Processamento de Imagens Digitais

Transformada de Fourier e Teorema da Convolução

Hugo Neves de Oliveira, Jefersson Alex dos Santos

{oliveirahugo, jefersson}@dcc.ufmg.br

DCC
DEPARTAMENTO DE
CIÊNCIA DA COMPUTAÇÃO

UFMG

Introdução

Introdução

Descrição

Grande parte das aplicações do processamento de imagens envolve operações no domínio espacial (i.e. convoluções, ajuste de brilho, sistemas de cor...).

Descrição

Entretanto, algumas formas de filtragem são mais eficientes e/ou interpretáveis por humanos se realizadas no domínio da frequência. Portanto, as próximas duas aulas abordarão o estudo da análise de imagens e sinais no domínio da frequência.

UFMG

Primórdios da Análise de Frequências

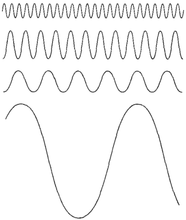
Começo

Théorie analytique de la chaleur (“A Teoria Analítica do Calor”) publicada pelo matemático e físico Joseph Fourier

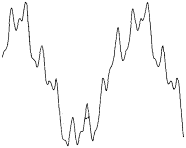
Contribuição Matemática

Qualquer função (contínua ou discreta) pode ser expandida em uma série possivelmente infinita de senos

Exemplo Visual



(a)



(b)

Figura: Soma de frequências (a) e sinal original gerado por elas (b).

UFMG

Aplicações

- Física
  - Mecânica Quântica
  - Termodinâmica
  - Teoria de Cordas
  - Ondas Gravitacionais
- Análise de Sinais e Imagens
  - Remoção de ruído
  - Detecção de aleatoriedade
  - Extração de características
- Compressão de Dados
  - Imagens
  - Áudio
  - Vídeo

UFMG

Aplicações

- Física
  - Mecânica Quântica
  - Termodinâmica
  - Teoria de Cordas
  - Ondas Gravitacionais
- Análise de Sinais e Imagens
  - Remoção de ruído
  - Detecção de aleatoriedade
  - Extração de características
- Compressão de Dados
  - Imagens
  - Áudio
  - Vídeo

UFMG

Aplicações

- Física
  - Mecânica Quântica
  - Termodinâmica
  - Teoria de Cordas
  - Ondas Gravitacionais
- Análise de Sinais e Imagens
  - Remoção de ruído
  - Detecção de aleatoriedade
  - Extração de características
- Compressão de Dados
  - Imagens
  - Áudio
  - Vídeo

Análise de Fourier

Transformada Contínua

- Fourier Transform
  - $F(u) = \int_{-\infty}^{\infty} f(x) e^{-j2\pi ux} dx$
- Inverse Fourier Transform
  - $f(x) = \int_{-\infty}^{\infty} F(u) e^{j2\pi ux} du$
- Legenda:
  - $f(x) =$ função expressa no domínio do original (i.e. espaço, tempo...)
  - $F(u) =$ função expressa no domínio da frequência

UFMG

Transformada Contínua

- 2D Fourier Transform
  - $F(u,v) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} f(x,y) e^{-j2\pi(ux+vy)} dx dy$
- 2D Inverse Fourier Transform
  - $f(x,y) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} F(u,v) e^{j2\pi(ux+vy)} du dv$

UFMG

Transformada Discreta

- Discrete Fourier Transform (DFT)
  - $F(u) = \frac{1}{M} \sum_{x=0}^{M-1} f(x) e^{\frac{-j2\pi ux}{M}}$,
    para $u = 0, 1, 2, ..., M-1$ e $j = \sqrt{-1}$
- Inverse Discrete Fourier Transform (IDFT)
  - $f(x) = \sum_{u=0}^{M-1} F(u) e^{\frac{j2\pi ux}{M}}$,
    para $x = 0, 1, 2, ..., M-1$ e $j = \sqrt{-1}$
- Fórmula de Euler
  - $e^{\pm j\theta} = \cos\theta \pm j \sin\theta$

UFMG

Transformada Discreta

- Discrete Fourier Transform (DFT)
  - $F(u) = \frac{1}{M} \sum_{x=0}^{M-1} f(x) e^{\frac{-j2\pi ux}{M}}$,
    para $u = 0, 1, 2, ..., M-1$ e $j = \sqrt{-1}$
- Inverse Discrete Fourier Transform (IDFT)
  - $f(x) = \sum_{u=0}^{M-1} F(u) e^{\frac{j2\pi ux}{M}}$,
    para $x = 0, 1, 2, ..., M-1$ e $j = \sqrt{-1}$
- Fórmula de Euler
  - $e^{\pm j\theta} = \cos\theta \pm j \sin\theta$

UFMG

Transformada Discreta

- Discrete Fourier Transform (DFT)
  - $F(u) = \frac{1}{M} \sum_{x=0}^{M-1} f(x) e^{\frac{-j2\pi ux}{M}}$,
    para $u = 0, 1, 2, ..., M-1$ e $j = \sqrt{-1}$
- Inverse Discrete Fourier Transform (IDFT)
  - $f(x) = \sum_{u=0}^{M-1} F(u) e^{\frac{j2\pi ux}{M}}$,
    para $x = 0, 1, 2, ..., M-1$ e $j = \sqrt{-1}$
- As fórmulas são muito parecidas
  - A transformada inversa é uma modificação da transformada direta

UFMG

Transformada Discreta

- Discrete Fourier Transform (DFT)
  - $F(u) = \frac{1}{M}\sum_{x=0}^{M-1} f(x)e^{\frac{-j2\pi ux}{M}}$,
    para $u = 0, 1, 2, ..., M-1$ e $j = \sqrt{-1}$
- Inverse Discrete Fourier Transform (IDFT)
  - $f(x) = \sum_{u=0}^{M-1} F(u)e^{\frac{j2\pi ux}{M}}$,
    para $x = 0, 1, 2, ..., M-1$ e $j = \sqrt{-1}$
- O sinal transformado terá o mesmo tamanho do sinal de entrada

UFMG

Transformada Discreta

- $F(u) = \frac{1}{M} \sum_{x=0}^{M-1} f(x)[cos(2\pi ux/M) - j\ sin(2\pi ux/M)]$,
  para $u = 0, 1, 2, ..., M-1$ e $j = \sqrt{-1}$
- Esse somatório é executado M vezes

UFMG

Transformada Discreta

- $F(u) = \frac{1}{M} \sum_{x=0}^{M-1} f(x)[cos(2\pi ux/M) - j\ sin(2\pi ux/M)]$,
  para $u = 0, 1, 2, ..., M-1$ e $j = \sqrt{-1}$
- A variável u assume M valores

UFMG

Transformada Discreta

- $F(u) = \frac{1}{M} \sum_{x=0}^{M-1} f(x)[cos(2\pi ux/M) - j\, sin(2\pi ux/M)]$,
  para $u = 0, 1, 2, ..., M-1$ e $j = \sqrt{-1}$
- Assim, as complexidades computacionais tanto da DFT quanto da IDFT são equivalentes a $\Theta(M^2)$

UFMG

Transformada Rápida de Fourier

DFT Original

A DFT é utilizada em muitas aplicações nas áreas de processamento de sinais e imagens, muitas delas que necessitam de processamento em tempo real. $\Theta(M^2)$ pode, portanto, ser uma complexidade computacional proibitiva para o algoritmo.

FFT

A Fast Fourier Transform (FFT) é um algoritmo que mitiga o problema da complexidade computacional se $M = 2^n$, com $n \in \mathbb{N}$. A FFT tem complexidade $\Theta(MlogM)$.

UFMG

Transformada Rápida de Fourier

DFT Original

A DFT é utilizada em muitas aplicações nas áreas de processamento de sinais e imagens, muitas delas que necessitam de processamento em tempo real. $\Theta(M^2)$ pode, portanto, ser uma complexidade computacional proibitiva para o algoritmo.

FFT

A Fast Fourier Transform (FFT) é um algoritmo que mitiga o problema da complexidade computacional se $M = 2^n$, com $n \in \mathbb{N}$. A FFT tem complexidade $\Theta(MlogM)$.

Exemplo Prático

Exemplo Prático - FFT 1D

1D_FFT.ipynb

UFMG

Transformada Discreta 2D

- 2D Discrete Fourier Transform (DFT)
  - $F(u,v) = \frac{1}{MN} \sum_{x=0}^{M-1} \sum_{y=0}^{N-1} f(x,y) e^{-j2\pi(\frac{ux}{M} + \frac{vy}{N})}$,
    para $u = 0,1,2,...,M-1$, $v = 0,1,2,...,N-1$ e $j = \sqrt{-1}$
- 2D Inverse Discrete Fourier Transform (IDFT)
  - $f(x,y) = \sum_{u=0}^{M-1} \sum_{v=0}^{N-1} F(u,v) e^{j2\pi(\frac{ux}{M} + \frac{vy}{N})}$,
    para $x = 0,1,2,...,M-1$, $y = 0,1,2,...,N-1$ e $j = \sqrt{-1}$
- Ambas fórmulas revelam uma complexidade computacional de $\Theta(M^2N^2)$

UFMG

Propriedades Úteis da DFT

- Entretanto a DFT e a IDFT são transformadas separáveis
- Usando a propriedade da separabilidade, é possível executar as DFTs 1D separadamente em todas as linhas da imagem e posteriormente nas colunas, obtendo o mesmo resultado da transformada bidimensional
- Sabemos que a DFT 1D possui complexidade $\Theta(M^2)$, logo, a complexidade das transformadas pode passar a ser $\Theta(NM^2)$ para as linhas e $\Theta(MN^2)$ para as colunas, resultando em $\Theta(NM^2 + MN^2)$
- Se for usada a FFT 1D no lugar da DFT 1D, a complexidade cai para $\Theta(NMlogM + MNlogN) = \Theta(NM(logN + logM))$

UFMG

Propriedades Úteis da DFT

- Entretanto a DFT e a IDFT são transformadas separáveis
- Usando a propriedade da separabilidade, é possível executar as DFTs 1D separadamente em todas as linhas da imagem e posteriormente nas colunas, obtendo o mesmo resultado da transformada bidimensional
- Sabemos que a DFT 1D possui complexidade $\Theta(M^2)$, logo, a complexidade das transformadas pode passar a ser $\Theta(NM^2)$ para as linhas e $\Theta(MN^2)$ para as colunas, resultando em $\Theta(NM^2 + MN^2)$
- Se for usada a FFT 1D no lugar da DFT 1D, a complexidade cai para $\Theta(NMlogM + MNlogN) = \Theta(NM(logN + logM))$

UFMG

Propriedades Úteis da DFT

- Entretanto a DFT e a IDFT são transformadas separáveis
- Usando a propriedade da separabilidade, é possível executar as DFTs 1D separadamente em todas as linhas da imagem e posteriormente nas colunas, obtendo o mesmo resultado da transformada bidimensional
- Sabemos que a DFT 1D possui complexidade $\Theta(M^2)$, logo, a complexidade das transformadas pode passar a ser $\Theta(NM^2)$ para as linhas e $\Theta(MN^2)$ para as colunas, resultando em $\Theta(NM^2 + MN^2)$
- Se for usada a FFT 1D no lugar da DFT 1D, a complexidade cai para $\Theta(NMlogM + MNlogN) = \Theta(NM(logN + logM))$

UFMG

Exemplo Prático

Exemplo Prático - FFT 2D

2D_FFT.ipynb

UFMG

Propriedades Úteis da DFT

- O componente de frequência de índice zero das DFTs (i.e. $F(0)$) tem uma propriedade interessante
  - $F(u) = \frac{1}{M}\sum_{x=0}^{M-1} f(x)e^{\frac{-j2\pi ux}{M}} = \frac{1}{M}\sum_{x=0}^{M-1} f(x)$, para $u = 0$
  - $F(0)$ é, portanto, equivalente à média do sinal
- Devido a essa propriedade, o componente de índice zero é chamado de componente DC (*Direct Current*)
- Todos os outros componentes de índices diferentes de zero (i.e. $F(1)$, $F(2)$...) são chamados de componentes AC (*Alternating Current*)

Exemplo Prático

Exemplo Prático - Zerando o Coeficiente DC

1D_FFT_DC-AC.ipynb

UFMG

Teorema da Convolução

UFMG

Teorema da Convolução

Teorema da Convolução

O Teorema da Convolução é a relação mais fundamental entre filtragens no domínio espacial e no domínio da frequência

UFMG

Convoluções

- Sinal: $f(.)$
  - i.e. sinal de ECG
  - X samples
- Kernel: $h(.)$
  - i.e. Kernel Gaussiano 1D
  - M valores
- $f(x) * h(x) = \frac{1}{M} \sum_{m=0}^{M-1} f(m)h(x-m)$, para $x = 0, 1, \ldots, X-1$
- $f(x) * h(x) = \Theta(MX)$

UFMG

Teorema da Convolução

Teorema da Convolução

$$f(x) * h(x) \leftrightarrow F(u).H(u)$$

UFMG

Teorema da Convolução

Teorema da Convolução

$$f(x) * h(x) \leftrightarrow F(u).H(u)$$

Teorema da Convolução

Domínio Espacial/Temporal

UFMG

Teorema da Convolução

Teorema da Convolução

$$f(x) * h(x) \leftrightarrow F(u).H(u)$$

Teorema da Convolução

Domínio da Frequência

UFMG

Teorema da Convolução

Teorema da Convolução

$$f(x) * h(x) \leftrightarrow F(u).H(u)$$

Teorema da Convolução

Convolução

Teorema da Convolução

Teorema da Convolução

$$f(x) * h(x) \leftrightarrow F(u) \;.\; H(u)$$

Teorema da Convolução

Multiplicação ponto-a-ponto

UFMG

Teorema da Convolução

Teorema da Convolução

$$f(x) * h(x) \leftrightarrow F(u).H(u)$$

Teorema da Convolução

$$f(x) * h(x) = \Theta(MX)$$

vs.

$$F(u).H(u) = \Theta(X)$$

UFMG

Exemplo Prático

Exemplo Prático - Teorema da Convolução 1D

1D_Convolution_Theorem.ipynb

UFMG

Convoluções 2D

- Imagem: $f(.)$
  - i.e. Imagem da Lena
  - $X \times Y$ pixels
- Kernel: $h(.)$
  - i.e. Kernel Gaussiano 2D
  - $M \times N$ valores
- $f(x,y) * h(x,y) = \frac{1}{MN} \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} f(m,n)h(x-m, y-n),$
  para $x = 0, 1, ..., X-1$ e $y = 0, 1, ..., Y-1$
- $f(x,y) * h(x,y) = \Theta(MNXY)$

UFMG

Teorema da Convolução 2D

Teorema da Convolução 2D

$$f(x,y) * h(x,y) \leftrightarrow F(u,v).H(u,v)$$

Teorema da Convolução

$$f(x,y) * h(x,y) = \Theta(MNXY)$$

vs.

$$F(u,v).H(u,v) = \Theta(XY)$$

UFMG

Exemplo Prático

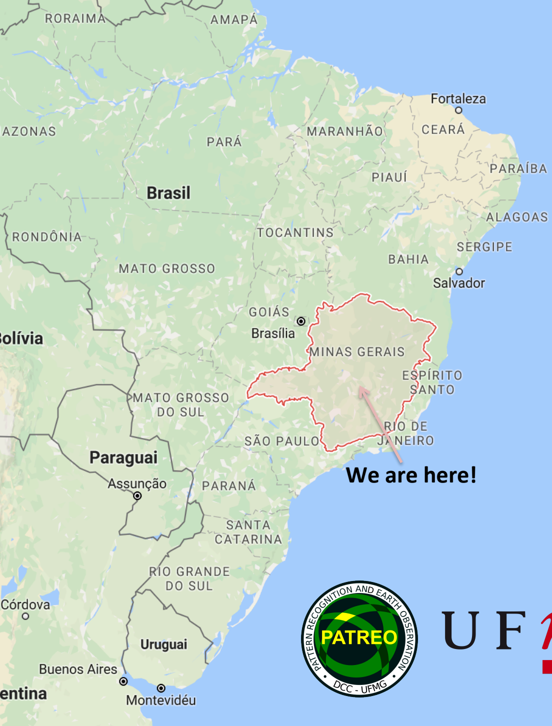
Exemplo Prático - Teorema da Convolução 2D

2D_Convolution_Theorem.ipynb

UFMG

Bibliografia

1. R. C. Gonzalez & R. E. Woods. Digital Image Processing, Addison-Wesley, 3rd Ed., 2007.
2. H. Pedrini & W. R. Schwartz. Análise de imagens digitais: princípios, algoritmos e aplicações. Thomson Learning, 2008.
3. A.S. Glassner. Principles of Digital Image Synthesis. Vols 1 and 2, Morgan Kauffman, 1995.
4. A.X.Falcão. Notas de aula em `http://www.ic.unicamp.br/~afalcao/mo443`
5. Diversos artigos tratando dos problemas relacionados à área de Processamento de Imagens, Visão Computacional e Reconhecimento de Padrões.



We are here!

And here:

www.patreo.dcc.ufmg.br