Fig. 2, será ilustrado um meio para transformação inversa correspondendo à Fig. 1, onde fora o bloco de valores de saída inteiros na saída do bloco 112 da Fig. 1, também o segundo bloco de valores somados é usado na saída do meio 106 da Fig. 1. Com referência à Fig. 4, a qual será discutida em mais detalhes abaixo, isso corresponde ao caso em que meramente os blocos 150 e 130 estão presentes, mas não o bloco de transformação 124.

[00052] A Fig. 2 mostra um aparato inventivo para a conversão inversa de um bloco de valores de saída inteiros da representação transformada, como ele será obtido na saída 114 da Fig. 1, e o segundo bloco de valores somados. O 2º bloco de valores somados é inserido em uma entrada 120 do aparato para conversão inversa, mostrado na Fig. 2. O bloco de valores de saída da representação transformada é inserido em uma entrada adicional 122 do aparato para conversão inversa.

[00053] O 2º bloco de valores somados é inserido em um meio 130 para processamento desse bloco através do uso da 2ª regra de transformação, quando a última regra de transformação usada durante a codificação for a 2ª regra de transformação. No lado da saída, o meio 130 fornece um primeiro bloco de valores transformados, o qual é fornecido a um meio 132 para arredondamento, o qual novamente gera um primeiro bloco de valores transformados arredondados no lado da saída. O primeiro bloco de valores transformados arredondados é em seguida subtraído a partir do bloco de valores de saída da representação transformada por um meio 134 de modo a obter o primeiro bloco de valores discretos em uma primeira saída 149 do aparato da Fig. 2.

[00054] Esse primeiro bloco de valores discretos é fornecido a um meio 150 para o processamento desse bloco através do uso da 1ª regra de transformação para obter um segundo bloco de valores transformados na saída do meio 150. Esse segundo bloco de valores subtraídos transformados é novamente arredondado no meio 152 para se obter um segundo bloco de valores transformados arredondados. Esse segundo bloco de valores transformados arredondados é em seguida subtraído a partir do 2º bloco de valores somados fornecido no lado da entrada, o qual foi inserido através da entrada 120, para se obter um segundo bloco de valores discretos em uma saída 136 no lado de saída.

[00055] Com respeito à relação de 1ª, 2ª e 3ª regras de transformação, assim como com respeito à implementação específica dos blocos de funções individuais na Fig. 2 por

unidades comuns de função e uma armazenagem de controle/temporária de seqüência correspondente, faz-se referência às explicações feitas com relação à Fig. 1.

[00056] A seguir, com referência à Fig. 3, será discutida uma configuração preferida do aparato para conversão em uma representação transformada geralmente ilustrada na Fig. 1. A configuração na Fig. 1 compreende uma transformação/arredondamento adicional comparada com a Fig. 1, de modo a gerar o bloco adicional de valores de saída inteiros do segundo bloco de valores somados.

[00057] A primeira entrada 100a compreende N linhas de entrada $x_0, \dots, x_{N-1}$ para a entrada de N valores do primeiro bloco de valores discretos. A segunda entrada 100b também compreende N linhas para a entrada de N valores $x_N, \dots, x_{2N-1}$ do segundo bloco de valores discretos. O meio 102 da Fig. 1 é mostrado na Fig. 3 como um transformador DCT-IV. O transformador DCT 102 é formado para gerar N valores de saída a partir de N valores de entrada, cada um dos quais será então arredondado pelo meio 104 através da especificação de arredondamento indicada pelo "[.]" conforme mostrado na Fig. 3. O meio 106 para a soma é ilustrado tal que ocorra uma soma por valor. Isso significa que o valor de saída do meio 102 com índice 0 é somado com o primeiro valor do segundo bloco de valores discretos tendo o índice N. Assim, geralmente o valor do primeiro bloco de valores transformados arredondados com um número ordinal i é somado individualmente na saída do meio de arredondamento 104 com o valor discreto

do segundo bloco de valores de saída com um número ordinal $N+i$, onde i é um índice variável que vai de 0 a $N-1$.

[00058] O meio 108 para processamento através do uso da 2ª regra de transformação também é projetado como um transformador DCT-IV. Na configuração preferida mostrada na Fig. 3, o meio 112 para subtração também é formado para realizar uma subtração por valor, na qual os valores de saída do arredondador 110, o que quer dizer os valores do segundo bloco de valores transformados arredondados, são subtraídos individualmente a partir do primeiro bloco de valores discretos. Na configuração mostrada na Fig. 3, é preferido realizar uma subtração correspondente, na qual um valor do segundo bloco é um número ordinal de $N+i$ subtraído de um valor do primeiro bloco com o número ordinal i , onde i novamente varia de 0 a $N-1$. Como alternativa, pode-se realizar outras somas/subtrações nas quais, por exemplo, um valor de um bloco, com o número ordinal $N-1$, seja subtraído do valor do outro bloco, com o número ordinal $N-1$, desde que este seja correspondentemente considerado na conversão inversa.

[00059] No lado de saída, o meio 112 para subtração já fornece um bloco de valores de saída inteiros da representação transformada remanescentes, o que significa os valores de saída inteiros y_0 a y_{N-1} da representação transformada. De modo a obter, caso opcionalmente se deseje, os valores de saída inteiros da representação transformada, o que significa o bloco adicional y_N a y_{2N-1} , o bloco de valores de saída inteiros da representação transformada aplicado à

saída 114 é submetido a uma transformação através do uso da terceira regra de transformação pelo transformador para frente 140, onde os valores de saída dele são novamente arredondados conforme ilustrado por um meio de arredondamento 142 para realizar uma adição desses valores com o segundo bloco de valores somados na saída do somador 106, conforme representado pelo número de referência 144 na Fig. 3. Os valores de saída do somador 144 representam um bloco adicional 146 de valores de saída inteiros da representação transformada, os quais são indicados por y_N a y_{2N-1} .

[00060] A seguir, um aparato inventivo para a conversão inversa da representação transformada de acordo com uma configuração preferida será ilustrado com referência à Fig. 4. Deve-se observar que as operações realizadas pelo aparato na Fig. 3 podem ser invertidas de uma maneira sem perdas pelo aparato ilustrado na Fig. 4. A Fig. 4 corresponde à Fig. 2, exceto pelo estágio adicional de transformação/arredondamento para a geração do segundo bloco de valores de saída transformados, o qual é inserido na entrada 120 na configuração mostrada na Fig. 2. Deve-se observar que a função de adição é respectivamente invertida pela função de subtração. Além disso, deve-se notar que um par adicionador/subtrator (144 da Fig. 3 e 128 da Fig. 4) também pode ser provido com quantidades de entrada invertidas em relação ao sinal, de modo que o adicionador 144 realize na verdade uma operação de subtração, quando um grupo de quantidades de entrada é provido com sinais negativos em comparação com o caso ilustrado, desde que isso seja

considerado na contraparte (128 na Fig. 4), a qual iria então na verdade realizar uma operação de adição.

[00061] O subtrator 128, o adicionador 134, assim como o subtrator adicional 154 mostrados na Fig. 4 são novamente conformados para realizar uma adição/subtração individual por valor, onde novamente o mesmo processamento de números ordinais é usado como foi explicado com relação à Fig. 3. Caso outro uso de números ordinais fosse realizado na Fig. 3, conforme é mostrado, isso seria correspondentemente considerado na Fig. 4.

[00062] O primeiro bloco de valores discretos 136, indicado por x_0 a x_{N-1} , já está presente na saída do subtrator 134. De modo a obter também o restante da representação transformada inversa, o primeiro bloco de valores discretos é fornecido ao transformador 150, o qual opera com a primeira regra de transformação, cujos valores no lado da saída são arredondados pelo arredondador 152 e subtraídos a partir do segundo bloco de valores subtraídos na saída do subtrator 128, de modo a finalmente obter também o segundo bloco de valores discretos 156, indicado por $x_N, \dots, x_{2N-1}$.

[00063] A seguir, com referência às Figs. 5 - 8, será discutida a fundamentação matemática para o aparato inventivo conforme ilustrado pelas Figs. 1 - 4. Através do aparato inventivo ilustrado para conversão e para conversão inversa, respectivamente, são providos os métodos para transformação de inteiros para codificação de áudio sem perdas, onde o erro de aproximação é reduzido. Acima disso, o esforço de cálculo também é considerado em que a abordagem conhecida de

aplicação do esquema de elevação a cada rotação de Givens não é mais usada, onde sempre ocorrem borboletas triviais de soma-diferença. Elas aumentam o esforço de cálculo significativamente em comparação com a versão não inteira original da transformação a ser reproduzida.

[00064] Normalmente, o esquema de elevação é usado para se obter uma aproximação de inteiro invertível de uma rotação de Givens.

$$\begin{pmatrix} \cos \alpha & -\sin \alpha \\ \sin \alpha & \cos \alpha \end{pmatrix} \cong \begin{pmatrix} 1 & \frac{\cos \alpha - 1}{\sin \alpha} \\ 0 & 1 \end{pmatrix} \begin{pmatrix} 1 & 0 \\ \sin \alpha & 1 \end{pmatrix} \begin{pmatrix} 1 & \frac{\cos \alpha - 1}{\sin \alpha} \\ 0 & 1 \end{pmatrix}$$

[00065] Essa aproximação de inteiro é obtida através do uso de uma função de arredondamento após cada uma das adições, o que significa após cada etapa de elevação.

[00066] O esquema de elevação também pode ser usado para uma aproximação de inteiro invertível de certas operações de escalonamento. Na publicação especializada R. Geiger e G. Schüller, "Integer low delay and MDCT filter banks" Proc. of the Asilomar Conf. On Signals, Systems and Computers, 2002, a seguinte separação por elevação de uma matriz de escalonamento de 2 x 2 com determinante igual a 1 é ilustrada e descrita:

$$\begin{pmatrix} d & 0 \\ 0 & d^{-1} \end{pmatrix} = \begin{pmatrix} -1 & 0 \\ d^{-1} & 1 \end{pmatrix} \begin{pmatrix} 1-d & 0 \\ 0 & 1 \end{pmatrix} \begin{pmatrix} 0 & 1 \\ 1 & d^{-1} \end{pmatrix}$$

[00067] De acordo com a invenção, essa separação por elevação, que é unidimensional, o que significa que ela se refere meramente a uma matriz de escalonamento de 2 x 2, é estendida para o caso multidimensional. Individualmente,

todos os valores da equação anterior são substituídos por matrizes de $n \times n$, onde n , que significa o número de valores discretos de um bloco, é maior do que ou igual a 2. Assim, o resultado é que para qualquer matriz T de $n \times n$, que seja preferivelmente invertível, é possível a seguinte separação em matrizes de bloco de $2n \times 2n$, onde E_n descreve a matriz identidade $n \times n$:

$$\begin{pmatrix} T & 0 \\ 0 & T^{-1} \end{pmatrix} = \begin{pmatrix} -E_n & 0 \\ T^{-1} & E_n \end{pmatrix} \begin{pmatrix} E_n & -T \\ 0 & E_n \end{pmatrix} \begin{pmatrix} 0 & E_n \\ E_n & T^{-1} \end{pmatrix}$$

[00068] Fora operações simples, tais como permutações ou multiplicações por -1 , todos os três blocos dessa separação possuem a seguinte estrutura geral:

$$\begin{pmatrix} E_n & 0 \\ A & E_n \end{pmatrix}$$

[00069] Para essa matriz em bloco de $2n \times 2n$, pode-se usar um esquema de elevação inventivamente generalizado, o qual também é subsequentelemente referenciado como uma elevação multidimensional.

[00070] Para um vetor dos valores $x = (x_0, \dots, x_{2n-1})$, a aplicação dessa matriz em bloco fornece a seguinte equação:

$$\begin{pmatrix} E_n & 0 \\ A & E_n \end{pmatrix} (x_0, \dots, x_{2n-1}) = ((x_0, \dots, x_{n-1}), (x_n, \dots, x_{2n-1}) + A \cdot (x_0, \dots, x_{n-1}))$$

[00071] Deve-se observar que há um vetor no lado direito da equação anterior, cuja dimensão, que significa o número de linhas, é igual a $2n$. Os primeiros n componentes, que significam os componentes de 0 a $n-1$, correspondem a x_0 a y_{n-1} . Os segundos n componentes, que significa a segunda

metade do vetor, o que resulta no lado direito da equação anterior, são iguais à soma do segundo bloco de valores discretos, o que significa $x_n, \dots, x_{2n-1}$, mas agora somados com a multiplicação da matriz A, o que corresponde à matriz de transformação das Figs. 1 - 4 e o primeiro bloco de valores discretos $x_0, \dots, x_{n-1}$. A matriz de transformação representa a primeira, a segunda e a terceira regras de transformação, respectivamente.

[00072] Similar ao esquema de elevação comum com matrizes de 2×2 na forma:

$$\begin{pmatrix} 1 & a \\ 0 & 1 \end{pmatrix}$$

essas duas matrizes de $2n \times 2n$ podem ser usadas para aproximações inteiras invertíveis da transformação T como segue. Para valores de entrada inteiros $(x_0, \dots, x_{2n-1})$, os valores de saída de ponto flutuante $(y_0, \dots, y_{n-1}) = A \cdot (x_0, \dots, x_{n-1})$ são arredondados para valores inteiros, antes de serem adicionados a valores inteiros $(x_n, \dots, x_{2n-1})$. O inverso da matriz em bloco resulta da seguinte maneira:

$$\begin{pmatrix} E_n & 0 \\ A & E_n \end{pmatrix}^{-1} = \begin{pmatrix} E_n & 0 \\ -A & E_n \end{pmatrix}$$

[00073] Assim, esse processo pode ser invertido sem erros através do simples uso da mesma matriz A e da mesma função de arredondamento e agora pela subtração dos valores resultantes ao invés da adição no processamento para frente. O processamento para frente é ilustrado na Fig. 6, enquanto o processamento para trás é ilustrado na Fig. 7. Deve-se

observar que a matriz de transformação na Fig. 6 é idêntica à matriz de transformação na Fig. 7, o que é preferido por razões de simplicidade da implementação.

[00074] Visto que os valores $(x_0, \dots, x_{n-1})$ não são modificados na etapa para frente mostrada na Fig. 6, eles ainda estão presentes para a etapa inversa, o que significa a etapa para trás na Fig. 7. Deve-se observar que não há restrições específicas para a matriz A. Assim, ela não necessariamente tem de ser invertível.

[00075] De modo a obter uma aproximação inteira invertível da MDCT conhecida, a MDCT é separada em rotações de Givens em um primeiro estágio, onde esse estágio é o estágio de janelamento, e em um subsequente estágio de DCT-IV. Essa separação é ilustrada na Fig. 10, a qual será discutida abaixo e descrita em detalhes em DE 10129240 A1.

[00076] Em oposição à técnica anterior, onde a DCT-IV é separada em vários estágios de rotações de Givens, de acordo com a invenção, a própria transformação é deixada e em seguida arredondada.

[00077] Assim, como é sabido, a aproximação inteira da DCT-IV é realizada por vários estágios de rotações de Givens baseados em elevação. O número de rotações de Givens é determinado pelo algoritmo rápido subjacente usado. Dessa maneira o número de rotações de Givens é dado por $O(N \log N)$ para uma transformação do comprimento N. O estágio de janelamento de cada separação de MDCT consiste apenas em $N/2$ rotações de Givens ou de $3N/2$ etapas de arredondamento. Assim, particularmente em altos comprimentos de

transformação, como os usados em aplicações de codificação de áudio (por exemplo, 1.024), a aproximação inteira da DCT-IV fornece a principal contribuição para o erro de aproximação.

[00078] A abordagem inventiva usa o esquema de elevação multidimensional descrito. Por conseguinte, o número de etapas de arredondamento na DCT_{IV} é reduzido para $3N/2$, o que significa que foi tornado igual ao número de etapas de arredondamento no estágio de janelamento, comparado com cerca de $2N \log_2 N$ etapas de arredondamento na abordagem convencional baseada em elevação.

[00079] De acordo com a invenção, a DCT-IV é aplicada simultaneamente a dois blocos de sinais. Por conseguinte, uma possibilidade é mostrada na Fig. 5, onde simplesmente, por exemplo, dois blocos de amostras subseqüentes no tempo são sujeitos a uma DCT-IV. Os dois blocos, os quais são sujeitos às duas transformações, também podem ser amostras de dois canais de um sinal multicanal.

[00080] A separação da equação de elevação multidimensional acima descrita é aplicada à regra de transformação, a qual também pode ser considerada como uma matriz $N \times N$. Visto que particularmente na DCT-IV o inverso é novamente a DCT-IV, resulta a seguinte separação para o conceito mostrado na Fig. 5:

$$\begin{pmatrix} \text{DCT}_{IV} & 0 \\ 0 & \text{DCT}_{IV} \end{pmatrix} = \begin{pmatrix} -E_N & 0 \\ \text{DCT}_{IV} & E_N \end{pmatrix} \begin{pmatrix} E_N & -\text{DCT}_{IV} \\ 0 & E_N \end{pmatrix} \begin{pmatrix} 0 & E_N \\ E_N & \text{DCT}_{IV} \end{pmatrix}$$

[00081] As permutações das multiplicações por -1 podem ser extraídas em matrizes em bloco individuais, resultando o seguinte contexto:

$$\begin{pmatrix} \text{DCT}_{IV} & 0 \\ 0 & \text{DCT}_{IV} \end{pmatrix} = \begin{pmatrix} -E_N & 0 \\ 0 & E_N \end{pmatrix} \begin{pmatrix} E_N & 0 \\ \text{DCT}_{IV} & E_N \end{pmatrix} \begin{pmatrix} E_N - \text{DCT}_{IV} & 0 \\ 0 & E_N \end{pmatrix} \begin{pmatrix} E_N & 0 \\ \text{DCT}_{IV} & E_N \end{pmatrix} \begin{pmatrix} 0 & E_N \\ E_N & 0 \end{pmatrix}$$

[00082] Assim, a aplicação de uma transformação a dois blocos de sinais, o que significa dois blocos de valores discretos, pode ser obtida com preferivelmente três etapas de elevação multidimensional:

$$\begin{pmatrix} E_N & 0 \\ \text{DCT}_{IV} & E_N \end{pmatrix} \begin{pmatrix} E_N - \text{DCT}_{IV} & 0 \\ 0 & E_N \end{pmatrix} \begin{pmatrix} E_N & 0 \\ \text{DCT}_{IV} & E_N \end{pmatrix}$$

[00083] A equação acima é graficamente ilustrada na Fig. 3 com relação a uma configuração preferida da presente invenção. A conversão inversa é, como foi explicado, ilustrada de maneira correspondente na Fig. 4.

[00084] Com a abordagem inventiva, duas transformações DCT-IV de comprimento N podem ser implementadas de uma maneira invertível, onde apenas 3N etapas de arredondamento são necessárias, o que significa 3N/2 etapas de arredondamento por transformação.

[00085] A DCT-IV nas três etapas de elevação multidimensionais pode ter uma implementação arbitrária, o que significa, por exemplo, uma implementação baseada em ponto flutuante ou em ponto fixo. Ela nem precisa ser invertível. Ela apenas tem de ser executada exatamente da mesma maneira nos processos para frente e para trás. Como resultado, esse conceito é adequado a altos comprimentos de transformação, tais como, por exemplo, 1.024, como eles são usados nas aplicações atuais de codificação de áudio.

[00086] Toda a complexidade de computação é igual a 1,5 vezes a complexidade de computação da implementação não

inteira das duas transformações DCT-IV. Essa complexidade de computação é ainda significativamente inferior às implementações inteiras convencionais baseadas em elevação, as quais são cerca de duas vezes mais complexas do que a DCT-IV convencional, visto que essas implementações têm de usar as triviais borboletas para mais/menos baseadas no esquema de elevação usado, de modo a obter conservação de energia, conforme é descrito em R. Geiger, T. Sporer, J. Koller e K. Brandenburg, "Audio Coding based on Integer Transforms" em 111th AES Convention, New York, 2001.

[00087] A abordagem ilustrada calculará pelo menos duas transformações DCT-IV simultaneamente, o que significa dentro de uma conversão. Isso pode, por exemplo, ser alcançado através do cálculo da transformação DCT-IV para dois blocos subseqüentes do sinal de áudio ou duas imagens subseqüentes de um sinal de imagem. No caso de um sinal estéreo de dois canais, isso também pode ser obtido através do cálculo da DCT-IV dos canais esquerdo e direito em uma ação de conversão e uma ação de conversão inversa, respectivamente. A primeira versão introduz um atraso adicional de um bloco no sistema. A segunda versão é possível para canais estéreos e geralmente para sinais multicanal, respectivamente.

[00088] Como alternativa, caso ambas as opções não sejam desejáveis, mas caso um comprimento de processamento de bloco normal de N valores deva ser mantido, a DCT-IV de comprimento N pode ser separada em duas transformações DCT-IV de comprimento $N/2$. Nesse contexto, faz-se referência a Y.

Zeng, G. Bi e Z. Lin, "Integer sinusoidal transforms based on lifting factorization", em Proc. OCASSP'01, maio de 2001, pp. 1.181 - 1.184, onde essa separação é realizada. Além das duas transformações DCT de comprimento $N/2$, são necessários vários estágios adicionais de rotações de Givens. Além disso, nesse algoritmo, é usada uma matriz em bloco

$$\begin{pmatrix} E_{N/2} & -E_{N/2} \\ E_{N/2} & E_{N/2} \end{pmatrix},$$

isto é, $N/2$ borboletas para mais/menos, uma matriz diagonal em bloco com $N/2$ rotações de Givens e, além disso, várias matrizes de permutação. Através do uso desses estágios adicionais de $N/2$ rotações de Givens, a abordagem de elevação multidimensional também pode ser usada para o cálculo de apenas uma DCT-IV de comprimento N . A estrutura fundamental desse algoritmo é ilustrada na Fig. 8, onde, fora o próprio estágio de conversão, onde são usadas duas transformações DCT-IV com o comprimento $N/2$, primeiro um estágio de borboleta está presente para calcular o primeiro e o segundo blocos de valores discretos, os quais normalmente têm um comprimento de $N/2$. No lado da saída, é provido um estágio de rotação para se obter os valores de saída $y_0, \dots, y_{N-1}$, os quais são iguais aos valores de saída de uma operação DCT-IV da Fig. 5, como pode ser visto a partir de uma comparação dos índices do lado de entrada e do lado de saída das Figs. 8 e 5, a partir do bloco de valores de saída da representação transformada e do bloco adicional de valores de saída da representação transformada, os quais agora têm cada um meramente $N/2$ valores.

[00089] Até agora, foi ilustrada meramente a aplicação da elevação multidimensional a matrizes em blocos da seguinte forma:

$$\begin{pmatrix} T & 0 \\ 0 & T^{-1} \end{pmatrix}$$

[00090] Todavia, também é possível separar outras matrizes em bloco em etapas de elevação multidimensionais. Por exemplo, a separação a seguir pode ser usada para implementar a combinação de um estágio com borboletas para mais/menos normalizadas e dois blocos de transformações DCT-IV através de três etapas de elevação multidimensional:

$$\frac{1}{\sqrt{2}} \begin{pmatrix} E_N & E_N \\ -E_N & E_N \end{pmatrix} \begin{pmatrix} \text{DCT}_{IV} & 0 \\ 0 & \text{DCT}_{IV} \end{pmatrix} = \begin{pmatrix} \frac{1}{\sqrt{2}} \text{DCT}_{IV} & \frac{1}{\sqrt{2}} \text{DCT}_{IV} \\ -\frac{1}{\sqrt{2}} \text{DCT}_{IV} & \frac{1}{\sqrt{2}} \text{DCT}_{IV} \end{pmatrix} =$$

$$\begin{pmatrix} E_N & 0 \\ E_N & -\sqrt{2} \text{DCT}_{IV} E_N \end{pmatrix} \begin{pmatrix} E_N & \frac{1}{\sqrt{2}} \text{DCT}_{IV} \\ 0 & E_N \end{pmatrix} \begin{pmatrix} E_N & 0 \\ E_N & -\sqrt{2} \text{DCT}_{IV} E_N \end{pmatrix}$$

[00091] Torna-se óbvio a partir da equação anterior que a primeira regra de transformação usada no colchete esquerdo da equação anterior e a segunda regra de transformação usada no colchete intermediário da equação anterior e a terceira regra de transformação usada no último colchete da equação anterior não têm de ser idênticas. Além disso, torna-se óbvio da equação anterior que podem ser separadas não apenas matrizes em bloco em que meramente os elementos da diagonal principal estão ocupados, mas que também matrizes plenamente ocupadas podem ser processadas de acordo com a invenção. Além disso, deve-se observar que a

presente invenção não se limita ao fato de que as regras de transformação, que são usadas na conversão em uma representação transformada, têm de ser idênticas ou precisam ter qualquer relação umas com as outras, tal que, por exemplo, a segunda regra de transformação seja o inverso da regra de transformação da primeira regra de transformação. Geralmente, três diferentes regras de transformação poderiam ser usadas, desde que isso seja considerado na representação inversa.

[00092] Nesse contexto, faz-se novamente referência à Fig. 3 e à Fig. 4. Ao se converter os valores discretos para uma representação transformada, o meio 102 pode ser conformado para implementar qualquer regra de transformação 1. Além disso, o meio 108 também pode ser conformado para usar qualquer outra regra de transformação ou a mesma, a qual é referenciada como regra de transformação 2. Além disso, o meio 140 pode ser conformado para usar em geral qualquer regra de transformação 3, a qual não necessariamente tem de ser igual à primeira ou à segunda regra de transformação.

[00093] No entanto, na conversão inversa da representação transformada, tem de ser encontrada uma adaptação às regras de transformação 1 - 3 ilustradas na Fig. 3, pois o primeiro meio 124 para conversão não realiza nenhuma regra de transformação além da regra de transformação 3 realizada no bloco 140 da Fig. 3. De maneira correspondente, o meio 130 na Fig. 4 tem de realizar a regra de transformação 2, a qual também tem de ser realizada pelo bloco 108 na Fig. 3. Finalmente, o meio 150 da Fig. 4 tem de

realizar a regra de transformação 1, a qual também tem de ser realizada pelo meio 102 da Fig. 3, de modo a ser obtida uma conversão inversa sem perdas.

[00094] A seguir, com referência às Figs. 9 - 11, a separação do janelamento MDCT é representada novamente, conforme descrita em DE 10129240 A1, onde a separação do janelamento MDCT em rotações de Givens com matrizes de elevação e correspondentes arredondamentos podem ser combinadas de forma vantajosa com o conceito ilustrado na Fig. 1 para a conversão e na Fig. 2 para a conversão inversa, de modo a obter uma aproximação MDCT inteira completa, o que significa uma MDCT inteira (IntMDCT) de acordo com a presente invenção, onde tanto um conceito de transformação para frente como um conceito de transformação para trás foi realizado com o exemplo de uma MDCT.

[00095] A Fig. 9 mostra um diagrama de visão geral para o aparato preferido inventivamente para processamento de amostras discretas no tempo, as quais representam um sinal de áudio, para se obter valores inteiros sobre os quais opera o algoritmo de transformação inteira Int-MDCT. As amostras discretas no tempo são janeladas pelo aparato mostrado na Fig. 9 e opcionalmente convertidas em uma representação espectral. As amostras discretas no tempo inseridas no aparato na entrada 10 são janeladas com uma janela w com um comprimento correspondente a $2N$ amostras discretas no tempo para se obter amostras janeladas inteiras em uma saída 12, as quais são adequadas para serem convertidas em uma representação espectral através de uma transformação e

particularmente o meio 14 para a realização de uma DCT inteira. A DCT inteira é conformada para gerar N valores de saída de N valores de entrada, o que é contrário à função MDCT 408 da Fig. 12a, a qual gera meramente N valores espectrais de 2N amostras janeladas devido à equação da MDCT.

[00096] Para o janelamento das amostras discretas no tempo, primeiro são escolhidas duas amostras discretas no tempo em um meio 16, as quais juntas representam um vetor de amostras discretas no tempo. Uma amostra discreta no tempo selecionada pelo meio 16 fica no primeiro quarto da janela. A outra amostra discreta no tempo fica no segundo quarto da janela como será discutido com mais detalhes com referência à Fig. 11. O vetor gerado pelo meio 16 agora é provido de uma matriz de rotação de dimensão 2 x 2, onde essa operação não é realizada imediatamente, mas através de várias das assim chamadas matrizes de elevação.

[00097] Uma matriz de elevação tem a característica de ter apenas um elemento dependendo da janela w e diferente de "1" ou "0".

[00098] A fatorização de transformadas do tipo wavelet em etapas de elevação é ilustrada na publicação especializada "Factoring Wavelet Transforms Into Lifting Steps", Ingrid Daubechies e Wim Sweldens, Pré-Publicação, Bell Laboratories, Lucent Technologies, 1996. Geralmente, um esquema de elevação é um relacionamento simples entre pares de filtros de reconstrução perfeitos, os quais têm o mesmo filtro de baixa passagem ou de alta passagem. Cada par de filtros complementares pode ser fatorizado em etapas de

elevação. Isso se aplica particularmente a rotações de Givens. Será considerado o caso em que a matriz de polifase é uma rotação de Givens. Aplica-se o seguinte:

$$\begin{pmatrix} \cos \alpha & -\sin \alpha \\ \sin \alpha & \cos \alpha \end{pmatrix} = \begin{pmatrix} 1 & \cos \alpha - 1 \\ & \sin \alpha \end{pmatrix} \begin{pmatrix} 1 & 0 \\ \sin \alpha & 1 \end{pmatrix} \begin{pmatrix} 1 & \cos \alpha - 1 \\ & \sin \alpha \end{pmatrix} \quad (1)$$

[00099] Cada uma das três matrizes de elevação à direita do sinal de igual possui o valor "1" como elemento da diagonal principal. Além disso, em cada matriz de elevação, um elemento da diagonal secundária é igual a 0 e um elemento da diagonal secundária depende do ângulo de rotação α .

[000100] O vetor é agora multiplicado pela terceira matriz de elevação, isto é, a matriz de elevação na extremidade direita da equação acima, para se obter um primeiro vetor de resultados. Isso é ilustrado na Fig. 9 através de um meio 18. Agora, o primeiro vetor de resultado é arredondado com uma função de arredondamento arbitrária, a qual mapeia a quantidade de números reais para a quantidade de números inteiros, como é ilustrado na Fig. 9 por um meio 20. Um primeiro vetor arredondado de resultados é obtido na saída do meio 20. O primeiro vetor arredondado de resultados é agora inserido no meio 22 para multiplicação dele pela matriz de elevação do meio, isto é, a segunda, para se obter um segundo vetor de resultados, o qual é de novo arredondado em um meio 24 para se obter um segundo vetor arredondado de resultados. O segundo vetor arredondado de resultados é agora inserido no meio 26 para multiplicação dele pela matriz de elevação mostrada na esquerda da equação acima, isto é, a primeira matriz de elevação, para se obter um terceiro vetor

de resultados, o qual é em seguida finalmente arredondado através de um meio 28 para se obter finalmente amostras janeladas inteiras na saída 12, as quais agora, caso se deseje uma representação espectral delas, têm de ser processadas pelo meio 14 para se obter valores espectrais inteiros em uma saída espectral 30.

[000101] O meio 14 é preferivelmente identificado como uma DCT inteira.

[000102] A transformada do co-seno discreto de acordo com o tipo 4 (DCT-IV) com um comprimento N é dada pela seguinte equação:

$$X_t(m) = \sqrt{\frac{2}{N}} \sum_{k=0}^{N-1} x(k) \cos\left(\frac{\pi}{4N} (2k+1)(2m+1)\right) \quad (2)$$

[000103] Os coeficientes da DCT-IV formam uma matriz N x N ortogonal. Cada matriz N x N ortogonal pode ser separada em N (N-1)/2 rotações de Givens conforme discutido na publicação especializada P. P. Vaidyanathan, "Multirate Systems And Filter banks", Prentice-Hall, Englewood Cliffs, 1993. Deve-se observar que também há outras separações.

[000104] Com relação às classificações dos diferentes algoritmos de DCT, deve-se fazer referência a H. S. Malvar, "Signal Processing With Lapped Transforms", Artech House, 1992. Geralmente, os algoritmos de DCT diferem pelo tipo de suas funções básicas. Enquanto o DCT-IV, que é o preferido aqui, compreende funções básicas não simétricas, isto é, um co-seno de quarto de onda, um co-seno de 3/4 de onda, um co-seno de 5/4 de onda, um co-seno de 7/4 de onda, etc., a transformada do co-seno discreta, por exemplo, do tipo II

(DCT-II), possui funções básicas simétricas no eixo e simétricas no ponto. A 0ª função base possui um componente direto, a primeira função básica é um meio co-seno de onda, a segunda função básica é um co-seno inteiro de onda, etc. Devido ao fato de que a DCT-II particularmente considera o componente constante, ela é usada em codificação de vídeo, mas não em codificação de áudio, visto que o componente constante não é relevante em codificação de áudio, ao contrário da codificação de vídeo.

[000105] A seguir, faz-se referência ao fato de como o ângulo de rotação α da rotação de Givens depende da função de janela.

[000106] Uma MDCT com um comprimento de janela de $2N$ pode ser reduzida a uma transformada do co-seno discreta do tipo IV com um comprimento N . Isso é obtido através da realização da operação TDAC explicitamente no domínio do tempo e em seguida aplicando-se a DCT-IV. Em uma sobreposição de 50%, a metade esquerda da janela para um bloco t se sobrepõe com a metade direita do bloco anterior, isto é, o bloco $t-1$. A parte da sobreposição dos dois blocos subsequentes $t-1$ e t é pré-processada da seguinte maneira no domínio do tempo, isto é, antes da transformação, isto é, entre a entrada 10 e a saída 12 da Fig. 9:

$$\begin{pmatrix} \tilde{x}_t(k) \\ \tilde{x}_{t-1}(N-1-k) \end{pmatrix} = \begin{pmatrix} w\left(\frac{N}{2}+k\right) & -w\left(\frac{N}{2}-1-k\right) \\ w\left(\frac{N}{2}-1-k\right) & w\left(\frac{N}{2}+k\right) \end{pmatrix} \begin{pmatrix} x_t\left(\frac{N}{2}+k\right) \\ x_t\left(\frac{N}{2}-1-k\right) \end{pmatrix} \quad (3)$$

[000107] Os valores indicados com til são valores na saída 12 da Fig. 9, enquanto os valores de x sem til na equação acima são os valores na entrada 10 e após o meio 16 para seleção, respectivamente. O índice variável k varia de 0 a N/2-1, enquanto w representa a função de janela.

A partir da condição TDAC para a função de janela w, aplica-se o seguinte contexto:

$$w\left(\frac{N}{2} + k\right)^2 + w\left(\frac{N}{2} - 1 - k\right)^2 = 1 \quad (4)$$

[000108] Para certos ângulos α_k , $k = 0, \dots, N/2-1$, esse pré-processamento no domínio do tempo pode ser escrito como uma rotação de Givens, conforme foi explicado.

[000109] O ângulo α da rotação de Givens depende da função de janela w da seguinte maneira:

$$\alpha = \arctan[w(N/2-1-k) / w(N/2 + k)] \quad (5)$$

[000110] Deve-se observar que é possível usar qualquer função de janela w, desde que ela preencha essa condição de TDAC.

[000111] A seguir, um codificador e decodificador em cascata é descrito com referência à Fig. 10. As amostras discretas no tempo $x(0)$ a $x(2N-1)$, que são "janeladas" juntas por uma janela são primeiro selecionadas de tal maneira pelo meio 16 da Fig. 9 que a amostra $x(0)$ e a amostra $x(N-1)$, isto é, uma amostra do primeiro quarto da janela e uma amostra do segundo quarto da janela são selecionadas para formar o vetor na saída do meio 16. As setas cruzadas representam esquematicamente as multiplicações de elevação e os subseqüentes arredondamentos dos meios 18, 20 e 22, 24 e 26,

28, respectivamente, para se obter as amostras janeladas inteiras na entrada dos blocos DCT-IV.

[000112] Quando o primeiro vetor tiver sido processado como descrito acima, além disso, um segundo vetor das amostras $x(N/2-1)$ e $x(n/2)$, isto é, novamente uma amostra do primeiro quarto da janela e uma amostra do segundo quarto da janela, é selecionada e novamente processada pelo algoritmo descrito na Fig. 9. De maneira análoga, todos os demais pares de amostras do primeiro e do segundo quartos da janela são processados. O mesmo processamento é realizado para o terceiro e o quarto quartos da primeira janela. Agora, $2N$ amostras inteiras janeladas estão presentes na saída 12, as quais são agora inseridas em uma transformação DCT-IV conforme ilustrado na Fig. 10. Particularmente, as amostras janeladas inteiras do segundo e terceiro quartos são inseridas em uma DCT. As amostras inteiras janeladas do primeiro quarto da janela são processadas em uma DCT-IV anterior em conjunto com as amostras inteiras janeladas do quarto quarto da janela anterior. De maneira análoga, os quartos quartos das amostras inteiras janeladas da Fig. 10 são inseridas em uma transformação DCT-IV com o primeiro quarto da próxima janela. A transformação DCT-IV integral intermediária mostrada na Fig. 10 agora fornece N valores espectrais inteiros $y(0)$ a $y(N-1)$. Esses valores espectrais inteiros agora podem ser facilmente codificados por entropia, sem exigir uma quantização intermediária, visto que o janelamento e a transformação fornecem valores de saída inteiros.

[000113] Um decodificador é ilustrado na metade direita da Fig. 10. O decodificador, consistindo em transformação inversa e "janelamento inverso", opera inversamente ao codificador. É sabido que uma DCT-IV inversa pode ser usada para a transformação inversa de uma DCT-IV, conforme ilustrado na Fig. 10. Os valores de saída do decodificador DCT-IV 34 são agora processados inversamente, conforme ilustrado na Fig. 10, com os valores correspondentes da transformação prévia e da transformação seguinte, respectivamente, de modo a gerar amostras de áudio discretas no tempo $x(0)$ a $x(2N-1)$ novamente a partir das amostras janeladas inteiras na saída do meio 34 e das transformações prévia e seguinte, respectivamente.

[000114] A operação do lado de saída é realizada por uma rotação de Givens inversa, isto é, tal que os blocos 26, 28 e 22, 24 e 18, 20, respectivamente, sejam passados na direção oposta. Isso é ilustrado em mais detalhes com relação à segunda matriz de elevação da equação 1. Caso (no codificador) o segundo vetor de resultado seja formado por uma multiplicação do primeiro vetor arredondado de resultado pela segunda matriz de elevação (meio 22), resultará a seguinte expressão:

$$(x,y) \mapsto (x, y + x \operatorname{sen} \alpha) \quad (6)$$

[000115] Os valores x , y no lado direito da equação 6 são inteiros. No entanto, isso não se aplica ao valor de $x \operatorname{sen} \alpha$. Aqui, a função de arredondamento r tem de ser introduzida, conforme ilustrado na equação subsequente:

$$(x,y) \mapsto (x, y+r(x \operatorname{sen} \alpha)) \quad (7)$$

[000116] Essa operação é realizada pelo meio 24.

[000117] O mapeamento inverso (no decodificador) é definido da seguinte maneira:

$$(x',y') \mapsto (x', y'-r(x' \operatorname{sen} \alpha)) \quad (8)$$

[000118] Devido ao sinal de menos antes da operação de arredondamento, torna-se óbvio que a aproximação inteira da etapa de elevação pode ser invertida sem a introdução de um erro. A aplicação dessa aproximação a cada uma das três etapas de elevação leva a uma aproximação inteira da rotação de Givens. A rotação arredondada (no codificador) pode ser invertida (no decodificador) sem se introduzir um erro e pela passagem através das etapas de elevação arredondadas em ordem reversa, isto é, quando o algoritmo da Fig. 9 é executado de baixo para cima durante a decodificação.

[000119] Quando a função de arredondamento r é simétrica no ponto, a rotação arredondada inversa é idêntica à rotação arredondada com o ângulo $-\alpha$ e é como segue:

$$\begin{pmatrix} \cos \alpha & \operatorname{sen} \alpha \\ -\operatorname{sen} \alpha & \cos \alpha \end{pmatrix} \quad (9)$$

[000120] As matrizes de elevação para o decodificador, isto é, para a rotação de Givens inversa, resulta neste caso imediatamente da equação (1) através da mera substituição da expressão " $\operatorname{sen} \alpha$ " pela expressão " $-\operatorname{sen} \alpha$ ".

[000121] A seguir, é ilustrada novamente a separação de uma MDCT comum com janelas sobrepostas 40 a 46 com relação à Fig. 11. Cada uma das janelas 40 a 46 se sobrepõe de 50%.

Por janela, as rotações de Givens são realizadas dentro do primeiro e segundo quartos de uma janela e dentro do terceiro e quarto quartos de uma janela, respectivamente, como é ilustrado esquematicamente pelas setas 48. Em seguida, os valores rotacionados, isto é, as amostras inteiras janeladas, são inseridos em uma DCT de N para N , tal que sempre o segundo e o terceiro quartos de uma janela e o quarto e o primeiro quartos de uma janela subsequente, respectivamente, sejam convertidos, em conjunto, em uma representação espectral através de um algoritmo de DCT-IV.

[000122] Assim, as rotações comuns de Givens são separadas em matrizes de elevação, as quais são realizadas em seqüência, onde uma etapa de arredondamento após cada multiplicação de matriz de elevação, tal que os números em ponto flutuante sejam arredondados imediatamente após a sua ocorrência, tal que o vetor de resultados tenha meramente inteiros antes de cada multiplicação de um vetor de resultados por uma matriz de elevação.

[000123] Assim, os valores de saída sempre permanecem inteiros, onde é preferível também usar valores de entrada inteiros. Isso não representa uma limitação, visto, por exemplo, que quaisquer amostras de PCM, quando elas são armazenadas em um CD, são valores inteiros cuja faixa de valores varia dependendo da largura de bit, isto é, dependendo de se os valores de entrada digitais discretos no tempo são valores de 16 bits ou valores de 24 bits. Todavia, o processo como um todo é invertível, como foi discutido, através da execução de rotações inversas em uma direção

inversa. Assim, existe uma aproximação inteira da MDCT com reconstrução perfeita, o que significa uma transformação sem perdas.

[000124] A transformação mostrada fornece valores de saída inteiros ao invés de valores de ponto flutuante. Ela fornece uma reconstrução perfeita, de modo que nenhum erro é introduzido quando é realizada uma transformação para frente e depois outra para trás. De acordo com uma configuração preferida da presente invenção, a transformação é uma substituição para a transformada do co-seno discreta modificada. Outros métodos de transformação também podem ser realizados de maneira inteira, desde que seja possível uma separação em rotações e uma separação das rotações em etapas de elevação.

[000125] A MDCT inteira tem as características mais favoráveis da MDCT. Ela possui uma estrutura de sobreposição, onde é obtida uma melhor seletividade de frequência do que com transformações de bloco sem sobreposição. Devido à função TDAC, que já é considerada durante o janelamento antes da transformação, é mantida uma amostragem crítica, de modo que o número total dos valores espectrais representando um sinal de áudio seja igual ao número total de amostras de entrada.

[000126] Comparada com uma MDCT normal, que fornece amostras em ponto flutuante, a transformação inteira preferida descrita mostra que comparada com a MDCT com nível baixo de sinal, o ruído é aumentado meramente na faixa espectral, enquanto esse aumento de ruído não é perceptível em níveis de sinal significativos. Ao invés disso, o

processamento inteiro presta-se a uma implementação de hardware eficiente, visto que são usados meramente etapas de multiplicação, as quais podem facilmente ser separadas em etapas de deslocamento/adição que podem ser implementadas em hardware de uma maneira simples e rápida. É claro que uma implementação por software também é possível.

[000127] A transformação inteira fornece uma boa representação espectral do sinal de áudio e ainda permanece na faixa dos números inteiros. Caso ela seja aplicada a partes tonais de um sinal de áudio, isso resulta em uma boa concentração de energia. Por conseguinte, pode-se construir um esquema de codificação eficiente sem perdas através do simples janelamento/transformação em cascata ilustrado na Fig. 9 com um codificador por entropia. É particularmente favorável o uso de codificadores empilhados usando valores de escape como o usado em MPEG AAC. É preferível escalonar para baixo todos os valores de uma certa potência de dois até que eles caibam em uma tabela de código desejável para em seguida codificar adicionalmente os bits menos significativos omitidos. Em comparação com a alternativa de usar tabelas de código maiores, a alternativa descrita é mais favorável em relação ao consumo de memória para armazenagem das tabelas de código. Também seria possível obter um codificador quase sem perdas através da simples omissão de certos bits menos significativos.

[000128] Particularmente para sinais tonais, a codificação por entropia dos valores espectrais inteiros permite um alto ganho de codificador. Para partes transientes

do sinal, o ganho do codificador é baixo, devido ao espectro plano dos sinais transientes, isto é, devido a um baixo número de valores espectrais serem iguais a ou próximos de 0. Conforme descrito em J. Herre, J. D. Johnston: "Enhancing the Performance of Perceptual Audio Coders by Using Temporal Noise Shapping (TNS)" 101. AES Convention, Los Angeles, 1996, Pré-Publicação 4384, essa planura pode ser usada através do uso de uma previsão linear na faixa de frequência. Uma alternativa é uma previsão com laço aberto. Outra alternativa é o previsor com laço fechado. A primeira alternativa, isto é, o previsor com laço aberto, é chamada de TNS. A quantização após a previsão leva a uma adaptação do ruído de quantização resultante à estrutura de tempo do sinal de áudio e assim evita pré-ecos em codificadores de áudio psicoacústicos. Para codificação de áudio sem perdas, a segunda alternativa, isto é, com um previsor com laço fechado, é mais adequada, visto que a previsão com laço fechado possibilita uma reconstrução exata do sinal de entrada. Quando essa técnica é aplicada a um espectro gerado, uma etapa de arredondamento tem de ser realizada após cada etapa do filtro de precisão para permanecer na faixa dos inteiros. Através do uso do filtro inverso e da mesma função de arredondamento, o espectro original pode ser exatamente reconstruído.

[000129] De modo a usar a redundância entre os dois canais para a redução de dados, uma codificação centro-lateral pode ser usada de uma maneira sem perdas, quando se usa uma rotação arredondada com um ângulo de $\pi/4$. Em

comparação com a alternativa de calcular a soma e a diferença dos canais esquerdo e direito de um sinal estéreo, a rotação arredondada tem a vantagem da conservação de energia. O uso das assim chamadas técnicas de codificação de estéreo conjunto pode ser ativado ou desativado para cada banda, como é feito em um MPEG AAC padrão. Também se pode considerar ângulos de rotação adicionais para reduzir a redundância entre dois canais de uma maneira mais flexível.

[000130] Dependendo das circunstâncias, o método inventivo para a conversão e para a conversão inversa, respectivamente, e para a transformação tempo-freqüência ou a transformação freqüência-tempo, respectivamente, pode ser implementado em hardware ou em software. A implementação pode ser configurada em mídia de memória digital, particularmente um disco ou CD com sinais de controle legíveis eletronicamente, os quais podem colaborar com um sistema de computador programável, tal que o método seja executado. Geralmente, a invenção assim também consiste em um produto de programa de computador com um código de programa armazenado em um portador legível por máquina para realização do método inventivo quando o programa for executado em um computador. Em outras palavras, a invenção pode ser concretizada como um programa de computador com um código de programa para a realização do método quando o programa é executado em um computador.

REIVINDICAÇÕES

1. Aparato para conversão de valores discretos em uma representação transformada com valores inteiros, caracterizado pelo fato de que os valores discretos possuem informações de áudio e/ou imagem, compreendendo:

um meio (102) para o processamento de um primeiro bloco de valores discretos através do uso de uma primeira regra de transformação para obter um primeiro bloco de valores transformados;

um meio (104) para o arredondamento do primeiro bloco de valores transformados para se obter um primeiro bloco de valores transformados arredondados;

um meio (106) para a soma do primeiro bloco de valores transformados arredondados a um segundo bloco de valores discretos para se obter um segundo bloco de valores somados;

um meio (108) para o processamento do segundo bloco de valores somados através do uso de uma segunda regra de transformação para se obter um segundo bloco de valores transformados;

um meio (110) para o arredondamento do segundo bloco de valores transformados para se obter um segundo bloco de valores transformados arredondados; e

um meio (112) para a subtração do segundo bloco de valores transformados arredondados a partir do primeiro bloco de valores discretos para se obter um bloco de valores de saída inteiros da representação transformada.

2. Aparato de acordo com a reivindicação 1, caracterizado por compreender, ainda:

um meio (140) para o processamento do bloco de valores de saída inteiros através do uso de uma terceira regra de transformação para se obter um bloco de valores de saída transformados;

um meio (142) para o arredondamento do bloco de valores de saída transformados para se obter um bloco de valores de saída transformados arredondados; e

um meio (144) para a soma do bloco de valores de saída transformados arredondados e do segundo bloco de valores somados para se obter um bloco adicional de valores de saída inteiros da representação transformada.

3. Aparato de acordo com as reivindicações 1 ou 2, caracterizado pelo fato de que a primeira regra de transformação, a segunda regra de transformação e a terceira regra de transformação, respectivamente, compreendem uma matriz de transformação.

4. Aparato de acordo com uma das reivindicações anteriores, caracterizado pelo fato de que:

a primeira regra de transformação, a segunda regra de transformação e a terceira regra de transformação, respectivamente, são idênticas entre si.

5. Aparato de acordo com uma das reivindicações anteriores, caracterizado pelo fato de que a primeira regra de transformação e a terceira regra de transformação são uma regra de transformação para frente,

enquanto a segunda regra de transformação é uma regra de transformação para trás.

6. Aparato de acordo com uma das reivindicações anteriores, caracterizado pelo fato de que:

o meio (104) para arredondamento do primeiro bloco e o meio (110) para arredondamento do segundo bloco e o meio (142) para arredondamento dos valores de saída transformados, respectivamente, são conformados para arredondar de acordo com a função de arredondamento que fornece resultados arredondados, a precisão dos quais é inferior à precisão de máquina de um computador que compreenda o aparato.

7. Aparato de acordo com uma das reivindicações anteriores, caracterizado pelo fato de que o primeiro bloco de valores discretos possui valores discretos de um número ordinal 0 até um número ordinal $N-1$, onde o segundo bloco de valores discretos possui valores discretos de um número ordinal N até um número ordinal $2N-1$,

onde o primeiro bloco de valores transformados arredondados possui um número ordinal de 0 a $N-1$ e onde o meio (106) para a soma é conformado para realizar uma adição por valor, de modo que um valor do bloco de valores transformados arredondados, com um número ordinal i , seja adicionado a um valor do segundo bloco de valores discretos, com um número ordinal $i+N$, onde i é um índice variável estendendo-se de 0 a $N-1$.

8. Aparato de acordo com uma das reivindicações anteriores, caracterizado pelo fato de que o

primeiro bloco de valores discretos possui valores discretos com um número ordinal 0 até um número ordinal $N-1$,

onde o segundo bloco de valores discretos possui valores discretos de um número ordinal 0 até um número ordinal $2N-1$, onde o segundo bloco de valores transformados arredondados possui valores entre um número ordinal N e $2N-1$,

onde o meio (112) para subtração é conformado para subtrair um valor do bloco de valores transformados arredondados, com o número ordinal $n+i$, de um valor do primeiro bloco, com o número ordinal i , onde i é um índice variável que se estende de 0 a $N-1$.

9. Aparato de acordo com uma das reivindicações anteriores, caracterizado pelo fato de que o primeiro e o segundo blocos de valores discretos compreendem valores discretos inteiros.

10. Aparato de acordo com uma das reivindicações anteriores, caracterizado pelo fato de que o meio (104) para o arredondamento do primeiro bloco de valores transformados ou o meio (110) para o arredondamento do segundo bloco de valores transformados é conformado para arredondar de acordo com a mesma especificação de arredondamento.

11. Aparato de acordo com a reivindicação 3, caracterizado pelo fato de a matriz de transformação subjacente à primeira regra de transformação, a matriz de transformação subjacente à segunda regra de transformação e a matriz de transformação subjacente à terceira regra de transformação, respectivamente, são, cada uma, uma matriz

quadrada tendo um número de linhas e colunas, respectivamente, que são iguais aos valores discretos no primeiro ou segundo blocos.

12. Aparato de acordo com uma das reivindicações anteriores, caracterizado pelo fato de que o meio (102) para processamento de acordo com a primeira regra de transformação e o meio (108) para processamento de acordo com a segunda regra de transformação são conformados em um transformador, e

onde é provida ainda uma seqüência de controle, a qual é conformada para fornecer ao transformador inicialmente o primeiro bloco de valores discretos e depois o segundo bloco de valores somados.

13. Aparato de acordo com uma das reivindicações anteriores, caracterizado pelo fato de que o meio (104) para arredondamento do primeiro bloco de valores transformados e o meio (110) para arredondamento do segundo bloco de valores transformados são conformados em um arredondador, e

onde é provida ainda uma seqüência de controle para fornecer ao arredondador inicialmente o primeiro bloco de valores discretos e depois o segundo bloco de valores transformados.

14. Aparato de acordo com uma das reivindicações anteriores, caracterizado pelo fato de que o primeiro e o segundo blocos de valores discretos são amostras de áudio do primeiro e segundo canais de um sinal de áudio multicanal.

15. Aparato de acordo com uma das reivindicações de 1 a 13, caracterizado pelo fato de que o primeiro bloco e o segundo bloco são amostras de tempo subseqüentes de um sinal de áudio.

16. Aparato de transformação para frente, de acordo com uma das reivindicações de 1 a 15, caracterizado por compreender : um meio (16, 18, 20, 22, 24, 26, 28) para o janelamento de um primeiro bloco de amostras representando um sinal de áudio ou um sinal de imagem, e para o janelamento de um segundo bloco de amostras representando um sinal de áudio ou um sinal de imagem através do uso de várias matrizes de elevação (18, 22, 26) e subseqüentes operações de arredondamento (20, 24, 28) para se obter um primeiro bloco de valores discretos e um segundo bloco de valores discretos; e

um aparato para se obter uma representação transformada do primeiro e o segundo blocos com valores de saída inteiros.

17. Método para a conversão de valores discretos em uma representação transformada com valores inteiros, caracterizado pelo fato de que os valores discretos possuem informações de áudio e/ou imagem, compreendendo as etapas de:

processamento (102) de um primeiro bloco de valores discretos através do uso de uma primeira regra de transformação para se obter um primeiro bloco de valores transformados;

arredondamento (104) do primeiro bloco de valores transformados para se obter um primeiro bloco de valores transformados arredondados;

soma (106) do primeiro bloco de valores transformados arredondados com um segundo bloco de valores discretos para obter um segundo bloco de valores somados;

processamento (108) do segundo bloco de valores somados através do uso de uma segunda regra de transformação para se obter um segundo bloco de valores transformados;

arredondamento (110) do segundo bloco de valores transformados para se obter um segundo bloco de valores transformados arredondados; e

subtração (112) do segundo bloco de valores transformados arredondados a partir do primeiro bloco de valores discretos para se obter um bloco de valores de saída inteiros da representação transformada.

18. Método de transformação para frente, de acordo com a reivindicação 17, caracterizado por compreender as etapas de:

janelamento (16, 18, 20, 22, 24, 26, 28) de um primeiro bloco de amostras representando um sinal de áudio ou um sinal de imagem, e

janelamento de um segundo bloco de amostras representando um sinal de áudio ou um sinal de imagem através do uso de várias matrizes de elevação (18, 22, 26) e subseqüentes operações de arredondamento (20, 24, 28) para se obter um primeiro bloco de valores discretos e um segundo

bloco de valores discretos; e método para se obter uma representação transformada do primeiro e segundo blocos com valores de saída inteiros.

19. Aparato para a conversão inversa de um bloco de valores de saída inteiros e um segundo bloco de valores somados, os quais são uma representação transformada dos valores discretos para se obter o primeiro e o segundo blocos de valores discretos, caracterizado pelo fato de que o bloco de valores de saída inteiros e o segundo bloco de valores somados são derivados do primeiro e segundo blocos de valores discretos, por processamento (102) de um primeiro bloco de valores discretos através do uso da primeira regra de transformação para se obter um primeiro bloco de valores transformados, arredondamento (104) do primeiro bloco de valores transformados para se obter um primeiro bloco de valores transformados arredondados, soma (106) do primeiro bloco de valores transformados arredondados com um segundo bloco de valores discretos para se obter o segundo bloco de valores somados, processamento (108) do segundo bloco de valores somados através do uso de uma segunda regra de transformação para se obter um segundo bloco de valores transformados, arredondamento (110) do segundo bloco de valores transformados para se obter um segundo bloco de valores transformados arredondados, e subtração (112) do segundo bloco de valores transformados arredondados do primeiro bloco de valores discretos para se obter um bloco de valores de saída inteiros da representação transformada, compreendendo:

um meio (130) para o processamento do segundo bloco de valores somados através do uso da segunda regra de transformação para se obter um primeiro bloco de valores de saída transformados;

um meio (132) para o arredondamento do primeiro bloco de valores de saída transformados para se obter um primeiro bloco de valores transformados arredondados;

um meio (134) para soma do primeiro bloco de valores de saída transformados arredondados com o bloco de valores de saída inteiros para se obter o primeiro bloco de valores discretos;

um meio (150) para o processamento do primeiro bloco de valores discretos através do uso da primeira regra de transformação para se obter um segundo bloco de valores transformados;

um meio (152) para o arredondamento do segundo bloco de valores transformados para se obter um bloco de valores transformados arredondados; e

um meio (154) para a subtração do bloco de valores transformados arredondados do segundo bloco de valores somados para se obter o segundo bloco de valores discretos.

20. Aparato de acordo com a reivindicação 19, caracterizado pelo fato de que um bloco adicional de valores de saída inteiros da representação transformada é calculado a partir do segundo bloco de valores somados pelo processamento (140) do bloco de valores de saída inteiros através do uso de

uma terceira regra de transformação para se obter um bloco de valores de saída transformados, pelo arredondamento (142) do bloco de valores de saída transformados para se obter um bloco de valores de saída transformados arredondados, e pela soma (144) do bloco de valores de saída transformados arredondados com o segundo bloco ou os valores somados para se obter o bloco adicional de valores de saída inteiros da representação transformada, o aparato compreendendo ainda:

um meio (124) para o processamento do bloco de valores de saída inteiros através do uso da terceira regra de transformação para se obter um primeiro bloco de valores de saída transformados;

um meio (126) para o arredondamento do primeiro bloco de valores de saída transformados para se obter um primeiro bloco de valores de saída transformados arredondados; e

um meio para subtração (128) do primeiro bloco dos valores de saída transformados arredondados a partir do bloco adicional de valores de saída para se obter o segundo bloco de valores somados.

21. Método para conversão inversa de um bloco de valores de saída inteiros e um segundo bloco de valores somados, os quais são uma representação transformada dos valores discretos, para obter um primeiro e um segundo blocos de valores discretos, caracterizado pelo fato de que o bloco de valores de saída inteiros e o segundo bloco de valores somados foram derivados do primeiro e do segundo blocos de valores discretos, através de processamento (102) de um

primeiro bloco de valores discretos através do uso de uma primeira regra de transformação para se obter um primeiro bloco de valores transformados, arredondamento (104) do primeiro bloco de valores transformados para se obter um primeiro bloco de valores transformados arredondados, soma (106) do primeiro bloco de valores transformados arredondados com um segundo bloco de valores discretos para se obter um segundo bloco de valores somados, processamento (108) do segundo bloco de valores somados através do uso de uma segunda regra de transformação para se obter um segundo bloco de valores transformados, arredondamento (110) do segundo bloco de valores transformados para se obter um segundo bloco de valores transformados arredondados, e subtração (112) do segundo bloco de valores transformados arredondados a partir do primeiro bloco de valores discretos para se obter um bloco de valores de saída inteiros da representação transformada, compreendendo as etapas de:

processamento (130) do segundo bloco de valores somados através do uso da segunda regra de transformação para se obter um primeiro bloco de valores de saída transformados;

arredondamento (132) do primeiro bloco de valores de saída transformados para se obter um primeiro bloco de valores transformados arredondados;

soma (134) do primeiro bloco de valores de saída transformados arredondados com o bloco de valores de saída inteiros para se obter o primeiro bloco de valores discretos;

processamento (150) do primeiro bloco de valores discretos através do uso da primeira regra de transformação para se obter um segundo bloco de valores transformados;

arredondamento (152) do segundo bloco de valores transformados para se obter um bloco de valores transformados arredondados; e

subtração (154) do bloco de valores transformados arredondados a partir do segundo bloco de valores somados para se obter o segundo bloco de valores discretos.

22. Aparato de transformação para trás, de acordo com as reivindicações 19 ou 20 caracterizado por compreender um meio para o janelamento inverso do primeiro e segundo blocos de valores discretos através do uso de matrizes de elevação (18, 22, 26) e subseqüentes operações de arredondamento (20, 24, 28) para se obter um primeiro e segundo blocos de valores originais representando amostras de áudio ou amostras de imagem.

23. Método de transformação para trás, de acordo com a reivindicação 21, caracterizado por compreender as etapas de: janelamento inverso do primeiro e segundo blocos de valores discretos através do uso de matrizes de elevação (18, 22, 26) e subseqüentes operações de arredondamento (20, 24, 28) para se obter um primeiro e segundo blocos de valores originais representando amostras de áudio ou amostras de imagem.

FIGURA 1

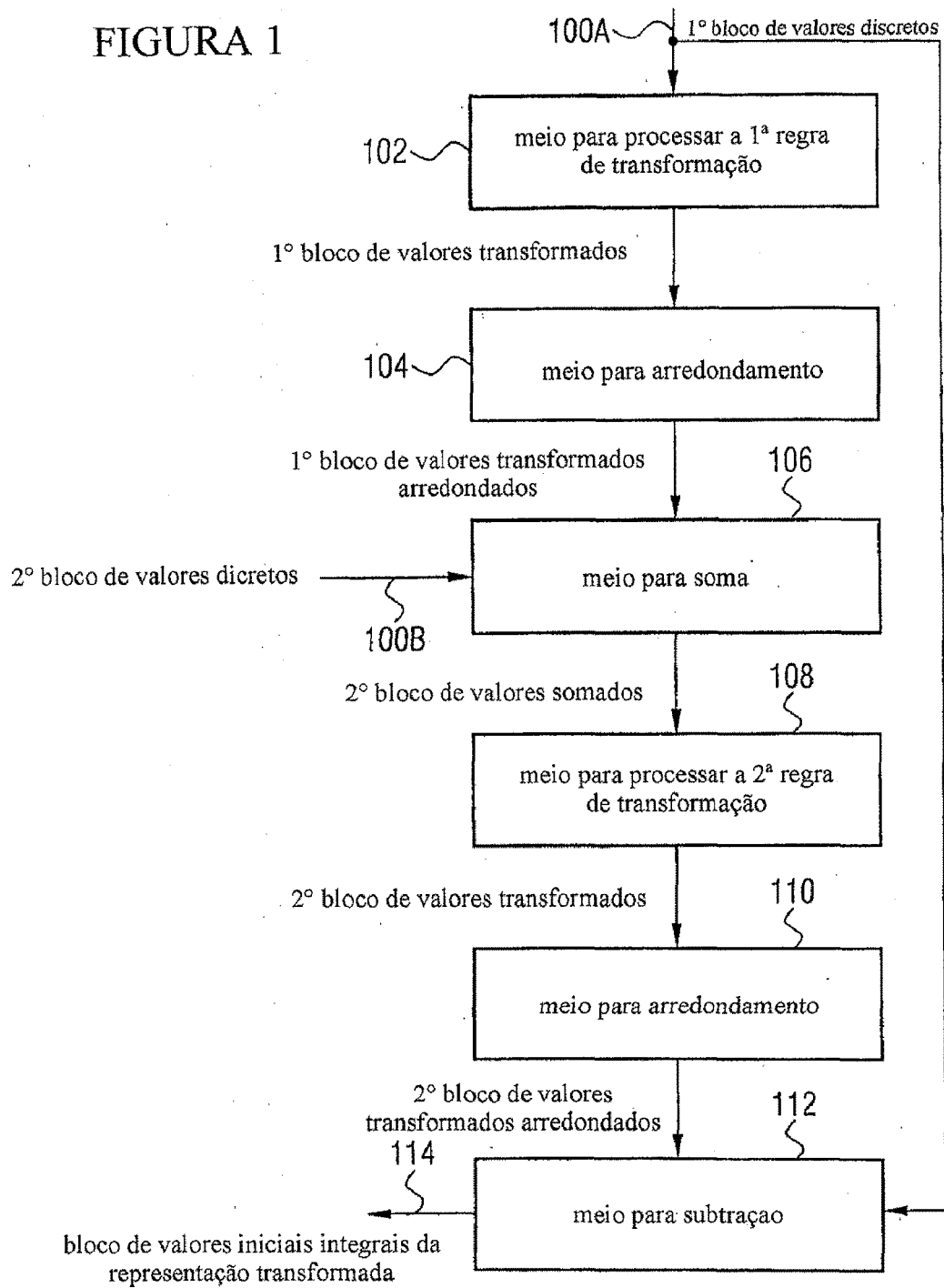


FIGURA 2

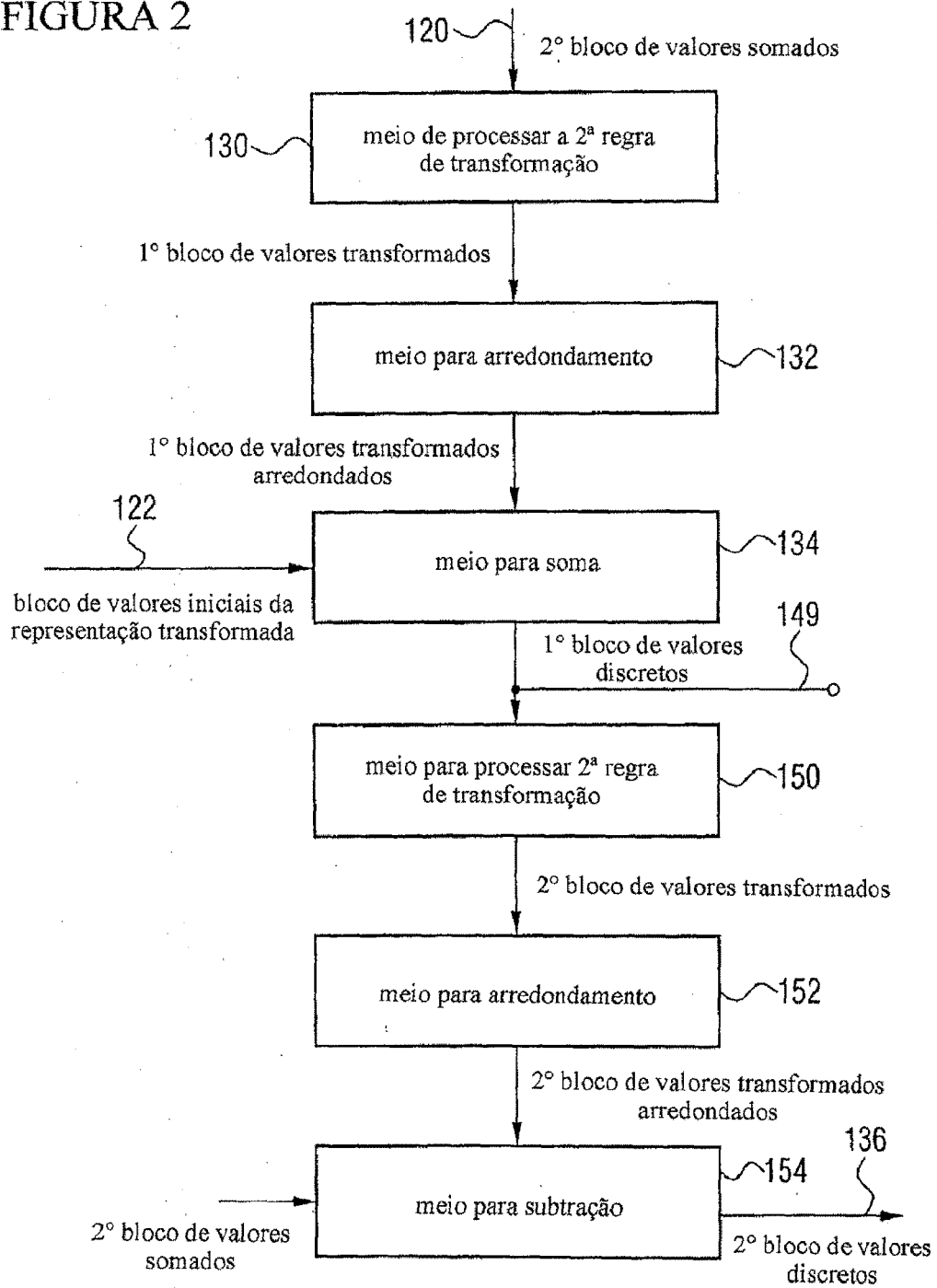


FIGURA 3

(conversão em
representação transformada)

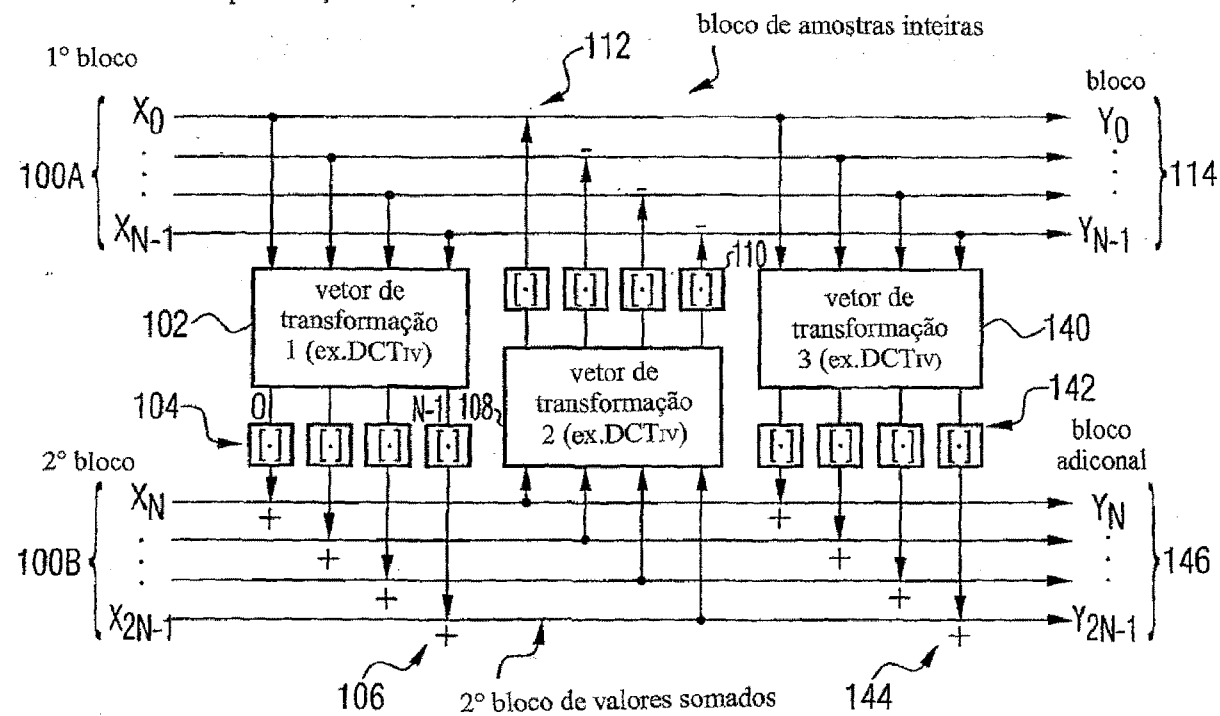


FIGURA 4

(conversão inversa da representação transformada)

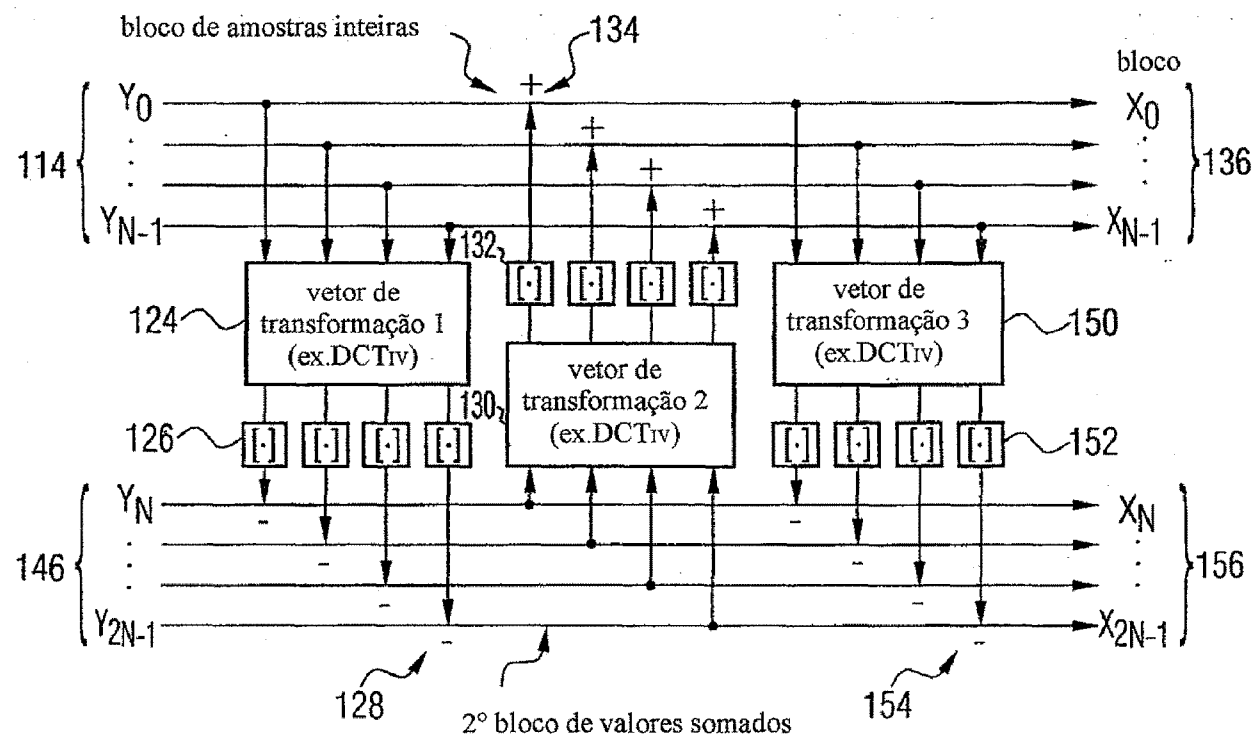


FIGURA 5

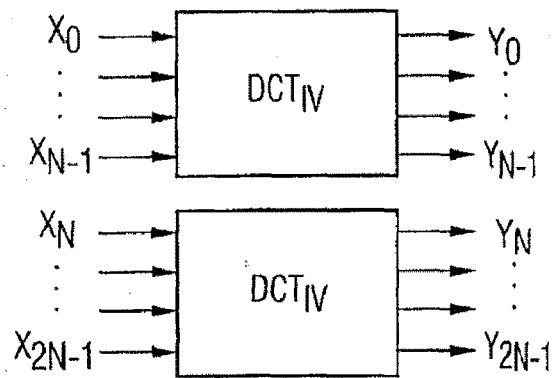


FIGURA 6

Etapa de elevação com matriz de transformação durante a conversão

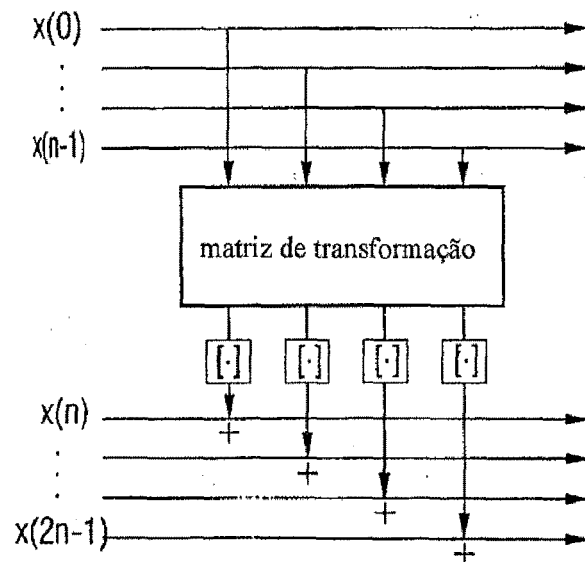


FIGURA 7

etapa de elevação inversa com matriz de transformação durante conversão inversa

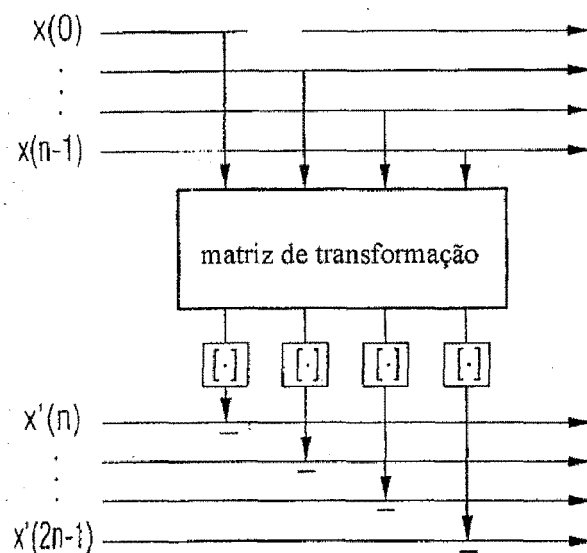
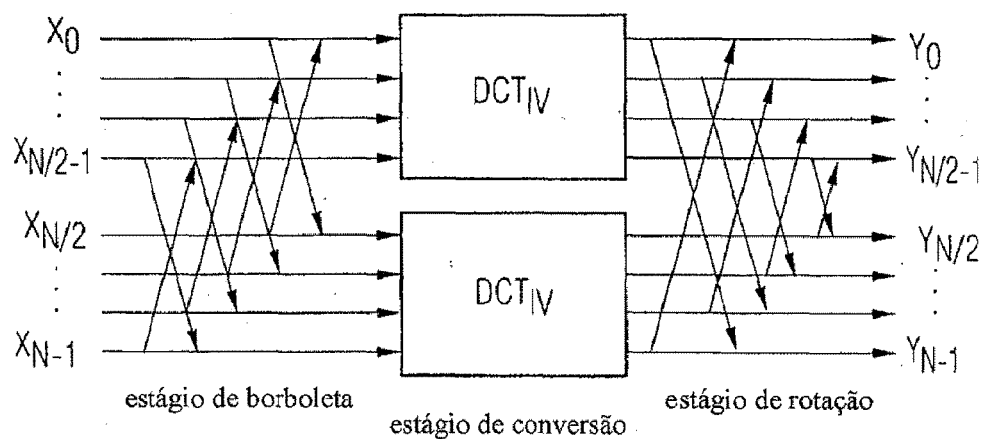


FIGURA 8



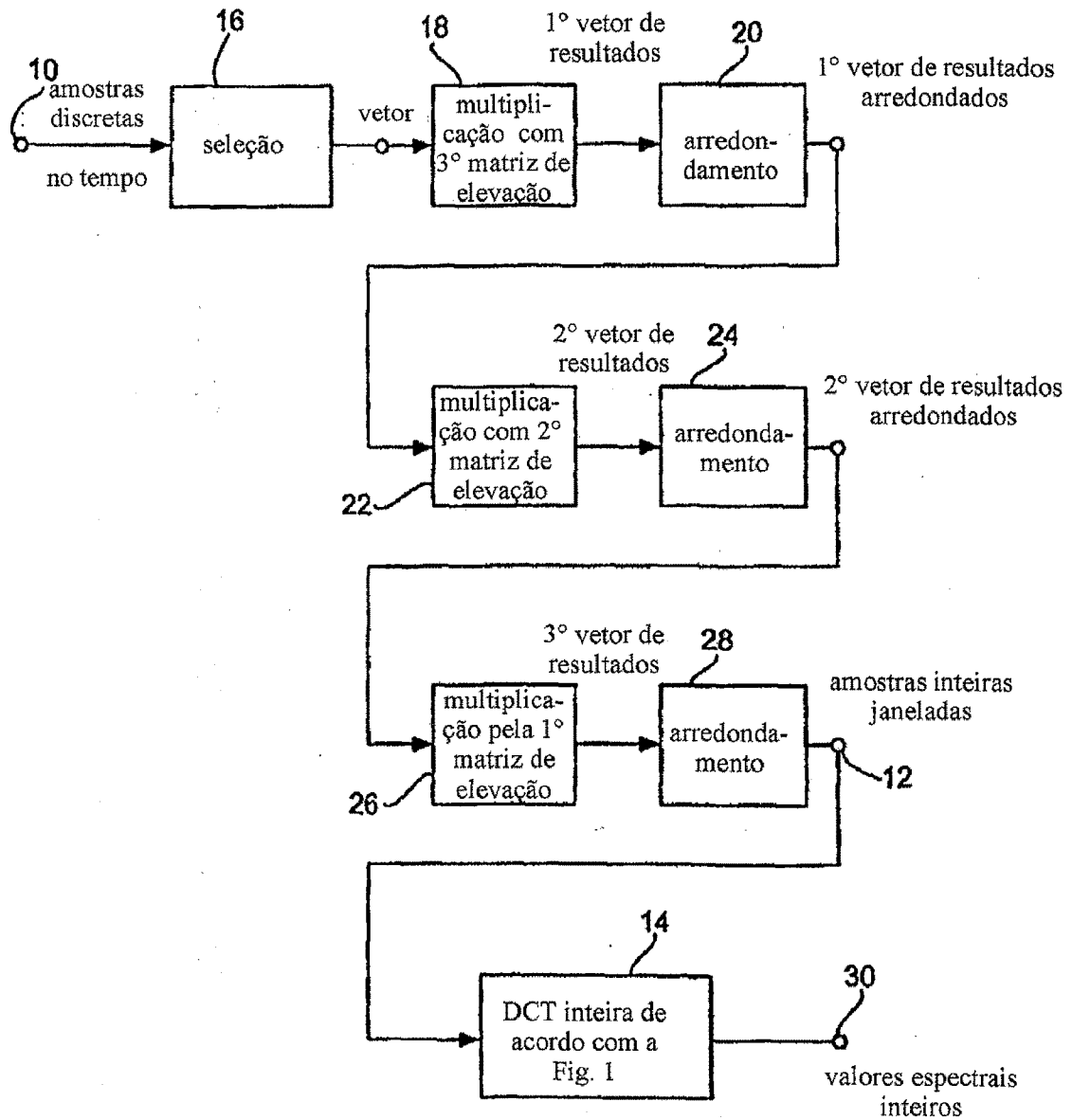


FIGURA 9

FIGURA 10

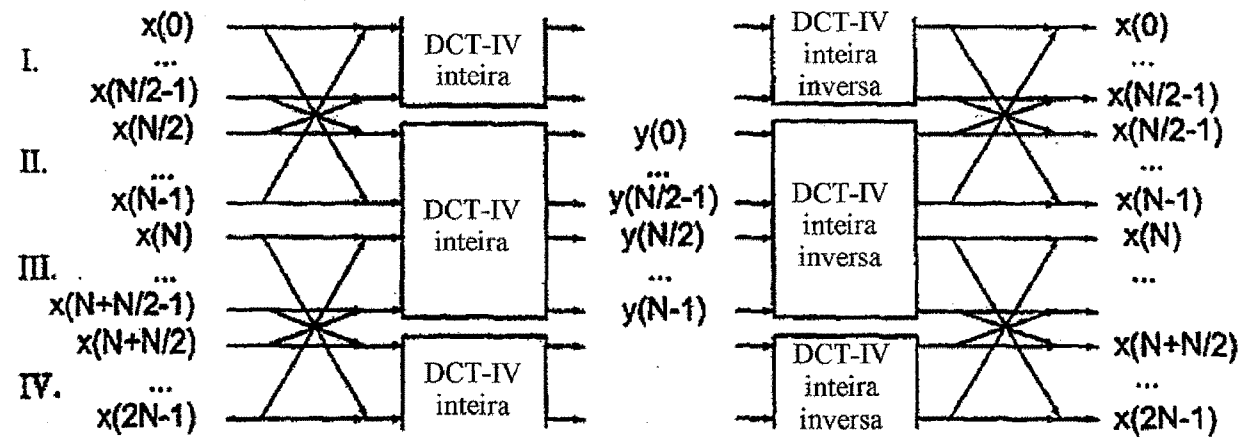


FIGURA 11

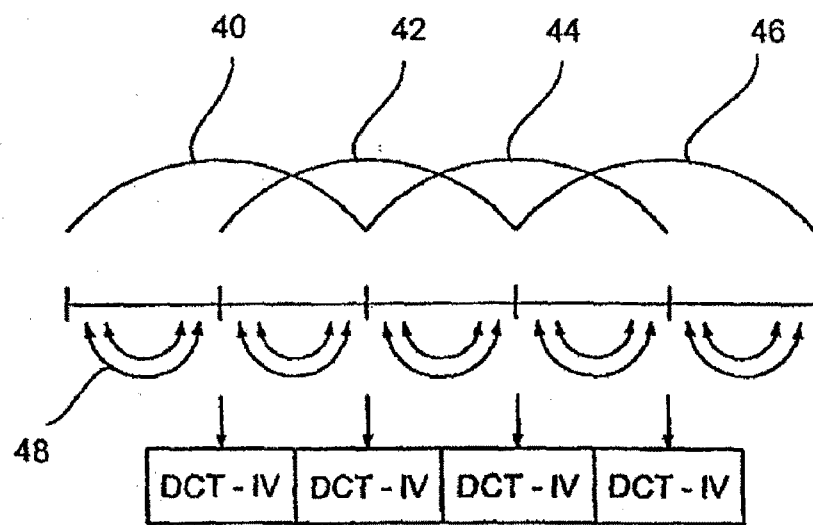


FIGURA 12A

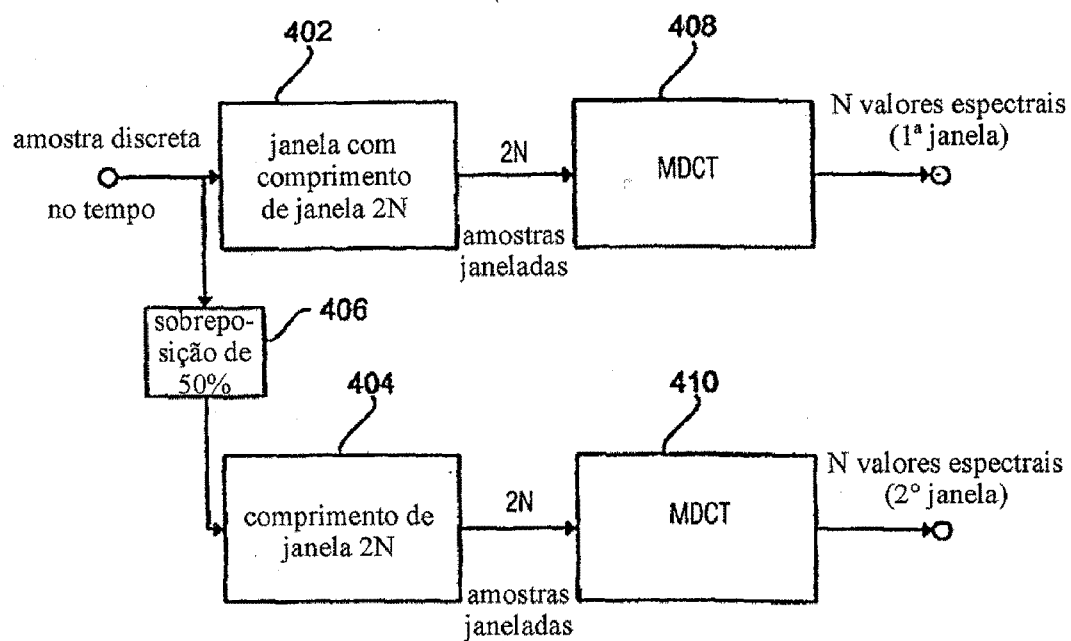


FIGURA 12B

