

# TRẢ LỜI CÂU HỎI BÀI TẬP TẠI LỚP – NHÀ

## 1. Các nhánh chính:

- Đồ họa máy tính:
  - Là một lĩnh vực của khoa học máy tính dùng để tạo, hiển thị, điều khiển hình ảnh trên máy tính. Mục đích chính của đồ họa máy tính là tăng tính trực quan của dữ liệu hình ảnh.
  - Input: Dữ liệu hình ảnh dạng thô.
  - Output: Các hình ảnh đã xử lý và có thể tương tác để có cái nhìn trực quan hơn.
- Thị giác máy tính:
  - Là một lĩnh vực khoa học liên ngành dùng để rút trích các thông tin hữu ích từ ảnh và video, bao gồm các phương pháp cơ bản như thu thập, xử lý, phân tích và rút trích thông tin.
  - Input: Dữ liệu ảnh hoặc video.
  - Output: các thông tin, kiến thức có giá trị từ dữ liệu đó.
- Xử lý ảnh và video số:
  - Là phương pháp thực hiện các phép toán trên ảnh hoặc video, để có được những ảnh hoặc video được tăng cường chất lượng hoặc lấy ra các thông tin hữu ích từ nó.
  - Input: Ảnh hoặc video dạng thô.
  - Output: Ảnh hoặc video được tăng chất lượng để mang lại nhiều thông tin hơn.

## 2. Các thành phần chính của ảnh là Object, Scene, Event.

- Object: Là vật thể có trong một bức ảnh. Vật thể này có thể là người, xe, con vật,...
- Scene: Là tập hợp các Object và có sự liên kết giữa các Object đó. Ví dụ, các xe đang dừng lại trước đèn đỏ.
- Event: Là chuỗi hoạt động của các Object trong một Scene. Các hoạt động này mang ý nghĩa và có liên kết chặt chẽ với Scene. Ví dụ, 2 boxer đang trong một trận đấu.
- Time: Thể hiện được mốc thời gian của một bức ảnh. Ví dụ, có được khoảng thời gian ước lượng từ hình ảnh mặt trời trên biển.
- Meaning: Thể hiện được nội dung của ảnh.

### 3. Input, Output:

	Input	Output	Ứng dụng
<b>Detection</b>	Ảnh grayscale	Boundary box	Xác định vật thể
<b>Recognition</b>	Ảnh grayscale	Định danh của vật	Nhận diện các vật thể các nhau
<b>Classification</b>	Ảnh grayscale	Loại hoặc class của vật	Phân loại vật thể
<b>Retrieval</b>	Ảnh grayscale	Các ảnh có vật thể/ cảnh tương tự	Tìm kiếm bằng hình ảnh
<b>Tracking</b>	Ảnh grayscale	Boundary box và định danh của vật	Camera giám sát
<b>Counting</b>	Ảnh grayscale	Dữ liệu thống kê	Thống kê sơ bộ trong nông nghiệp

**4. Mục đích của việc xử lý ảnh:** Để chỉnh sửa, xử lý hình ảnh phù hợp với mục đích sử dụng của cá nhân hay tổ chức nào đó để đáp ứng nhu cầu trong một hay nhiều lĩnh vực nào đó (an ninh, giáo dục, giải trí,...).

### 5. Các ứng dụng của xử lý ảnh:

- Trong lĩnh vực an ninh: các ứng dụng cụ thể của xử lý ảnh có thể là nhận dạng (khuôn mặt) tội phạm tự động tại các điểm giao thông công cộng (nhà ga, sân bay, trên phố,...), nhận dạng vân tay, phát hiện và cảnh báo xâm nhập trái phép, tìm kiếm đối tượng mất tích, nhận dạng và dự đoán hành vi phạm tội,...
- Trong lĩnh vực giao thông thông minh, xử lý ảnh giúp phát hiện tự động các phương tiện vi phạm luật giao thông (chạy quá tốc độ, vượt đèn đỏ,...), ước lượng mật độ xe lưu thông trên đường (cảnh báo tắc nghẽn), nhận dạng biển số xe, tự động thu phí các phương tiện giao thông,...
- Trong lĩnh vực giáo dục, các ứng dụng tiềm năng bao gồm lớp học ảo (mô phỏng các ca phẫu thuật ảo), chấm bài thi trắc nghiệm tự động, xử lý tự động các phiếu điều tra, khảo sát, nhận dạng văn bản và công thức toán.
- Trong lĩnh vực y tế: xử lý ảnh giúp phát triển các hệ thống chẩn đoán hình ảnh siêu âm, hình ảnh chụp cắt lớp, lọc nhiễu và làm nổi các chi tiết bất thường trong

ảnh y tế, phát hiện dấu hiệu gây ung thư sớm và nhiều căn bệnh khác bằng cách phân tích các vùng ảnh bất thường (màu da, kết cấu, giá trị xám,...).

- Ngoài ra, xử lý ảnh cũng được ứng dụng rộng rãi trong các lĩnh vực khác như giải trí (thiết kế game, các hiệu ứng ảnh động), truyền hình số (nén ảnh), thể thao (phát hiện các banner quảng cáo trong video bóng đá, hay các thương hiệu logo nổi tiếng,...).

## 6. Phân biệt ảnh gray scale và ảnh depth:

- Grayscale image: là ảnh thông thường được thể hiện dưới hệ màu gray, tức là giá trị màu nằm trong khoảng 0-255.
- Depth image: là ảnh thể hiện độ sâu dựa vào khoảng cách đến điểm đo. Ảnh này không thể hiện giá trị màu của các điểm ảnh mà thể hiện khoảng cách từ điểm nhìn (điểm gốc để đo giá trị) đến vị trí thực tế của điểm ảnh.

## 7. Chuyển ảnh RGB sang ảnh Grayscale:

Từ ảnh RGB, áp dụng công thức để chuyển sang ảnh Grayscale:

$$*R+*G+*B$$

, với các hệ số được người dùng chỉ định. Các giá trị thường được sử dụng nhiều nhất là (0.2989, 0.5870, 0.1140).

## 8. Chuyển ảnh từ Grayscale sang RGB:

Ngược lại, chuyển từ hệ màu gray sang hệ màu RGB khó hơn nhiều. Có một vài phương pháp đã được đề xuất, nhưng đơn giản nhất là công thức:

$$R=G=B=Gray$$

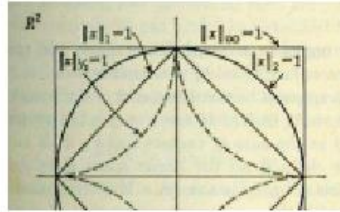
## 9. Định lượng màu:

- Định lượng màu là phương pháp được áp dụng vào miền không gian giá trị màu của ảnh với mục đích giảm số lượng giá trị màu được dùng trong một ảnh, và ảnh kết quả càng giống với ảnh gốc càng tốt.
- Cần phải dùng đến định lượng màu vì tùy với nhu cầu của người dùng, có nhiều trường hợp ta không cần giá trị màu của điểm ảnh quá chi tiết, chỉ cần biết là nó có giá trị màu khác với những điểm ảnh khác. Mà việc này không cần đến số lượng giá trị màu cao như hệ màu RGB, nên ta dùng đến định lượng màu để giảm không giá giá trị màu mà vẫn bảo toàn được đặc tính khác biệt giữa các điểm ảnh.

### Khoảng cách giữa các điểm ảnh

$$d(p, q) = \left( \sum_{i=1}^n |p_i - q_i|^r \right)^{1/r},$$

$p_i, q_i$  là tọa độ thứ  $i$  của điểm  $p, q$



10.

11. Sau khi định lượng màu, ta mất đi một lượng lớn giá trị màu, nhưng ta vẫn chấp nhận điều đó bởi vì việc mắt người rất khó phân biệt độ chênh lệch giữa các giá trị màu gần giống nhau, từ đó việc gộp các giá trị màu này lại không ảnh hưởng đến cảm nhận của con người. Bên cạnh đó, việc bỏ đi một phần chi tiết của ảnh để tiết kiệm được một lượng lớn không gian bộ nhớ lưu trữ rất đáng để cân nhắc.

### 12. Công thức tính đặc trưng màu (lược đồ màu):

- Công thức:

$$H_t(i) = \frac{N_t(i)}{N(t)}$$

với  $H(i)$  là giá trị thành phần thứ  $i$  trong lược đồ màu trong khung hình thứ  $t$ ,  $N(i)$  là số lượng điểm ảnh của bin màu  $i$  và  $N(t)$  là tổng số điểm ảnh trong khung hình thứ  $t$ .

- Input: dữ liệu dạng ảnh.
- Output: một lược đồ màu với mỗi cột của lược đồ là tỉ lệ xuất hiện của một bin màu.

13. Nếu bỏ đi mẫu số trong công thức lược đồ màu, ta vẫn có được một lược đồ màu khác. Tuy nhiên lược đồ này không thể hiện được tỉ lệ xuất hiện của bin màu trong ảnh, mà thể hiện số lần xuất hiện của bin màu đó. Việc này sẽ gây ra một vài vấn đề khi tổng số lượng điểm ảnh trong khung hình không nhất quán. Ví dụ, trong một khung hình có độ lớn  $10 \times 10$  pixel, màu đỏ xuất hiện ở 10 pixel (10%), nhưng trong khung hình khác có độ lớn  $1000 \times 1000$  pixel, cũng 10 pixel đó nhưng tỉ lệ giờ là 0.001%. Ta thấy được dù là cùng là 10 pixel nhưng tỉ lệ khác biệt rất lớn, từ đó việc không dùng đến mẫu số có ảnh hưởng rất lớn để việc thể hiện tính chất của ảnh.

### 14. Từ hệ màu HSV, làm sao để map vào mảng 1 chiều

- Đầu tiên, ta cần định lượng màu để giảm số lượng bin màu cho 3 kênh màu HSV. Gọi  $NH, NS, NV$  lần lượt là số bin màu cho các bin màu,  $H(i), S(i), V(i)$  là các giá trị bin màu của màu thứ  $i$ . Tạo một mảng có độ lớn  $NH * NS * NV$  phần tử.
- Từ HSV lưu vào mảng một chiều:

$$His[i] = H(i) * (NS * NV) + S(i) * NV + V(i)$$

**15. Công thức lược đồ hệ số góc dùng để thể hiện đặc trưng đáng:**

- Gồm có 73 phần tử.
- 72 phần tử đầu tỉ lệ các điểm ảnh tương tự nhau.
- Phần tử cuối là số lượng điểm ảnh không nằm trên biên.

$$H_i = \frac{m(I_0)}{n_{I_0}}$$

với  $m(I)$  là số lượng điểm ảnh có hệ số góc  $\alpha$ .

- Có những trường hợp các ảnh khác nhau nhưng lại có chung lược đồ màu hoặc lược đồ hệ số góc. Điều này là có thể tại vì các ảnh đó có tỉ lệ xuất hiện của các màu tương tự nhau nhưng vị trí xuất hiện trong ảnh khác nhau dẫn đến ảnh khác nhau. Tương tự như lược đồ hệ số góc.

**16. Ma trận đồng hiện để thể hiện đặc trưng vân:**

- Công thức:

$$C[i, j] = |r, c|f(r, c) = if(r + r_x, c + r_y) = j|$$

- Đặc trưng vân Haralick:

$$N_d(i, j) = \frac{C_d(i, j)}{\sum_i \sum_j C_d(i, j)}$$

$$N_{d,1}(i) = \sum_j N_d(i, j)$$

$$N_{d,2}(j) = \sum_i N_d(i, j)$$

$$\mu_i = \sum_i i N_{d,1}(i)$$

$$\sigma_i = \sum_i (i - \mu_i)^2 N_{d,1}(j)$$

$$\mu_j = \sum_j j N_{d,2}(j)$$

$$\sigma_j = \sum_j (j - \mu_j)^2 N_{d,1}(j)$$

$$Energy = \sum_i \sum_j N_d^2(i, j)$$

$$Entropy = \sum_i \sum_j N_d(i, j) \log_2(N_d(i, j))$$

$$Contrast = \sum_i \sum_j (i - j)^2 N_d(i, j)$$

$$homogeneity = \sum_i \sum_j \frac{N_d(i, j)}{1 + |i - j|}$$

$$Correlation = \frac{\sum_i \sum_j (i - \mu_i)(j - \mu_j) N_d(i, j)}{\sigma_i \sigma_j}$$



**21. Sự khác nhau giữa phép biến đổi phi tuyến và tuyến tính.**

Sự khác nhau cơ bản là độ lớn của sự thay đổi của kết quả sau khi áp dụng phép biến đổi. Trong phép biến đổi tuyến tính, độ lớn này là không thay đổi (xét trường hợp các khoảng cách của input là như nhau), còn trong phi tuyến, độ lớn thay đổi tùy theo phép biến đổi đang áp dụng.

**\*(bổ sung) Phép biến đổi Fourier làm sao áp dụng vào làm trơn ảnh và phát hiện biên cạnh?**

Phép biến đổi Fourier chuyển ảnh qua miền tần số rồi dùng bộ lọc thông thấp để làm trơn ảnh và thông cao để phát hiện biên cạnh, cuối cùng chuyển ảnh về lại miền không gian.

**22. Giải thuật cho toán tử trung bình trong làm trơn ảnh trên miền không gian.**

- Xét ảnh gốc  $f$
- Với mỗi điểm ảnh không phải biên, tính trung bình giá trị màu của điểm ảnh đó và các điểm ảnh trong lân cận 8. Đưa kết quả nhận được vào ảnh kết quả.
- Với các điểm ảnh là biên của ảnh gốc, có nhiều phương pháp để xử lý những điểm ảnh này. Điển hình một phương pháp là bỏ qua các điểm này, lấy giá trị tại ảnh gốc và đưa vào ảnh kết quả. Ngoài ra còn có phương pháp tính trung bình các điểm ảnh có giá trị (các lân cận mà nằm ngoài ảnh thì ta bỏ qua).

**23. Bản chất của toán tử Gaussian trong làm trơn ảnh.**

- Làm trơn ảnh bằng toán tử Gaussian là phương pháp áp dụng công thức Gaussian cho toàn bộ các điểm ảnh của ảnh gốc để có được ảnh kết quả.

$$G(x, y) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{x^2 + y^2}{2\sigma^2}}$$

- Công thức:
- Bản chất của toán tử Gaussian là phép biến đổi tuyến tính dựa phân bố xác suất chuẩn hóa (Normal Distribution). Phân bố này rất quan trọng vì nó xuất hiện rất nhiều trong thế giới thực. Hầu hết mọi thứ đều nằm trong phân phối xác suất này. Vì thế, nếu tận dụng được sự phân phối này trong xử lý ảnh, thì ta sẽ sử dụng được rất tốt nguồn thông tin của ảnh đó.

**24. Toán tử trung vị trong làm trơn ảnh.**

- Toán tử trung vị là một phương pháp làm trơn ảnh dựa vào giá trị màu của các lân cận của điểm ảnh đó tương tự như toán tử trung bình. Nhưng thay vì tính giá trị trung bình, trong toán tử trung vị, ta sẽ lấy giá trị median.
- Giải thuật:

- Xét ảnh gốc  $f$ .
- Với các điểm ảnh không phải là biên ảnh:
  - Lấy giá trị màu của các điểm ảnh lân cận.
  - Sắp xếp các giá trị này theo giá trị màu.
  - Lấy giá trị median của mảng đã sắp xếp này và lưu kết quả vào ảnh kết quả.
- Với các ảnh là biên ảnh, ta làm tương tự nhưng chỉ khác ở cách lấy các điểm ảnh lân cận. Tùy theo nhu cầu của người dùng, ta sẽ có các cách lấy lân cận khác nhau, một cách là lấy các lân cận theo hình chữ thập (lấy 2 điểm ảnh theo 4 hướng của điểm ảnh đang xét)

## 25. Bảng so sánh 3 toán tử làm trơn.

	Nguyên lý	Công thức tính	Phương pháp	Ưu	Nhược	Tổng quát
Toán tử trung bình	- Phép biến đổi cục bộ dựa trên tích chập với hàm lọc	$g(x,y) = \sum_i \sum_j g(x-i,y-j)h(i,j)$ với $x, y \in 0$ $h$ là hàm lọc	- Dùng phép tính trung bình cộng - Tính giá trị tổng hợp	- Nhanh - Dễ dàng	- Dễ dàng nhiễu do phụ thuộc nhiều các điểm lân cận	Gaussian, median
Toán tử Gaussian	—//—	$h(x,y) = \frac{1}{N(\sigma)}$ $h = \frac{1}{2\pi\sigma^2} e^{-\frac{x^2+y^2}{2\sigma^2}}$	- Dùng hàm lọc Gaussian - pixel lân cận đóng góp nhiều hơn	- Pixel kq - Khử nhiễu - Các giá trị đóng góp nhiều hơn pixel lân cận	- Khử nhiễu - pixel lân cận đóng góp nhiều hơn pixel trung tâm	
Toán tử trung vị	- Phép biến đổi cục bộ dựa trên phép lọc chọn điểm trung vị	$g(x,y) = \text{med}\{g(x-i,y-j)   h(i,j) \neq 0\}$	- Dùng hàm lọc chọn trung vị	- Pixel kq - Vẫn giữ được biên ảnh	- Vẫn giữ được biên ảnh - Khử nhiễu qua biên ảnh	

## 26. Ví dụ của việc làm trơn ảnh:

- Khử nhiễu trong dữ liệu ảnh. Vd: ảnh thu được từ vệ tinh, từ tàu thám hiểm trên Sao Hỏa,...
- Làm mờ nội dung. Vd: làm mờ ảnh thu được từ khu quân sự hoặc những vùng chính trị,...
- Tăng tính thẩm mỹ của ảnh. Vd: làm mờ những vật thể xung quanh để làm nổi bật vật thể chính trong một bức ảnh.

## 27. Làm sao để phát hiện những điểm có sự biến thiên lớn về độ sáng



Ta tính gradient bình phương hay dùng mặt nạ Laplace rất nhạy cảm với biến đổi để phát hiện thay đổi.

## 28. Những cách tính gradient.

- Cách 1:
  - $f_x(x, y) = f(x, y) - f(x + 1, y)$
  - $f_y(x, y) = f(x, y) - f(x, y + 1)$
- Cách 2:
  - $f_x(x, y) = f(x, y) - f(x + 1, y + 1)$
  -
- Cách 3:
  - $f_x = \frac{1}{k+2}[(A_2 + kA_3 + A_4) - (A_0 + kA_7 + A_6)]$
  - $f_y = \frac{1}{k+2}[(A_0 + kA_1 + A_2) - (A_6 + kA_5 + A_4)]$

## 29. Khuyết điểm của toán tử gradient

- Khuyết điểm chí mạng nhất của các cách tính gradient đơn giản như cách 1 hoặc cách 2 là sự ảnh hưởng của nhiễu. Với việc có cách tính đơn giản như vậy, chỉ xét đến giá trị của một điểm ảnh gần đó để tính giá trị gradient thì chỉ cần một lượng nhỏ điểm ảnh nhiễu là có thể ảnh hưởng đến kết quả nhận được.
- Để khắc phục điều này, có những cách tính khác được đề xuất (như cách 3). Cách tính này sử dụng giá trị của nhiều điểm ảnh để khắc phục sự phụ thuộc vào một điểm ảnh duy nhất (phòng trường hợp điểm ảnh này là nhiễu).

## 30. Giải thích sự khác nhau của hệ số k trong các cách tính gradient

- Với cách tính của Prewitt (k=1): sử dụng các lân cận với các trọng số như nhau để tính gradient tại một điểm. Chính vì thế, phương pháp tính này rất nhạy với nhiễu.
- Với cách tính của Sobel (k=2): sử dụng các lân cận với tổng số khác nhau (các điểm liền kề ở 4 hướng có trọng số lớn hơn). Phương pháp này rất nhạy với các đường biên chéo nhưng rất hiệu quả trong hầu hết các trường hợp.
- Với cách tính của Frei-chen (k=sqrt(2)): cải tiến từ phương pháp của Sobel và sử dụng hệ số k tốt hơn. Với hệ số k mới này, việc phát hiện biên cạnh ở các đường chéo tốt hơn. Chính vì vậy, phương pháp Frei-chen dễ dàng phát hiện ra được các biên nhỏ và tạo ra được các đường biên nhỏ.

## 31. Các công thức bộ lọc với từng hệ số k khác nhau

- Với k = 1:
 
$$G_x = \frac{1}{3} \begin{bmatrix} -1 & 0 & 1 \\ -1 & 0 & 1 \\ -1 & 0 & 1 \end{bmatrix}$$

$$G_y = \frac{1}{3} \begin{bmatrix} 1 & 1 & 1 \\ 0 & 0 & 0 \\ -1 & -1 & -1 \end{bmatrix}$$

- Với  $k = 2$ :

$$G_x = \frac{1}{4} \begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix}$$

$$G_y = \frac{1}{4} \begin{bmatrix} 1 & 2 & 1 \\ 0 & 0 & 0 \\ -1 & -2 & -1 \end{bmatrix}$$

- Với  $k = 2$ :

$$G_x = \frac{1}{2\sqrt{2}} \begin{bmatrix} -1 & 0 & 1 \\ -\sqrt{2} & 0 & \sqrt{2} \\ -1 & 0 & 1 \end{bmatrix}$$

$$G_y = \frac{1}{2\sqrt{2}} \begin{bmatrix} 1 & \sqrt{2} & 1 \\ 0 & 0 & 0 \\ -1 & -\sqrt{2} & -1 \end{bmatrix}$$

### 32. Input, Output của việc tính gradient

- Input: Ảnh nguồn.
- Công thức:  $\nabla f(x, y) = [\partial f / \partial x, \partial f / \partial y] = [f_x, f_y]^T$
- Output: Các thông tin cơ bản của ảnh như:
  - Độ lớn:  $e(x, y) = \sqrt{f_x^2(x, y) + f_y^2(x, y)}$
  - Góc gradient:  $\phi(x, y) = \arctan(f_x, f_y)$

### 33. Phương pháp Karhunen Loeve

- Phát biểu bài toán: Cần tìm phép biến đổi tuyến tính để:
  - Giảm số chiều không gian đặc trưng nhưng vẫn giữ được được các đặc trưng chính của ảnh.
  - Cực tiểu hóa việc mất mát thông tin, đồng nghĩa với việc giữ được nhiều thông tin nhất có thể từ ảnh gốc.

- Phương pháp:

- $y = ATx$

- Tìm xấp xỉ của  $x$  để giảm số chiều nhưng vẫn thỏa yêu cầu cực tiểu mất mát thông tin.

$$\hat{x} = \sum_{i=0}^{m-1} y_i a_i$$

$$\text{Cực tiểu } E(\|x - \hat{x}\|^2)$$

- Các bước thực hiện:

- Tính  $ATB = \frac{1}{M} \sum_{i=1}^M I_i(x, y)$

- Tính ảnh độ lệch  $X_i = I_i - ATB$
- Tìm M giá trị riêng của  $C'_x = X^T X$  và giữ lại K giá trị lớn nhất
- Tính mật riêng là các vector riêng của  $C_x = XX^T$ 
  - $F = X.F'_M$
  - $F_i(j) = \sum_{l=1}^M X_i(j)F'_j(l)$
- Chiếu tập ảnh xuống không gian vector con hợp bởi K vector riêng của  $C_x$ .
  - $Y = A'^T.X$
  - $Y_i(j) = \sum_{l=1}^N F_j(l)X_i(l)$
  - A' chính là A nhưng chỉ gồm K vector riêng
- Xấp xỉ ảnh bởi hình chiếu của nó xuống không gian con hợp bởi các vector riêng của  $C_x$ 
  - $I_i = ATB + \sum a_{ij}F_j$
  - Vector đặc trưng của  $I_i$  là  $\{a_{ij}, j = 1, 2, \dots, k\}$