

XỬ LÝ ẢNH VÀ VIDEO SỐ

BÀI TẬP LỚP NHÀ



Giáo viên: PGS. TS. Lý Quốc Ngọc

Nhóm Depression:

- Chung Kim Khánh
- Lê Nguyễn Anh Khôi
- Bùi Ngọc Chính

MỤC LỤC

I.	THÔNG TIN NHÓM.....	2
II.	TUẦN 1 (09/10/2021)	3
III.	TUẦN 2 (16/10/2021).....	6
IV.	TUẦN 3 (06/11/2021).....	12
V.	TUẦN 4 (13/11/2021).....	15
VI.	TUẦN 5 (27/11/2021).....	18
VII.	TUẦN 6 (04/12/2021).....	21
VIII.	TUẦN 7 (11/12/2021).....	25
IX.	TUẦN 8 (18/12/2021).....	28

I. THÔNG TIN NHÓM

Tên nhóm: Depression

MSSV	Họ Tên	Email
19127644	Chung Kim Khánh	ckkhanh19@clc.fitus.edu.vn
19127451	Lê Nguyễn Anh Khôi	lnakhoi19@clc.fitus.edu.vn
19127109	Bùi Ngọc Chính	bnchinh19@clc.fitus.edu.vn

II. TUẦN 1 (09/10/2021)

1. Thông minh là gì? Hiểu sao về câu “Thông minh là khả năng thích ứng với sự thay đổi”.

Là khi có sự thay đổi thì vẫn có thể linh hoạt thích nghi với sự thay đổi đó, ví dụ như hệ thống giám sát thông minh: khi trời mưa vẫn có thể giám sát, trời nắng gắt vẫn có thể giám sát, khi con người già đi vẫn nhận dạng được,...

2. Cho một ví dụ hệ thống thị giác thông minh.

- **System name:** Hệ thống mắt thần (của Trung Quốc)

Input: video hành động của mọi người trên đường. (dãy ảnh)

Output:

_Data: Ảnh đã tăng chất lượng

_Information: Xác định được các đối tượng (người, xe, cây, ...)

_Knowledge: Biết thông tin người, giới tính, loại cây, xe hãng nào, ...

_Intelligence: Đánh giá hành vi tốt hay xấu, xác định được cảm xúc, đánh giá độ tuổi, chủng tộc, ...

_Wisdom: Dự đoán hành vi đối tượng trong tương lai.

- **System name:** Hệ thống giám sát giao thông (Đồng Nai)

_Data: Ảnh đã tăng chất lượng

_Information: Xác định được các đối tượng (người, xe, cây, ...)

_Knowledge: Biết thông tin xe bao gồm: biển số, hãng, chủ xe, ...

_Intelligence: Đánh giá từng xe xem có vi phạm luật giao thông hay không

_Wisdom: Điều khiển phân luồng giao thông, tự gửi giấy báo phạt vào điện thoại.

3. Kỹ thuật mô phỏng sẽ được áp dụng vào đâu, dữ liệu output ở tầng nào?

- Output ở mức data.
- Áp dụng vào: phim ảnh, bắt động sản, thể thao, thiết kế nội thất, ...

4. Theo em trong ảnh có gì? Ảnh là gì? Nội dung chứa gì?

Trong ảnh mang 2 thuộc tính:

- Đặc tính về thị giác (visual level/visual information): color, texture, shape.
- Đặc tính về ngữ nghĩa (semantic level):

- o Object (đối tượng):

- Specific object (đặc thù): hiệu xe hơi Innova/Mercedes
- General object (tổng quát): Xe hơi (knowledge)
- Abstract object (trừu tượng): phương tiện giao thông

- o Scene (cảnh - gồm nhiều objects hợp lại):

- Specific scene: Vịnh Cam Ranh/San Francisco (Intelligence)
- General scene: Vịnh - gồm có bãi biển, bãi cát, núi, mây, cây...
- Abstract scene: Cảnh thiên nhiên
- Event: (relationship giữa objects và scenes)
 - Specific event: Cảnh ghi bàn trong trận chung kết World Cup 2022
 - General event: Cảnh ghi bàn
 - Abstract event: Cảnh quan trọng trong thể thao

5. Task của computer vision (trong video)? Tác vụ? Mức độ thông minh của computer vision?

Person tracking (Theo vết người):

Input: Video số quay từ camera đặt tại nhiều vị trí

Dùng đồ họa máy tính để mô phỏng lại thế giới thực từ input, qua đó giúp cho CV nhận biết và đánh giá đối tượng và hành vi của đối tượng đó.

Output:

_Data: Video có từng khung hình đã qua xử lý video số (task của DVP).

_Information: Xác định đâu là người ở trong video.

_Knowledge: Xác định thông tin và hành vi của đối tượng.

_Intelligence: Đánh giá hành vi của đối tượng.

_Wisdom: Dự đoán hành vi trong tương lai gần của đối tượng.

Trong video thì hệ thống đó thông minh ở mức Intelligence. Vì hệ thống có xác định được đối tượng nguy hiểm (hiển thị khung màu đỏ thay vì màu xanh).

Tác vụ của CV:

Visual Data Understanding:

- Object Detection: tìm kiếm con người trong khung hình.
- Object Recognition: nhận dạng con người đã được phát hiện trong khung hình.
- Object Retrieval: truy xuất thông tin đối tượng.
- Object Classification: phân loại người cần truy vết và không cần truy vết.
- Object Tracking: theo dõi đối tượng cần truy vết qua chuỗi các khung hình.

Xe tự hành: Những điểm lấy được là mức intelligence.

Input: Video số quay từ camera đa hướng trên xe.

Dùng đồ họa máy tính để mô phỏng lại thế giới thực từ input, qua đó giúp cho CV nhận biết và đánh giá đối tượng và hành vi của đối tượng đó.

Output:

- _Data: Chuỗi khung hình đã tăng chất lượng từ các video.
- _Information: Các vật thể xung quanh xe.
- _Knowledge: Xác định vị trí các vật thể, biển báo.
- _Intelligence: Lập kế hoạch chạy (chọn đoạn đường tối ưu nhất, tốc độ phù hợp nhất, ...).
- _Wisdom: Hệ thống xử lý mà không cần sự can thiệp của con người.

Tác vụ của CV:

Visual Data Understanding:

- Object, Event Detection: phát hiện vật thể và sự kiện diễn ra trong các khung hình.
- Object, Event Recognition: nhận dạng vật thể và sự kiện đã được phát hiện trong khung hình.
- Object, Event Classification: phân loại các vật thể (di chuyển hay không) và các sự kiện (nguy hiểm hay không).
- Object, Event Tracking: theo dõi vật thể, diễn biến sự kiện qua chuỗi các khung hình.

3D object reconstruction: Một thiết bị dùng để tạo bản đồ 3D của những vật thể xung quanh chiếc xe bằng cách phát ra những tia laser ở mọi hướng (LiDAR).

Visual Slam:

- Visual Localization, Mapping: tạo bản đồ 3D môi trường xung quanh đồng thời định vị trí dựa trên hình ảnh được chụp từ camera
- Obstacle detection & avoidance: phát hiện và né tránh các vật thể trên đường đi.
- Moving object tracking: theo dõi các vật thể chuyển động.

Hệ thống thu hoạch trái trong nông nghiệp:

Ứng dụng: Thu hoạch ớt chuông (Trái cây)

Tác vụ: Phát hiện (detection object), 3D object, recognition object (phân biệt chín, hư), visual location (Vị trí của trái, cây).

Input: chuỗi hình ảnh số quay từ camera.

Dùng đồ họa máy tính để mô phỏng lại thế giới thực từ input, qua đó giúp cho CV nhận biết và đánh giá đối tượng và hành vi của đối tượng đó.

Output:

- _Data: Chuỗi khung hình đã tăng chất lượng từ các chuỗi hình ảnh số.
- _Information: Các vật thể trong khung hình.
- _Knowledge: Xác định vị trí, độ chín của các quả ớt chuông.
- _Intelligence: Chọn ra quả chín để thu hoạch.
- _Wisdom: Dự báo thời gian chín của quả ớt, đánh giá tình trạng của cây.

III. TUẦN 2 (16/10/2021)

1. Tập trái gồm 200k ảnh, tập phải gồm 200k frames. Xử lý như thế nào?

Bản thân ảnh mang **2 thuộc tính**:

Hình ảnh: Spatial. (không gian)

Video: Spatial + Temporal. (không gian và thời gian).

Điểm chung: Cùng cố gắng khai thác visual cues và semantic cues

Đối với tập trái: chúng ta chỉ cần xử lý về mặt không gian (Spatial) xem nội dung của ảnh là gì? Và các ảnh độc lập với nhau.

Đối với tập phải: chúng ta cần phải xử lý về mặt không gian và cả thời gian để các khung hình có mối liên kết với nhau đúng theo thời gian diễn ra.

2. Tại sao xe tự hành, làm xe ra để lái, rồi giờ muốn nó tự đi. Liệu có hợp logic không?

Xe tự hành không giảm tai nạn. Không công ty bảo hiểm nào ký kết vì không biết người nào sẽ chịu trách nhiệm khi có tai nạn.

Làm xe để lưu thông, phục vụ nhu cầu di chuyển. Chứ không phải để lái.

- Phục vụ cho con người trong những lúc cần thiết. Ví dụ khi đang chạy xe bị choáng hay chóng mặt... thì xe tự hành sẽ đảm bảo an toàn cho người lái.
- Vì nhu cầu của con người ngày càng tăng cao nên xe tự hành đáp ứng nhu cầu của con người tốt hơn xe tự lái nên được áp dụng nhiều hơn.

Chốt lại:

Theo CM 4.0 thì muốn thông minh hoá sản xuất và quản lý xã hội. Xe tự hành thực hiện thông minh hoá xã hội bằng cách chuyển đổi số trong giao thông (AI).

Xe mỗi ng tự ý tự đi thì sẽ tăng lượng khí thải → gây ra hiệu ứng nhà kính từ đó dẫn đến thủng tầng ozone. Không cần lượng xe lưu thông quá nhiều trên đường phố.

3. Xử lý dữ liệu ảnh 2 chiều, video 2 chiều như hiện nay, giả sử có một ứng dụng/tác vụ xử lý ảnh video 3 chiều (lập băng). So sánh 2 tập dữ liệu đó: ưu khuyết điểm là gì? Cái nào buộc cần 3 chiều. Cái nào không cần thiết?

Đầu tư camera trong ngành Y tại nó khó và chuyên nghiệp.

Ảnh / Video 2D	
Ưu điểm	Nhược điểm

<ul style="list-style-type: none"> + Gọn nhẹ + Dễ thực thi + Góc nhìn 1 hướng + Thích hợp chọn nhu cầu cơ bản (hình lưu niệm, selfie, ...) 	<ul style="list-style-type: none"> + Chỉ nhận diện được màu sắc, hành động. + Khó nhận biết được chính xác khoảng cách và kích thước.
--	---

Ảnh / Video 3D	
Ưu điểm	Nhược điểm
<ul style="list-style-type: none"> + Góc nhìn đa hướng, tổng quan + Thích hợp cho những công việc cần góc nhìn đa hướng (VAR, hình ảnh y học...) + Có thể xác định khoảng cách, kích thước của các vật thể trong ảnh. 	<ul style="list-style-type: none"> + Nặng hơn + Khó thực thi

2D chỉ là cái bóng của thế giới thực (3D) chiếu xuống, làm mất đi kênh thông tin liên lạc giữa môi trường ảo và môi trường thật, bù lại 2D giảm số chiều dữ liệu, đủ đáp ứng nhiều tác vụ.

4. Nếu màn hình máy tính, mỗi pixel là 1 điểm RGB có cường độ khác nhau, giúp mắt người nhìn được màu sắc hiển thị trên màn hình, vậy một tờ giấy, không có pixel, thì làm sao mắt người có thể nhìn được màu sắc trên tờ giấy?

Một trang giấy thì có CMY (cyan, magenta, yellow). Giấy chỉ in đc 3 màu này -> nó đối trọng (tương ứng) với RGB (màu ánh sáng phản xạ). Phóng lớn giấy lên thì thấy hạt **Tram** - phân bố các hạt CMY lớn nhỏ phân bố vị trí góc nghiêng để hấp thụ và phản xạ ánh sáng ra RGB. CMY hợp lại theo lý thuyết thì ra màu đen. Nhưng thực tế nó thấm xuống tờ giấy (bị tiêu hao năng lượng) thì lại ra màu nhạt hơn một tí. Vì vậy phải thêm một màu mực BK (**black**).

5. Tính mối quan hệ u, v và x, y, z trong thế giới thực

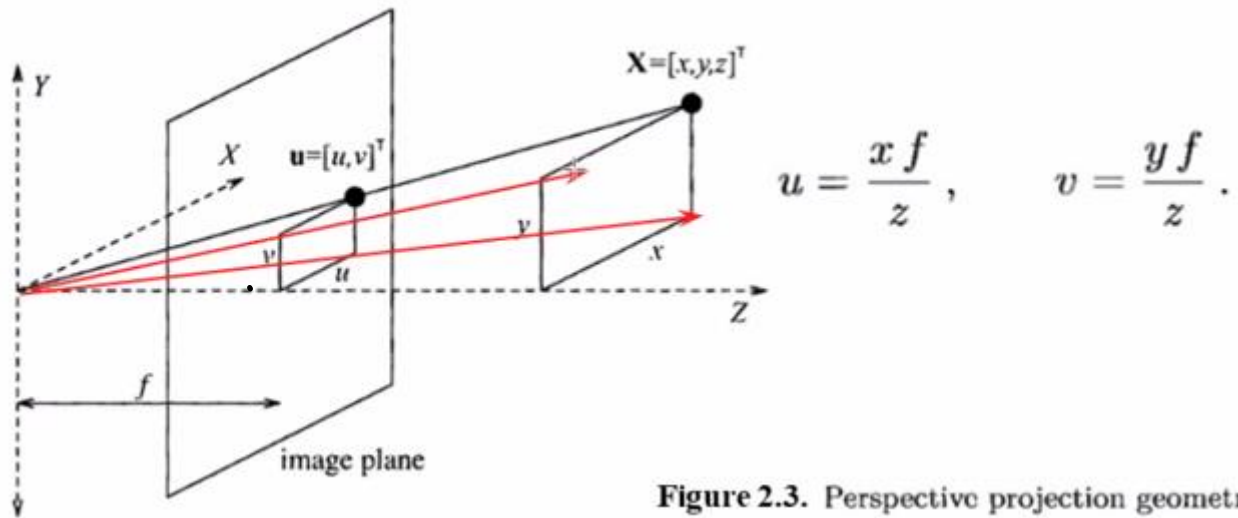


Figure 2.3. Perspective projection geometry.

6. Ví dụ về ảnh đa phổ?

Ảnh RGB với:

$$f(x, y) = \{R(x, y), G(x, y), B(x, y)\}$$

Hoặc CMY:

$$f(x, y) = \{C(x, y), M(x, y), Y(x, y)\}$$

→ Với mỗi R, G, B hoặc C, M, Y là 1 ảnh đơn phổ.

7. Viết chương trình chuyển ảnh màu RGB → Grayscale

Phương pháp:

Duyệt tất cả các pixels của ảnh và sử dụng công thức biến đổi giá trị R, G, B của pixels.

Công thức biến đổi:

$$0.299 * Red + 0.587 * Green + 0.114 * Blue$$

Thuật toán:

```
If image is empty
    Then return 1
Else {
    Traverse all the pixels of the image
    For i = 0, i < image rows, increase i by 1
        For j = 0, j < image cols, increase j by 1
            Vec3b pixel = image.at<Vec3b>(i,j)
            assign B = pixel.val[0];
            assign G = pixel.val[1];
            assign R = pixel.val[2];

            use the formular to assign back each pixel with
            new RGB value

            outputImage.at<uchar>(i,j) = 0.299*R +
            0.587*G + 0.114*B;

        return 0;
    }
```

8. Tính độ dị biệt giữa 2 màu trong không gian RGB ($d(c_i, c_j)$)

$$d(c_i, c_j) = \sqrt{(R_1 - R_2)^2 + (B_1 - B_2)^2 + (G_1 - G_2)^2}$$

9. Đưa ảnh Grayscale chuyển sang ảnh RGB.

Phương pháp: Gán các giá trị pixels R, G, B = giá trị của grayscale.

10. Viết chương trình chuyển RGB \rightarrow HSV và ngược lại

a) RGB \rightarrow HSV

Phương pháp:

Duyệt tất cả các pixels của ảnh và sử dụng công thức để tính toán giá trị H, S, V theo R, G, B cho từng pixels.

Thuật toán:

```
For the Value:
    •  $V = \max(R, G, B)$ 
For the S (saturation) value:
    • If  $V \neq 0$  then  $S = \frac{(V - \min(R, G, B))}{V}$ 
    • Else  $S = 0$ 
For the H (Hue) value:
    • If  $V = R$  then  $H = \frac{60(G - B)}{V - \min(R, G, B)}$ 
    • If  $V = G$  then  $H = 120 + \frac{60(B - R)}{V - \min(R, G, B)}$ 
    • If  $V = B$  then  $H = 240 + \frac{60(R - G)}{V - \min(R, G, B)}$ 
    • If  $H < 0$  then  $H += 360$ 
```

b) HSV \rightarrow RGB

Phương pháp: duyệt từng pixel của ảnh, Tính các giá trị C, X, m

Với:

$$m = V - C$$

$$C = V * S$$

$$X = C * (1 - |\left(\frac{H}{60 \text{ degree}}\right) \bmod 2 - 1|)$$

Tính lại các giá trị RGB mới theo các trường hợp của H như sau:

$$(R', G', B') = \begin{cases} (C, X, 0) & , 0^\circ \leq H < 60^\circ \\ (X, C, 0) & , 60^\circ \leq H < 120^\circ \\ (0, C, X) & , 120^\circ \leq H < 180^\circ \\ (0, X, C) & , 180^\circ \leq H < 240^\circ \\ (X, 0, C) & , 240^\circ \leq H < 300^\circ \\ (C, 0, X) & , 300^\circ \leq H < 360^\circ \end{cases}$$

$$(R, G, B) = ((R'+m) \times 255, (G'+m) \times 255, (B'+m) \times 255)$$

11. Tính độ dị biệt giữa 2 điểm màu trong HSV

$$d(c_i, c_j) = \sqrt{(H_1 - H_2)^2 + (S_1 - S_2)^2 + (V_1 - V_2)^2}$$

12. Thay vì xài RGB có ~16tr8 màu mà quy về 1600 màu HSV có hợp lý không? (1/1000)

Do mắt người khó nhận ra được sự khác biệt giữa các màu gần nhau, nên có thể gom nhóm các màu gần nhau lại để giảm thiểu số liệu tính toán.

Màu ở thực tế biến thiên liên tục → giá trị số mà máy mô phỏng thực tế đọc được cũng biến thiên liên tục và có hàng chục ngàn màu có giá trị gần giống nhau.

Do bản chất dữ liệu là màu sắc, nên quy về HSV là hợp lý.

13. $r = 1$, $r = 2$, $r = \infty$ cực trong công thức Minkowski

$$\begin{aligned} d_1(p, q) &= |p_1 - q_1| + |p_2 - q_2| \\ d_2(p, q) &= \sqrt{(|p_1 - q_1|^2 + |p_2 - q_2|^2)} \\ d_\infty(p, q) &= \max\{|p_1 - q_1|, |p_2 - q_2|\} \end{aligned}$$

Ghi chú: $\|x\|_2 = d_2(x, O)$

IV. TUẦN 3 (06/11/2021)

1. Màu là gì? Vân là gì? Dáng là gì?

- Màu gồm Red Green Blue trong 1 pixel.
- Vân là texture. Là tập điểm phân bố vị trí và màu khác nhau. (Haralick có 6 đặc trưng, Wavelet khảo sát vân tay, Curvelet,...)
- Dáng là shape: có từ 2 pixels trở lên.

2. Ngoài màu, vân, dáng còn đặc trưng nào nữa?

Biên cạnh, độ sáng, độ tương phản.

3. Pixel tròn hay vuông?

Trên thực tế pixel không có hình dạng nhất định. Nhưng trên máy tính, pixel lại có hình dạng cố định là hình vuông để vừa với màn hình máy tính.

4. Xấp xỉ từ 16 triệu 8 xuống còn 2 nghìn chấp nhận được không?

Do mắt người khó nhận ra được sự khác biệt giữa các màu gần nhau, nên có thể gom nhóm các màu gần nhau lại để giảm thiểu số liệu tính toán.

Màu ở thực tế biến thiên liên tục → giá trị số mà máy mô phỏng thực tế đọc được cũng biến thiên liên tục và có hàng chục ngàn màu có giá trị gần giống nhau.

Do bản chất dữ liệu là màu sắc, nên quy về HSV là hợp lý.

5. Định lượng hcolor. Chỉ số index m. Định lượng màu.

x là kích thước theo phương ngang, y là phương dọc.

6. Có thể nào nhiều ảnh khác nhau có cùng một histogram hay không?

Có, histogram chỉ là sơ đồ tần số màu, nên vẫn có thể có nhiều hình dùng chung tần số màu.

7. Xem trong color histogram và edge histogram đã bỏ qua điều gì (thiếu sót gì) mà nhiều ảnh khác nhau mà cùng đặc trưng (song ánh mà không phải đơn ánh). Vậy sao phân biệt và nhận biết.

Color: Histogram chỉ nhận về độ phổ biến từng màu, không có nhận về phân bố màu sắc trong ảnh (nghĩa là dùng chung 1 tập màu, nhưng các màu phân bố khác nhau, tạo ra nhiều vật thể khác nhau).

Edge: Histogram dùng vector tiếp tuyến với mỗi 5 độ để lưu thông tin về các điểm biên của hình, nhưng mà lại không có độ phổ biến màu, vị trí các điểm biên, vị trí màu sắc. Dẫn đến vẫn có thể có ảnh dùng chung tập vector tiếp tuyến.

8. Để ảnh bên trái sáng như ảnh bên phải? Làm cách nào? Có nhiều cách làm sáng. Làm sáng nền hoặc làm sáng chủ thể.

Tăng giá trị RGB của pixel.

9. Vì sao nhân với hệ số a ở Contrast modification lại gây ra sự tương phản.

Gọi delta là độ tương phản của 2 pixel.

Xét 2 pixel $f(x_i, y_i)$ và $f(x_j, y_j)$, ta có: $\Delta = |f(x_i, y_i) - f(x_j, y_j)|$

\Rightarrow Tăng delta lên a lần thì độ tương phản cũng tăng lên a lần. Với a thuộc $[0, +\infty)$

10. Vì sao cộng với hệ số b lại tăng hoặc giảm độ sáng.

Màu trắng: 255,255,255

Màu đen: 0,0,0

\Rightarrow Số màu càng cao thì càng sáng (tiến về trắng). Ngược lại thì càng tối (tiến về đen).

11. Có một tập ảnh đang ở mức xám $[f_1, f_2]$ chuyển thành mức $[g_1, g_2]$ bằng cách nào? Đưa ra phương pháp và công thức chuẩn hoá.

Tăng độ contract để lấy chính xác chủ thể cho nhận dạng chính xác.

Ví dụ: vật mẫu có contract là 5.

Mà mẫu để tìm kiếm trong file học có contract là 9 thì tăng cho đồng đều.

Z thì từ $g_x = f_x + b \Rightarrow b = g_x - f_x$

Contrast thì $g_x = a.f_x \Rightarrow a = g_x / f_x$

Công thức là: $g(x) = a.f(x) + b$


Nếu chỉnh độ sáng không thì $a=1 \Rightarrow g(x) = f(x) + b$

$\Rightarrow b = g(x) - f(x) \Leftrightarrow b = g(i,j) - f(i,j)$

12. Có thể dùng ánh xạ tuyến tính hoặc phi tuyến cho 3.1.3 Probability Density Function-based mapping để chỉnh sửa không?

Có

13. Tính tay cân bằng vùng sáng 3.1.3.1. Histogram Equalization. Ex: Histogram equalization of f.

 3.1.3. Probability Density Function-based mapping

3.1.3.1. Histogram Equalization

$nG=15, N.M=64$

f	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14
H	0	4	16	5	2	7	8	4	5	4	4	1	2	1	1
T	0	4	20	25	27	34	42	46	51	55	59	60	62	63	64
TR	0	1	4	5	6	7	9	10	11	12	13	13	14	14	14

$T[0] = H[0]$
 $T[p] = T[p-1] + H[p], p = 1, 2, \dots, nG-1$
 $T[p] = \text{round}((nG-1/NM)T[p])$

Associate Prof. Lý Quốc Ngọc

32

V. TUẦN 4 (13/11/2021)

1. Về lập bảng thống kê so sánh các phương pháp thay đổi, tăng cường, xử lý chất lượng ảnh.

Phương pháp	Nội dung	Nguyên nhân
Biến đổi tuyến tính từng đoạn	Xử lý những ảnh có độ tương phản thấp	_Do thiếu sáng hoặc ánh sáng không đều. _Do tính phi tuyến hoặc biến động nhỏ của bộ cảm nhận.
Tách nhiễu và lấy ngưỡng ảnh	Tách nhiễu là trường hợp riêng của giãn độ tương phản khi $\alpha = \gamma = 0$.	Giảm nhiễu khi biết tín hiệu nằm trong khoảng $[a, b]$.
	Lấy ngưỡng ảnh là trường hợp riêng của tách nhiễu khi $a = b = \theta$ là ngưỡng ảnh.	Biến đổi từ ảnh đa mức xám về ảnh nhị phân.
Biến đổi Histogram	Phân bố mức xám trong ảnh.	Đánh giá gần đúng hàm mật độ phân bố xác suất.

2. Tiêu chí thực hiện, độ phức tạp tính toán, ưu khuyết điểm.

Nhiều pixel mới map lại cùng một pixel cũ nên gây ra hiện tượng blocky (tạo ra các dạng khối, đường trên ảnh)

3. Khác biệt cơ bản của giữa biến đổi một đối tượng trong ảnh với đối tượng trong đồ họa máy tính.

Trong ĐHMT có 1 đối tượng được tạo từ tập các điểm cấu thành nên đối tượng đó và được mô tả bằng phương trình toán học. Đối tượng được mô tả bởi tổ hợp các vectors. Khi thực hiện một phép biến đổi thì sẽ tác động trực tiếp lên các điểm. Sau đó từ phương trình toán học tô lại đối tượng.

⇒ Độc lập với độ phân giải, dù phóng to thu nhỏ thì vẫn không bị ảnh hưởng.

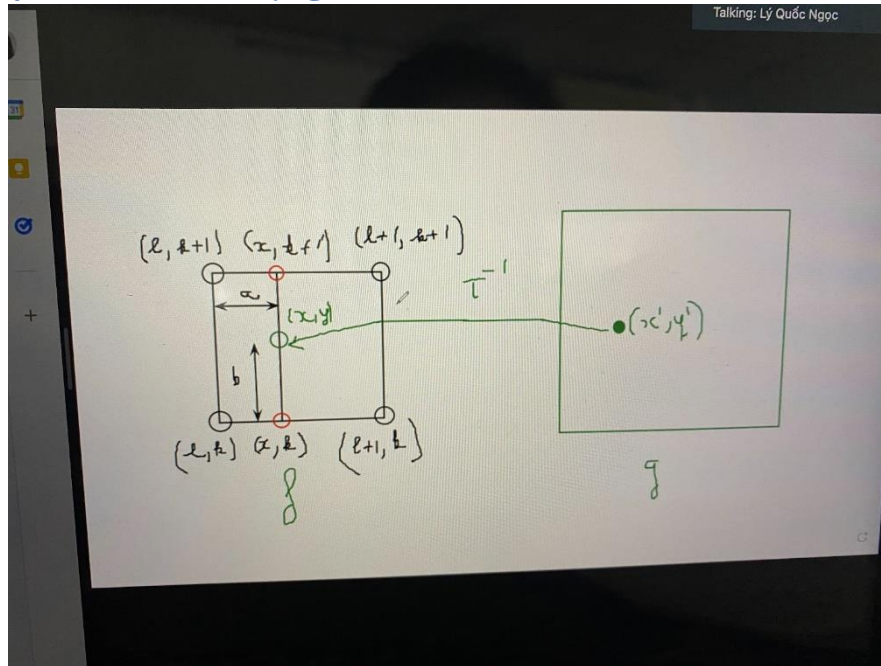
Bên ảnh muốn thay đổi đối tượng thì phải thay đổi, di chuyển từng pixel trong ảnh.

⇒ Không độc lập với độ phân giải. Khi phóng to quá cỡ sẽ ảnh hưởng chất lượng ảnh.

4. Dùng backward mapping có ưu nhược điểm gì trong việc scale ảnh?

- **Nhược:** Gây ra hiện tượng blocky, do nhiều pixel kết quả nhận cùng 1 giá trị từ pixel gốc.
- **Ưu:** Tính toán nhanh.

5. Nội suy tuyến tính. Tìm trọng số.



$$l = \text{round}(x)$$

$$k = \text{round}(y)$$

$$a = x - l$$

$$b = y - k$$

$$g(x', y') = b \cdot (a \cdot f(l+1, k+1) + (1-a)f(l, k+1)) + (1-b) \cdot (a \cdot f(l+1, k) + (1-a) \cdot f(l, k))$$

6. Lấy giá trị màu như thế nào để bỏ vào x', y' ?

Lấy từ 4 góc sử dụng trung bình cộng có trọng số. Ai gần hơn thì có tác động mạnh hơn.

7. Nếu làm trơn ảnh (lọc nhiễu) sẽ gặp những ưu và khuyết điểm gì?

Nhược: Gây ra hiện tượng blocky, do nhiều pixel kết quả nhận cùng 1 giá trị từ pixel gốc

Ưu: Tính toán nhanh

8. Tính giá trị Median

$$g(x, y) = \text{med} \{f(x+i, y+j), (i, j) \in O\}$$

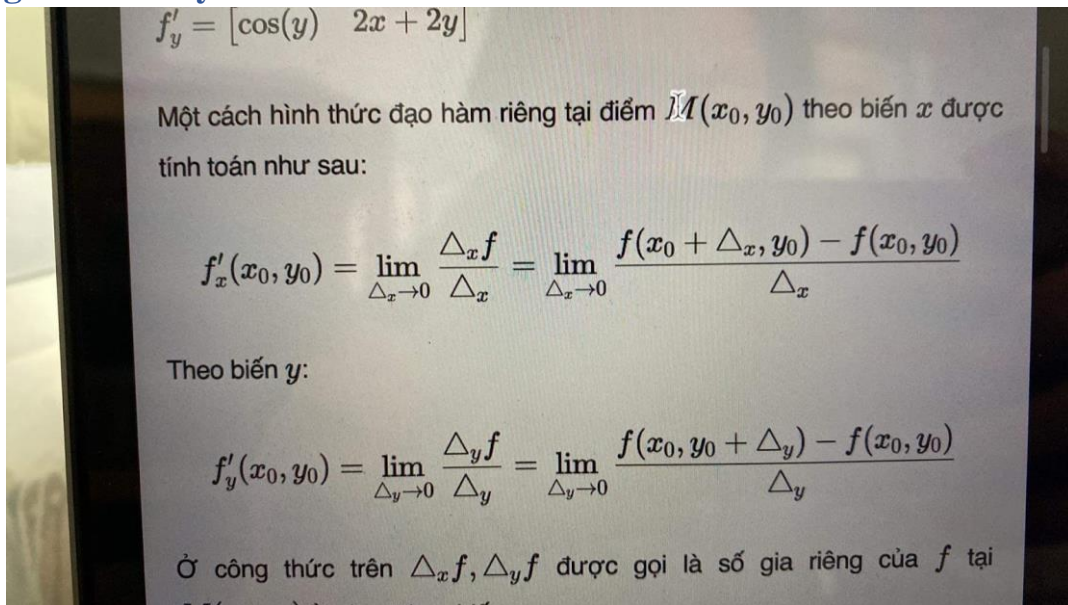
Suppose that $\{f(x+i, y+j), (i, j) \in O\}$

is sorted in ascending order and redefined as

follows: $I_1 < I_2 < \dots < I_n, n = 2\nu + 1$,

Then $\text{med}(I_i) = I_{\nu+1}$

9. Tính Gradient Operator? Xấp xỉ đạo hàm riêng của f theo x bằng với đạo hàm riêng của f theo y .



Cho $\Delta = 1$ (hoặc $h = 1$):

- $\frac{\partial f}{\partial x}(x, y) = \lim_{h \rightarrow 0} ([f(x + h) - f(x, y)]/h)$

VI. TUẦN 5 (27/11/2021)

1. Thay $f(x + L)$ vào $f(x)$ thì hàm có thay đổi không?

7.2. Fourier Theory

7.2.1. Fourier series (Jean Baptiste Joseph Fourier)

Any periodic function can be represented as a sum of these simpler sinusoids.

$$f(x) = \sum_{n=0}^{\infty} a_n \cos\left(\frac{2\pi nx}{L}\right) + b_n \sin\left(\frac{2\pi nx}{L}\right)$$

$$= a_0 + \sum_{n=1}^{\infty} a_n \cos\left(\frac{2\pi nx}{L}\right) + b_n \sin\left(\frac{2\pi nx}{L}\right)$$

Period: L , Frequency: n

$f(x+L) = f(x)$

Associate Prof. Lý Quốc Ngọc

$$f(x) = a_0 \sum_{n=1}^{\infty} a_n \cos\left(\frac{2\pi nx}{L}\right) + b_n \sin\left(\frac{2\pi nx}{L}\right)$$

$$f(x + L) = a_0 \sum_{n=1}^{\infty} a_n \cos\left(\frac{2\pi nx}{L} + 1\right) + b_n \sin\left(\frac{2\pi nx}{L} + 1\right)$$

$$= a_0 \sum_{n=1}^{\infty} a_n \cos\left(\frac{2\pi nx}{L}\right) + b_n \sin\left(\frac{2\pi nx}{L}\right)$$

→ Không thay đổi

2. Tầm quan trọng của định lý Convolution theorem (dòng đầu) của phép biến đổi Fourier trong Làm trơn ảnh và phát hiện biên cạnh.

Chuyển đổi từ việc lấy từng phần tử trong h tích với các phần tử trong f qua tính trực tiếp phép biến đổi Fourier ngược của $F.H$:

$$(f * h)(x, y) = \xi^{-1}(F(u, v) \cdot H(u, v))$$

3. Định lý Convolution điều 2 dùng ở đâu? Dùng trong việc lọc ảnh trong miền tần số.

$$\xi\{f(x, y) \cdot h(x, y)\} = (F * H)(u, v)$$

4. Độ phức tạp tính toán của Convolution Theroem: $O((N*N)*(M*M))$ nếu trong miền không gian, khi dùng Convolution Theorem bằng bao nhiêu?

$$O((N * N))$$

5. Biện luận Ideal Lowpass Filters. Nếu D_0 càng lớn/càng bé chuyện gì sẽ xảy ra?

$$H(u, v) = \begin{cases} 1 & \text{if } D(u, v) \leq D_0 \\ 0 & \text{if } D(u, v) > D_0 \end{cases}$$

$$D(u, v) = \left[\left(u - \frac{M}{2} \right)^2 + \left(v - \frac{N}{2} \right)^2 \right]^{\frac{1}{2}}$$

D_0 là tần số cắt. D_0 càng lớn thì sẽ khử nhiều kém chỉ khử về 0 với những tần số cao hơn nhưng những chỗ không bị nhiễu sẽ được bảo toàn.

D_0 càng bé thì khử nhiễu tốt nhưng những chỗ không bị nhiễu sẽ bị khử theo. Nên ảnh sẽ mờ.

6. Butterworth Lowpass Filter khi $D(u,v)$ bằng bao nhiêu thì H tiến về 0 và ngược lại khi nào H tiến về 1?

$$H(u, v) = \frac{1}{1 + \left[\frac{D(u, v)}{D_0} \right]^{2n}}$$

$$D(u, v) = \left[\left(u - \frac{M}{2} \right)^2 + \left(v - \frac{N}{2} \right)^2 \right]^{\frac{1}{2}}$$

Từ công thức $H(u,v)$, ta dễ dàng thấy: $D(u, v)$ càng cao thì $H(u,v)$ sẽ càng tiến về 0 và ngược lại $D(u,v)$ càng thấp thì $H(u,v)$ sẽ tiến về 1

7. Cho 1 hàm $f_p(t) = \begin{cases} 0 & \text{nếu } t < 0 \\ e^{-at} & \text{nếu } t \geq 0 \end{cases}$ với $a > 0$. Tính $f(t), F(\omega)$:

$$\begin{aligned} F(\omega) &= \frac{1}{2\pi} \int_{-\infty}^0 f(t) e^{-i\omega t} dt = \frac{1}{2\pi} \int_{-\infty}^0 e^{-at-i\omega t} dt = \frac{-e^{-t(a+i\omega)}}{2\pi(a+i\omega)} = \frac{1 - \lim_{t \rightarrow \infty} e^{-t(a+i\omega)}}{2\pi(a+i\omega)} \\ &= \frac{1}{2\pi(a+i\omega)} (\lim_{t \rightarrow \infty} e^{-t(a+i\omega)} = 0) \end{aligned}$$

$$f(t) = \int_{-\infty}^{+\infty} F(\omega) e^{i\omega t} d\omega$$

VII. TUẦN 6 (04/12/2021)

1. So sánh độ phức tạp tính toán giữa phương pháp lọc ảnh trong miền không gian và miền tần số.

- Trong miền không gian: $g = f * h$, nên độ phức tạp tính toán là (n^2, m^2)

$$\sum \sum f(x-i, y-j) h(i, j)$$

- Trong miền tần số:

- $\text{fourier}\{f\} = f(u, v) \rightarrow \frac{O(n^2 * n^2)}{\text{Fast fourier transform: } O(n^2 * \log n)}$
- $\text{fourier}\{h\} = h(u, v) \rightarrow O(n^2)$
- $f(u, v) * h(u, v) \rightarrow O(n^2)$
- $\text{fourier}_{-1}\{F.H\} \rightarrow \frac{O(n^2, n^2)}{\text{Fast fourier transform: } O(n^2 * \log n)}$

\Rightarrow Trong trường hợp m,n nhỏ thì miền không gian nhanh hơn vì độ phức tạp ít hơn, nhưng khi m, n lớn thì tần số có độ phức tạp nhỏ hơn.

2. Chứng minh $x = Ay$ là

1. Phát biểu bài toán

Cần tìm phép biến đổi tuyến tính để biến đổi vector đặc trưng gốc sang vector đặc trưng mới đạt:

- Giảm số chiều không gian đặc trưng nhưng vẫn giữ được các đặc trưng chính (có phương sai lớn nhất có thể).
- Cực tiểu hóa việc mất mát thông tin.

2. PP

$$y = A^T x \quad (A^T = A^T)$$

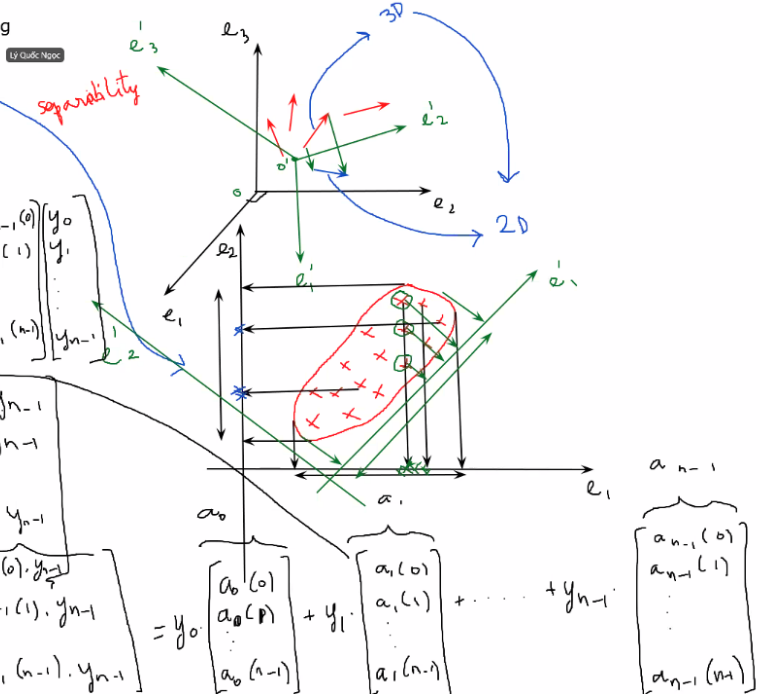
$$x = (A^T)^{-1} y = (A^{-1})^T y = Ay$$

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

$$= \begin{bmatrix} a_0(0) \cdot y_0 + a_0(1) \cdot y_1 + \dots + a_0(n-1) \cdot y_{n-1} \\ a_1(0) \cdot y_0 + a_1(1) \cdot y_1 + \dots + a_1(n-1) \cdot y_{n-1} \\ \vdots \\ a_{n-1}(0) \cdot y_0 + a_{n-1}(1) \cdot y_1 + \dots + a_{n-1}(n-1) \cdot y_{n-1} \end{bmatrix}$$

$$= \begin{bmatrix} a_0(0) \cdot y_0 \\ a_0(1) \cdot y_0 \\ \vdots \\ a_0(n-1) \cdot y_0 \end{bmatrix} + \begin{bmatrix} a_0(0) \cdot y_1 \\ a_0(1) \cdot y_1 \\ \vdots \\ a_0(n-1) \cdot y_1 \end{bmatrix} + \dots + \begin{bmatrix} a_{n-1}(0) \cdot y_{n-1} \\ a_{n-1}(1) \cdot y_{n-1} \\ \vdots \\ a_{n-1}(n-1) \cdot y_{n-1} \end{bmatrix}$$

$$= y_0 \begin{bmatrix} a_0(0) \\ a_0(1) \\ \vdots \\ a_0(n-1) \end{bmatrix} + y_1 \begin{bmatrix} a_1(0) \\ a_1(1) \\ \vdots \\ a_1(n-1) \end{bmatrix} + \dots + y_{n-1} \begin{bmatrix} a_{n-1}(0) \\ a_{n-1}(1) \\ \vdots \\ a_{n-1}(n-1) \end{bmatrix}$$



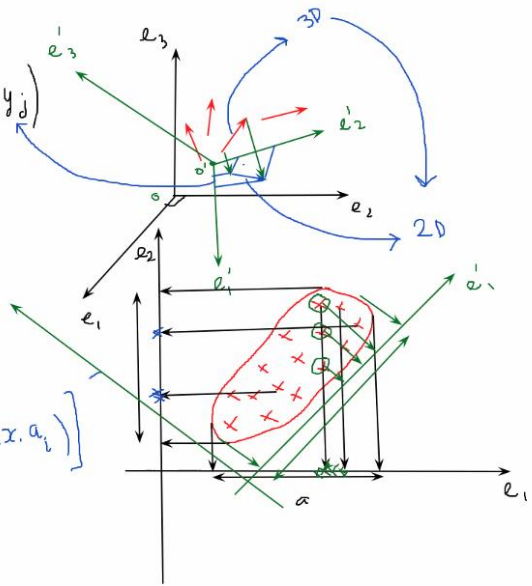
- $x = \sum_{i=0}^{n-1} y_i a_i$ a_i ?

($x \cdot a_j = (\sum_{i=0}^{n-1} y_i a_i) \cdot a_j = \sum_{i=0}^{n-1} y_i \underbrace{a_i \cdot a_j}_{\delta_{ij}} = \sum_{i=0}^{n-1} y_i \delta_{ij} = y_j$)

- χ^2
($x \approx \hat{x} = \sum_{i=0}^{m-1} y_i a_i$ ($m < n$))

- Tính sai số:
 $\|x - \hat{x}\|^2 = \left\| \sum_{i=m}^{n-1} y_i a_i \right\|^2 = \sum_{i=m}^{n-1} y_i^2$

(SS) $= E(\|x - \hat{x}\|^2) = E\left[\sum_{i=m}^{n-1} y_i^2\right] = E\left[\sum_{i=m}^{n-1} (x \cdot a_i) \cdot (x \cdot a_i)\right]$
 $= \sum_{i=m}^{n-1} a_i^T E[xx^T] \cdot a_i +$
 $\left\{ \begin{array}{l} a_i^T a_i = 1 \\ a_i^T a_j = 0 \quad i \neq j \end{array} \right.$

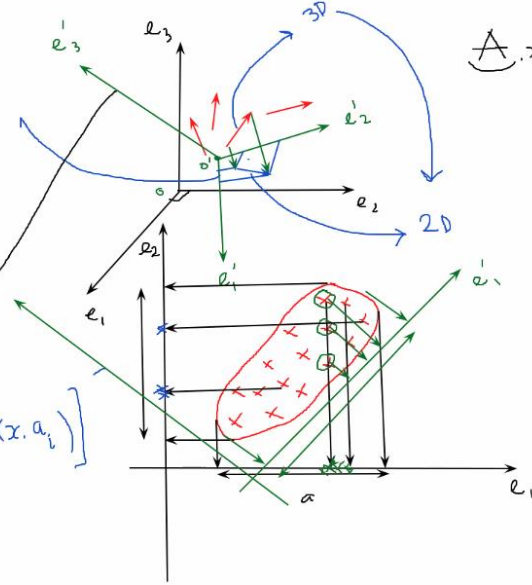


PP nhân tử Lagrange:
 $F(a_i, \lambda_i) = \sum_{i=m}^{n-1} a_i^T E[xx^T] \cdot a_i + \sum_{i=m}^{n-1} \lambda_i (1 - a_i^T a_i)$

ĐK cần:
 $\frac{\partial F}{\partial a_i} = 0 \Rightarrow a_i^T E[xx^T] - a_i^T \lambda_i = 0$
 $\frac{\partial F}{\partial \lambda_i} = 0 \Rightarrow E[xx^T] a_i = \lambda_i a_i$

$\underline{A} \cdot x = \lambda \cdot x$
 $\downarrow \quad \downarrow$
 $gh \quad vt$

(SS) $= E(\|x - \hat{x}\|^2) = E\left[\sum_{i=m}^{n-1} y_i^2\right] = E\left[\sum_{i=m}^{n-1} (x \cdot a_i) \cdot (x \cdot a_i)\right]$
 $= \sum_{i=m}^{n-1} a_i^T E[xx^T] \cdot a_i +$
 $\left\{ \begin{array}{l} a_i^T a_i = 1 \\ a_i^T a_j = 0 \quad i \neq j \end{array} \right.$



3. .

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

$$E(\|x - \hat{x}\|^2) = \sum_{i=0}^{m-1} a_i^T E[xx^T] a_i$$

$$= \sum_{i=0}^{m-1} a_i^T x_i a_i$$

$$= \sum_{i=0}^{m-1} x_i \rightarrow$$

$x_0, x_1, x_2, \dots, x_m, \dots, x_{m+1}, \dots, x_{n-1}$

$- a_i: \text{vết } E[xx^T]$
 $- \text{Giữ lại } m \text{ vết với } m \text{ lớn hơn } I$

$- \frac{T_0 T_{0C}}{E[yy^T]} = E[A^T x x^T A] = A^T E[xx^T] A, A = \Lambda = \begin{bmatrix} \lambda_0 & & \\ & \ddots & \\ & & \lambda_{m-1} \end{bmatrix}$
 $+ \sigma^2(y_i) \rightarrow \sigma^2(y_i) = E[y_i^2] = E[(a_i^T x)(x^T a_i)] = a_i^T E[xx^T] a_i = a_i^T x_i a_i$

$\vec{A} \cdot x = x \cdot \vec{A}$
 $\downarrow \quad \downarrow$
 $g^h \quad v^h$

4. Rút trích đặc trưng tập ảnh mặt người

$C_Y = A^T X^T X A = A^T C_X A = \Lambda$
 $(N \times M) \times (M \times N) \rightarrow N \times N$
 Ma trận A có các cột là các vector riêng của $C_X \rightarrow C_Y$ là ma trận chéo

Do kích thước C_X quá lớn ($N \times N$), thay vì tìm giá trị riêng của C_X , tìm $C_X^{-1} = X^T X$ ($M \ll N$)
 $M \times N \quad N \times M$
 Vector riêng C_X^{-1} là $F^T \Rightarrow F = X \cdot F^T$
 C_X là F

Chọn K vector riêng ứng với K giá trị riêng lớn nhất của C_X làm vector cơ sở để xấp xỉ X_i ($K < M$), làm cực tiểu sai số xấp xỉ:

$SS = \sum_{i=1}^K \|X_i^k - x_i\|^2 = \sum_{i=1}^K x_i^T x_i$
 $x_i^k = \sum_{l=1}^K a_{il} F_l$ là xấp xỉ của x_i , tại đây có số gồm K vector riêng $F_l, l=1, 2, \dots, K$
 $\{a_{il}, l=1, 2, \dots, K\} : \text{TPC của } x_i^k$

Giả sử tập ảnh: T_1, T_2, \dots, T_M
 Số chiều: $N \times R \times C$
 $T_i^T = [T_i(1) \ T_i(2) \ \dots \ T_i(N)]$

Xét tập ảnh độ lệch so với ảnh TB:
 $x_i = T_i - ATB$
 $ATB = \frac{1}{M} \sum_{i=1}^M T_i$

Xét phép biến đổi TT A chuyển X thành Y:
 $Y = A^T X$

Cần tìm A sao cho Y giữ được các đặc trưng chính, các đặc trưng không tương quan để loại bớt dư thừa thông tin của vector đặc trưng
 $C_Y = Y \cdot Y^T$
 $Y = \begin{bmatrix} Y_1(1) & Y_1(2) & \dots & Y_1(M) \\ Y_2(1) & Y_2(2) & \dots & Y_2(M) \\ \vdots & \vdots & \ddots & \vdots \\ Y_N(1) & Y_N(2) & \dots & Y_N(M) \end{bmatrix}$

BÀI TẬP LỚP NHÀ

bang 1.docx - Microsoft Word

Home Insert Page Layout References Mailings Review View

Print Layout Reading Layout Document Views

Full Screen Web Outline Draft Document Views

Ruler Document Map Thumbnails

Message Bar Show/Hide

Zoom 100% Page Width

New Window Arrange All Split

View Side by Side Synchronous Scrolling

Switch Windows

Macros

Who can see what you share here?

Research Search for:

All Reference Books

Type text to look up in a variety of research and reference services.

You can also press Alt F and click a word to start a search.

Get updates to your services...

Research options...

Page: 1 of 3 Words: 0 English (United States)

$\{T_i, i=1, 2, \dots, M\}, T_i \in \mathbb{R}^N$, tìm độ lệch K của \mathbb{R}^N
 được sinh bởi hệ K vector nguyên trực giao ứng với
 K giá trị $\{F_i, i=1, 2, \dots, K\}$ của $C_X = X \cdot X^T$
 T_i độ lệch X_i bởi hình chiếu của nó xuống \mathbb{R}^K
 $T_i \approx ATB + \sum_{j=1}^K a_{ij} F_j, i=1, 2, \dots, M$
 F_j : một trong các ảnh cơ sở để tái tạo lại
 tập $T_i, i=1, 2, \dots, M$
 Bộ Vector $\{a_{ij}, j=1, 2, \dots, K\}$: vector đặc trưng T_i
 $(K \ll M \ll N)$

bang 1.docx - Microsoft Word

Home Insert Page Layout References Mailings Review View

Print Layout Reading Layout Document Views

Full Screen Web Outline Draft Document Views

Ruler Document Map Thumbnails

Message Bar Show/Hide

Zoom 100% Page Width

New Window Arrange All Split

View Side by Side Synchronous Scrolling

Switch Windows

Macros

Who can see what you share here?

Research Search for:

All Reference Books

Type text to look up in a variety of research and reference services.

You can also press Alt F and click a word to start a search.

Get updates to your services...

Research options...

Page: 2 of 3 Words: 0 English (United States)

1. Tính ATB
 $ATB = \frac{1}{M} \sum_{i=1}^M T_i$
 2. Tính ảnh độ lệch
 $X_i = T_i - ATB, i=1, 2, \dots, M$
 3. Tính M giá trị riêng của $C_X = X \cdot X^T$ và giữ lại K giá trị lớn nhất
 Tính K vector nguyên ứng với K giá trị lớn nhất
 $\{F_i, i=1, 2, \dots, K\}$
 4. Tính K một trong các vectơ-nhập của $C_X = X \cdot X^T$
 $F = X \cdot F_i$
 $F_i(j) = \sum_{l=1}^N X_l(j) F_i(l), i=1, 2, \dots, K, j=1, 2, \dots, N$
 5. Chuyển tập ảnh mẫu X xuống không gian bởi K vector của C_X
 Tập ảnh chiếu: Y
 $Y = X^T \cdot X$ (A' xấp xỉ A , gồm các giá trị K lớn nhất)
 $Y_i(j) = \sum_{l=1}^N F_j(l) \cdot X_i(l), i=1, 2, \dots, M, j=1, 2, \dots, K$
 6. Xấp xỉ mỗi ảnh T_i bằng hệ của
 ảnh độ lệch xấp xỉ K và ảnh bởi K của
 của C_X :
 $T_i \approx ATB + \sum_{j=1}^K a_{ij} F_j$
 $a_{ij} = Y_i(j)$
 $i=1, 2, \dots, M$
 $j=1, 2, \dots, K$
 Vector đặc trưng của T_i : $\{a_{ij}, j=1, 2, \dots, K\}$

VIII. TUẦN 7 (11/12/2021)

1. Viết giải thuật của region growing và code

Giải thuật:

- **Bước 1:** Đọc ảnh, tìm pixel đầu tiên không thuộc về và đặt nó là (x_0, y_0)
- **Bước 2:** Lấy (x_0, Y_0) làm trung tâm, xem xét 4 pixel lân cận (x, y) của (x_0, Y_0) , nếu (x_0, Y_0) đáp ứng tiêu chí tăng trưởng, hợp nhất (x, y) và (x_0, Y_0) trong cùng một vùng, và đẩy (x, y) lên ngăn xếp.
- **Bước 3:** Lấy một pixel từ ngăn xếp và đưa nó trở lại bước 2 là (x_0, Y_0) .
- **Bước 4:** Nếu như ngăn xếp trống thì quay về Bước 1
- **Bước 5:** Lặp lại các bước trên cho tới khi mỗi điểm trong ảnh có ghi nhận.

Code:

```

import numpy as np
import cv2

class Point(object):
    def __init__(self,x,y):
        self.x = x
        self.y = y

    def getX(self):
        return self.x
    def getY(self):
        return self.y

def getGrayDiff(img,currentPoint,tmpPoint):
    return abs(int(img[currentPoint.x,currentPoint.y]) - int(img[tmpPoint.x,tmpPoint.y]))

def selectConnects(p):
    if p != 0:
        connects = [Point(-1, -1), Point(0, -1), Point(1, -1), Point(1, 0), Point(1, 1), \
                    Point(0, 1), Point(-1, 1), Point(-1, 0)]
    else:
        connects = [ Point(0, -1), Point(1, 0),Point(0, 1), Point(-1, 0)]
    return connects

def regionGrow(img,seeds,thresh,p = 1):
    height, weight = img.shape
    seedMark = np.zeros(img.shape)
    seedList = []
    for seed in seeds:
        seedList.append(seed)
    label = 1
    connects = selectConnects(p)
    while(len(seedList)>0):
        currentPoint = seedList.pop(0)

        seedMark[currentPoint.x,currentPoint.y] = label
        for i in range(8):
            tmpX = currentPoint.x + connects[i].x
            tmpY = currentPoint.y + connects[i].y
            if tmpX < 0 or tmpY < 0 or tmpX >= height or tmpY >= weight:
                continue
            grayDiff = getGrayDiff(img,currentPoint,Point(tmpX,tmpY))
            if grayDiff < thresh and seedMark[tmpX,tmpY] == 0:
                seedMark[tmpX,tmpY] = label
                seedList.append(Point(tmpX,tmpY))
    return seedMark

img = cv2.imread('lean.png',0)
seeds = [Point(10,10),Point(82,150),Point(20,300)]
binaryImg = regionGrow(img,seeds,10)
cv2.imshow(' ',binaryImg)
cv2.waitKey(0)

```

2. Lấy ngẫu nhiên w_i . Đọc K-mean++ để xem khác biệt lấy dữ liệu như thế nào?

Sự khác nhau giữa K-mean và K-Mean ++: Sự khác biệt là ở việc Lựa chọn các trung tâm xung quanh các cụm có tham gia. K-means ++ loại bỏ nhược điểm của K, có nghĩa là nó sẽ phụ thuộc vào việc khởi tạo tâm.

Gán (chọn) $w_j = i_l$ như thế nào? i_l lấy từ đâu ra?

Điều kiện dừng của vòng lặp: Until($(w_{1m} = w_{1m-1}) \& (w_{2m} = w_{2m-1}) \dots (w_{km} = w_{km-1})$)

Điều kiện kết nạp i_l với C_j^* :

```

1. for l in range(1,n){
2.     minDistance= distance(i[l],w[i])
3.     minIndex=1
4.     for j* in range(1,k){
5.         if(distance(i[l],w[j*]<minDistance)){
6.             minDistance=distance(i[l],w[j*]);
7.             minIndex=j*;
8.         }
9.     }
10.    C[minIndex].add(i[l])

```

IX. TUẦN 8 (18/12/2021)

TỔNG HỢP SEMINAR CÁC NHÓM

1. Nhóm Depression - Phân đoạn ảnh dựa trên superpixel

- Phân đoạn ảnh là quá trình chia một bức ảnh số thành nhiều phần khác nhau (các tập điểm ảnh) có sự đồng nhất về thuộc tính chung
- Superpixel được xem là một nhóm nhỏ các pixel với màu sắc đồng nhất và có chung ý nghĩa, màu, vân, độ sâu, thường được áp dụng cho phân đoạn ảnh hay nhận dạng vật thể
- Phân đoạn ảnh dựa trên superpixel là phương pháp chia ảnh thành các nhóm pixel hay còn gọi là superpixel theo nguyên lý chia để trị

a. Động lực nghiên cứu

- Về khoa học:
- Giảm độ phức tạp tính toán vì chỉ cần quan tâm đến superpixel
- Mở ra khả năng xử lý ảnh với mức megapixel - ảnh chứa hàng triệu pixel
- Tăng độ chính xác của kết quả
- Ứng dụng:

Phát hiện vật thể

- Chẩn đoán bệnh, xác định khối u trên ảnh chụp X-Quang, ảnh nội soi
- Giúp nhận dạng đối tượng, giám sát giao thông qua hình ảnh vệ tinh và camera an ninh

b. Phương pháp

Phương pháp	Group
SLIC	Gradient-based
SEEDS	Gradient-based
FH (Felzenszwalb Huttenlocher)	Graph-based
CRS (Contour Relaxed Superpixels)	Gradient-based
CS / CIS	Gradient-based

- Phương pháp đề xuất là SLIC và FH

2. Nhóm I3C - Phân đoạn ảnh dựa trên mặt nạ R-CNN

a. Động lực nghiên cứu

- Trong khoa học thường được dùng để phân loại, phát hiện đối tượng một cách linh hoạt
- Ứng dụng: dùng trong an ninh dùng để đếm và nhận dạng con người và hành vi của họ, bắt lỗi, quản lý giao thông, xe tự hành, giúp đỡ trong y tế,...

b. Phương pháp

R-CNN, FCIS (Fully Convolutional Instance-aware semantic segmentation), MNC (Multi-task Network Cascade)

- Phương pháp đề xuất: Dùng mặt nạ CNN (Convolutional Neural Network)

3. Nhóm Siu Nhon - YOLO

a. Động lực nghiên cứu

- Các mô hình nhận diện vật thể cần thực hiện 2 tác vụ chính đó là xác định vị trí của vật thể và dán nhãn cho vật thể đó. Gồm 2 nhóm mô hình:
- **Two stage detector (RCNN)** các mô hình này sẽ phân tách 2 tác vụ trên thành 2 giai đoạn riêng biệt. Đầu tiên là dùng một thuật toán có sẵn để xác định vật thể (vẽ bounding box cho vật thể) ví dụ như sử dụng thuật toán đề xuất khu vực Region Proposal hoặc mạng đề xuất khu vực RPN. Sau đó sử dụng một thuật toán khác để xác định nhãn của vật thể đã tìm được ở bước đầu tiên
- **One stage detector (YOLO)** các mô hình này sẽ gộp 2 tác vụ trên để xử lý đồng thời trên một mạng học sâu được huấn luyện end-to-end tức vừa xác định vật thể, vừa xác định nhãn của vật thể đó. Do 2 tác vụ được xử lý đồng thời nên tốc độ của mô hình này nhanh hơn mô hình Two stage detector

Sự phát triển của mạng học sâu CNN được áp dụng vào lĩnh vực xử lý hình ảnh. Tuy nhiên thuật toán nhóm R-CNN trước đó lại chưa áp dụng triệt để mạng CNN trong xử lý hình ảnh khi phân tách 2 tác vụ là định vị vật thể và dán nhãn vật thể thành hai tác vụ riêng biệt dẫn đến tốc độ xử lý không thể đáp ứng tốc độ của các máy quay phim hiện tại để nhận diện vật thể theo thời gian thực

- Ứng dụng: Xây dựng hệ thống nhận diện vật thể theo thời gian thực với độ chính xác cao, ví dụ như:

Xe tự hành: xe tự hành cần một hệ thống có độ phản hồi rất nhanh để xử lý được các tình huống khẩn cấp nhằm đảm bảo an toàn cho người dùng

Giải trí: các trò chơi thực tế ảo tăng cường (AR) cần sự phân tích môi trường, nhận diện vật thể nhanh để có thể đưa ra những phản hồi nhanh nhất cho người dùng khi chơi game

An ninh: cần hệ thống nhận diện vật thể nhanh và chính xác để đưa ra những cảnh báo cho người điều khiển để đảm bảo an ninh

b. Phương pháp

- Mô hình huấn luyện YOLO

4. Nhóm 404-noname - Ghép ảnh - Image Stitching

- Ghép ảnh là quá trình kết hợp nhiều ảnh chụp với các trường xem chồng lên nhau để tạo ra ảnh toàn cảnh được phân đoạn hoặc ảnh có độ phân giải cao.

a. Động lực nghiên cứu

- Ứng dụng trong:
- Ảnh toàn cảnh trong bản đồ và hình ảnh vệ tinh với độ phân giải cao
- Hình ảnh dùng trong y tế
- UAV viễn thám
- Thực tế ảo (VR)

b. Phương pháp

- Reference:
- Application of Migration Image Registration Algorithm Based on Improved SURF in Remote Sensing Image Mosaic
- Image Mosaic Algorithm Based on PCA-ORB Feature Matching
- SIFT Feature Image Stitching Based on Improved Cuckoo Algorithm
- Study and Performance Evaluation Binary Robust Invariant Scalable Keypoints (BRISK) for Underwater Image Stitching
- Automatic Stitching of Medical Images Using Feature Based Approach
- Phương pháp sử dụng: Image stitching model based on features based techniques (Mô hình ghép ảnh dựa trên kỹ thuật features based)

5. Nhóm TVT - Ứng dụng Seamless cloning để sao chép đối tượng trông tự nhiên hơn

- a. Động lực nghiên cứu
 - Về khoa học:
 - Chép một đối tượng trong ảnh sang một vị trí khác hoặc một ảnh khác.
 - Giữ cho phân bố màu chuyển tiếp tự nhiên ở phần đối tượng và biên cạnh.
 - Làm cho vùng biên không còn phân cách nữa, và có sự hoà trộn giữa đối tượng sao chép và hình nền.
 - Ngoài ra, mục đích của việc này là nghiên cứu về “sự giả mạo” để tìm ra phương pháp nhận biết giả mạo, chống các hành vi xấu (forensics)
 - Ứng dụng:
 - Nhân bản liên mạch
 - Che dấu đối tượng
 - Đổi kết cấu của đối tượng
 - Ghép hình
- b. Phương pháp
 - Chỉnh sửa ảnh bằng cách giải phương trình Poisson (dựa vào lý thuyết chỉnh sửa Poisson và sự biến thiên gradient của đối tượng và ảnh nền)

6. Nhóm Noodles - Ứng dụng Inpainting để sao chép đối tượng trông tự nhiên hơn

- Inpainting là một quá trình bảo tồn trong đó các phần bị hư hỏng, xuống cấp hoặc bị thiếu của các tác phẩm nghệ thuật để tạo ra một hình ảnh hoàn chỉnh.
- Quy trình inpainting có thể được áp dụng cho cả phương diện về nghệ thuật và kỹ thuật số như tranh sơn dầu hoặc acrylic, bản in ảnh hóa học, tác phẩm điêu khắc hoặc hình ảnh và video kỹ thuật số.
- a. Động lực nghiên cứu
 - Về khoa học:
 - Xử lý các pixel bị thiếu một cách hiệu quả hơn
 - Xử lý các khu vực bị thiếu bất thường

- Khôi phục các vùng hư hỏng, xuống cấp (thay đổi màu sắc,...) hoặc bị thiếu pixel của ảnh trông tự nhiên hơn bằng cách ước lượng giá trị từ thông tin màu sắc được cung cấp
- Ứng dụng:

Khảo cổ: Khôi phục tài liệu bị ố vàng, rách nát

Xử lý ảnh: Khôi phục ảnh cũ

b. Phương pháp

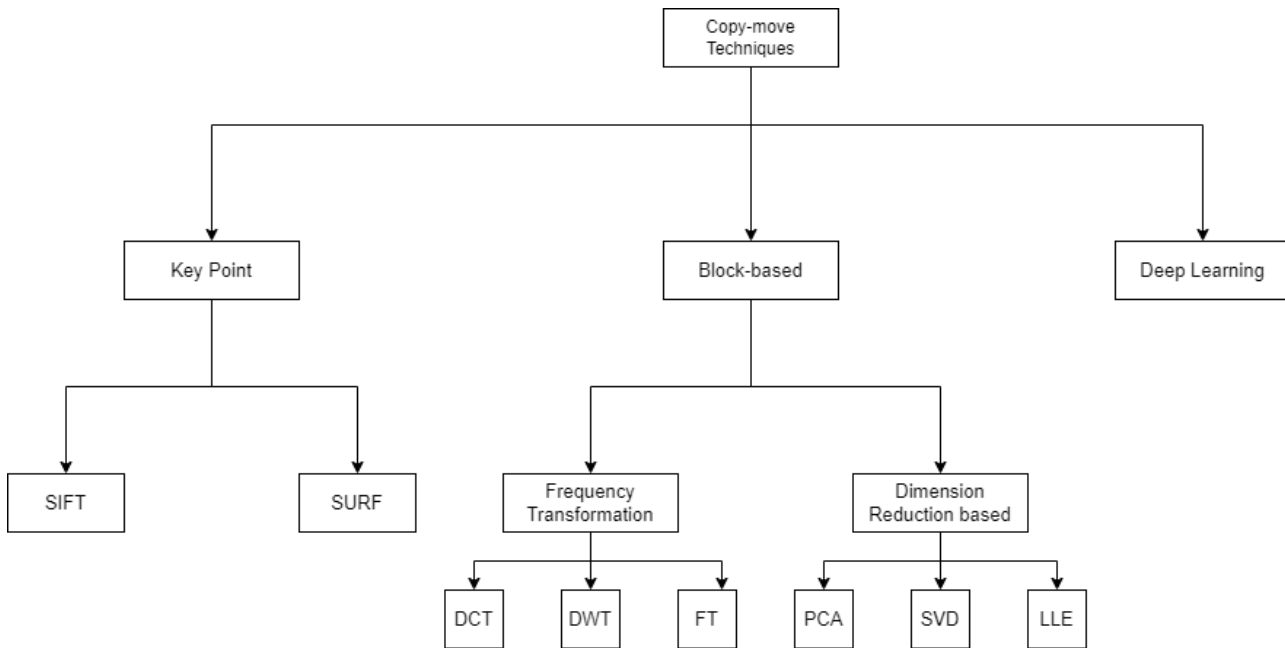
- Thuật toán Hole Filling SWA (Spiral weighted average algorithm)
- Giải thuật Gradient
- Thuật toán Fast Digital Image Processing
- Image Inpainting with Deep Learning

7. Nhóm Triocate - Phát hiện ảnh giả mạo (forensics)

a. Động lực nghiên cứu

- Về khoa học: Nghiên cứu sử dụng các thuật toán trên pixel giúp làm tiền đề nhận biết, giải mã và nhận dạng các loại ảnh giả mạo.
- Ứng dụng:
- Tìm các tội phạm pháp luật
- Nhận dạng tài liệu tài liệu điện tử cần thông qua kiểm định là thật
- Xác nhận ảnh chứng minh kinh nghiệm làm việc
- Nhận biết tài liệu chứng từ (CMND, sổ đỏ) không phải là giả
- Qua nhận diện ảnh giả mạo, bảo vệ quyền lợi của người dân

b. Phương pháp



- Phương pháp đề xuất là DCT (Discrete Cosine Transform – Phép biến đổi Cosin rời rạc)

8. Nhóm Nhóm 1 - Siêu phân giải (Super Resolution)

a. Động lực nghiên cứu

- Về khoa học:
- Là một trong những lĩnh vực nghiên cứu mới nổi ổn định nhất trong lĩnh vực xử lý hình ảnh. (Ruikar, 2020)
- Hình ảnh có độ phân giải cao cung cấp các chi tiết được tái tạo một cách cải thiện hơn của các cảnh và các đối tượng cấu thành, rất quan trọng đối với nhiều lĩnh vực nghiên cứu khác nhau như thị giác máy tính, đồ họa máy tính, v.v.
- Ứng dụng:

Với sự phát triển của các ứng dụng xử lý hình ảnh, nhu cầu lớn về hình ảnh có độ phân giải cao vì hình ảnh High Resolution không chỉ cung cấp cho người xem một hình ảnh đẹp mắt mà còn cung cấp các chi tiết bổ sung quan trọng cho việc phân tích trong nhiều ứng dụng. Các trường yêu cầu độ phân giải cao hơn; hình ảnh / video y tế, hệ thống nhận dạng sinh trắc học, viễn thám, theo dõi và nhận dạng đối tượng, giám sát an ninh, truyền thông phương tiện, giám sát và phân tích video, nhận dạng mẫu trong thị giác máy tính, v.v. (Ruikar, 2020)

b. Phương pháp

- iSeeBetter sử dụng:
- Mean Squared Error - Sai số toán phương trung bình
- Peak Signal-To-Noise Ratio - Tỷ số tín hiệu cực đại trên nhiễu
- Structural Similarity Index Measure - Chỉ mục cấu trúc tương tự

9. Nhóm Central Circus (Không thực hiện đồ án)

10. Nhóm 2 Dose Vaccine - Phục hồi ảnh (restoration)

- Image restoration (Phục hồi ảnh): là sự thay đổi mật độ pixel trên mỗi inch ảnh (PPI). Hồi phục ảnh sẽ làm tăng mật độ điểm ảnh trên mỗi inch ảnh, làm cho hình ảnh chất lượng cao hơn, sắc nét hơn.
- **Động lực khoa học:**
 - Tái tạo thông tin ảnh, giúp cho những tầng sau cải tiến về chất lượng thông tin cũng như số lượng thông tin trong một ảnh. Thông tin trong ảnh đóng vai trò quan trọng trong việc sử dụng ảnh ở những tầng trên, nếu thông tin càng chất lượng, càng nhiều thì những tầng trên sẽ càng dễ xử lý thông tin.
 - Lấy lại, dự đoán thông tin ảnh bị mờ, bị nhiễu, giúp cho thông tin trong ảnh phần nào được rõ ràng hơn, dễ cho việc nghiên cứu sau này. Làm nền tảng cho việc làm rõ, làm nét lại ảnh của những nghiên cứu sau này, làm nền tảng cho nghiên cứu làm rõ ảnh bị rung, bị lắc.
 - Làm nền tảng cho nghiên cứu bắt ảnh khoảng khắc nhanh, ở tốc độ cao, giúp cho việc lưu lại thông tin thông qua hình ảnh tốt hơn và nhanh hơn.
- **Động lực ứng dụng:**
 - Ứng dụng làm rõ ảnh trong giao thông, truy lùng tội phạm, giúp nhận rõ khuôn mặt người vi phạm, người phạm pháp, góp phần đảm bảo an toàn xã hội và con người.
 - Ứng dụng trong việc chụp ảnh ở tốc độ cao, ảnh bị rung, làm rõ ảnh trong hệ thống cảm biến sensor trong điện thoại và máy ảnh.
 - Ứng dụng trong việc đo độ loạn của mắt con người, giúp cho giới hạn về thị giác của những người bị loạn xa hơn, tốt hơn.
 - Ứng dụng trong việc xác định vị trí, làm rõ ảnh từ trên cao thông qua vệ tinh, giúp cho việc giám sát từ trên cao xuống trở nên rõ ràng hơn.
- **Phương pháp:**
 - Median filter
 - Inverse filter
 - Wiener filter
- Phương pháp đề xuất wiener filter.

11. Nhóm Noob - Ứng dụng saliency map trong phát hiện vùng quan tâm trong ảnh

- **Động lực khoa học:**

- Nhận biết đối tượng là một trong những giai đoạn đầu tiên trước khi bước vào giai đoạn xử lý ảnh. Để có thể nhận biết được đối tượng trong ảnh thì trước tiên phải xác định được đối tượng.
- Bản chất của việc xác định đối tượng trong ảnh là chúng ta bóc tách đối tượng ra khỏi nền của ảnh trước khi thực hiện nhận biết đối tượng. Vấn đề đặt ra là làm thế nào để hệ thống thị giác máy tính có thể xác định được vùng nổi bật của ảnh (Saliency region hay Saliency map) từ một ảnh nền chưa biết.
- Như vậy nhận biết và xác định các Saliency map là bước đầu tiên nhằm mục đích nhận biết đâu là thành phần quan trọng của ảnh, điểm nổi bật của ảnh để tiện cho các xử lý phức tạp sau này. Đặc biệt, đối với những bài toán xử lý để nâng cao chất lượng ảnh thì việc sử dụng phương pháp saliency maps là điều tất yếu, hỗ trợ tìm kiếm được phần quan trọng và loại bỏ những yếu tố dư thừa của ảnh đó.
- **Động lực ứng dụng:**
 - Phát hiện và nhận dạng đối tượng: Thay vì áp dụng thuật toán phức tạp về tính toán cho toàn bộ ảnh, có thể sử dụng thuật toán chỉ các vùng nổi bật nhất của hình ảnh nơi có nhiều khả năng chứa một đối tượng nhất.
 - Image retargeting (làm hình ảnh có khả năng hiển thị trên các màn hình có các kích thước khác nhau): dựa trên các thông tin thị giác sẵn có của Saliency Map để ước tính chính xác tất cả các chi tiết ảnh có tính nổi bật.
 - Nén hình ảnh và video Đánh giá chất lượng video Cải thiện chất lượng ảnh

12.Nhóm KFC - Phát hiện điểm neo trên khuôn mặt

- **Động lực khoa học:**
 - Các thuật toán phát hiện điểm neo trên khuôn mặt đã và đang góp phần vào các công trình nghiên cứu khác trong lĩnh vực nhận diện khuôn mặt và phát hiện khuôn mặt
 - Góp phần khảo sát và nghiên cứu về học sâu.
 - Tiền đề để tái tạo khuôn mặt con người trong môi trường máy tính
- **Động lực ứng dụng:**
 - Ứng dụng vào các thuật toán nhận diện khuôn mặt, từ đó sẽ có nhiều ứng dụng có ích khác như trong bảo mật có những phương pháp mã khóa bằng khuôn mặt 2D và 3D trên các thiết bị smartphone hay các hệ thống bảo mật, chấm công...
 - Chỉnh sửa ảnh chân dung.
 - Phát hiện và nhận diện cảm xúc con người.
 - Phát hiện cử chỉ khuôn mặt, làm tiền đề cho các ứng dụng điều khiển bằng cử chỉ.
 - Trong nhiếp ảnh có ứng dụng giúp bắt nét vào mặt hoặc mắt của chủ thể.
 - Ứng dụng trong các thiết bị follow tự động, tracking chủ thể tự động trong các thiết bị drone và các thiết bị hay không người lái
- **Phương pháp đề xuất: sử dụng CNN thư viện MTCNN**
 - CNN áp dụng những bộ lọc kernel vào những tấm ảnh trước khi huấn luyện nhận diện. Những kernel và trọng số sẽ được máy tính tự thiết lập và điều chỉnh qua mỗi lần học. Khi cho mỗi tấm ảnh qua những bộ lọc, chúng ta sẽ nhận được những giá trị đầu ra. Những giá trị này tiếp tục đi qua nhiều lớp nữa, cuối cùng sẽ giúp mạng neuron đưa ra các dự đoán dán nhãn cho các ảnh. Máy ảnh sẽ so sánh các nhãn này với các nhãn đúng trong tập huấn luyện.

13.Nhóm Nhóm 3 - Xác định điểm neo trên cơ thể

- **Động lực khoa học**
 - HPE là một công nghệ dựa trên thị giác máy tính để xác định và phân tích để hiểu thông tin hình học và chuyển động trên các khớp nối của cơ thể con người
 - HPE về cơ bản là một cách để nắm bắt một tập hợp các tọa độ cho mỗi khớp, được gọi là điểm chính, có thể mô tả tư thế của một người. Các mối kết nối giữa các điểm này là một cặp kết nối, không phải tất cả các điểm đều có thể tạo thành một cặp.
 - Mục đích của HPe là tạo dựng lại bộ xương người và được xử lý cho các ứng dụng cụ thể
- **Động lực ứng dụng:**
 - Theo dõi sự thay đổi trong tư thế của một người trong một khoảng thời gian cũng có thể được sử dụng để nhận dạng hoạt động, cử chỉ và dáng đi. Có một số trường hợp sử dụng cho cùng một trường hợp, bao gồm:
 - 1Chụp chuyển động (Motion Capture) và Thực tế tăng cường (Augmented Reality)
 - Ứng dụng công nghệ mô phỏng hình ảnh bằng máy tính
 - Huấn luyện Robot
 - Thay vì lập trình thủ công cho rô-bốt đi theo quỹ đạo (follow trajectories), rô-bốt có thể được chế tạo để tự đi theo quỹ đạo của điểm neo của bộ xương người đang thực hiện một hành động.
 - 1Theo dõi chuyển động (Motion Tracking) cho bảng điều khiển (consoles)
 - Ứng dụng để theo dõi chuyển động của đối tượng con người để chơi trò chơi tương tác (interactive gaming).
- **Phương pháp đề xuất: Hrnet**

14.Nhóm DIP - Xác định điểm neo trên bàn tay

- **Động lực khoa học:**
 - Ý nghĩa khoa học: có thể phát hiện bàn tay con người và cùng với các điểm neo vào các vị trí thích hợp trong môi trường có nhiều cao
 - qua các bài toán phát hiện đối tượng, ta thấy bàn tay không chỉ là một vật linh hoạt mà với từng tư thế khác nhau, chúng ta có thể tạo ra và tiếp nhận nhiều thông tin với nó. Và bàn tay là một vật thể xuất hiện ở khắp mọi nơi. Vì vậy việc phát hiện tốt tay, cũng như gắn các điểm neo tốt sẽ giúp chúng ta có cũng như tạo được nhiều thông tin hữu ích.
- **Động lực ứng dụng:**
 - Ứng dụng thực tế: Nhận diện thủ ngữ, chụp ảnh bằng tay (samsung), Điều khiển drone bằng tay, điều khiển máy tính bằng tay (human-computer interaction), ... +
 - Triển vọng: hiện nay mọi thứ dần trở nên ảo hóa, việc nhận diện bàn tay có thể giúp ta tạo ra bàn tay ảo trong môi trường 3D có rất ích rất nhiều trong các công nghệ thực tế ảo.
- **Phương pháp đề xuất: PoseNet**
 - Xử dụng hitmap và bộ dữ liệu Rendered Handpose Dataset chứa 41258 hình ảnh huấn luyện và 2728 hình ảnh ví dụ. Sử dụng segmentation masks để tách hình ảnh bàn tay ra riêng và 21 điểm neo trên bàn tay, từ những vị trí này sử dụng làm Gaussian peak để chiếu thành một hitmap

15.Nhóm VPN (Không thực hiện đồ án)

16.Nhóm 3 Bích - Truy vấn người (Person re-identification)

- **Động lực về mặt khoa học**
 - Là xác định xem một hoặc nhiều người đã xuất hiện ở một nơi nào đó vào một thời điểm khác nhau hay chưa. Những người này có thể được theo dõi bởi 1 hoặc nhiều camera khác nhau hoặc 1 camera ở những thời điểm khác nhau.
- **Động lực ứng dụng:**
 - Ứng dụng trong giám sát người , truy vấn người
- **Phương pháp đề xuất**
 - Pyramid Spatial-dTemporal Aggregation for Video-based Person Re-Identification:

17.Nhóm TMT (Không thực hiện đồ án)