Transformadas de Imagens

Professor: Wemerson D. Parreira.
parreira@univali.br

Universidade do Vale do Itajaí
Escola do Mar, Ciência e Tecnologia

2022

Neste capítulo abordaremos:

- as transformadas de imagens e algumas aplicações (extração de características, compressão de dados, segmentação e filtragem de imagens);
- os Conceitos gerais das transformações de coordenadas;
- a Transformada de Fourier (principais conceitos e propriedades);
- a Transformada Rápida de Fourier;
- Outras Transformadas (Wash, de Hadamard, do Cosseno Discreta).

Frequência

- em **sinais unidimensionais** (ex.: ondas eletromagnéticas, fala, ...): número de vibrações em um determinado intervalo de tempo.
- em **sinais bidimensionais** (ex.: imagens): quantidade de variações na intensidade de pixels, considerando regiões de tamanhos específicos.

Frequência Espacial – medida da periodicidade de um conjunto de dados com respeito a uma medida de distância. Mudanças periódicas em valores de brilho em uma imagem são definidos em termos de frequência espacial.

➪ regiões da imagem que apresentam homogeneidade nos tons de cinza apresentam frequência espacial baixa, ou simplesmente frequência baixa.

Trata-se de um tipo de **transformação linear**

➪ Uma transformação obtida pela combinação linear dos dados da entrada com os coeficientes pré-definidos (varia de acordo com a transformação)

Transformações de Coordenadas

- São classificadas, quanto a dimensionalidade da entrada, como:
  - **Unidimensional:** quando os dados da entrada é um vetor

$$\mathbf{y}^\top = \mathbf{A}\mathbf{x}^\top \text{ ou } y_u = \sum_{n=0}^{N-1} a_{u,n} x_n \text{ com } u = 0, 1, \ldots N-1$$

em que $\mathbf{A}$ contém o núcleo da transformada e $\mathbf{x}$ e $\mathbf{y}$ representam respectivamente a entrada e saída da transformação

  - **Bidimensional:** quando os dados da entrada é uma matriz

$$\mathbf{Y} = \mathbf{B}(u, v)\mathbf{X} \text{ ou } \mathbf{y}_{u,v} = \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} b^{(u,v)}(m, n) x_{m,n}$$

$$\text{com } u = 0, 1, \ldots, M-1 \text{ e } v = 0, 1, \ldots, N-1$$

em que $\mathbf{B}(u, v)$ contém o núcleo da transformada e $\mathbf{X}$ e $\mathbf{Y}$ representam respectivamente a entrada e saída da transformação

Núcleo da transformada

➪ O núcleo da transformada será a matriz da base de destino ou sua inversa.

1. Caso unidimensional: O núcleo de uma transformada unidimensional é representado por uma matriz $\mathbf{A}$, $N \times N$ contendo em suas linhas os vetores $\boldsymbol{a}_u = [a_{u,0},\ a_{u,1},\ \ldots,\ a_{u,N-1}]$ em que $u = 0, 1, \ldots, N-1$.

Núcleo da transformada

Exemplo: Considere a transformada cosseno, que apresenta os vetores do núcleo definidos por:

$$\boldsymbol{a}_{u,n} = k(n)\sqrt{\frac{2}{N}}\cos\left(\frac{\pi n(u+0.5)}{N}\right)$$

em que $n = 0, 1, \ldots, N-1$ e $k(n) = 1/\sqrt{2}$ se $n = 0$ e $k(n) = 1$, caso contrário.

Assim, o núcleo da transformada discreta do cosseno para um vetor de entrada com 8 componentes é presentado pela matriz $\mathbf{A}$,

$$A = \frac{1}{\sqrt{8}}\begin{bmatrix} a_{0,0} & a_{0,1} & a_{0,2} & \ldots & a_{0,7} \\ a_{1,0} & a_{1,1} & a_{1,2} & \ldots & a_{1,7} \\ a_{2,0} & a_{2,1} & a_{2,2} & \ldots & a_{2,7} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ a_{7,0} & a_{7,1} & a_{7,2} & \ldots & a_{7,7} \end{bmatrix}$$

➪ A matriz $\mathbf{A}$ pode ser representada graficamente por meio de N planos cartesianos, um para cada vetor no núcleo da transformação.

Núcleo da transformada

Exercício:

(a) Calcule os elementos da matriz $\mathbf{A}$ representada anteriormente usando a linguagem de programação de sua escolha.

(b) Represente graficamente cada uma das componentes do núcleo.

2. Caso bidimensional: O núcleo de uma transformada bidimensional é composto por uma função matricial matriz $\mathbf{B}(u,v)$, que depende das coordenadas do ponto para qual a transformada está sendo executada.

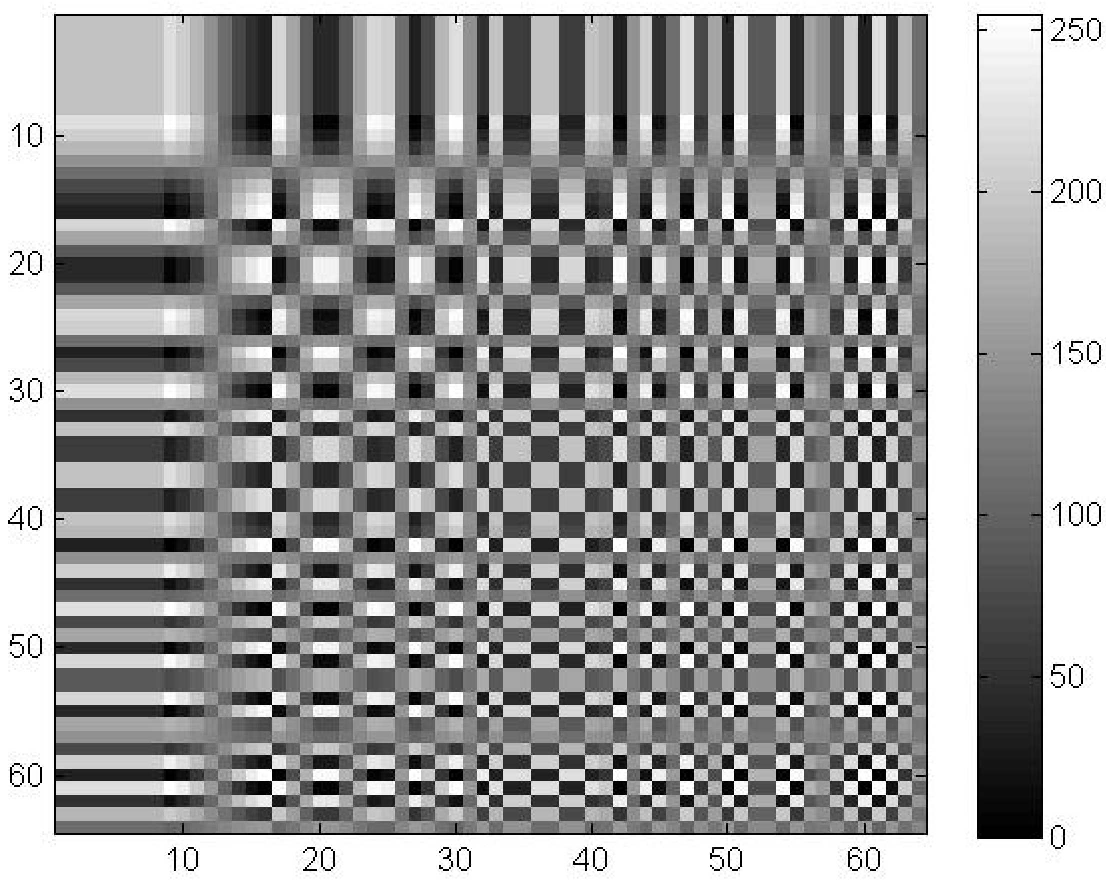
$$\mathbf{B}(u,v) = \begin{bmatrix} b^{(u,v)}(0,0) & b^{(u,v)}(0,1) & \dots & b^{(u,v)}(0,N-1) \\ b^{(u,v)}(1,0) & b^{(u,v)}(1,1) & \dots & b^{(u,v)}(1,N-1) \\ \vdots & \vdots & \ddots & \vdots \\ b^{(u,v)}(M-1,0) & b^{(u,v)}(M-1,1) & \dots & b^{(u,v)}(M-1,N-1) \end{bmatrix}$$

em que $u, v = 0, 1, \dots, N-1$ representam as coordenas do ponto em que está sendo executada a transformada e cada componente $\boldsymbol{b}^{(u,v)}(m,n)$ depende da transformada em questão.

➪ Uma representação gráfica da função matricial $\mathbf{B}(u,v)$ pode ser obtida por meio de uma imagem em tons de cinza composta por $U \times V$ blocos, cada um composto por $M \times N$ pixels, apresentando intensidade proporcional ao valores obtidos pela instanciação da função $\mathbf{B}(u,v)$.

Núcleo da transformada

Exemplo: o núcleo da transformada discreta do cosseno para uma matriz de entrada com $N = M = U = V = 8$ é presentado pela imagem,



Exercício:

(a) Calcule os elementos da matriz $\mathbf{B}(u, v)$ representada anteriormente usando a linguagem de programação de seu interesse.

(b) Represente graficamente o núcleo da transformada discreta do cosseno para uma matriz de entrada de ordem 8 como apresentado anteriormente (use normalização com 256 níveis de cinza).

Núcleo da transformada

Separabilidade:

O núcleo de algumas transformadas bidimensionais pode ser decomposto em duas matrizes C e $\mathbf{D}$, permitindo que sua execução seja efetuada em duas etapas, inicialmente sobre as linhas da matriz de entrada e então sobre as colunas da matriz resultante da primeira aplicação

➪ As transformadas que apresentam essa propriedade são denominadas transformadas separáveis.

Exemplo: O núcleo da transformada bidimensional de Fourier é obtido por meio da função $b^{(u,v)}(m,n) = \exp(-2\pi j(mu + nv)/N)$, que pode ser decomposto em exponenciais que dependem apenas de n,u e m,v

$$b^{(u,v)}(m,n) = \exp(-2\pi jv/N)\exp(-2\pi jmu/N)$$

➪ A vantagem de poder executar esse tipo de operação está na redução do custo computacional. (posteriormente voltaremos a este assunto)

Transformada Discreta de Fourier – DFT

- Capaz de descrever funções a partir de exponenciais complexas.
- Trata-se de uma especificação da série de Fourier (DFS – *Discrete Fourier Series*).
- Usada inicialmente em processamento de sinais para determinar as frequências presentes em uma dada função unidimensional.
- A versão bidimensional, usada em PDI, provê informações a respeito das variações na intensidade de cinzas.
    - Trata-se de uma transformação de coordenadas que resulta em componentes pertencentes ao conjunto dos números $\mathbb{C}$ (Complexos), em que cada coeficiente é obtido pela combinação linear dos elementos da entrada com o núcleo da transformada.

- Apesar dos componentes complexos resultantes da DFT não serem normalmente utilizados na forma $z = a + bj$, utiliza-se
  - a fase, representada pelo argumento de z

$$arg\, z = \text{arctan}\left(\frac{b}{a}\right)$$

    contendo informações essenciais sobre a estrutura da imagem;
  - o espectro de Fourier, o módulo de z,

$$|z| = \sqrt{a^2 + b^2}$$

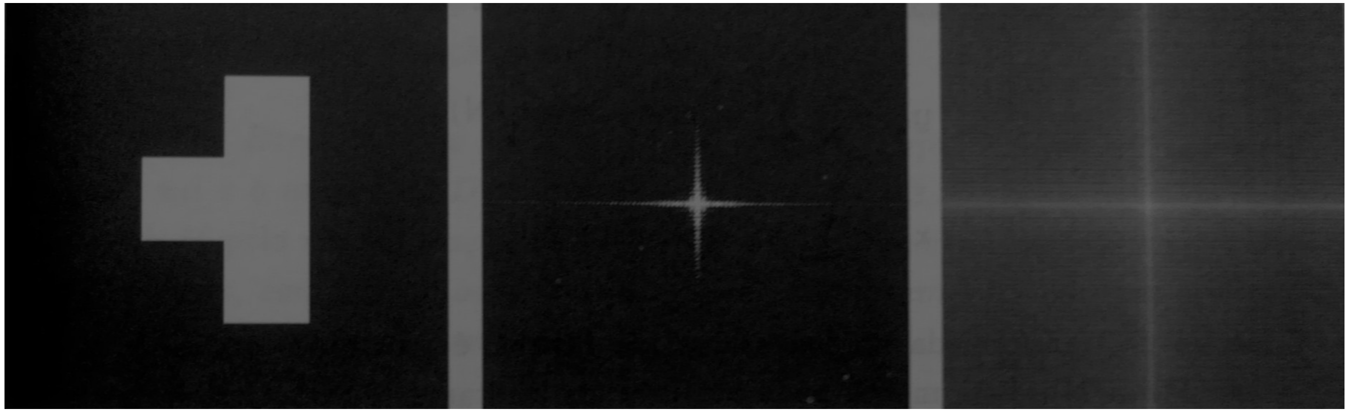
    que descreve informações sobre as frequências especiais contidas na imagem.

- Dado que os coeficientes do espectro de Fourier apresentam valores com magnitude distintas, a visualização dos elementos mais claros é possível somente quando representamos o espectro de Fourier na forma de imagem em tons de cinza.

- Aplica-se a função

$$\log\left[1+\sqrt{(a^2+b^2)}\right]$$

Exemplo: Representação do espectro de Fourier para uma imagem monocromática:



- Se uma sequência não periódica de tamanho finito for representada pela série discreta de Fourier, esta será chamada de transformada discreta de Fourier.

$$\textbf{DFT: } \overline{\mathbf{x}}[n] = \begin{cases} \mathbf{x}[n], & 0 \leq n \leq N-1 \\ 0, & \text{caso contrário} \end{cases}$$

$$\textbf{IDFT: } \overline{\mathbf{y}}[k] = \begin{cases} \mathbf{y}[k], & 0 \leq k \leq N-1 \\ 0, & \text{caso contrário} \end{cases}$$

- No caso de imagens, geralmente se desconsidera a existência de periodicidade, ou seja, $f(x,0) \neq f(x+N,0)$, para uma imagem $N \times N$.
- Como se tem o interesse apenas no intervalo finito $0 \leq x \leq N-1$ e $0 \leq y \leq N-1$, a DFT pode ser utilizada para análise da f.

- Núcleo da transformada unidimensional de Fourier

$$
\boldsymbol{F} = \frac{1}{N}\begin{bmatrix} 1 & \omega_1^{-1} & \omega_1^{-2} & \dots & \omega_1^{-(N-1)} \\ 1 & \omega_2^{-1} & \omega_2^{-2} & \dots & \omega_2^{-(N-1)} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & \omega_{N-1}^{-1} & \omega_{N-1}^{-2} & \dots & \omega_{N-1}^{-(N-1)} \end{bmatrix}
$$

com $\omega_k^n = \exp(2\pi jkn/N)$.

- A matriz com o núcleo da T.F. pode ser utilizada para determinação dos coeficientes de Fourier de uma função discreta unidimensional
- Tais coeficientes descrevem o comportamento desta função por meio de exponenciais complexas

Exercício

Implemente a transformada de Fourier e a transformada Inversa de Fourier para sinais unidimensionais discretos de dimensão $N \times 1$ usando ma matriz núcleo da transformação de Fourier – $\boldsymbol{F}$.

$$\mathcal{F}\{\mathbf{x}\} = \boldsymbol{F}\,\mathbf{x}$$

e

$$\mathcal{F}^{-1}\{\mathbf{y}\} = \boldsymbol{F}^{-1}\,\mathbf{y}$$

(a) Exiba para $N = 8$ as matrizes $\boldsymbol{F}$ e $\boldsymbol{F}^{-1}$

(b) Gere uma sequência aleatória de valores para testar a sua implementação.

- O resultado DFT unidimensional expressa quais frequências estão presentes na função de entrada

  ➪ quanto maior a frequência apresentada pela função de entrada, o espectro de Fourier apresentará componentes em regiões mais distantes da origem

- Em processamento de imagem, o objetivo normalmente está na obtenção de informações sobre quais frequências espaciais presentes na imagem ou em qual orientação ocorrem as maiores variações de intensidade de tons de cinza

  ➪ Após este procedimento, pode -se por exemplo, aplicar um filtro no domínio da frequência visando eliminar as componentes que apresentam frequências indesejadas ao experimento efetuado

3.2 Transformada Discreta de Fourier: caso bidimensional

- Extensão da versão unidimensional, a transformada bidimensional de Fourier pode ser aplicada diretamente em imagens sem que haja a necessidade de transformar os dados de entrada em um vetor antes da obtenção dos coeficientes de Fourier.
- Transformada Fourier:

$$\mathbf{Y}[u,v] = \frac{1}{N^2}\sum_{m=0}^{N-1}\sum_{n=0}^{N-1}\mathbf{X}[m,n]\exp(-2\pi j(mu+nv)/N)$$

- Transformada Inversa de Fourier:

$$\mathbf{X}[m,n] = \sum_{v=0}^{N-1}\sum_{u=0}^{N-1}\mathbf{Y}[u,v]\exp(2\pi j(mu+nv)/N)$$

- Assim como na versão unidimensional, a expressão $\exp(-2\pi j(mu+nv)/N)$ que compõe o núcleo da DFT-2D, pode ser decomposta na soma $cos(-2\pi(mu+nv)/N)+jsen(-2\pi(mu+nv)/N)$.

Exercício

As componentes do núcleo da DFT-2D podem ser presentadas separadamente por meio de duas imagens monocromáticas, uma representado as componentes reais do núcleo e outra representando as componentes complexas do núcleo. Obtenha essas imagens para uma matriz de entrada de ordem 8.

(exercício análogo ao feito em sala para a Transformada Discreta do Cosseno)

Transformada Discreta de Fourier: caso bidimensional

- O Espectro de Fourier obtido a partir da DFT bidimensional prove informações sobre a orientação das estruturas presentes na imagem de entrada.

- Efetua o mapeamento das variações nos tons de cinza dos pixels.

  ➪ Um número pequeno de variações de intensidade de cinza em um determinado espaço indica a presença de regiões de baixa frequência espacial, enquanto um número maior de variações indica a presença de regiões de frequência espacial alta na imagem.

1 – Separabilidade: Esta propriedade mostra a DFT pode ser executada em duas etapas.

$$\mathbf{Y}[u,v] = \frac{1}{N^2}\sum_{m=0}^{N-1}\exp(-2\pi jmu/N)\sum_{n=0}^{N-1}\mathbf{X}[m,n]\exp(-2\pi jnv/N)$$

com núcleos dados por

$$\boldsymbol{E}[m,v] = \sum_{n=0}^{N-1}\mathbf{X}[m,n]\exp(-2\pi jnv/N)$$

$$\boldsymbol{Y}[u,v] = \frac{1}{N^2}\sum_{m=0}^{N-1}\boldsymbol{E}[m,v]\exp(-2\pi jmu/N)$$

as quais executam a transformada sobre as colunas e linhas da imagem.

➪ Este tipo de propriedade reduz o número de operações de $\mathcal{O}(N^4)$ para $\mathcal{O}(N^3)$.

2 – Periodicidade e simetria conjugada:
Dado
$$\mathcal{F}\{f(x,y)\} = F(u,v)$$
A transformada discreta de Fourier e sua inversa são periódicas, com período N. Ou seja,

$$F(u,v) = F(u+N,v) = F(u,v+N) = F(u+N,v+N)$$

Se $f(x,y)$ é real, sua transformada de Fourier exibe também a propriedade conhecida como simetria conjugada:

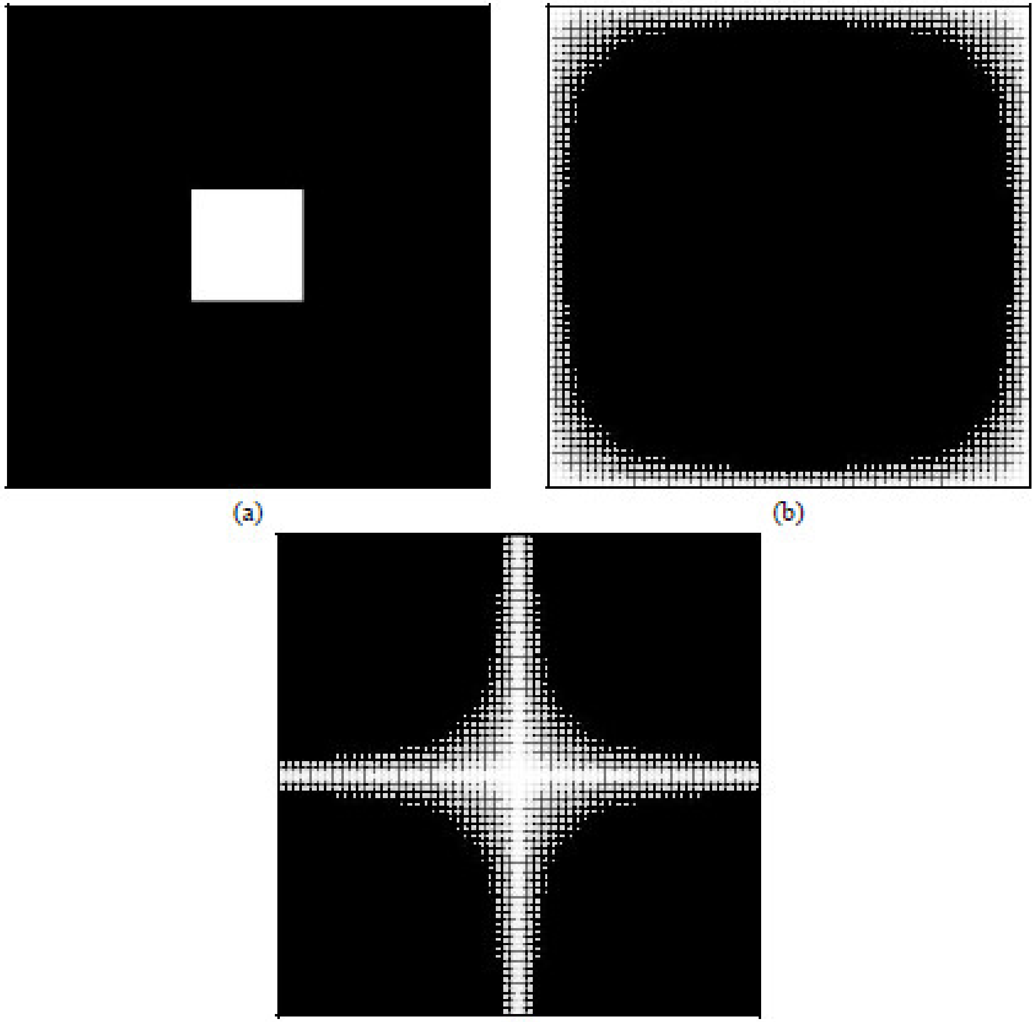
$$F(u,v) = F^*(-u,-v)$$

ou

$$|F(u,v)| = |F(-u,-v)|$$

em que $F^*(u,v)$ é o conjugado complexo de $F(u,v)$.

➪ A combinação das propriedades da translação e da periodicidade e a conveniência de sua utilização para fins de visualização podem ser ilustradas



(a) (b) (c)

3 – Distributividade:

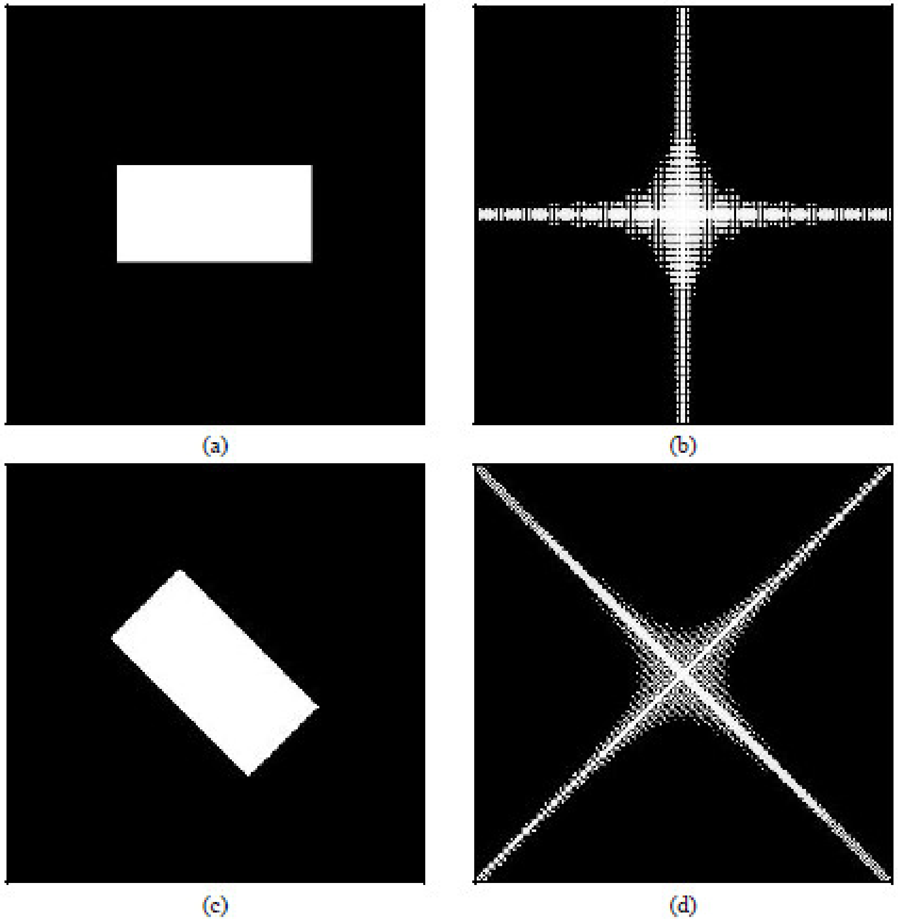
A FT obedece à propriedade distributiva para a adição, mas não para a multiplicação, ou seja:

$$\mathcal{F}\{f(x,y)+g(x,y)\}=\mathcal{F}\{f(x,y)\}+\mathcal{F}\{g(x,y)\}=F(u,v)+G(u,v)$$

➪ Em geral $\mathcal{F}\{f(x,y).g(x,y)\}\neq F(u,v).G(u,v)$.

4 – Rotação:

A propriedade da rotação estabelece que, se uma imagem $f(x, y)$ for rotacionada de um certo ângulo θ_0, sua transformada, $F(u, v)$, será rotacionada do mesmo ângulo, θ_0.



5 – Escala:

Sejam dois escalares a e b. Pode-se mostrar que:

$$af(x, y) \Leftrightarrow aF(u, v)$$

e

$$f(ax, by) \Leftrightarrow \frac{1}{|ab|} F\left(\frac{u}{a}, \frac{v}{b}\right)$$

6 – Valor médio:

O valor médio de uma função bidimensional $f(x,y)$ é dado por:

$$\overline{f}(x,y) = \frac{1}{N^2}\sum_{x=0}^{N-1}\sum_{y=0}^{N-1} f(x,y)$$

Pela propriedade da separabilidade temos

$$F(u,v) = \frac{1}{N}\sum_{x=0}^{N-1}\exp\left[\frac{-j2\pi xu}{N}\right]\sum_{y=0}^{N-1}\exp\left[\frac{-j2\pi yv}{N}\right] f(x,y)$$

com $u = v = 0$ obtemos

$$F(0,0) = \frac{1}{N}\sum_{x=0}^{N-1}\sum_{y=0}^{N-1} f(x,y)$$

Logo, o valor médio de uma função 2-D está relacionado à sua FT a partir da relação

$$\overline{f}(x,y) = \frac{1}{N}F(0,0).$$

7 – Laplaciano:

O laplaciano de uma função de duas variáveis f(x,y) é definido como:

$$\nabla^2 f(x,y) = \frac{\partial^2 f}{\partial x^2} + \frac{\partial^2 f}{\partial y^2}$$

A FT do laplaciano de uma função bidimensional é

$$\mathcal{F}\{\nabla^2 f(x,y)\} \Leftrightarrow -(2\pi)^2(u^2+v^2)F(u,v)$$

➪ O laplaciano é um operador útil para detecção de bordas

8 – Convolução:

Pelo teorema da convolução, caso unidimensional temos:

$$\begin{cases} f(x) * g(x) \Leftrightarrow F(u).G(u) \\ f(x).g(x) \Leftrightarrow F(u) * G(u). \end{cases}$$

Para o caso bidimensional, temos:

$$\begin{cases} \mathcal{F}\{f(x,y) * g(x,y)\} \Leftrightarrow F(u,v).G(u,v) \\ \mathcal{F}\{f(x,y).g(x,y)\} \Leftrightarrow F(u,v) * G(u,v). \end{cases}$$

Filtragem

- Filtragem Passa-baixa
- Filtragem Passa-faixa
- Filtragem Passa-alta

Algumas Aplicações da Tranformada de Fourier

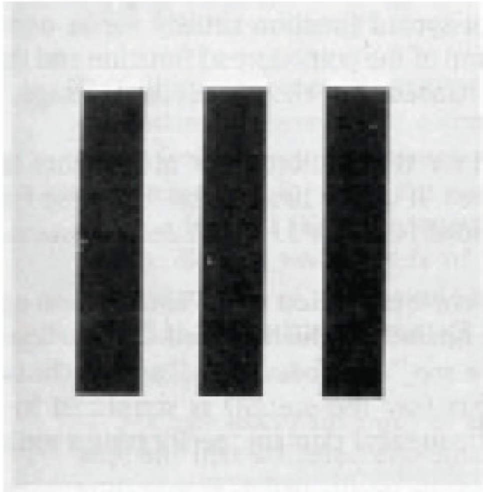


Imagem Original

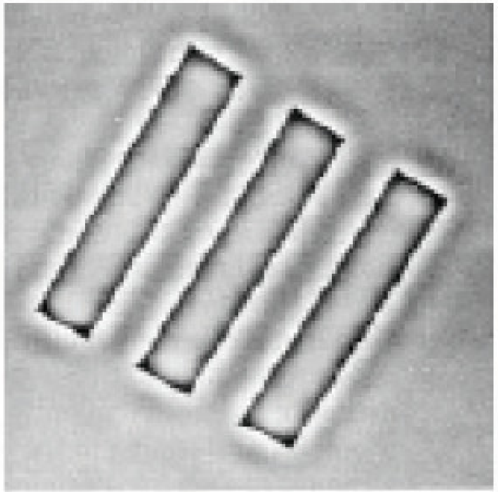


Imagem Filtrada
(Passa-Alta)

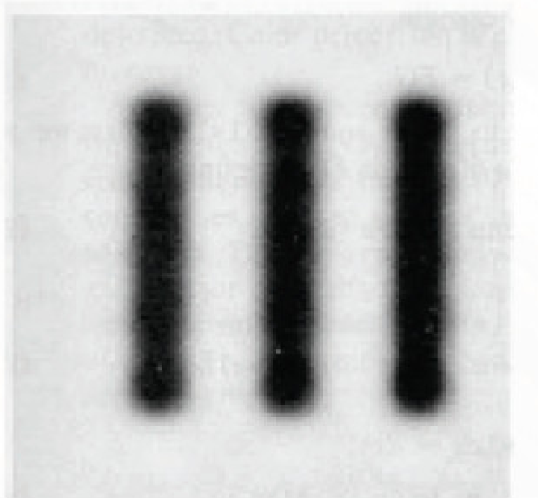
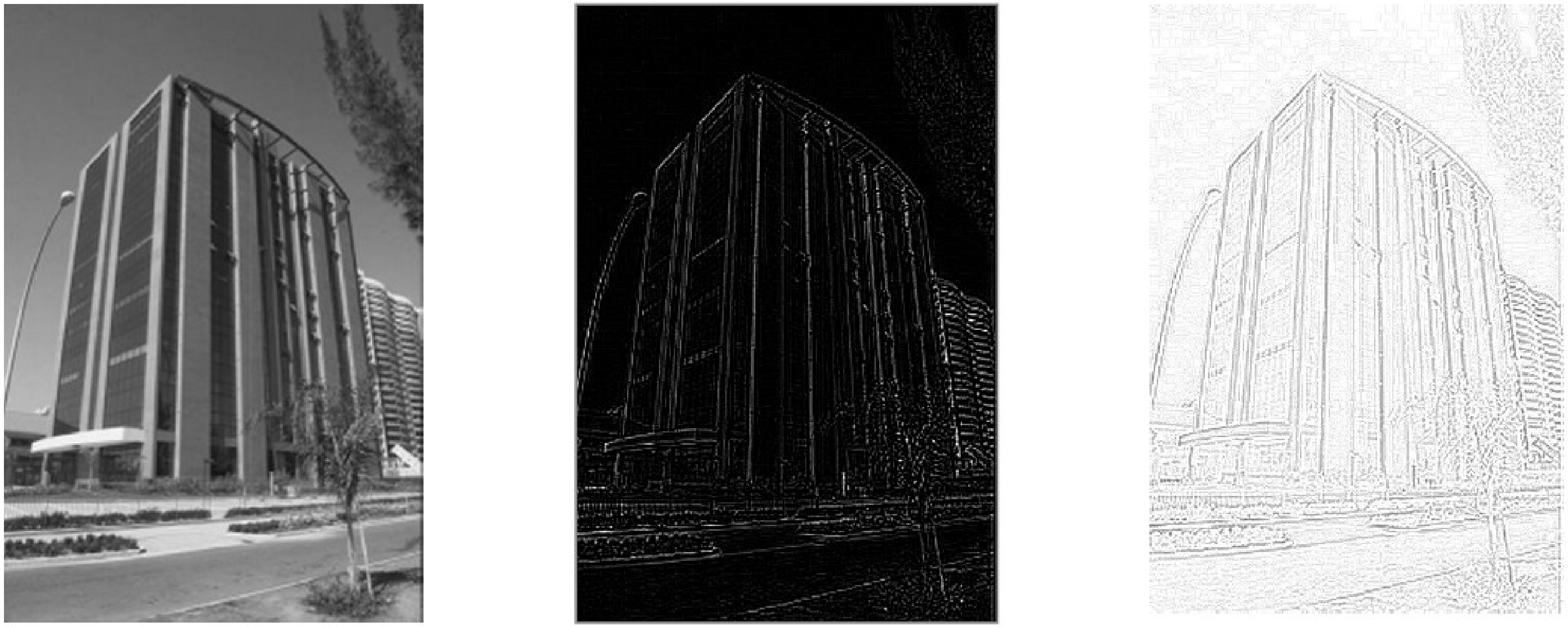


Imagem Filtrada
(Passa-Baixa)

Filtragem Passa-Alta
(realce de contornos, bordas):



- Trata-se de um algoritmo cujo principal objetivo é reduzir o custo computacional do cálculo da FT de N pontos
- É obtida substituindo o processo convencional de cálculo, no qual o número de multiplicações e adições é proporcional a $(\mathcal{N}^2)$ por um engenhoso arranjo que combina diversas transformadas parciais
- Cada transformada parcial com pequeno número de pontos, em que o número de adições e multiplicações é proporcional a $N \log_2 N$
- Para se poder apreciar a diferença em velocidade entre os algoritmos, pode-se supor $N = 512$ pontos, verificando que neste caso a FFT é mais de 56 vezes mais rápida
- A maioria das implementações prontas usa o algoritmo FFT para calcular a DFT.

Exercício

Usando a linguagem de programação do seu interesse e uma imagem também escolhida por você em níveis de cinza, execute os seguintes procedimentos:

a. Aplique a transformada de Fourier na imagem escolhida, I;
b. Obtenha o valor médio da imagem I, a partir da Transformada de Fourier e verifique;
c. Verifique a propriedade de Separabilidade;
d. Verifique a propriedade de Rotação (escolha o ângulo $\theta_0 > 0^o$) para o teste.
e. Verifique a propriedade de mudança de escala.

Considerando $N = 2^n$, a transformada discreta de Walsh de uma função $f(x)$, denotada por $W(u)$, é obtida por,

$$W(u) = \frac{1}{\sqrt{N}} \sum_{x=0}^{N-1} f(x) \prod_{k=0}^{n-1} (-1)^{b_k(x) b_{n-1-k}(u)}$$

e sua inversa dada por,

$$f(x) = \frac{1}{\sqrt{N}} \sum_{u=0}^{N-1} W(u) \prod_{k=0}^{n-1} (-1)^{b_k(x) b_{n-1-k}(u)}$$

em que

$\frac{1}{\sqrt{N}} \prod_{k=0}^{n-1} (-1)^{b_k(x) b_{n-1-k}(u)}$ representa o núcleo da transformação discreta de Walsh e

$b_p(z)$ é o p-ésimo bit na representação binária de z (*Ex.:* se $z = 6$ e portanto 110 em binário temos $b_0(z) = 0$, $b_1(z) = 1$ e $b_2(z) = 1$).

➪ Note que o núcleo da transformação direta e inversa são idênticos.

Exemplo: Considerando $N = 8$ o núcleo da transformação discreta de Walsh – DWT será:

$$\mathbf{K}_W = \frac{1}{\sqrt{8}} \begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & -1 & -1 & -1 & -1 \\ 1 & 1 & -1 & -1 & 1 & 1 & -1 & -1 \\ 1 & 1 & -1 & -1 & -1 & -1 & 1 & 1 \\ 1 & -1 & 1 & -1 & 1 & -1 & 1 & -1 \\ 1 & -1 & 1 & -1 & -1 & 1 & -1 & 1 \\ 1 & -1 & -1 & 1 & 1 & -1 & -1 & 1 \\ 1 & -1 & -1 & 1 & -1 & 1 & 1 & -1 \end{bmatrix}$$

Observações:

- A matriz $\mathbf{K}_W$, matriz núcleo da transformação de Wash é uma matriz simétrica tendo linhas e colunas ortogonais.
- Diferentemente da FT, que se baseia em exponenciais complexas, a WT se baseia em expansão por séries de funções cujos valores são $+1$ e -1.
- Considerando a igualdade entre os núcleos, da transformação direta e inversa, qualquer algoritmo que calcula DWT também calcula a sua inversa – IDWT.

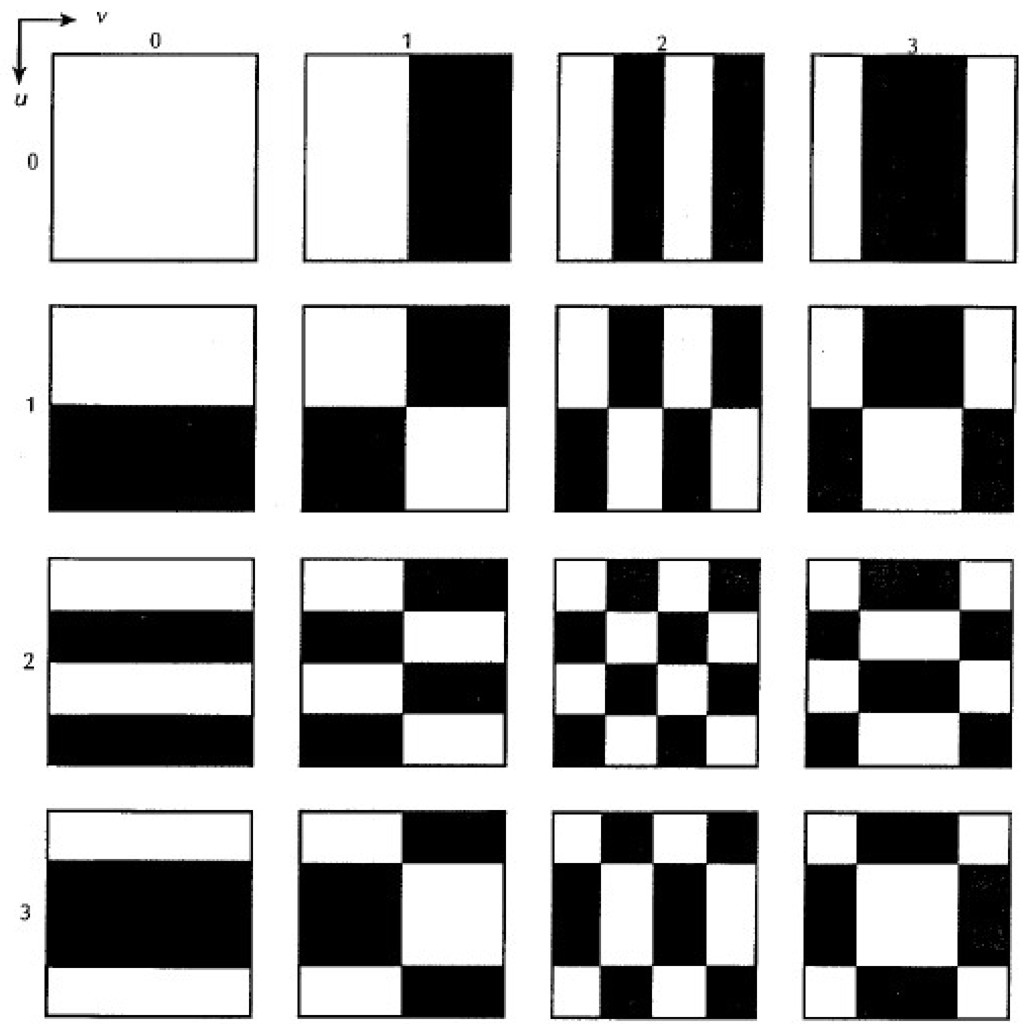
Os núcleos da DWT-2D são separáveis e simétricos por que

$$\begin{aligned}
[\mathbf{K}_W]_{x,y,u,v} &= \frac{1}{N}\prod_{k=0}^{n-1}(-1)^{[b_k(x)b_{k-1-i}(u)+b_k(y)b_{k-1-i}(v)]} \\
&= \frac{1}{N}\prod_{k=0}^{n-1}(-1)^{b_k(x)b_{k-1-i}(u)}(-1)^{b_k(y)b_{k-1-i}(v)} \\
&= \left[\frac{1}{\sqrt{N}}\prod_{k=0}^{n-1}(-1)^{b_k(x)b_{k-1-i}(u)}\right]\left[\frac{1}{\sqrt{N}}\prod_{k=0}^{n-1}(-1)^{b_k(y)b_{k-1-i}(v)}\right].
\end{aligned}$$

➪ Portanto a DWT - 2D possui a propriedade de separabilidade. Logo, $W(u,v)$ e sua inversa podem ser computadas por aplicações sucessivas da transformada de Walsh unidimensional.

➪ Note também que, o núcleo das transformações direta e inversa, dependem apenas dos índices x, y, u e v de modo que os núcleos servem como um conjunto de funções cuja natureza fica completamente determinada pelas dimensões da imagem.

Exemplo: Funções base de Walsh para $N = 4$. Cada bloco consiste de elementos 4×4, correspondendo a x e y variando de 0 a 3. A origem de cada bloco no seu canto superior esquerdo. Branco e preto denotam $+1$ e -1.



Transformada Discreta de Hadamard

Considerando $N = 2^n$, a transformada discreta de Hadamard de uma função $f(x)$, denotada por $H(u)$, é obtida por,

$$H(u) = \frac{1}{\sqrt{N}} \sum_{x=0}^{N-1} f(x)(-1)^{\sum_{k=0}^{n-1} b_k(x) b_k(u)}$$

e sua inversa dada por,

$$f(x) = \frac{1}{\sqrt{N}} \sum_{u=0}^{N-1} H(u)(-1)^{\sum_{k=0}^{n-1} b_k(x) b_k(u)}$$

em que

$\frac{1}{\sqrt{N}}(-1)^{\sum_{k=0}^{n-1} b_k(x) b_k(u)}$ representa o núcleo da transformação discreta de Walsh e

$b_p(z)$ é o p-ésimo bit na representação binária de z.

➪ Note que o núcleo da transformação direta e inversa também são idênticos (como na DWT).

Exemplo: Considerando $N = 8$ o núcleo da transformação discreta de Hadamard – DHT unidimensional será:

$$\mathbf{K}_H = \frac{1}{\sqrt{8}} \begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & -1 & 1 & -1 & 1 & -1 & 1 & -1 \\ 1 & 1 & -1 & -1 & 1 & 1 & -1 & -1 \\ 1 & -1 & -1 & 1 & 1 & -1 & -1 & 1 \\ 1 & 1 & 1 & 1 & -1 & -1 & -1 & -1 \\ 1 & -1 & 1 & -1 & -1 & 1 & -1 & 1 \\ 1 & 1 & -1 & -1 & -1 & -1 & 1 & 1 \\ 1 & -1 & -1 & 1 & -1 & 1 & 1 & -1 \end{bmatrix}$$

Observações:

- A matriz $\mathbf{K}_H$, matriz núcleo da transformação de Hadamard, assim como $\mathbf{K}_W$ também é uma matriz simétrica tendo linhas e colunas ortogonais.

  ➪ Apesar de possuir as mesmas entradas que a DWT, $+1$ e -1, a disposição é diferente.

  ➪ Da mesma maneira que a DWT, o cálculo da transformação direta e inversa pode ser feita pelo mesmo algoritmo, ou seja, o algoritmo que calcula DHT também calcula a sua inversa – IDHT.

Transformada Bidimensional Discreta de Hadamard – DHT-2D

Considerando $N = 2^n$, a transformada discreta de Hadamard de uma função $f(x,y)$, denotada por $H(u,v)$, é obtida por,

$$H(u,v) = \frac{1}{N}\sum_{x=0}^{N-1}\sum_{y=0}^{N-1} f(x,y)(-1)^{\sum_{k=0}^{n-1}[b_k(x)b_k(u)+b_k(y)b_k(v)]}$$

e sua inversa dada por,

$$f(x) = \frac{1}{N}\sum_{u=0}^{N-1}\sum_{u=0}^{N-1} H(u,v)(-1)^{\sum_{k=0}^{n-1}[b_k(x)b_k(u)+b_k(y)b_k(v)]}$$

em que

$\frac{1}{N}(-1)^{\sum_{k=0}^{n-1}[b_k(x)b_k(u)+b_k(y)b_k(v)]}$ representa o núcleo da THD - 2D e

$b_p(z)$ é o p-ésimo bit na representação binária de z.

➪ Da mesma maneira que na DWT-2D, o núcleo da transformação direta e inversa são idênticos na THD-2D.

Transformada Bidimensional Discreta de Hadamard – DHT-2D

Os núcleos da DHT-2D são separáveis e simétricos por que

$$
\begin{aligned}
[\mathbf{K}_W]_{x,y,u,v} &= \frac{1}{N}(-1)^{\sum_{k=0}^{n-1}[b_k(x)b_k(u)+b_k(y)b_k(v)]} \\
&= \frac{1}{N}(-1)^{\sum_{k=0}^{n-1} b_k(x)b_k(u)+\sum_{k=0}^{n-1} b_k(y)b_k(v)} \\
&= \left[\frac{1}{\sqrt{N}}(-1)^{\sum_{k=0}^{n-1} b_k(x)b_k(u)}\right]\left[\frac{1}{\sqrt{N}}(-1)^{\sum_{k=0}^{n-1} b_k(y)b_k(v)}\right].
\end{aligned}
$$

➪ Portanto a DHT - 2D também possui a propriedade de separabilidade. Logo, $H(u,v)$ e sua inversa podem ser computadas por aplicações sucessivas da transformada de Hadamard unidimensional.

Transformada Bidimensional Discreta de Hadamard – DHT-2D

Observações

- Diferentemente da Transformada Walsh que pode ser formulada para qualquer valor inteiro N, a tranformada de Hadamard é válida sempre que N for uma potência de 2, caso contrário só temos garantias da existêcia da DHT, na literatura, até $N = 200$.
- Como a maioria das aplicações de transformada em processamento de imagens são baseadas em $N = 2^n$ amostras por linha ou coluna de uma imagem, a terminologia Transformadas Walsh e Hadamard não contém distinção na literatura de processamento de imagens, o termo Transformada Walsh-Hadamard frequentemente é usada para denotar ambas as transformadas.

Transformada Bidimensional Discreta de Hadamard – DHT-2D

- Embora a Transformada Wash permita a possibilidade uma versão rápida FWT, com pequenas modificações na FFT o mesmo não acontece com a TH. Porém, uma relação recursiva simples pode ser usada para gerar a matrizes de Hadamard – H.

$$H_2 = \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix},$$

$$H_4 = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & -1 & 1 & -1 \\ 1 & 1 & -1 & -1 \\ 1 & -1 & -1 & 1 \end{bmatrix} = \begin{bmatrix} H_2 & H_2 \\ H_2 & -H_2 \end{bmatrix}$$

$$\vdots$$

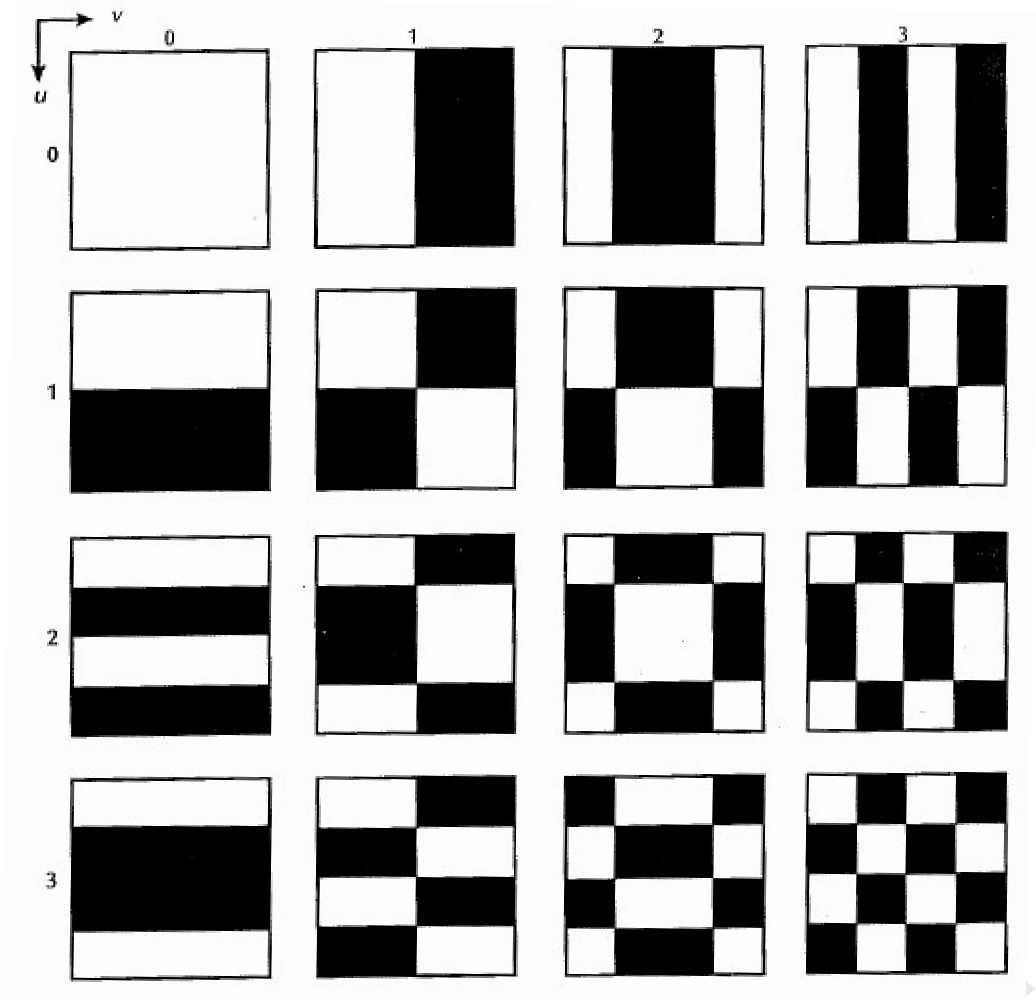
$$H_{2M} = \begin{bmatrix} H_M & H_M \\ H_M & -H_M \end{bmatrix}.$$

Para ordem $N = 2M$ obtemos a matriz núcleo da transformação fazendo:

$$\mathbf{K}_H = \frac{1}{\sqrt{2M}} H_{2M}.$$

Transformada Bidimensional Discreta de Hadamard – DHT-2D

Exemplo: Funções base de Hadamard para $N = 4$. Cada bloco consiste de elementos 4×4, correspondendo a x e y variando de 0 a 3. A origem de cada bloco no seu canto superior esquerdo. Branco e preto denotam $+1$ e -1.



A transformada Discreta do Cosseno – DCT, denotada por $C(u)$ é definida como:

$$C(u) = \alpha(u) \sum_{x=0}^{N-1} f(x) \cos\left[\frac{(2x+1)u\pi}{2N}\right]$$

com $u = 0, 1, 2, \ldots, N-1$ e sua inversa, IDCT dada por,

$$f(x) = \sum_{u=0}^{N-1} \alpha(u) C(u) \cos\left[\frac{(2x+1)u\pi}{2N}\right]$$

com $x = 0, 1, 2, \ldots, N-1$, em que

$$\alpha(u) = \begin{cases} \sqrt{\frac{1}{N}}, & \text{para } u = 0 \\ \sqrt{\frac{2}{N}}, & \text{para } u = 1, 2, 3, \ldots, N-1. \end{cases}$$

Exemplo: Considerando $N = 8$ obtenha o núcleo da DCT

$$\mathbf{K}_C = \begin{bmatrix} \\ \\ \\ \\ \\ \\ \\ \\ \end{bmatrix}$$

Transformada Bidimensional Discreta do Cosseno – DCT-2D

A transformada discreta do Cosseno, DCT-2D de uma função $f(x, y)$, denotada por $C(u, v)$, é obtida por,

$$C(u,v) = \alpha(u)\alpha(v)\sum_{x=0}^{N-1}\sum_{y=0}^{N-1} f(x,y)\cos\left[\frac{(2x+1)u\pi}{2N}\right]\cos\left[\frac{(2y+1)v\pi}{2N}\right]$$

com u, $v = 0, 1, 2, \ldots, N-1$ e sua inversa, IDCT-2D dada por,

$$f(x,y) = \sum_{u=0}^{N-1}\sum_{u=0}^{N-1} \alpha(u)\alpha(v)C(u,v)\cos\left[\frac{(2x+1)u\pi}{2N}\right]\cos\left[\frac{(2y+1)v\pi}{2N}\right]$$

com x, $y = 0, 1, 2, \ldots, N-1$, em que

$$\alpha(s) = \begin{cases} \sqrt{\frac{1}{N}}, & \text{para } s = 0 \\ \sqrt{\frac{2}{N}}, & \text{para } s = 1, 2, 3, \ldots, N-1. \end{cases}$$

Transformada Bidimensional Discreta do Cosseno – DCT-2D

- A DCT-2D adminte a propriedade da separabilidade. Verifique!

Aplicações:

- A Transformada Discreta do Cosseno (DCT) é utilizada no padrão JPEG de compressão de imagens para a exploracão da redundância psicovisual (com perdas), num processo conhecido como codificação por transformadas.
- No padrão JPEG, o processo de codificação por transformadas mapeia os pixels da imagem para um espaço de coeficientes mais compacto. (dicutiremos com mais detalhes posteriomente em compactação de imagens)

Transf. Discreta do Cosseno – DCT-2D

Exemplo: Funções base DCT para $N = 4$. Cada bloco consiste de elementos 4×4, correspondendo a x e y variando de 0 a 3. A origem de cada bloco está no seu canto superior esquerdo. O maior valor mostrado em branco. Outros valores mostrados em tons de cinza, sendo os menores valores representados em tons mais escuros.