ỦY BAN NHÂN DÂN THÀNH PHỐ HỒ CHÍ MINH
TRƯỜNG ĐẠI HỌC SÀI GÒN

BÁO CÁO TỔNG KẾT
ĐỒ ÁN MÔN HỌC
PHÂN TÍCH VÀ XỬ LÝ ẢNH

Xử lý ảnh trên miền tần số (Image Enhancement in the Frequency Domain)

Giảng viên hướng dẫn: PGS.TS Phạm Thế Bảo

Thành phố Hồ Chí Minh, tháng 11 năm 2025

ỦY BAN NHÂN DÂN THÀNH PHỐ HỒ CHÍ MINH
TRƯỜNG ĐẠI HỌC SÀI GÒN

BÁO CÁO TỔNG KẾT
ĐỒ ÁN MÔN HỌC
PHÂN TÍCH VÀ XỬ LÝ ẢNH

Xử lý ảnh trên miền tần số
(Image Enhancement in the Frequency Domain)

MSV	Họ và tên
3122480001	Trần Đức Anh
3122480034	Nguyễn Thành Nam
3122480042	Bùi Tấn Phát

Thành phố Hồ Chí Minh, tháng 11 năm 2025

Lời cam đoan

Tôi tên là Nguyễn Thành Nam, Sinh viên lớp DTU1221, Khoa Toán-Ứng dụng, Khóa 22, thuộc trường Đại học Sài Gòn.

Tôi xin cam đoan toàn bộ nội dung được trình bày trong bản báo cáo này đều do chính tôi thực hiện dưới sự hướng dẫn của *PGS.TS Phạm Thế Bảo*. Những kết quả nghiên cứu của tác giả khác được sử dụng trong đề tài đều có trích dẫn đầy đủ. Tôi xin hoàn toàn chịu trách nhiệm nếu có các nội dung sao chép không hợp lệ hoặc vi phạm quy chế đào tạo.

Tp. HCM, tháng 11 năm 2025

Tác giả

Nguyễn Thành Nam

Lời cảm ơn

Tiểu luận này được hoàn thành tại trường Đại Học Sài Gòn dưới sự hướng dẫn của *PGS.TS Phạm Thế Bảo*. Em xin bày tỏ lòng biết ơn chân thành và sâu sắc về sự tận tâm và nhiệt tình của Thầy trong suốt quá trình tác giả thực hiện đề tài.

Xin cám ơn Phòng Đào tạo và Khoa Toán - Ứng dụng trường Đại học Sài Gòn, gia đình, bạn bè và các thầy cô đã tạo nhiều điều kiện thuận lợi, giúp em hoàn thành đồ án môn học này.

Tp. HCM, tháng 11 năm 2025

Tác giả

Nguyễn Thành Nam

Mục lục

Danh mục các từ viết tắt

DFT	Discrete Fourier Transforms
IDFT	Inverse Discrete Fourier Transforms
	Phương án tối ưu
	Phương án
	Bài toán

Danh mục các bảng, hình vẽ

Hình, bảng	Trang

Lời nói đầu

Tp. HCM, tháng 5 năm 2025

Tác giả

Nguyễn Thành Nam

Chương 1

Xử lý ảnh trong miền tần số

1.1. Giới thiệu và đôi nét về miền tần số

Trong bài tiểu luận này chúng tôi nghiên cứu về xử lý ảnh trên miền tần số. Sử dụng công cụ toán học để tính toán và ứng dụng cho xử lý ảnh/ lọc ảnh trên miền tần số. Ảnh đầu vào của các phép xử lý sẽ là ảnh xám của 1 bức hình bình thường, sau đó dùng các phương pháp xử lý ảnh trên miền tần số để làm sắc nét ảnh hơn hoặc làm mờ ảnh đi để bảo mật.

Khái niệm lọc ảnh dễ dàng nhận thấy trong miền tần số. Giúp tăng cường ảnh $f(x, y)$. Ảnh tăng cường được xây dựng nên thông qua tích chập sau:

$$\overbrace{g(m,n)}^{\text{Ảnh tăng cường}} = \underbrace{h(x,y)}_{PSS} * \underbrace{f(x,y)}_{\text{Ảnh}}.$$

Ta cũng có thể cài đặt tăng cường ảnh trong miền tần số và thiết kế hàm tương ứng như:

$$\overbrace{\hat{g}(u,v)}^{\text{Ảnh tăng cường}} = \underbrace{\hat{h}(u,v)}_{\text{Hàm biến đổi}} * \overbrace{\hat{f}(u,v)}^{\text{Ảnh}},$$

với các kí hiệu hàm trên là các kết quả của các biến đổi Fourier.

Ta sẽ tìm hiểu khái niệm miền tần số trong không gian 1 chiều.

1. Chu kỳ của $\cos t$ là 2π giây, nghĩa là tín hiệu sẽ được lặp lại sau mỗi 2π giây.

2. Chu kỳ của $\cos(2\pi t)$ là 1 giây.

3. Ta ký hiệu chu kỳ là T.

4. Đơn vị đo cho tần số là Hz (Herts), ký hiệu là f, tương ứng với số chu kỳ xảy ra trong 1 giây
$$f = \frac{1}{T}$$

5. Hàm $\cos(2\pi \cdot ft)$ có chu kỳ $1/T$ và tần số f.

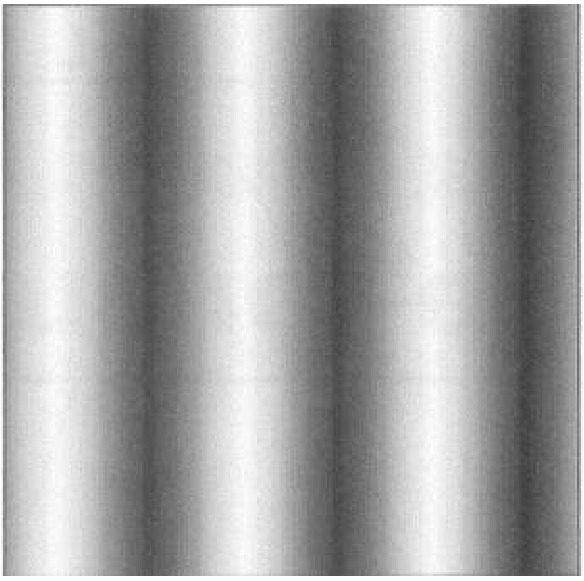
6. Đơn vị đo cho tần số góc là rad/s, ký hiệu là ω
$$\omega = \frac{2\pi}{T}$$

Ví dụ: Giả sử một bức ảnh có các điểm ảnh thỏa hàm số
$$f(x, y) = 128 + A \sin\left(\frac{22\pi ux}{N-1} + \phi\right)$$

Khi đó, ta biết được ảnh có mức xám trung bình là 128, biên độ $A \in [1, 127]$, độ rọng của ảnh là N, ϕ là pha va u là tần số không gian(số vòng hàm sin "vừa với độ rộng của hình, chia cho $N \to$ tần số không gian trong 1 vòng đơn vị trên điểm ảnh).

Cho hàm
$$f(x, y) = 128 + 127 \sin\left(\frac{2\pi \cdot 3x}{100-1} + 0\right).$$

Ta được ảnh tương ứng của hàm số $f(x, y)$ là



Hình 1.1: Ảnh $f(x, y)$

1.2. Khác biệt giữa miền không gian và miền tần số

Trong miền không gian, ta xử lý trực tiếp trên từng điểm ảnh, còn trong miền tần số ta xử lý dựa trên tốc độ thay đổi giá trị của điểm ảnh trên miền không gian.

1. Miền không gian: Ma trận ảnh đầu vào → Xử lý → Ma trận ảnh đầu ra.

2. Miền tần số: Ảnh vào → Phân bố tần số → Xử lý → Chuyển đổi ngược → Ảnh ra.

   Miền tần số không gian có thể tạo ra mối quan hệ chu kỳ rõ ràng trong miền không gian, trong miền tần số, một số toán tử xử lý ảnh sẽ trở nên hiệu quả hơn.

   Trong nhiều trường hợp, người ta dùng chuyển đổi Fourier để chuyển ảnh từ miền không gian sang miền tần số.

1.3. Khái niệm chuỗi Fourier và biến đổi Fourier

Chuỗi Fourier (Fourier series) được nhà Toán học người Pháp tên Jean Baptiste Joseph Fourier đưa ra vào thế kỷ 19. Ông khẳng định rằng với bất kỳ hàm số $f(t)$

tuần hoàn với chu kỳ T đều có thể biểu diễn được dưới dạng tổng của các hàm số sine và cosine với những tần số khác nhau, mỗi hàm số nhân với một hệ số tương ứng. Khi đó, ta gọi tổng các chuỗi hàm số sine và cosine này là chuỗi Fourier.

Giống như chuỗi Taylor, chuỗi Fourier là một dạng đặc biệt của khai triển hàm số. Với chuỗi Taylor, ta quan tâm đến tập các hàm số đặc biệt dạng như sau:

$$1, x, x^2, x^3, \dots \tag{a}$$

còn với chuỗi Fourier, ta quan tâm đến các dạng hàm đặc biệt lượng giác:

$$1, \cos x, \cos 2x, \cos 3x, \dots, \sin x, \sin 2x, \sin 3x, \dots \tag{b}$$

Do đó, khai triển chuỗi Fourier của một hàm số f sẽ có dạng tổng các hàm lượng giác sin và cos cùng các hệ số:

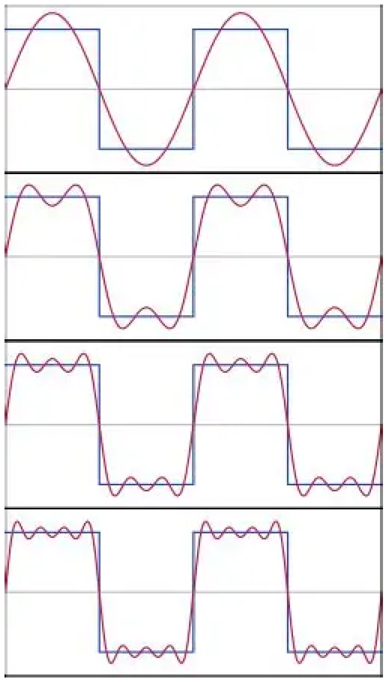
$$f(x) = a_0 + \sum_{n=1}^{+\infty} a_n \cos nx + b_n \sin nx, \tag{c}$$

với

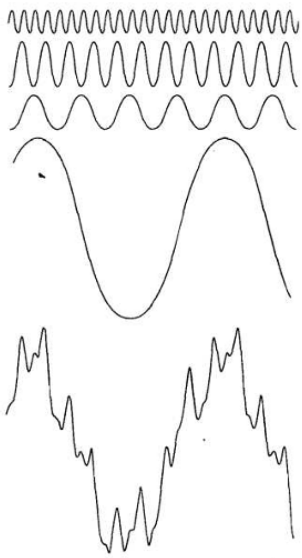
$$a_0 = \frac{1}{2\pi} \int_{-\pi}^{\pi} f(x) dx,$$

$$a_n = \frac{1}{\pi} \int_{-\pi}^{\pi} f(x) \cos nx dx, \quad n = 1, 2, \dots$$

$$b_n = \frac{1}{\pi} \int_{-\pi}^{\pi} f(x) \sin nx dx, \quad n = 1, 2, \dots$$



Hình 1.2: Minh họa về chuỗi Fourier và các dạng hàm sóng tương ứng từ trên xuống với chỉ số n càng lớn.



Hình 1.3: Minh họa chuỗi Fourier

1.3.1 Biến đổi Fourier cho hàm liên tục

1.3.1.1 Biến đổi Fourier cho hàm một biến số

Lấy f là hàm trơn từng khúc tuần hoàn chu kì T. Dạng phức của chuỗi Fourier của f là:

Định lý 1.1. ***(Dạng phức của chuỗi Fourier)***

$$f(t) = \sum_{n=-\infty}^{\infty} c_n e^{i\frac{2\pi n}{T}t}, \tag{1.3.1}$$

với hệ số Fourier c_n được cho bởi

$$c_n = \frac{1}{T}\int_{-\frac{T}{2}}^{\frac{T}{2}} f(t)e^{-i\frac{2\pi n}{T}t}dt, \quad n = 0, \pm 1, \pm 2, \pm 3, ... \tag{1.3.2}$$

Phương trình (1.3.1) là cách khai triển hàm số sin và cos theo công thức Euler trong trường số phức:

$$e^{i\varphi} = \cos\varphi + i\sin\varphi \tag{1.3.3}$$

Đối với những hàm số không có tính tuần hoàn, nhưng diện tích dưới đường cong của hàm số đó là hữu hạn, ta có thể biểu diễn hàm số đó dưới dạng tích phân của hàm sin và cos nhân với hàm trọng số. Biểu thức thu được gọi là biến đổi Fourier. Ta xác định phương trình biến đổi Fourier của một hàm số liên tục $f(t)$ có biến t liên tục như sau:

$$\mathcal{F}\{f(t)\}(\mu) = \int_{-\infty}^{+\infty} f(t)e^{-2\pi i\mu t}dt \tag{1.3.4}$$

với μ là biến liên tục. Vì khi lấy xong tích phân sẽ mất t nên ta chỉ còn lại μ, vậy ta sẽ ký hiệu lại phương trình biến đổi Fourier cho rõ ràng hơn như sau:

$$\hat{f}(\mu) = \int_{-\infty}^{+\infty} f(t)e^{-2\pi i\mu t}dt \tag{1.3.5}$$

Ngược lại, cho trước $\hat{f}(\mu)$, ta có thể tìm lại $f(t)$ bằng cách sử dụng chuyển đổi ngược Fourier (inverse Fourier transform), $f(t) = \mathcal{F}^{-1}\{\hat{f}(\mu)\}$, viết là:

$$f(t) = \int_{-\infty}^{+\infty} \hat{f}(\mu)e^{2\pi i\mu t}d\mu, \tag{1.3.6}$$

sử dụng công thức Euler, ta viết lại phương trình (1.3.5) như sau:

$$\hat{f}(\mu) = \int_{-\infty}^{+\infty} f(t)[\cos(2\pi\mu t) - i\sin(2\pi\mu t)]dt. \tag{1.3.7}$$

Biến còn lại khi lấy tích phân là μ, chính là tần số của hàm lượng giác nên miền của chuyển đổi Fourier là miền tần số.

1.3.1.2 Biến đổi Fourier cho hàm hai biến số

Ta có công thức biến đổi Fourier và biến đổi Fourier ngược cho hàm hai biến như sau:

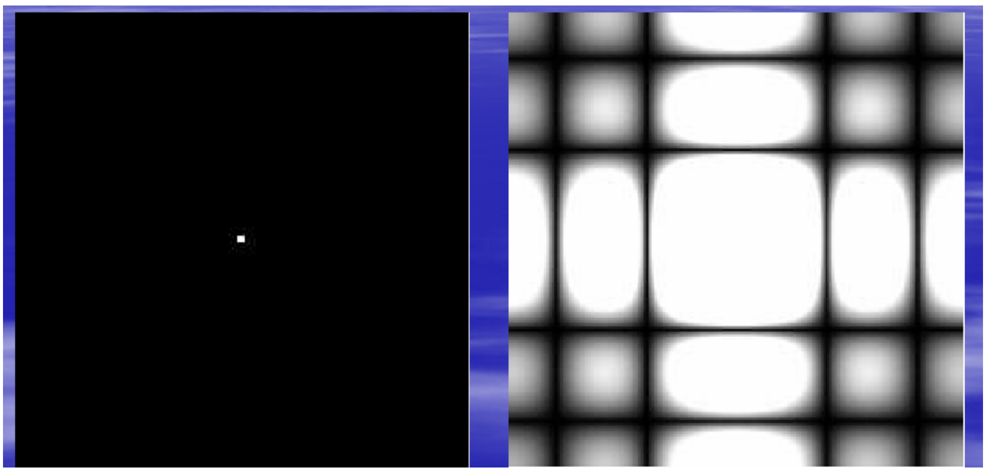
Biến đổi Fourier thuận:

$$\mathcal{F}(u,v) = \int_{-\infty}^{+\infty}\int_{-\infty}^{+\infty} f(x,y)e^{-2\pi i(ux+vy)}dxdy \tag{1.3.8}$$

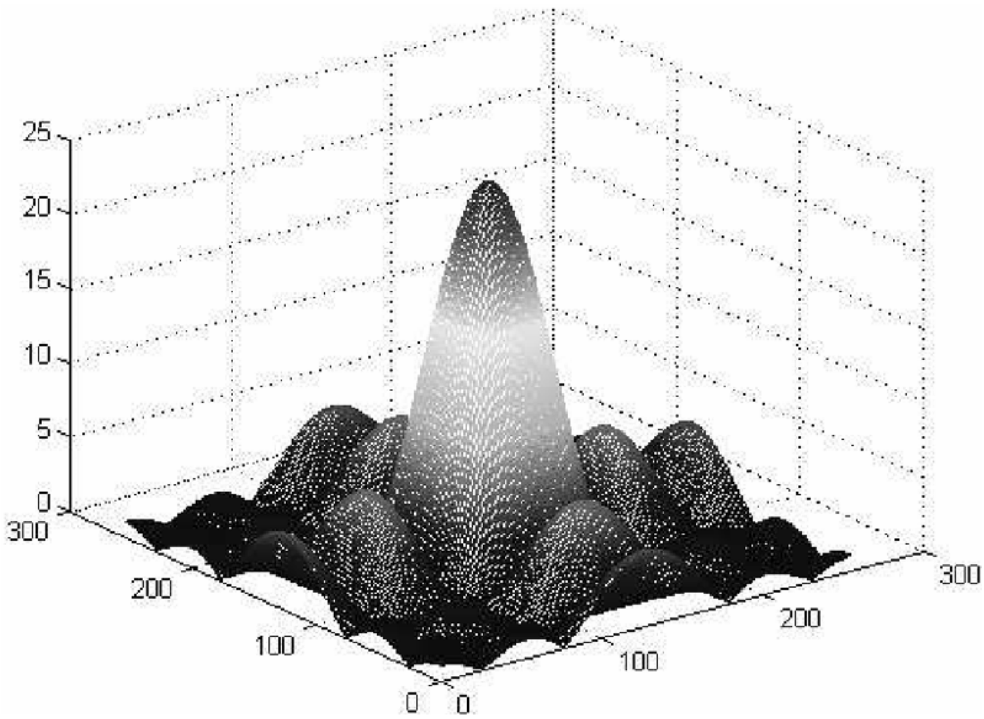
Biến đổi Fourier ngược:

$$\mathcal{F}^{-1}(u,v) = f(x,y) = \int_{-\infty}^{+\infty}\int_{-\infty}^{+\infty} f(x,y)e^{2\pi i(ux+vy)}dudv \tag{1.3.9}$$

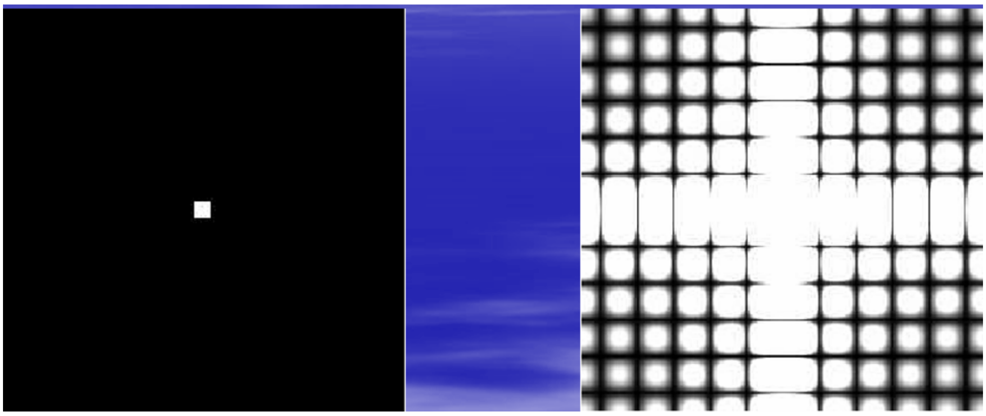
Một số ví dụ về biến đổi Fourier với ảnh.



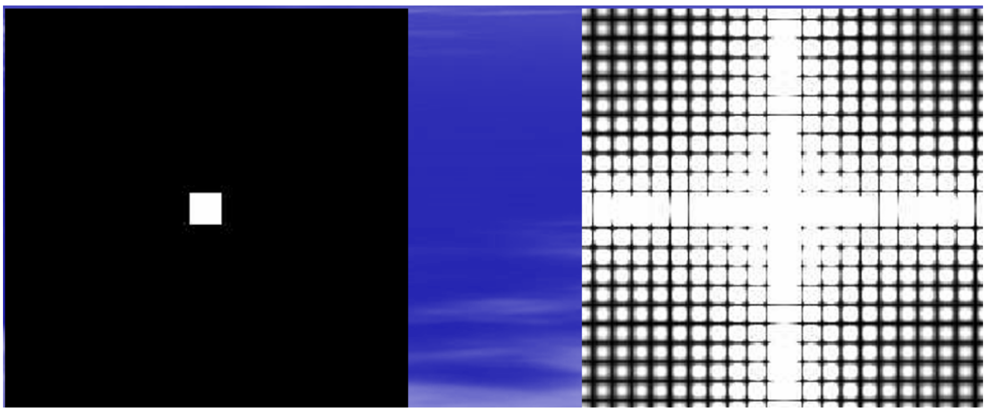
Hình 1.4: Ảnh-$f(x,y)$ (bên trái) và Phổ độ lớn $\hat{f}(u,v)$ (bên phải)



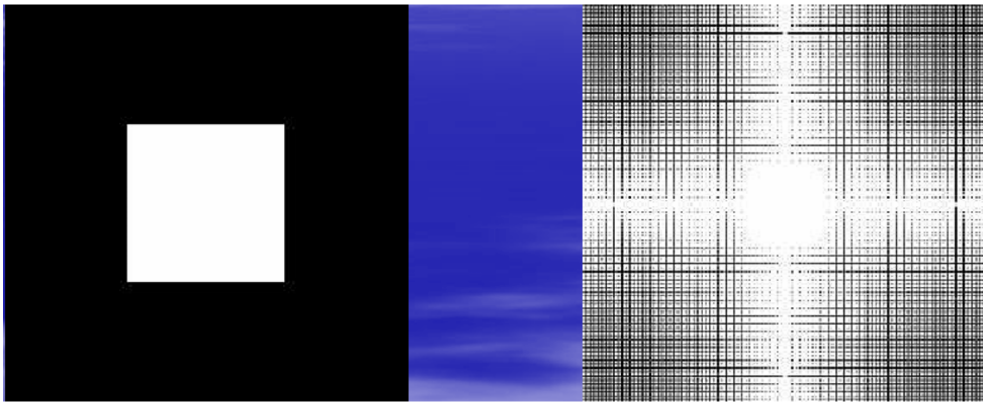
Hình 1.5: $\hat{f}(u,v)$ trong không gian 3 chiều



Hình 1.6: Ảnh-$f(x,y)$ (bên trái) và Phổ độ lớn $\hat{f}(u,v)$ (bên phải)



Hình 1.7: Ảnh-$f(x,y)$ (bên trái) và Phổ độ lớn $\hat{f}(u,v)$ (bên phải)



Hình 1.8: Ảnh-$f(x,y)$ (bên trái) và Phổ độ lớn $\hat{f}(u,v)$ (bên phải)

Nhận xét: Ta thấy khi kích thước đối tượng tăng lên trong miền không gian thì tương ứng "kích thước" - mức độ biến động lên xuống trong miền tần số giảm. Nhưng có xu hướng dao động rộng hơn với nhịp độ rất nhỏ. Hay nói cách khác là các tần số thấp nhiều hơn hẳn ảnh kích thước nhỏ.

1.3.2 Biến đổi Fourier cho hàm rời rạc - DFT

Vì các điểm ảnh là các điểm dữ liệu rời rạc nên ta áp dụng biến đổi Fourier, ta cần xây dựng một công thức sử dụng cho các biến rời rạc.

1.3.2.1 Hàm một biến số

Giả sử ra có bộ dữ liệu dãy f_n, (với $n = 1, 2, 3, ..., N$) là dãy-vector-ảnh 1-D, ta xác định DFT(Discrete Fourier Transforms) cho f_n như sau:

$$f_n = \frac{1}{N}\sum_{i=1}^{N} \hat{f}(k) e^{\frac{-ik2\pi n}{N}}, \quad n = \overline{1, N}. \tag{1.3.10}$$

Biến đổi Fourier ngược(IDFT) là

$$\hat{f}(k) = \sum_{k=1}^{N} f_n e^{\frac{ik2\pi n}{N}}, \quad n = \overline{1, N}. \tag{1.3.11}$$

Lưu ý:

- $\hat{f}(k)$ là phức dù cho f_n là thực.

- Ta cài đặt trực tiếp DFT thì có độ phức tạp là $O(N^2)$.

- Ta có thể dùng FFT được nói đến sau đây để cài đặt DFT hiệu quả hơn. Nhưng đây không phải là một biến đổi mới.

1.3.2.2 Hàm hai biến số

Giả sử hàm $f(x,y), x = 1,\ldots,M, y = 1,\ldots,N$ là ảnh đầu vào rời rạc $M \times N$, DFT cho hàm hai biến (DFT 2-D) này là:

$$\hat{f}(u,v) = \sum_{x=1}^{M}\sum_{y=1}^{N} f(x,y)e^{-2\pi i\left(\frac{ux}{M}+\frac{vy}{N}\right)} \tag{1.3.12}$$

$$\mathcal{F}^{-1}\{\hat{f}(u,v)\} = f(x,y) = \frac{1}{MN}\sum_{u=1}^{M}\sum_{v=1}^{N} \mathcal{F}(u,v)e^{2\pi i\left(\frac{ux}{M}+\frac{vy}{N}\right)} \tag{1.3.13}$$

1.3.3 Biến đổi Fourier nhanh(FFT)

1.3.3.1 Hàm một biến số

Để chuyển ảnh từ miền không gian sang miền tần số bằng cách sử dụng chuyển đổi Fourier thông thường đòi hỏi chi phí lớn ($O(N^2)$ với N là số điểm ảnh).

Thuật toán FFT (Fast Fourier Transforms) thông dụng nhất là thuật toán do J.W.Cooley và John Tukey đề xuất, tính chuyển đổi Fourier cho các giá trị rời rạc bằng cách sử dụng đệ quy tính các giá trị ở vị trí chẵn lẻ.

$$\mathcal{X}_k = \underbrace{\sum_{m=1}^{\frac{N}{2}} x_{2m} e^{-\frac{2\pi i}{N}(2m)k}}_{\text{DFT cho phần chẵn}} + \underbrace{\sum_{m=1}^{\frac{N}{2}} x_{2m+1} e^{-\frac{2\pi i}{N}(2m+1)k}}_{\text{DFT cho phần lẻ}}. \tag{1.3.14}$$

$$\mathcal{X}_k = \sum_{m=1}^{\frac{N}{2}} x_{2m} e^{-\frac{2\pi i}{N/2}mk} + e^{-\frac{2\pi i}{N}k} \sum_{m=1}^{\frac{N}{2}} x_{2m+1} e^{-\frac{2\pi i}{N/2}mk}. \tag{1.3.15}$$

Ta có thể viết lại X_k cho gọn hơn như sau:

$$\mathcal{X}_k = E_k + e^{-\frac{2\pi i}{N}k} O_k, \tag{1.3.16}$$

với E_k, O_k lần lượt là tổng các phần chẵn và phần lẻ.

Do tính tuần hoàn có chu kỳ của DFT nên

$$E_{k+\frac{N}{2}} = E_k \tag{1.3.17}$$

và

$$O_{k+\frac{N}{2}} = O_k \tag{1.3.18}$$

Khi đó, ta viết lại phương trình (1.3.14) và (1.3.15) thành

$$\mathcal{X}_k = \begin{cases} E_k + e^{-\frac{2\pi i}{N}k} O_k, & 0 \le k < \frac{N}{2} \\ E_{k-N/2} + e^{-\frac{2\pi i}{N}k} O_{k-N/2}, & \frac{N}{2} \le k < N \end{cases} \tag{1.3.19}$$

Mặt khác, $e^{-\frac{2\pi i}{N}k}$ hình thành từ:

$$e^{-\frac{2\pi i}{N}(k+N/2)} = e^{-\frac{2\pi i}{N}k - \pi i} = e^{-\pi i} e^{-\frac{2\pi i}{N}k} = -e^{-\frac{2\pi i}{N}k}$$

Khi đó, ta có thể giảm khối lượng tính toán xuống một nửa. Với $0 \le k < N/2$, từ phương trình (1.3.19) ta được:

$$\mathcal{X}_k = E_k + e^{-\frac{2\pi i}{N}k} O_k \tag{1.3.20}$$

$$\mathcal{X}_{k+N/2} = E_k - e^{-\frac{2\pi i}{N}k} O_k \tag{1.3.21}$$

Listing 1.1: Thuật toán FFT đệ quy dạng ditfft2

```python
import cmath

def ditfft2(x, N, s):
    """
    x: list day so (co the la phuc)
    N: kich thuoc FFT (phai la luy thua cua 2)
    s: buoc nhay (stride)
    """
    if N == 1:
        return [x[0]]  # Truong hop co so

    # Tinh FFT phan chan va le
    X_even = ditfft2(x[0::2*s], N//2, 2*s)  # (x0, x2s, x4s, ...)
    X_odd  = ditfft2(x[s::2*s], N//2, 2*s)  # (xs, x3s, x5s, ...)

    X = [0] * N
    for k in range(N//2):
        t = X_even[k]
        exp_term = cmath.exp(-2j * cmath.pi * k / N) * X_odd[k]
        X[k] = t + exp_term
        X[k + N//2] = t - exp_term
```

```python

    return X
```

1.3.3.2 Hàm hai biến số

Từ ý tưởng của hàm một biến, ta có thể tính được FFT của hàm hai biến bằng cách tính theo một chiều với mỗi giá trị của biến x(theo từng cột). Sau đó tính ngược lại theo y(theo từng hàng) với giá trị thu được ở trên.

Trong Python, ta có thể sử dụng hàm `fft2` để đưa ảnh sang chuỗi Fourier và dùng `ifft2` để trả ngược lại ảnh lúc đầu.

1.3.4 Định lý tích chập

Trong xử lý ảnh hoặc tăng cường ảnh trong miền không gian, người ta sử dụng khái niệm về tích chập , trong đó ảnh sau khi được xử lý bằng tích chập của ảnh ban đầu và bộ lọc ảnh tạo thành. Về mặt Toán học, ta giả sử hai hàm số liên tục, ta có định nghĩa tích chập $*$ của hai hàm số này như bên dưới.

$$f(t) * g(t) = \int_{-\infty}^{+\infty} f(\tau)g(t-\tau)d\tau. \tag{1.3.22}$$

Ta áp dụng tích chập cho hàm hai biến sau khi biến đổi Fourier như sau:

$$\mathcal{F}\{f(t) * g(t)\} = \int_{-\infty}^{+\infty} \left[\int_{-\infty}^{+\infty} f(\tau)g(t-\tau)d\tau \right] e^{-2\pi i\mu t}dt, \tag{1.3.23}$$

từ đây ta có:

$$\begin{aligned}
\mathcal{F}\{f(t) * g(t)\} &= \int_{-\infty}^{+\infty} f(\tau) \left[\int_{-\infty}^{+\infty} g(t-\tau)e^{-2\pi i\mu t}d\tau \right] dt \\
&= \int_{-\infty}^{+\infty} f(\tau)e^{-2\pi i\mu t}d\tau \int_{-\infty}^{+\infty} g(u)e^{-2\pi i\mu u}du \\
&= \mathcal{F}(f)(\mu)\mathcal{F}(g)(\mu) = \hat{f}(\mu)\hat{g}(\mu).
\end{aligned}$$

Vậy giả sử $\mathcal{F}\{f(t)\} = \hat{f}(\mu), \mathcal{F}\{g(t)\} = \hat{g}(\mu)$, khi đó:

$$\mathcal{F}\{f(t) * g(t)\} = \hat{f}(\mu)\hat{g}(\mu). \tag{1.3.24}$$

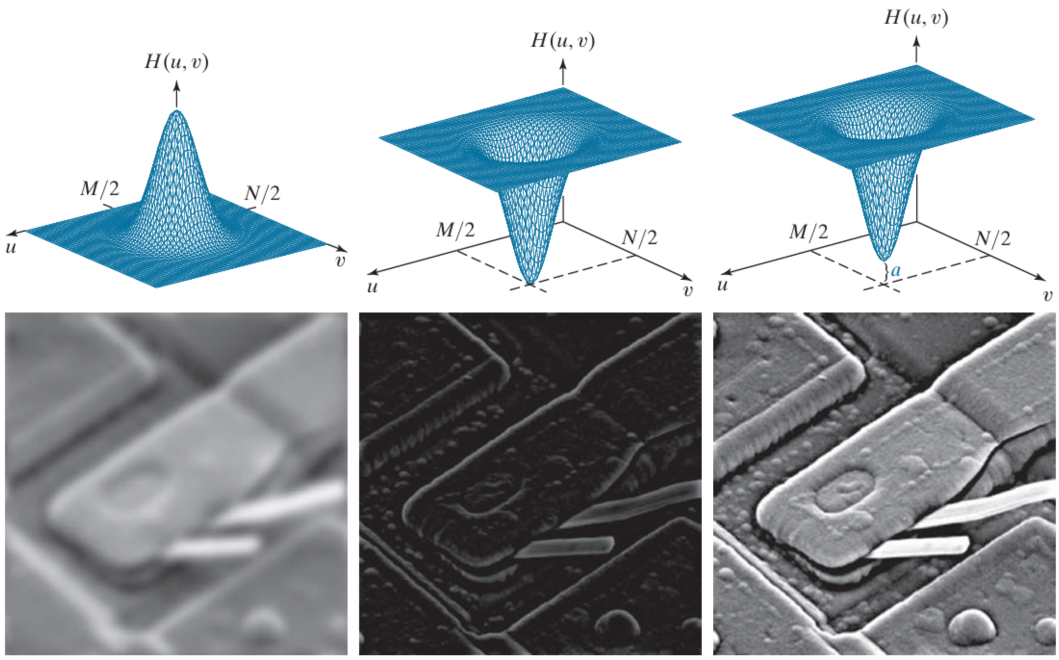
Áp dụng cho trường hợp hai chiều, giả sử $f(x,y)$ là ảnh đầu vào, $h(x,y)$ là bộ lọc ảnh, $\mathcal{F}\{f(x,y)\} = \hat{f}(\mu,\eta)$, $\mathcal{F}\{g(x,y)\} = \hat{g}(\mu,\eta)$, khi đó

$$\mathcal{F}\{f(x,y) * g(x,y)\} = \hat{f}(\mu,\eta)\hat{g}(\mu,\eta). \tag{1.3.25}$$

1.4. Xử lý ảnh trong miền tần số

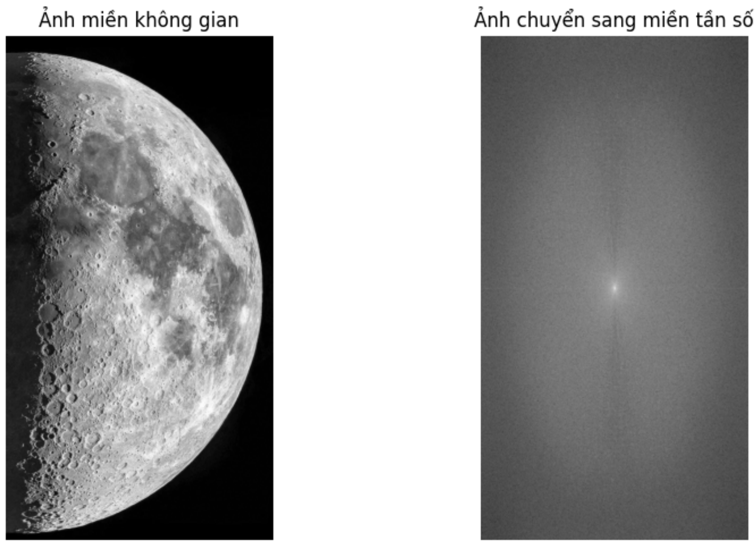
1.4.1 Khái niệm về xử lý ảnh trong miền tần số

Khi chụp ảnh, ta có thể thu được các ảnh có tần số thấp (sự thay đổi mức xám của ảnh ít, ví dụ như ảnh một bức tường) hay ảnh có tần số cao (ví dụ như biên của vật thể). Vì vậy, ta cần có một bộ lọc $H(u,v)$ lọc bỏ các tần số cao và giữ lại tần số thấp (được gọi là lọc thông thấp) làm cho ảnh mờ đi hoặc loại bỏ các tần số thấp giữ lại tần số cao (được gọi là lọc thông cao) làm tăng cường chi tiết vật thể và đồng thời giảm độ tương phản. Hình ảnh mô tả điều này bên dưới



Hình 1.9: Ảnh phía trên là đồ thị hàm tần số ứng với ảnh phía dưới.

Để biết được ảnh trong miền tần số như thế nào ta quan sát ảnh sau:



Hình 1.10: Ảnh gốc (bên trái) và ảnh trong miền tần số khi dùng hàm `fft2` trong Python để biến đổi (ảnh bên phải)

Định lý tích chập cho ta mối quan hệ giữa miền không gian và miền tần số, cụ thể, thông qua tích chập, một ảnh trong miền không gian có thể chuyển qua miền tần số và ngược lại.

Quy trình lọc ảnh trong miền tần số như sau:

$$\text{Ảnh} \rightarrow \text{Biến đổi Fourier} \rightarrow \text{Lọc} \rightarrow \text{Biến đổi Fourier ngược} \rightarrow \text{Ảnh}$$

1. Bước 1: Xử lý ảnh trong miền không gian, tức tăng hoặc giảm độ sáng của ảnh.
2. Bước 2: Lấy DFT của ảnh.
3. Bước 3: Canh giữa DFT, tức mang DFT từ góc ảnh ra giữa ảnh, trong Matlab ta sử dụng hàm shififft để thực hiện.
4. Bước 4: Thực hiện tích chập với hàm lọc.
5. Bước 5: Trượt DFT từ giữa ảnh ra góc.
6. Bước 6: Lấy chuyển đổi ngược IDFT, tức chuyển ảnh từ miền tần số sang miền không gian.

1.4.2 Các bộ lọc băng thông trong miền tần số

Khái niệm bộ lọc trong miền tần số tương tự như khái niệm mặt nạ trong miền không gian.

Sau khi chuyển ảnh sang miền không gian, ta áp dụng một số bộ lọc trong quy trình lọc ảnh nhằm làm mờ ảnh, giảm nhiễu ảnh hoặc làm rõ-làm nét ảnh.

Các bộ lọc thông dụng:

1. Lọc thông thấp/cao Ideal (Ideal low/high pass filter)

2. Lọc thông thấp/cao Gaus (Gaussian low/high pass filter).

3. Lọc thông thấp/cao Butterworth (Butterworth low/high pass filter).

Ta sẽ đi qua một số ứng dụng của bộ lọc này.

Biến đổi Fourier 1 biến số

Biến đổi fourier 2 biến số

ví dụ về biến đổi fourier

biến đổi fourier của ảnh "gạo"

biến đổi fourier của ảnh

pha của ảnh

biến đổi fourier rời rạc Discrete fourier transforms : 1D,2D hoặc f(x) hay f(x,y)

tính chất của DFT

Lọc highpass và lowpass

lọc trên miền tần số trên miền không gian

lọc Butterworth low/high

lọc Gauss low/high

1.4.3 Làm mờ ảnh trong miền tần số

Ta có thể làm mờ ảnh bằng cách giảm các tần số cao, tức ta sử dụng phép lọc thông thấp.

Phép mờ ảnh có những tính chất sau:

- Tất cả giá trị trong mặt nạ mờ đều dương.

- Tổng các giá trị bằng 1.

- Làm giảm phần cạnh bằng mặt nạ mờ.

- Khi kích thước mặt nạ tăng.

Ta có thể làm mờ ảnh bằng cách giảm hoặc loại bỏ các thành phần tần số cao trong ảnh, vì các tần số cao tương ứng với các chi tiết nhỏ của ảnh và các biên cạnh ảnh.

Phương pháp này tương đương với việc dùng bộ lọc thông thấp (Low pass filter) trong miền tần số.

Khi ta biến đổi ảnh $f(x, y)$ sang miền tần số, ta được:

$$\hat{f}(u, v) = \mathcal{F}\{f(x, y)\} = \sum_{x=1}^{M}\sum_{y=1}^{N} f(x, y)e^{-2\pi i\left(\frac{ux}{M}+\frac{vy}{N}\right)}. \tag{1.4.1}$$

Sau đó, ta nhân với một bộ lọc thông thấp $\hat{h}(u, v)$:

$$\hat{g}(u, v) = \hat{h}(u, v) \cdot \hat{f}(u, v), \tag{1.4.2}$$

cuối cùng, ta biến đổi ngược để thu được ảnh mờ :

$$g(x, y) = \mathcal{F}^{-1}\{\hat{g}(u, v)\} \tag{1.4.3}$$

1.4.3.1 Lọc thông thấp Ideal

Là phép lọc 2 chiều đi qua tất cả tần số mà không làm giảm chúng trong bán kính đường tròn D_0 tính từ tâm phép lọc và "chặt cụt"tất cả tần số bên ngoài hình tròn này.

Bộ lọc này xác định như sau:

với D_0 là hằng số dương và $D(u, v)$ là khoảng cách giữa điểm (u, v) trong miền tần số tâm của hình chữ nhật tần số, tức:

1.4.3.2 Lọc thông thấp Gauss

Đặc trưng cho nhiễu đó là hàm mật độ xác suất thể hiện sự phân bố của nhiễu. Ta sử dụng hàm phân phối Gauss làm bộ lọc nhằm làm mờ ảnh và giảm nhiễu. Trong trường hợp 1 chiều, phân phối Gauss có công thức:

công thức

với σ là độ lệch chuẩn của phân phối, Ta giả sử phân phối này có trung bình là 0.

hình ảnh

Khi xử lý ảnh, ta sẽ sử dụng hàm phân phối Gauss cho 2 chiều, hình thành bằng tích của 2 hàm Gauss 1 chiều x và y

công thức

Bộ lọc thông thấp Gauss có dạng:

công thức

với $D(u, v)$ là khoảng cách từ điểm (u, v) đến tâm hình.

Ta sẽ xử lý ảnh sau với phép lọc thông thấp Gauss

hình ảnh

khởi tạo bộ lọc

dòng code

Ảnh của bộ lọc ở miền khong gian chuyển sang miền tần số là

hình ảnh

Sử dụng thuật toán 1.5 - 3, ta tính tích chập của ảnh gốc trong miền tần số và bộ lọc trong miền tần số, sao đó lấy biến đổi ngược.

Ta được ảnh sau khi lọc là:

hình

Ta quan sát phổ của ảnh ban đầu và sau khi lọc (hướng trên - phải xuống dưới - trái)

hình

Ta thấy rằng sau khi lọc, độ biến thiên tần số của ảnh ít hơn ảnh gốc, đồng thời các giá trị tần số ảnh sau khi lọc cao hơn ảnh gốc.

Ảnh sau khi lọc không xảy ra hiệu ứng chuông như phép lọc thông thấp Ideal. Phép lọc thông thấp Gauss này cho ảnh mượt hơn.

1.4.3.3 Lọc thông thấp Butterworth

Bộ lọc này bao gồm các tính chất của lọc thông thấp Ideal và lọc thông thấp Gauss. Bộ lọc thông thấp Butterworth có dạng sau:

công thức

Trong đó:

- $D(u,v)$ là khoảng cách từ tâm ảnh đến điểm (u,v) (giống như trong lọc thông thấp Ideal).

- D_0 là tần số chặt cụt.

-n là cấp của bộ lọc.

Từ dạng của bộ lọc, ta có những tính chất sau

- Với $D(u,v) \ll D_0, H \approx 1$

- Với $D(u,v) \gg D_0, H \approx 0$

- Với $D(u,v) = D_0, H = \dfrac{1}{2}$

Bây giờ, ta sử dụng bộ lọc này để lọc ảnh Mặt Trăng

ảnh

Khởi tạo bộ lọc có kích thước $P \times Q$ (bằng với kích thước hình ban đầu)

code

Khi đó, ta được ảnh bộ lọc trong miền không gian và miền tần số là

ảnh

Sử dụng thuật toán 1.5 - 3, ta tính tích chập của ảnh gốc trong miền tần số và bộ lọc trong miền tần số, sau đó lấy chuyển đổi ngược.

Ta được ảnh sau khi lọc là:

ẢNH

Ta quan sát phổ của ảnh gốc và sau khi lọc

ảnh

Với bộ lọc này, ảnh sau khi lọc có độ biến thiên tần số ít hơn, không làm tăng giá trị tần số, hiệu ứng chuông ảnh hưởng không đáng kể.

1.4.4 Ứng dụng phép lọc làm mờ ảnh

Phép lọc này có nhiều ứng dụng như:

- Nhận dạng ký tự trong tri thức máy.

- Dùng trong công nghiệp in ấn.

- Xử lý ảnh trên vệ tinh hoặc trên không trung.

- ...

Hình ảnh dưới đây cho thấy một đoạn văn có độ phân giải thấp, một vài ký tự bị đứt gãy.

Sử dụng phép lọc thông thấp Gauss, ta được ảnh đoạn vă rõ nét hơn.

ảnh

Mặc dù mất người dễ dàng nhận diện nét chữ đứt gãy nhưng hệ thống nhận dạng của máy thì không thấy như vậy. Một cách giải quyết đó là ta bắc cầu các mảnh nhỏ lại với nhau bằng cách làm mờ chúng. Hình bên phải cho thấy các kí tự rất nét bằng bộ lọc thông thấp Gauss với $D_0 = 80$.

Lọc thông thấp còn được áp dụng trong công nghiệp in ấn với nhiều hàm số dùng để "sơ chế"ảnh, bao gồm ảnh chưa xác định rõ hình dạng. Hình ảnh dưới đây là một ứng dụng của lọc thông thấp, giúp ảnh mượt hơn, dễ nhìn hơn.

ảnh

Với ảnh người, mục tiêu hướng đến là làm giảm độ nét của đương da mảnh và các nhược điểm nhỏ. Trong hình nền, ta đã làm giảm độ nét đường da quanh mắt, giúp cho ánh nhìn mượt mà hơn.

Ảnh sau cho ta 2 ứng dụng của lọc thông thấp trên cùng một hình nhưng với các đối tượng khác nhau. Ảnh bên trái có độ phân giải bức xạ rất cao, đó là ảnh Vịnh Mexico (tối) và Florida (sáng) do vệ tinh NOAA chụp lại. Biên của các vật thể trong nước được chụp bởi vòng lặp dòng. Ảnh này dùng để minh họa ảnh viễn thám với cảm biến có xu hướng đưa ra các dòng quét rõ ràng theo hướng quét cảnh.

ảnh

Lọc thông thấp cho kết quả thô nhưng đây là cách đơn giản để làm giảm hiệu ứng đường sọc ngang này. Hình giữa sử dụng lọc thông thấp Gauss với $D_0 = 50$. Làm giảm các hiệu ứng đường quét này giúp đơn giản hóa việc phát hiện các đặc tính như các biên giữa dòng nước biển.

Ảnh bên phải là kết quả của phép lọc thông thấp Gauss với $D_0 = 20$. Ảnh này cho thấy vật thể bị làm mờ nhiều chi tiết nhất có thể, để lại các đặc tính nhận dạng lớn. Cụ thể, cách lọc này là một phần của bước chuẩn bị cho hệ thống phân tích hình ảnh nhằm tìm kiếm các đặc tính ở rìa ảnh.

Phép lọc thông thấp giúp đơn giản hóa khi phân tích bằng cách tính trung bình các đặc tính nhỏ hơn đặc tính mong muốn.

1.4.5 Lọc sắc ảnh

Ta có thể làm mờ ảnh bằng cách làm giảm các tần số cao khi sử dụng chuyển đổi Fourier. Bây giờ, để làm sắc các biên ảnh, ta sẽ lọc các tần số cao và làm giảm các phần có tần số thấp bởi vì đường biên vật thể trong ảnh có độ biến thiên tần số lớn, ứng với các tần số cao.

ảnh

Khi đi qua đường biên, độ biến thiên tần số lớn.

Cho bộ lọc thông thấp, ta thu được bộ lọc thông cao bằng công thức:

$$\hat{h}_{HP}(u,v) = 1 - \hat{h}_{LP}(u,v) \tag{1.4.4}$$

với $\hat{h}_{HP}(u,v)$ là hàm chuyển đổi lọc thông thấp. Công thức này có nghĩa rằng khi lọc thông thấp. Công thức này có nghĩa rằng khi lọc thông thấp làm giảm tần số thì lọc thông cao bỏ qua điều đó và ngược lại.

Bây giờ, ta sẽ tìm hiểu một số bộ lọc thông cao.

1.4.5.1 Lọc thông cao Ideal

Lọc thông cao Ideal có dạng như sau:

$$\hat{h}(u,v) = \begin{cases} 0 \text{ nếu } \hat{d}(u,v) \le D_0 \\ 0 \text{ nếu } \hat{d}(u,v) > D_0 \end{cases} \tag{1.4.5}$$

Ta thấy rằng dạng của lọc thông cao Ideal ngược với dạng của lọc thông thấp Ideal. Ta dùng phép lọc này để lọc hình sau:

ảnh

Khởi tạo bộ lọc

code

Ta được hình ảnh bộ lọc từ miền không gian sang miền tần số như sau:

ảnh

Sử dụng thuật toán 1.5-3, ta tính tích chập của ảnh gốc trong miền tần số và bộ lọc trong miền tần số, sau đó lấy chuyển đổi ngược.

Ta được ảnh sau khi lọc là:

Ảnh

Nhìn vào ảnh, ta thấy rằng phép lọc này gây ra hiệu ứng chuông. Ta quan sát phổ của ảnh gốc và ảnh sau khi lọc (trên - phải xuống dưới - trái)

ảnh

Phép lọc này biến các giá trị tần số khác biên trở nên gần như nhau, đồng thời làm nổi bật tần số ở biên quả bí ngô.

Ảnh sau đây minh họa hiệu ứng chuông khi sử dụng phép lọc thông cao Ideal với giá trị D_0 (từ trái sang) lần lượt là 15, 30 và 80

Ảnh

1.4.5.2 Lọc thông cao Gauss

Phép lọc thông cao Gauss 2 chiều (2-D) có dạng:

$$\hat{g}(u,v) = 1 - e^{-\dfrac{\hat{d}^2(u,v)}{2\sigma^2}}. \tag{1.4.6}$$

Ta dùng phép lọc này để lọc ảnh quả dưa hấu sau:

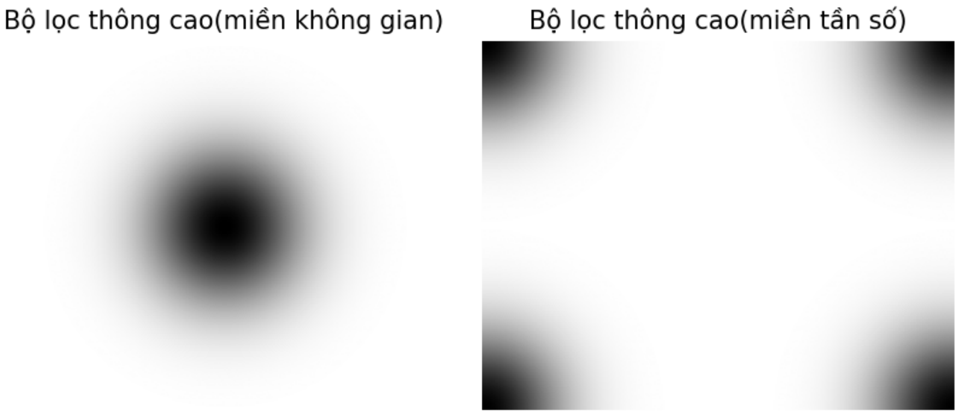
Hình 1.11: Ảnh quả dưa hấu

Khởi tạo bộ lọc trong python với `matplotlib`:

```
import numpy as np
import matplotlib.pyplot as plt
from PIL import Image

# 1 Doc anh goc
img = Image.open("img/duahau.png").convert("L")
f = np.array(img)
P, Q = f.shape

# 2 Tao bo loc thong cao Gauss
sig = 40
b = 2 * sig * sig

h = np.zeros((P, Q))
for i in range(P):
    for j in range(Q):
        D = (i - P/2)**2 + (j - Q/2)**2
        h[i, j] = 1 - np.exp(-D / b)

# Dich tam bo loc ve giua
H = np.fft.fftshift(h)

# 3 Bien doi Fourier anh va ap bo loc
F = np.fft.fft2(f)
F_shift = np.fft.fftshift(F)
G = H * F_shift

# 4 Bien doi nguoc de lay anh sau loc
G_ishift = np.fft.ifftshift(G)
filtered_img = np.abs(np.fft.ifft2(G_ishift))

# 5 Hien thi ket qua
plt.figure(figsize=(10, 6))

# Bo loc trong mien khong gian
plt.subplot(1, 3, 1)
plt.imshow(h, cmap='gray')
plt.title("Bo loc thong cao mien khong gian")
plt.axis('off')

# Bo loc trong mien tan so
plt.subplot(1, 3, 2)
```

```
plt.imshow(H, cmap='gray')
plt.title("Bo loc thong cao mien tan so")
plt.axis('off')

# Anh sau khi loc
plt.subplot(1, 3, 3)
plt.imshow(filtered_img, cmap='gray')
plt.title("Anh sau loc")
plt.axis('off')

plt.tight_layout()
plt.show()
```

Ta được ảnh bộ lọc trong miền không gian và miền tần số như sau:



Hình 1.12: Ảnh lọc thông cao Gauss miền không gian(bên trái) sang miền tần số(bên phải)

1.4.5.3 Lọc thông cao Butterworth

Phép lọc thông cao Butterworth trong không gian 2 chiều có cấp n và tần số chặt cụt D_0 xác định bởi công thức

công thức

Với $D(u, v)$ là khoảng cách từ tâm hình ảnh đến tọa độ (u, v).

Ta dùng bộ lọc này để lọc hình quả bí ngô:

hình ảnh

Khởi tạo bộ lọc

Code

Ta được ảnh của bộ lọc trong miền không gian và miền tần số như sau:

ảnh

Sử dụng thuật toán 1.5-3, ta tính tích chập của ảnh gốc trong miền tần số và bộ lọc trong miền tần số, sau đó lấy chuyển đổi ngược.

ảnh

Ta quan sát phổ của ảnh trước và sau khi lọc

ảnh

Ảnh sau khi lọc có xảy ra hiệu ứng chuông nhưng nhẹ hơn lọc thông cao Ideal, phổ tần số và ảnh cho thấy phần biên có tần số cao hơn hẳn những phần khác.

Hình ảnh sau là ví dụ của phép lọc thông cao Butterworth cấp 2 với các mức D_0 lần lượt (trái sang phải) là 15, 30 và 80

ảnh

1.4.6 Ứng dụng phép lọc sắc nét ảnh

Các phép lọc ảnh tần số cao này còn được sử dụng để nhận diện dấu vân tay con người. Một yếu tố quan trọng để máy móc nhận diện được vân tay đó là làm sao để tăng cường các đường vân và giảm thiểu đi các vết bẩn trên vân.

Để tăng cường độ sắc của đường vân ta chú tâm vào yếu tố quan trọng chính là đường vân tay chứa các tần số cao, không thay đổi khi ta sử dụng phép lọc thông cao. Mặt khác phép lọc này còn giảm các thành phần chứa tần số thấp ví dụ như phông nền hoặc các vết bẩn. Vì vậy, ta thu được ảnh tăng cường bằng cách giảm tất cả đặc trưng khác ngoại trừ đặc trưng có tần số cao.

Hình ảnh vân tay này là sử dụng phép lọc thông cao Butterworth cấp 4 và tần số chặt cụt là 50.

hình 1.5-36

Để có thể quan sát, ta sẽ biến đổi ảnh bên phải theo quy tắc điểm ảnh là giá trị âm sẽ có màu đen, còn dương thì sẽ có màu trắng

hình 1.5-37

Sử dụng phép lọc thông cao, ta loại bỏ những vết bẩn có trong ảnh ta đang xét và thu về ảnh vân tay rõ nét, thuận lợi hơn nếu công tác trong điều tra vụ án.

1.4.7 Lọc chặn

Đây là một ứng dụng của lọc ảnh trong miền tần số. Phép lọc chặn loại bỏ (hoặc đi qua) các tần số riêng biệt nào đó, ví dụ như nhiễu chu kỳ ứng với các nhánh hay các đường trong miền tần số, ta sẽ thiết kế một bộ lọc có tần số 0 tại những vị trí đó sẽ loại bỏ các tần số nhiễu.

Ví dụ về nhiễu chu kỳ như khảm ảnh khi kết hợp nhiều ảnh lại để tạo khảm, nhiễu dòng quét khi dùng máy ảnh quét, hay nhiễu bán sắc (kiểu gợn sóng) của bức ảnh trong tờ báo dưới đây.

hình 1.5-38

Chuyển ảnh vào miền tần số, ta được ảnh sau:

hình 1.5-39

Ta có thể thấy trên ảnh có những đỉnh nhỏ, các đỉnh này tương ứng với dạng nhiễu chu kì của ảnh bên miền không gian.

Các bước lọc chặn:

1. Nhìn vào các phổ $|\hat{f}(u,v)|$ của ảnh nhiễu $f(x,y)$, tìm vị trí tần số có liên quan đến nhiễu.

2. Tạo ảnh mặt nạ $\widehat{m}(u,v)$ với vết khuyết (các số 0) tại vị trí đó, những vị trí còn lại có giá trị bằng 1.

3. Lấy tích mặt nạ với ảnh ban đầu đã được chuyển đổi, các giá trị 0 sẽ làm mất các tần số nhiễu:
$$\hat{g}(u,v) = \widehat{m}(u,v)\hat{f}(u,v).$$

4. Lấy biến đổi Fourier ngược để thu về ảnh khôi phục:
$$g(x,y) = \mathcal{F}^{-1}\left(\hat{g}(u,v)\right).$$

Ta xác định vị trí có vết khuyết trong ảnh miền tần số sau:

hình 1.5-40

Lấy tích chập, chuyển đổi ngược, ta được kết quả:

hình1.5-41

Như vậy ta đã loại bỏ các đường quết của ảnh ban đầu.

Một ví dụ sau về phép lọc ảnh

hình 1.5-42

Kết luận

Sau khi tìm hiểu về lọc ảnh trong miền tần số, ta thấy phép lọc này có chức năng lọc những tần số thấp và cao trong ảnh.

1. Với thành phần có tần số cao liên quan đến biên vật thể trong ảnh, làm rõ biên. Các phép lọc trong cao làm giảm các thành phần số và bỏ qua thành phần tần số cao.

2. Với thành phần có thành số thấp liên quan đến vùng mượt trong ảnh, làm mờ ảnh. Cho phép lọc thông thấp giảm các thành phần tần số cao và bỏ qua thành phần tần số thấp.

Ngoài ra, phép lọc chặn giúp chúng ta loại bỏ các nhiễu lặp trong ảnh, làm giảm các tần số chọn trược (và một vài lân cận) và bỏ qua các tần số nhiễu khác.

Quy trình chung khi lọc trong miền tần số:

1. Dùng thuật toán FFT chuyển ảnh $f(x, y)$ sang miền tần số thành ảnh $\hat{f}(u, v)$.

2. Tạo bộ lọc $h(x, y)$ cùng kích thước với ảnh cần lọc, dùng FFT chuyển sang miền tần số thành $\hat{h}(u, v)$.

3. Thực hiện phép tính $\hat{g}(u, v) = \hat{h}(u, v)\hat{f}(u, v)$.

4. Lấy chuyển sổ đổi ngược IFFT của $\hat{g}(u, v)$, ta được hình $g(x, y)$ là hình sau khi lọc.

Bài báo cáo này đạt được các vấn đề sau đây:

- Chương 1:

- Chương 2:
- Chương 3:

Tài liệu tham khảo

[1] Michael Bartholomew-Biggs - Nonlinear Optimization with Engineering Applications (2008, Springer).