

III SIMPÓSIO BRASILEIRO DE COMPUTAÇÃO GRÁFICA E PROCESSAMENTO DE IMAGENS.

Hotel Serrano, Gramado, RS • 30 de maio a 1º de junho de 1990

ANAIS

Editores: Carla Dal Sasso Freitas e Ronaldo Marinho Persiano

DIZIMACÃO E INTERPOLAÇÃO PROGRESSIVAS PARA CODIFICAÇÃO EM PIRÂMIDE

Ricardo L. de Queiroz e João B.T. Yabu-uti

Depto. de Comunicações - Faculdade de Engenharia Elétrica
Universidade Estadual de Campinas - UNICAMP

C.P. 6101 - Campinas-SP - 13081

RESUMO - A codificação em pirâmide tem sido estudada ultimamente visando transmissão/codificação progressiva de uma imagem. Aqui propomos novo esquema de dizimação, adequado à interpolação através de filtros FIR bidimensionais. Estruturas polifásicas destes são apresentadas, bem como interpoladores bidimensionais baseados na Convolução Cúbica.

1. INTRODUÇÃO

A codificação em pirâmide consiste na representação da imagem original com diferentes resoluções. Com isto, é possível perfazer uma transmissão progressiva e, simultaneamente, reduzir a taxa de bits/pixel para codificação. Tal procedimento pode ser de grande utilidade para consulta a bancos de imagens via canais de faixa estreita ou sistemas de processamento lento, porém nada impede de aplicar esta codificação para simplesmente reduzir o montante de dados requeridos e melhor codificar a imagem.

Imaginaremos a imagem como uma matriz de $N \times N$ pixels de forma que N seja da ordem de 512. Esta matriz, denotaremos por $X(n_1, n_2)$ ou $X(n)$, $n = (n_1, n_2)$. Por todo o trabalho, utilizaremos notações familiares a [1] para representação de sinais e sistemas discretos multidimensionais.

2. PIRÂMIDE REDUZIDA

Se prefiltrarmos $X(n)$ adequadamente, podemos representar esta versão filtrada, fielmente, com $1/4$ das amostras originais, obtendo

$$X_{0F}(n) = X(n) \text{ filtrado} \quad (E1)$$

$$X_1(n) = X_{0F}(2n) \quad (E2)$$

$$L_0(n) = X_0(n) - X_{0F}(n) \quad (E3)$$

$L_0(n)$ seria o primeiro nível da pirâmide e, repetindo o processo com $X_1(n)$, obteríamos novo nível e daí por diante. De forma geral

$$X_{i+1}(n) = X_{iF}(2n) \quad (E4)$$

$$L_i(n) = X_i(n) - X_{iF}(n) \quad i=0,1,\dots,M-1 \quad (E5)$$

Este é o princípio das pirâmides Laplacianas ou Gaussianas [2] e neste referência o princípio pode ser melhor compreendido. Em [3], esforços são realizados de forma a definir pirâmides reduzidas, nas quais o número de amostras (nodos) representados em todos os níveis se soma a N^2 , ou seja, o número de pixels originais. No primeiro caso, se $M \gg 1$, então o número de nodos é cerca de $1/3$ maior que o original, ou

$$N_{\text{NODOS}} = \sum_{k=0}^{M-1} N^2/4^k = \left(N^2 + \frac{N^2}{3} \right) (1 - 1/4^{M-1}) = N^2 + N^2/3 \quad (E6)$$

Para reduzir esta total a N^2 , basta não prefiltrar o sinal, sub-amostrá-lo e interpolá-lo, tomando $L_k(n)$ como

$$L_k(n) = X_k(n) - X_{k+1}^{INT}(n) \quad (E7)$$

Porém, $1/4$ das amostras de $L_k(n)$ devem ser descartadas pois, como não houve prefiltragem e a interpolação não deturpou as amostras originais, tais amostras serão sempre nulas. Daí, o número de nodos total se reduzirá a N^2 . As referências [2] e [3] melhor explicam codificação em pirâmide e [4] e [5] tratam dos interpoladores para sinais unidimensionais, respeitando a restrição acima citada. A extensão bidimensional destes será feita posteriormente.

3. ESTRUTURA POLIFÁSICA BIDIMENSIONAL DO INTERPOLADOR

Na seção anterior, foi suposto que a sub-amostragem ou decimação era retangular, como na Figura 1, e, não havendo pre-filtragem, $X_{k+1}(n) = X_k(\lfloor n \rfloor)$ e $X_{k+1}(n) = X(\lfloor n \rfloor)$. Sendo Y uma matriz de sub-amostragem definida como

$$Y = \begin{bmatrix} Q_1 & 0 \\ 0 & Q_2 \end{bmatrix} \quad (\text{no caso } Q_1=Q_2=2) \quad (E8)$$

Um filtro FIR interpolador unidimensional pode ser dividido em seus sub-filtros polifásicos definidos por [4] e [5]

$$\begin{aligned} h(n) &= \text{interpolador FIR fase nula} \\ h_1(n) &= h(nQ+1) \quad i=0,1,\dots,Q-1 \end{aligned} \quad (E9)$$

Em cada período de Q amostras, uma é a original, $Q-1$ devem ser interpoladas e existem Q sub-filtros polifásicos, um para cada amostra a ser interpolada. Como uma é original e deve ser mantida, $h_0(n) = \delta(n)$, restando-nos projetar os $Q-1$ restantes.

Para extensão ao caso bidimensional, seja (E8) e:

$$\begin{aligned} \underline{Q}^T &= (Q_1, Q_2) \\ \underline{Q}^T &= (0, 0) \end{aligned}$$

Além disto, seja S o operador de superposição (para lidar com as diferentes taxas de amostragem entre $h(n)$ e os sub-filtros) suplantando a somatória. Teremos:

$$h_{1j}(n_1, n_2) = h(Q_1 n_1 + i, Q_2 n_2 + j) \quad (E10)$$

$$h(n_1, n_2) = \sum_{i=0}^{Q_1-1} \sum_{j=0}^{Q_2-1} h(Q_1 n_1 + i, Q_2 n_2 + j) \quad (E11)$$

Em notação vetorial, sendo $\underline{k}^T = (i, j)$, $0 \leq i < Q_1$, $0 \leq j < Q_2$, temos

$$h(\underline{n}) = \sum_{\underline{k}=\underline{Q}}^{\underline{Q}} h_k(\underline{n}) = \sum_{\underline{k}=\underline{Q}}^{\underline{Q}} h(\underline{Q}\underline{n} + \underline{k}) \quad (E12)$$

E, novamente,

$$h_0(\underline{n}) = \delta(\underline{n}) \quad (E13)$$

O nosso caso de interesse, se restringe a $Q_1=Q_2=2$ e $\underline{V} = 2\underline{I}$, sendo \underline{I} a matriz identidade 2×2 . O número de sub-filtros polifásicos, bem como o tamanho do período de sub-amostragem, é dado por $D = |\det(\underline{V})| = Q_1 Q_2 = 4$. Como conhecemos um destes por (E13), restam-nos 3 filtros a serem calculados. A Figura 2 ilustra um destes períodos e denota três amostras no plano por \underline{a} , \underline{b} e \underline{c} , sendo que cada uma corresponde a um sub-filtro polifásico, sendo interpolada por este. Suponha que o interpolador seja 7×7 , então a Figura 3 mostra as amostras de $h(\underline{n})$ correspondentes a cada sub-filtro. Nesta figura, 0's e 1's são valores reais da amostra, já computando $h_0(\underline{n})$. A análise desta figura nos leva a concluir que, para cada amostra, devemos realizar uma filtragem com 12 ou 16 coeficientes de $h(\underline{n})$, ao invés dos 49 iniciais.

Para simplificar a notação, indicaremos os coeficientes de um filtro unidimensional por um pseudo-vetor e por uma pseudo-matriz o filtro 2D. Como $h_0(n) = \delta(n)$, então:

$$h_1(n) = [a_2 \ 0 \ a_1 \ 1 \ a_1 \ 0 \ a_2] \quad (E14)$$

Para projetar o interpolador 2D, podemos utilizar Resposta Impulsiva Separável ou a Projeto pela Transformação de Mclellan [4], como em [6]. Porém, o último, se aplicado a resposta impulsiva de (E14), formará um filtro FIR 2D que não atenderá (E13). Se tentarmos encontrar uma transformação com isopotenciais menos arredondadas, iremos nos aproximar de um filtro com Resposta Impulsiva Separável. Por isto simplificamos, e fixamos

$$h(n_1, n_2) = h_1(n_1)h_1(n_2) \quad (E15)$$

$$h(n_1, n_2) = \begin{bmatrix} a_2^2 & 0 & a_1 a_2 & a_2 & a_1 a_2 & 0 & a_2^2 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ a_1 a_2 & 0 & a_1^2 & a_1 & a_1^2 & 0 & a_1 a_2 \\ a_2 & 0 & a_1 & 1 & a_1 & 0 & a_2 \\ a_1 a_2 & 0 & a_1^2 & a_1 & a_1^2 & 0 & a_1 a_2 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ a_2^2 & 0 & a_1 a_2 & a_2 & a_1 a_2 & 0 & a_2^2 \end{bmatrix} \quad (E16)$$

$$\bar{h}_g(n_1, n_2) = \begin{bmatrix} 0 & 0 & 0 & 0 \\ a_2 & a_1 & a_1 & a_2 \\ 0 & 0 & 0 & 0 \end{bmatrix} = h_g(n_1, n_2)^T \quad (E17)$$

$$\bar{h}_b(n_1, n_2) = \begin{bmatrix} a_2^2 & a_1 a_2 & a_1 a_2 & a_2^2 \\ a_1 a_2 & a_1^2 & a_1^2 & a_1 a_2 \\ a_1 a_2 & a_1^2 & a_1^2 & a_1 a_2 \\ a_2^2 & a_1 a_2 & a_1 a_2 & a_2^2 \end{bmatrix} \quad (E18)$$

Note que \bar{a} e \bar{g} seriam interpolados unidimensionalmente, mas iremos modificar a sub-amostragem de forma a sempre só utilizar $\bar{h}_b(n)$.

4. DIZIMÃO E INTERPOLAÇÃO PROGRESSIVAS

4.1. Dizimão

Nas seções anteriores consideramos, apenas, sub-amostragem retangular. Se dizimarmos progressiva e sucessivamente com $\bar{Q} = (2, 2)$ teríamos as seguintes matrizes de dizimão (sub-amo-

tragem):

$$\bar{V}_0 = \bar{I} \quad X(\bar{V}_0, n) = \text{imagem original}$$

$$\bar{V}_1 = 2\bar{I}$$

$$\bar{V}_k = 2^k \bar{I}$$

(E19)

Em $X(\bar{V}_k, n)$, a densidade de amostras coletadas por originais seria

$$d_k = \frac{1}{D} = \frac{1}{2^{2k}} \quad (E20)$$

Se, ao invés, fizéssemos

$$\bar{V}_1 = \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix}; \quad \bar{V}_2 = \begin{bmatrix} 2 & 0 \\ 0 & 2 \end{bmatrix}; \quad \bar{V}_3 = \begin{bmatrix} 2 & 2 \\ 2 & -2 \end{bmatrix}; \quad \bar{V}_4 = \begin{bmatrix} 4 & 0 \\ 0 & 4 \end{bmatrix}; \dots$$

$$\bar{V}_{2k-1} = \begin{bmatrix} 2^{k-1} & 2^{k-1} \\ 2^{k-1} & -2^{k-1} \end{bmatrix} \quad \bar{V}_{2k} = \begin{bmatrix} 2^k & 0 \\ 0 & 2^k \end{bmatrix} \quad (E21)$$

Notariamos que

$$d_{2k-1} = \frac{1}{2} \cdot \frac{1}{2^{2(k-1)}}$$

$$d_{2k} = 1/2^{2k} \quad \text{ou}$$

$$d_k = 2^{-k} \quad (E22)$$

Isto implica num maior número de passos para a formação da pirâmide, porém obteríamos duas vantagens básicas para redução da entropia da codificação. Para analisar a primeira vantagem, voltemos à figura 2. Note que com as dizimações sucessivas de (E21), reconstruímos o valor de b , para em seguida interpolar \bar{a} e \bar{g} .

Seja $V = V_1$ e Q_2 o erro de interpolação de \hat{q} . Daí

$$H(Q_2|V=V_1) = H(Q_2|V=V_1; \hat{q}) \quad (E23)$$

com $H(e)$ sendo a entropia do erro de interpolação.

V₁ configuraria dizimação hexagonal que permitiria uma interpolação mais eficiente para sinais circularmente limitados. O sinal, na realidade, não é limitado, mas a mesma afirmação vale para banda base circular, ou aproximadamente. Isto é tolerável, admitindo um modelo isotrópico para imagens.

Na figura 4, pode-se ver a dizimação hexagonal para re-composição de \hat{q} , mas todos os resultados acima podem ser estendidos para \hat{q} . Além disso o raciocínio aqui aplicado também pode ser estendido para qualquer $X(V_{2K-1}, n)$ e $X(V_{2K}, n)$.

4.2. Interpolação

O segundo fator de redução de entropia, se baseia no fato que para aproximar $X(V_{2K-1}, n)$ a partir de $X(V_{2K}, n)$ só precisamos utilizar $h_b(n)$, que é um filtro polifásico que pondera amostres bidimensionalmente e é mais eficiente. Por outro lado, para aproximar $X(V_{2K}, n)$ por $X(V_{2K-1}, n)$, também podemos utilizar apenas $h_b(n)$. Voltamos a figura 4 para interpolar $X(n)$ via $X(V_1, n)$, e, no caso desejamos interpolar \hat{q} . Adotando novos eixos de coordenadas modificamos os índices para obter $X(n_1', n_2') = X(n')$. A partir daí é fácil ver que novamente poderíamos aplicar um $h_b(n')$, desta vez ao longo das direções inclinadas.

Portanto, $h_b(n)$ seria o único interpolador utilizado na prática, já que h_0, h_a e h_c seriam dispensados. Com isto teríamos interpolações eficientes, reduzindo a entropia dos erros que compõem a pirâmide.

5. INTERPOLADORES PRÁTICOS

Como vimos, só necessitamos projetar $h_b(n)$, que deve levar em conta os recursos computacionais disponíveis. Tomamos como exemplo um filtro $h_b(n)$ de ordem 7×7 , mas, para $\hat{q} = (2, 2)$ e in-

terpolando uma das 3 amostras restantes, podemos ter ordens $2M+1 \times 2M+1$ (M ímpar) à escolha; gerados a partir de

$$h_1(n) = \left[a_{M+1}, 0, \dots, a_2, 0, a_1, 1, a_1, 0, a_2, \dots, 0, a_{M+1} \right]$$

Tal interpolador unidimensional pode ter aproximação de ordem $\frac{M+1}{2}$. Em muitos casos é necessário reduzir M ou aplicar outro esquema. Para $M=3$, podemos utilizar uma extensão discreta do Núcleo de Convolução Cúbica de Keys [7] [5], neste

$$h_1(n) = [-1/16 \quad 0 \quad 9/16 \quad 1 \quad 9/16 \quad 0 \quad -1/16]$$

gerando

$$h_b(n) = \begin{bmatrix} 1 & -9 & -9 & 1 \\ -9 & 81 & 81 & -9 \\ -9 & 81 & 81 & -9 \\ 1 & -9 & -9 & 1 \end{bmatrix} \times \frac{1}{256}$$

Se fizemos $M=1$, a melhor opção seria uma aproximação planar, ou

$$h_1(n) = [1/2 \quad 1 \quad 1/2] \quad h_b(n) = \begin{bmatrix} 1/4 & 1/4 \\ 1/4 & 1/4 \end{bmatrix}$$

Neste último caso, a amostra a ser interpolada é, simplesmente, a média das amostras próximas a seu redor. Outra forma de interpolação atendendo (E13), mas fazendo interpolação apenas ao longo de duas direções ortogonais, pode levar a bons resultados mas utilizando 8 coeficientes ao invés de 16 (no caso $M=3$). Utilizando $M=3$ e a Convolução Cúbica, teríamos

$$h(\vec{r}) = \begin{bmatrix} 0 & 0 & 0 & -1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 9 & 0 & 0 & 0 \\ -1 & 0 & 9 & 16 & 9 & 0 & -1 \\ 0 & 0 & 0 & 9 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & -1 & 0 & 0 & 0 \end{bmatrix} \times \frac{1}{16}$$

Este filtro deve ser utilizado nas sub-amostragens hexagonais nas quais o elemento central é desprezado, já que não interpolaremos amostras já existentes.

6. CONCLUSÃO

Propusemos, aqui, um método para implementação da codificação em pirâmide, baseando-nos na estrutura polifásica de um interpolador. Com isto formamos uma pirâmide reduzida (N^2 nodos), com uma redução de entropia dada pela eficiência dos filtros FIR interpoladores em face da dilatação hexagonal. A principal ideia era a de conduzir o processo de forma a sempre recarmos neste procedimento de sub-amostragem e, por tal motivo, a compressão de dados na codificação será maior do que nos métodos convencionais que exploravam dilatações retangulares e interpolações menos trabalhadas. Resta-nos, por último, ressaltar que o método aqui descrito é apenas o primeiro passo na intenção de desenvolver uma codificação em pirâmide com alto desempenho.

BIBLIOGRAFIA

- [1] D.E. Dudgeon, R.M. Mersereau; "Multidimensional Digital Signal Processing", Englewood Cliffs, NJ, Prentice-Hall, 1983.
- [2] F.J. Burt, E.H. Adelson; "The Laplacian Pyramid as a Compact Image Code", IEEE Trans. on Comm., COM-31, pp. 532-540, April 1983.

- [3] L. Wang, M. Goldberg; "Reduced-Difference Pyramid", Optical Engineering, Vol. 28, 7, July 1989.
- [4] R.E. Crochiere, L.R. Rabiner; "Multirate Digital Signal Processing", Englewood Cliffs, NJ, Prentice-Hall, 1983.
- [5] R.L. Queiroz, J.B. Yabu-uti; "Técnicas de Projeto de Filtros FIR para Dilatação e Interpolação de Sinais Discretos, RT-178, Contrato 208/89, UNICAMP/TELEBRÁS, Setembro 1989.
- [6] R.J.P. de Figueiredo, T.C. Chen; "Image Decimation and Interpolation Techniques Based on Frequency Domain Analysis", IEEE Trans. on Comm., COM-32, April 1984.
- [7] R.G. Keys; "Cubic Convolution Interpolation for Digital Image Processing", IEEE Trans. on ASSP, ASSP-29, pp. 746-749, June 1983.

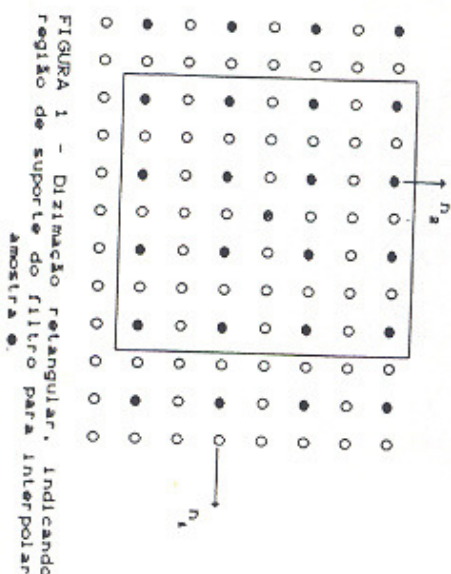


FIGURA 1 - Dilatação retangular, indicando região de suporte do filtro para interpolar amostra.

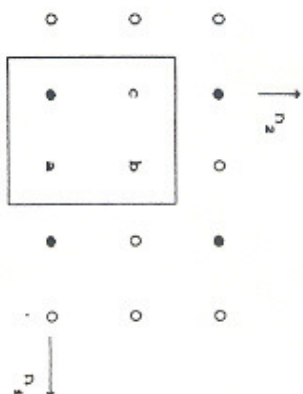


FIGURA 2 - Período de subamostragem, indicando amostras a serem interpoladas.

b	a	b	a	b	a	b
c	0	c	0	c	0	c
b	a	b	a	b	a	b
c	0	c	1	c	0	c
b	a	b	a	b	a	b
c	0	c	0	c	0	c
b	a	b	a	b	a	b

FIGURA 3 - Posição das amostras dos subfiltros de um interpolador 7x7 para diâmetro retangular com $Q=(2,2)$.

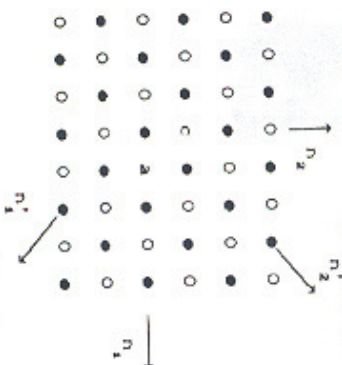


FIGURA 4 - Diâmetro Hexagonal, destacando amostras faltantes da diâmetro retangular e eixos auxiliares inclinados.

Imprimindo Imagens Coloridas em um Dispositivo Monocromático

*Claudio Esperança
João Luiz Dittl Comba*

*COPPE-Sistemas, UFRJ
Caixa Postal 68511
Rio de Janeiro - RJ - 21945*

RESUMO: Este trabalho apresenta técnicas de discretização para exibição de imagens originalmente coloridas em dispositivos de impressão monocromáticos. Uma variante do algoritmo de difusão de erros é apresentada com o fim de corrigir eventuais distorções introduzidas pelo mecanismo de impressão.

1. Introdução

Uma facilidade frequentemente desejada em um ambiente de síntese de imagens é a de reproduzir uma cópia em papel de uma imagem cromática obtida a partir de um processo qualquer. No nosso caso, por exemplo, as imagens foram obtidas por um sistema de "Ray Tracing".

Ovviamente, o meio preferencial para obter cópias em papel é o processo fotográfico. Entretanto, esse recurso é caro e em muitas ocasiões pouco prático. Nesse caso, é costume optar pela utilização de impressoras e, novamente, vemos-nos na situação em que o dispositivo ideal, isto é, impressoras coloridas, não estão normalmente disponíveis. As imagens apresentadas neste trabalho, por exemplo, foram reproduzidas numa impressora *Apple LaserWriter II*. Deparamo-nos então com o seguinte problema: dada uma imagem colorida, gerar uma representação correspondente usando apenas duas cores - branco e preto.

Claramente, a solução do problema envolve duas etapas:

- Transformar as cores dos pontos da imagem original em tons de cinza
- Dada a imagem expressa agora em tons de cinza, empregar um processo de discretização gerando uma imagem composta apenas de pontos brancos e pretos.

2. Avaliando tons de cinza

O problema de associar um tom de cinza a uma cor está sujeito a uma série de fatores, uma vez que durante o processo há necessariamente perda de informação. Há casos em que se deseja preservar visualmente o efeito de graduações de cores, ou seja, o efeito de suavidade. Em outros casos, deseja-se preservar a distinção cromática, isto é, fazer por exemplo com que tons de vermelho e verde com o mesmo grau de "luminosidade" sejam distinguíveis na imagem em tons de cinza.

Para obter resultados para todos os gostos, há basicamente dois fatores que devem ser considerados:

- Qual sistema de cores (RGB, HSV, HSB, etc.) deve ser empregado para representar as cores da imagem
- Qual função a ser empregada para, dados as três componentes de cor primária, avaliar o tom de cinza correspondente

Fazemos experimentos com os sistemas de cores RGB (*Red Green Blue*) e HSV (*Hue Saturation Value*). Observamos que no sistema RGB, cores com intensidades próximas tendem a se apresentar em