

DIGITAL IMAGE AND VIDEO PROCESSING



Lưu Thị Hồng Ngọc

Hồ Thiên Phước

Table of Contents

Tuần 1 (9/10/2021).....	2
1. Thông minh là gì? Hiểu như thế nào về câu nói của Stephen Hawking.....	2
2. Cho ví dụ hệ thống thị giác thông minh? cho biết output ở 5 cấp độ.	2
3. Kỹ thuật tương tác video có thể đc áp dụng vào đâu? (trả về output ở tầng nào?).....	2
4. Trong ảnh chứa gì? Xử lý ảnh là xử lý gì trong đó?	3
5-6-7. Cho biết tác vụ với mức độ thông minh của hệ thống trong video? Dùng kiến thức nào của 3 môn? (3 video) (Dùng kiến thức thị giác máy tính hay đồ hoạ máy tính hay xử lí ảnh?)	5
REFERENCES	5
Tuần 2 (16/10/2021).....	5
1. Ý nghĩa của xe tự hành? Ban đầu làm ra xe để lái, sau lại không muốn lái nữa mà đề xuất ra xe tự hành không người lái? Như vậy, logic có tự nhiên, bình thường hay không, có hợp logic không?	5
2. Lập bảng so sánh dữ liệu ảnh/video 2D như hiện nay với dữ liệu ảnh/video 3D? Cái nào tốt hơn? Ưu, nhược điểm? (Hint: ảnh RGB 2D và kênh độ sâu cho bit vị trí mỗi điểm ảnh cho đến vị trí của , lidar cho tập điểm tọa độ)	6
3. Nhìn màu trên màn hình máy tính, mỗi điểm ảnh, mắt ta tiếp nhận thông tin RGB khác nhau, vậy còn trên tờ giấy, làm thế nào chúng ta có thể nhìn ra màu?.....	7
4. Viết chương trình chuyển ảnh màu RGB sang Grayscale (công thức)?	7
Phương pháp Weighted hoặc phương pháp luminosity	7
5. Tính độ dị biệt giữa 2 màu trong không gian RGB $d(c_i, c_j)$, tính bằng gì? Đưa ảnh xám Grayscale sang ảnh màu RGB?	8
6. Đổi ảnh màu mô hình RGB sang HSV và ngược lại? Tính độ dị biệt màu giữa 2 điểm ảnh trong mô hình HSV?	9
7. Xét công thức Minkowski	10
REFERENCES	11
Tuần 3 (6/11/2021).....	11
Tuần 4 (13/11/2021).....	14
Tuần 5 Giải đáp thắc mắc về Project (không có BTLN) (20/11/2021)	17
Tuần 6 (27/11/2021).....	17
Tuần 7 Buổi giảng bài tập trên lớp (không có BTLN) (4/12/2021)	19
Tuần 8 (11/12/2021).....	19
Tuần 9 SEMINAR (18/12/2021).....	24

Tuần 1 (9/10/2021)

1. Thông minh là gì? Hiểu như thế nào về câu nói của Stephen Hawking...

Thông minh là liên quan đến cách bạn đạt được kiến thức và kỹ năng, cũng như khả năng tương tác với thế giới xung quanh

Thông minh thường gắn liền với khả năng hoặc chỉ số IQ bẩm sinh.

"Thông minh là khả năng thích ứng với sự thay đổi."

Thông minh là mong muốn học hỏi và làm nhiều hơn, tốt hơn, hữu ích hơn, nhanh hơn.

Thông minh là luôn luôn tiến bộ.

Khả năng thích ứng là một trong những thước đo quan trọng của thông minh. Cuộc sống là hàng loạt các thay đổi, có hoặc không có sự tác động của bản thân mình. Nếu muốn mọi thứ diễn ra theo cách mình muốn, bản thân phải có tác động đủ mạnh để điều chỉnh hướng của sự thay đổi. Thông minh cho phép bản thân chú ý đến mọi thứ xung quanh và nắm bắt được những cơ hội, vật chất, tinh thần... bản thân cần để tồn tại.

2. Cho ví dụ hệ thống thị giác thông minh? cho biết output ở 5 cấp độ.

Camera tích hợp AI trên iPhone 13

Task: Quay video xóa phông tự động

Data: Những khung hình

Information: Đối tượng trong khung hình

Knowledge: Xác định đối tượng cần xóa trong số nhiều đối tượng trong ảnh

Intelligence: Phân biệt đâu là chủ thể đâu là background vô cùng chính xác VD: tóc của phụ nữ

Wisdom: Tự chuyển đối tượng cần xóa phông khi chủ thể hiện tại có xu hướng biến mất khỏi khung hình, xóa phông ngay cả khi đối tượng di chuyển tốc độ tương đối nhanh

3. Kỹ thuật tương tác video có thể đc áp dụng vào đâu? (trả về output ở tầng nào?)

Kỹ thuật này áp dụng từ lâu vào phẫu thuật thẩm mỹ, nhằm mô phỏng kết quả sau khi phẫu thuật ngay lập tức cho khách hàng.

Output: Information

4. Trong ảnh chứa gì? Xử lý ảnh là xử lý gì trong đó?

Ảnh gồm hai thuộc tính:

1. Đặc tính về thị giác (Visual level) (Visual Information): Color, Texture, Shape
2. Đặc tính ngữ nghĩa (Semantic level)

-Object:

Specific Object: xe hơi Innova | Mercedes

General Object: xe hơi

Abstract Object: phương tiện giao thông

-Scene:

Specific Scene: Vịnh Cam ranh | San Francisco

General Scene: vịnh

Abstract Scene: Cảnh thiên nhiên

-Event:

Specific Event: Cảnh ghi bàn trong trận chung kết World Cup 2020

General Event: Cảnh ghi bàn trong bóng đá

Abstract Event: Cảnh quan trọng trong thể thao

Xử lý ảnh qua các bước:

Thu nhận, ảnh có thể nhiễu, độ tương phản thấp nên cần xử lý ảnh để nâng cao chất lượng.

Chức năng chính của bộ xử lý ảnh là lọc nhiễu, nâng độ tương phản để ảnh rõ nét hơn.

Phân đoạn (Segmentation) hay phân vùng ảnh là tách một ảnh đầu vào thành các vùng thành phần để biểu diễn phân tích, nhận dạng ảnh. Ví dụ: để nhận dạng chữ (hoặc mã vạch)

trên phong bì thư cho mục đích phân loại bưu phẩm, cần chia các câu, chữ về địa chỉ hoặc tên người thành chữ, số (hoặc vạch) riêng biệt. Đây là phần phức tạp khó khăn nhất trong xử lý ảnh và dễ gây lỗi, làm mất độ chính xác của ảnh.

Biểu diễn ảnh (Image Representation) Đầu ra ảnh sau phân đoạn chứa các điểm ảnh của vùng ảnh (ảnh đã phân đoạn) cộng với mã liên kết với các vùng lân cận. Việc biến đổi các số liệu này thành dạng thích hợp là cần thiết cho xử lý tiếp theo bằng máy tính. Việc chọn các tính chất để thể hiện ảnh gọi là trích chọn đặc trưng (Feature Selection) gắn với việc tách các đặc tính của ảnh dưới dạng các thông tin định lượng hoặc làm cơ sở để phân biệt lớp đối tượng này với đối tượng khác trong phạm vi ảnh nhận được. Ví dụ: trong nhận dạng ký tự trên phong bì thư, phân biệt ký tự này với ký tự khác.

Nhận dạng và nội suy ảnh (Image Recognition and Interpretation). Nhận dạng ảnh là quá trình xác định ảnh, thường thu được bằng cách so sánh với mẫu chuẩn đã được học (hoặc lưu) từ trước. Nội suy là phán đoán theo ý nghĩa trên cơ sở nhận dạng. Ví dụ: một loạt chữ số và nét gạch ngang trên phong bì thư có thể được nội suy thành mã điện thoại. Ảnh được phân theo hai loại nhận dạng ảnh cơ bản: - Nhận dạng theo tham số. - Nhận dạng theo cấu trúc. Nhận dạng chữ in, chữ viết tay, chữ ký điện tử, nhận dạng văn bản (Text), nhận dạng vân tay, mã vạch, mặt người...

Cơ sở tri thức (Knowledge Base) Ảnh là một đối tượng khá phức tạp về đường nét, độ sáng tối, dung lượng điểm ảnh, môi trường để thu ảnh phong phú kéo theo nhiều. Trong nhiều khâu xử lý và phân tích ảnh ngoài việc đơn giản hóa các phương pháp toán học đảm bảo tiện lợi cho xử lý, người ta mong muốn bắt chước quy trình tiếp nhận và xử lý ảnh theo trí tuệ con người.

Mô tả (biểu diễn ảnh), ảnh sau khi số hoá sẽ được lưu vào bộ nhớ, hoặc chuyển sang các khâu tiếp theo để phân tích. Nếu lưu trữ ảnh trực tiếp từ các ảnh thô, đòi hỏi dung lượng bộ nhớ cực lớn và không hiệu quả. Thông thường, các ảnh thô đó được đặc tả (biểu diễn) lại (hay đơn giản là mã hoá) theo các đặc điểm của ảnh như: biên ảnh (Boundary), vùng ảnh (Region). Một số phương pháp: • Biểu diễn bằng mã chạy (Run-Length Code) • Biểu diễn

bằng mã xích (Chaine -Code) • Biểu diễn bằng mã tứ phân (Quad-Tree Code) Biểu diễn bằng mã chạy thường biểu diễn cho vùng ảnh và ảnh nhị phân.

5-6-7. Cho biết tác vụ với mức độ thông minh của hệ thống trong video? Dùng kiến thức nào của 3 môn? (3 video) (Dùng kiến thức thị giác máy tính hay đồ họa máy tính hay xử lý ảnh?)

Video	Tác vụ	Mức độ thông minh	Kiến thức
Hitachi tracking	Tracking	<=Wisdom	Thị giác máy tính
The Sweeper Robot	Classification,Detection	<=Intelligence	Thị giác máy tính
Little Yellow Horse	Detection	<=Knowledge	Thị giác máy tính

REFERENCES

1. [Look out, Deliveroo and Uber Eats – in China, robots are the latest in food delivery trends - YP | South China Morning Post \(scmp.com\)](#)
2. [Hitachi AI people tracker - YouTube](#)
3. [The SWEEPER robot \(English\) - YouTube](#)

Tuần 2 (16/10/2021)

1. Ý nghĩa của xe tự hành? Ban đầu làm ra xe để lái, sau lại không muốn lái nữa mà đề xuất ra xe tự hành không người lái? Như vậy, logic có tự nhiên, bình thường hay không, có hợp logic không?

Bảo vệ môi trường - Giảm thiểu được lượng khí thải Cacbon, giải quyết vấn đề tầng ozon

Giảm số lượng xe - Giảm lượng xe lưu thông

Giảm chi phí - Chỉ khi cần thì mới gọi

Tối ưu thời gian - Giảm thời gian chết khi về đêm, xe thông thường để bên trong garage mà không được sử dụng, do vậy xe sẽ được sử dụng tối đa công năng

Giảm ách tắc giao thông - Hành khách có thể gọi xe nhanh chóng qua một ứng dụng di động. Đặc biệt hơn, mô hình này sẽ bao gồm dịch vụ chia sẻ xe giữa những người cùng lộ trình để giảm chi phí cũng như lượng xe lưu thông. Theo các số liệu thống kê, tổng quãng đường các phương tiện di chuyển trong năm 2015, 2016 tại Mỹ đạt kỷ lục gần 5.000 tỷ km. Tuy nhiên, con số này được dự báo sẽ tăng lên 12.800 tỷ km trong năm 35 năm tới khi xe tự lái được sử dụng ngày càng phổ biến, đặc biệt trong hai nhóm người độ tuổi trên 65 và từ 16-24. Bên cạnh đó, tình trạng ách tắc còn do xe tự lái vẫn phải chung đường với ô tô thường trong nhiều thập kỷ nữa. Theo một nghiên cứu hồi năm 2015 của Diễn đàn Vận tải quốc tế tại Lisbon (Bồ Đào Nha), nếu một nửa số xe di chuyển vẫn là ô tô thường, tổng quãng đường di chuyển của ô tô vẫn tăng từ 30 đến 90%. Do đó, việc người dân thường xuyên sử dụng thêm xe tự lái sẽ khiến tình trạng tắc nghẽn giao thông tồi tệ hơn trong thời gian dài.

Giảm thiểu tai nạn - Tuy nhiên, hơn một chục tai nạn ở Mỹ khiến gần 20 người thương vong. Vụ việc có thể ảnh hưởng hàng trăm ngàn xe đã bán của Tesla. Theo NHTSA, trong các tai nạn trên, có vụ xe Tesla đã đâm vào xe cứu hỏa hoặc xe cảnh sát, một số vụ khác đâm vào xe đang dừng và đã bật tín hiệu khẩn cấp. Ít nhất 17 người bị thương và 1 người chết sau các tai nạn này.

Tính đồng bộ - Bộ óc điều khiển xe tự hành thì tương đối đồng bộ hơn bộ óc của mỗi người riêng lẻ, hành vi của xe tự hành của cùng một số hãng sẽ thống nhất, đồng nhất.

Thích ứng rủi ro - Đặc biệt là trong tình huống Covid trên diện rộng, xe tự hành cho thấy được sự thích ứng hơn.

Như vậy, logic ở mệnh đề trên là bình thường. Không phải người ta tạo ra cái để người ta lái. Bản chất người ta tạo ra chiếc xe để phục vụ cho nhu cầu di chuyển, lưu thông, chứ không quan trọng ở vấn đề được lái hay không lái. Cho nên việc đề xuất xe tự hành không có gì là vô so với mục tiêu ban đầu là lái xe là để lưu thông chứ không chỉ để là lái.

Theo cuộc CMCN 4.0, người ta muốn thông minh hóa qui trình sản xuất và quản lý xã hội bằng cách chuyển đổi số trong giao thông công cộng dưới nền tảng là AI.

2.Lập bảng so sánh dữ liệu ảnh/video 2D như hiện nay với dữ liệu ảnh/video 3D? Cái nào tốt hơn? Ưu, nhược điểm? (Hint: ảnh RGB 2D và kênh độ sâu cho

bit vị trí mỗi điểm ảnh cho đến vị trí của , lidar cho tập điểm tọa độ)

	dữ liệu ảnh/video 2D	dữ liệu ảnh/video 3D
Ưu	Hình thức giảm chiều dữ liệu, tạo tiền đề cho những toán tử sau xử lý nhanh hơn	-Có thể xem được một sự kiện trong khung hình với nhiều góc nhìn -Tái tạo đối tượng từ dãy ảnh 2D
Khuyết	-Không gian ảnh trên mặt phẳng chỉ phản ánh cái bóng của thế giới, mất đi thông tin quan trọng là kích thước, mất đi kênh thông tin liên lạc giữa môi trường thực và môi trường	

3. Nhìn màu trên màn hình máy tính, mỗi điểm ảnh, mắt ta tiếp nhận thông tin RGB khác nhau, vậy còn trên tờ giấy, làm thế nào chúng ta có thể nhìn ra màu?

Khi ánh sáng trắng chiếu vào tờ giấy, gặp những hạt mực CMY phản xạ ngược vào mắt người, mực sẽ trả về bước sóng ánh sáng qua mắt.

Người thấy được màu trên trang vì trên trang có các hạt Trám (tập hợp hạt CMY) được sắp xếp phân bố theo khoảng cách tùy ý và góc nghiêng nhất định với size tùy ý.

4. Viết chương trình chuyển ảnh màu RGB sang Grayscale (công thức)?

Phương pháp Weighted hoặc phương pháp luminosity

Phương pháp trung bình Grayscale=(R+G+B)/3 làm mất đi độ bóng của bức ảnh. Phương pháp trọng số có một giải pháp cho vấn đề đó. Vì red có nhiều bước sóng hơn trong cả ba màu, và green là màu không chỉ có ít bước sóng hơn red mà còn có green là màu mang lại hiệu ứng dịu mắt hơn. Nó có nghĩa là chúng ta phải giảm sự đóng góp của red, và tăng sự đóng góp của green, và đưa sự đóng góp của blue vào giữa hai màu này. Vì vậy, phương trình Gauss 3x3 có dạng:

Hình ảnh thang độ xám mới = ((0,2989 * R) + (0,587 * G) + (0,114 * B)).

Theo phương trình này, red đóng góp ~30%, green đóng góp ~59%, cao hơn trong cả ba màu và blue đóng góp ~11%.


```
#python
def gray(src):
    Mat gray,
    vector<Mat> img_rgb
    split(src, img_rgb)

    img_rgb[0] = img_rgb[0] * 0.1140
    img_rgb[1] = img_rgb[1] * 0.5870
    img_rgb[2] = img_rgb[2] * 0.2989

    gray = img_rgb[0] + img_rgb[1] + img_rgb[2]
    return gray
```

```
//C++
#include "stdafx.h"
int rgb2gray(cv::Mat &s, cv::Mat &d){
    if (s.data ==NULL)return 0;

    int w=s.col, h=s.row, sChannel=s.channels(), dChannel=d.channels();
    d=cv::Mat(h,w,CV_8UC1);
    for(int i=0;i<h;i++){
        uchar*pSR=s.ptr<uchar>(i);
        uchar*pDR=d.ptr<uchar>(i);
        for(j=0;j<w;j++,pSR+=sChannel,pDR+=dChannel){
            uchar B=pSR[0];
            uchar G=pSR[1];
            uchar R=pSR[2];
            uchar gray=(uchar)(B*0.114+G*0.587+R*0.2989)/3;
            pDR[0]=gray;
        }
    }return 1;
}
```

5. Tính độ dị biệt giữa 2 màu trong không gian RGB $d(c_i, c_j)$, tính bằng gì? Đưa ảnh xám Grayscale sang ảnh màu RGB?

$$colour_1 = (R_1, G_1, B_1)$$

$$colour_2 = (R_2, G_2, B_2)$$

$$difference = \sqrt{(R_2 - R_1)^2 + (G_2 - G_1)^2 + (B_2 - B_1)^2}$$

Gray sang RGB

Khó có thể chuyển từ màu xám sang RGB chính xác, nên ở đây ta sẽ tạm chấp nhận việc giá trị 3 kênh màu RGB sẽ bằng giá trị của kênh màu xám

```
int gray2rgb(const cv::Mat& s, cv::Mat& d) {
    int col = s.cols, row = s.rows;
    Mat image(row, col, CV_8UC3);
    for (int i = 0; i < row; i++){
        for (int j = 0; j < col; j++){uint8_t gray = s.data[i * col + j];

            image.data[i * col * 3 + j * 3 + 0]= gray;
            image.data[i * col * 3 + j * 3 + 1]= gray;
            image.data[i * col * 3 + j * 3 + 2]= gray;
        }
    }
    d=image;
    return 1;
};
```

6. **Đổi ảnh màu mô hình RGB sang HSV và ngược lại? Tính độ dị biệt màu giữa 2 điểm ảnh trong mô hình HSV?**

RGB sang HSV:

$$H = \begin{cases} \theta, (B \leq G) \\ 360 - \theta, (B > G) \end{cases} \quad \text{với}$$

$$\theta = \cos^{-1} \left\{ \frac{\frac{1}{2}[(R - G) + (R - B)]}{[(R - G)^2 + (R - B)(G - B)]^{\frac{1}{2}}} \right\}$$

$$S = 1 - \frac{3}{(R + G + B)} [\min(R, G, B)]$$

$$V = \max\left(\frac{R}{255}, \frac{G}{255}, \frac{B}{255}\right)$$

HSV sang RGB:

$$(R, G, B) = \left(\frac{R}{255} + V - VS, \frac{G}{255} + V - VS, \frac{B}{255} + V - VS\right)$$

Độ dị biệt:

$$dh = \frac{\min(|h1 - h0|, 360 - |h1 - h0|)}{180}$$

$$ds = |s1 - s0|$$

$$dv = \frac{|v1 - v0|}{255}$$

$$\text{distance} = \sqrt{dh * dh + ds * ds + dv * dv}$$

7. Xét công thức Minkowski

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

$$\text{Với } r=1, n=2: d_1(p, q) = |p_1 - q_1| + |p_2 - q_2|$$

$$\text{Với } r=2, n=2: d_2(p, q) = \sqrt{|p_1 - q_1|^2 + |p_2 - q_2|^2}$$

$$\text{Với } r=\infty, n=2: d_\infty(p, q) = \max\{|p_1 - q_1|, |p_2 - q_2|\}$$

a) Biết $\|x_2\| = d_2(x, O) = 1$, CMR $x \in (O; R=1)$

$$\text{Cho } x = (x_x, x_y) \Rightarrow (x_x - 0)^2 + (x_y - 0)^2 = 1 \Rightarrow x_x^2 + x_y^2 = 1$$

b) Biết $||x_1|| = d_1(x, O) = 1$, CMR $x \in$ hình vuông mà $(O; R=1)$ ngoại tiếp

$$\text{Cho } x = (a, b) \Rightarrow |a - 0| + |b - 0| = 1 \Rightarrow |a| + |b| = 1$$

$$\Rightarrow \begin{cases} a + b = 1, (\exists a, b > 0) \\ -a + b = 1, (\exists a < 0, \exists b > 0) \\ a - b = 1, (\exists a > 0, \exists b < 0) \\ -a - b = 1, (\exists a, b < 0) \end{cases}$$

c) Biết $||x_\infty|| = d_\infty(x, O) = 1$, CMR $x \in$ hình vuông mà $(O; R=1)$ nội tiếp

$$\text{Cho } x = (c, d) \Rightarrow \max\{|c - 0|, |d - 0|\} = 1 \Rightarrow \max\{|c|, |d|\} = 1$$

$$\Rightarrow \begin{cases} c = \pm 1 \\ -1 < d < 1 \\ d = \pm 1 \\ -1 < c < 1 \end{cases}$$

Vậy nếu $||x_2|| = d_2(x, O) = a$ thì nó sẽ là tập điểm thế nào? (mở rộng phạm vi lấy điểm lân cận)

Cmtt với đường tròn lượng giác, $d_2(x, O) = a$ sẽ thuộc tập điểm nằm trên đường tròn tâm O có bán kính $R=a$

$d_1(x, O) = a$ sẽ thuộc tập điểm nằm trên hình vuông mà $(O; R=a)$ ngoại tiếp

$d_\infty(x, O) = a$ sẽ thuộc tập điểm nằm trên hình vuông mà $(O; R=a)$ nội tiếp

REFERENCES

[metrics - Calculate distance between colors in HSV space - Stack Overflow](#)

[Colour Difference Formula | 101 Computing](#)

Tuần 3 (6/11/2021)

1. Màu là gì? Vân là gì? Dáng là gì? Làm sao nhìn cái hình mà nó ra màu?

Làm sao nhìn được dáng? Dáng là tập hợp điểm như thế nào?

- Đối với màn hình: Màn hình phát sáng chiếu trực tiếp đến mắt, tại 1 pixel của màn

hình gồm R,G,B, từ đó não bộ s tổng hợp ra màu.

- Đối với tờ giấy: mực CMYK sẽ hấp thụ bớt ánh sáng ở ngoài chiếu vào và phản xạ ra mắt người.
- Vân là phân bố tập điểm, phân bố về màu sắc, vị trí mà các tập điểm cấu thành, có thể mô tả bằng biểu thức toán.

(hint: Haralick, Wavelet, Gabor)

- Dáng của các đối tượng, sự kiện là do 2 hay nhiều điểm cấu thành, tại các điểm đó có biến thiên lớn về độ sáng. Một trường vector tiếp xúc với biên để mô tả dáng.

2. Ngoài màu, vân, dáng, còn đặc trưng nào khác? (Ảnh tĩnh)

Đặc trưng kết cấu

Trích xuất nội dung ảnh theo kết cấu nhằm tìm ra mô hình trực quan của ảnh và cách thức chúng được xác định trong không gian. Kết cấu được biểu diễn bởi các texel mà sau đó được đặt vào một số các tập phụ thuộc vào số kết cấu được phát hiện trong ảnh. Các tập này không chỉ xác định các kết cấu mà còn chỉ rõ vị trí các kết cấu trong ảnh. Việc xác định các kết cấu đặc biệt trong ảnh đạt được chủ yếu bằng cách mô hình các kết cấu như những biến thể cấp độ xám 2 chiều.

3. Pixel tròn hay vuông?

Pixel không có hình dạng, kích thước cụ thể.

4. Xấp xỉ $16 \times 8 \approx 2000$ có chấp nhận được không?

Có thể chấp nhận được. Trong khối 1800 màu có thể lấy ra 1 màu đại diện vì màu biến thiên liên tục nên nó sẽ rất giống nhau.

5. Lập bảng xem Histogram có bền vững theo mỗi góc không? (hint: scale (Position, Size, Rotation), viewpoint, lighting)

$$h^{color}[m] = \frac{1}{I_x I_y} \sum_{x=0}^{I_x-1} \sum_{y=0}^{I_y-1} \begin{cases} 1, & (Q^{color}(I^C(x, y)) = m) \\ 0, & \text{otherwise} \end{cases}$$

6. Nhiều ảnh khác nhau có cùng 1 histogram không? Trong cách tính color histogram và edge histogram, người ta đã bỏ lỡ gì mà lại xảy ra hiện tượng ánh xạ không đơn ánh này?

Có trường hợp như vậy vì không xét đến yếu tố vị trí không gian, người ta chỉ xét tần số xuất hiện của màu đó.

7. Tăng độ sáng. Tăng độ sáng mà vẫn giữ độ tương phản khi nhân với a trong $g(x, y) = a \cdot f(x, y) + b$

-Tăng đều tại mỗi điểm ảnh 1 giá trị b

-Khi có nhân với a:

$$|g(x_i, y_i) - g(x_j, y_j)| = |a| \cdot |f(x_i, y_i) - f(x_j, y_j)|$$

$$\Rightarrow \begin{cases} |g(x_i, y_i) - g(x_j, y_j)| < |f(x_i, y_i) - f(x_j, y_j)|, (|a| < 1) \\ |g(x_i, y_i) - g(x_j, y_j)| \geq |f(x_i, y_i) - f(x_j, y_j)|, (|a| \geq 1) \end{cases}$$

8. Ép mức xám từ khoảng này [f1 f2] sang khoảng khác [g1 g2]

Phương pháp

$$g(x, y) = a \cdot f(x, y) + b$$

$$\text{với } a = \frac{g_2 - g_1}{f_2 - f_1}$$

$$\begin{cases} g_1 = a f_1 + b \\ g_2 = a f_2 + b \end{cases}$$

9. Cân bằng lược đồ xám

2	2	2	2	2	2	2	5
2	6	6	6	6	5	5	5
1	7	7	6	6	5	5	5
1	7	7	1	1	8	6	4
2	9	8	8	8	8	6	4
2	9	10	10	11	12	12	3
2	9	9	10	10	14	13	3
2	2	2	2	2	3	3	3

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

$$T[0] = H[0]$$

$$T[p] = T[p - 1] + H[p], p \in [1; nG - 1]$$

$$T[p] = \text{round}\left(\frac{nG - 1}{N \cdot M} \cdot T[p]\right)$$

$$\Rightarrow g(x, y) = T[f(x, y)]$$

4	4	4	4	4	4	4	7
4	9	9	9	9	7	7	7
1	10	10	9	9	7	7	7
1	10	10	1	1	11	9	6
4	12	11	11	11	11	9	6
4	12	13	13	13	14	14	5
4	12	12	13	13	14	14	5
4	4	4	4	4	5	5	5

TR board

Tuần 4 (13/11/2021)

1. Khác biệt cơ bản giữa biến đổi ảnh của DHMT và XLA

DHMT	XLA
Cho những điểm cấu thành nên thực thể, mà giữa những điểm cấu thành đó được vẽ ra bởi phương trình toán	Giữa các pixel màu là độc lập, không có bất kì mối liên hệ nào

học. Đối tượng được miêu tả dưới dạng tổ hợp các vector tạo thành. Độc lập với độ phân giải.	
---	--

2. Thống kê bảng các tiêu chí: phương pháp tăng cường chất lượng ảnh dựa vào màu sắc, độ phức tạp, ưu khuyết,...

	Linear	NonLinear	PDF
Phương pháp	Dựa trên vị trí tại mỗi điểm ảnh khi thay đổi giá trị màu (tăng, giảm..) hoặc theo công thức hàm số.		Dựa trên phân bố xác suất màu của toàn ảnh để quyết định
	<p>Bright: $g(x, y) = f(x, y) + b$</p> <p>Contrast: $g(x, y) = a \cdot f(x, y)$</p> <p>Bright+Contrast: $g(x, y) = a \cdot f(x, y) + b$</p> <p>Range of grey levels onto new range: $g(x, y) = a \cdot f(x, y) + b$</p> <p>với $a = \frac{g_2 - g_1}{f_2 - f_1}$</p> <p>$\begin{cases} g_1 = a f_1 + b \\ g_2 = a f_2 + b \end{cases}$</p>	<p>Logarithmic: $g(x, y) = c \cdot \log f(x, y)$</p> <p>Exponential: $g(x, y) = e^{f(x, y)}$</p>	<p>Histogram equalization: $g = T(f)$ $\Rightarrow p_g(g) = \frac{p_f(f)}{\left \frac{dT(f)}{df} \right }$ For granted $p_g(g) = 1$</p> <p>$\Rightarrow \frac{dT(f)}{df} = p_f(f)$</p> <p>$\Rightarrow \int_0^f \frac{p_f(w)}{p_f(f)} dw$ $= \int_0^f p_f(f) df$</p> <p>$\Rightarrow T(f) = \int_0^f p_f(w) dw$</p> <p>$g_k = T(f_k)$ $= \sum_{j=0}^k p_j(f_j)$ $= \sum_{j=0}^k \frac{n_j}{n}$ $= \sum_{j=0}^{k-1} \frac{n_j}{n} + \frac{n_k}{n}$ $= g_{k-1} + \frac{n_k}{n}$</p> <p>Histogram specification: $s =$</p>

			$T(f) = \int_0^f p_f(w)dw$ $z = G(g)$ $= \int_0^g p_g(w)dw$ $g = G^{-1}(z) = G^{-1}(s)$ $= G^{-1}[T(f)]$
Độ phức tạp	$O(n)$	Nonlinear	$O(n)$ theo độ dài của chuỗi truyền, nhưng $O(2^n)$ trong độ dài bộ nhớ kênh
Ưu	Dễ hiểu Ứng dụng phổ biến hơn.	Hiệu suất xử lý ảnh vượt xa những gì có thể đạt được bởi đối tác Linear	
Khuyết	Hiệu suất xử lý ảnh thấp hơn Nonlinear	Các hệ thống phi tuyến gần như không được hiểu rõ, do đó phạm vi ứng dụng của chúng bị hạn chế.	

3. Dùng phương pháp Nearest neighbor có ưu khuyết gì trong vấn đề scale ảnh

Ưu	Khuyết
Tính rất nhanh	Blocky (pixel lọt vào vùng buộc phải lấy những màu pixel lân cận ở gần nó nhất), gây ra hiện tượng ảnh không được mượt, có nhiều hình vuông nhỏ.

4. Trung bình trọng số

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

5. Tính trực tiếp

$$f(l, k)\sqrt{a^2 + b^2} + f(l+1, k)\sqrt{(1-a)^2 + (1-b)^2} + f(l+1, k+1)\sqrt{(1-a)^2 + (1-b)^2} + f(l, k+1)\sqrt{a^2 + (1-b)^2}$$

6. Ưu nhược của image smoothing khi dùng Averaging

Ưu	Nhược
Phương pháp Averaging lọc tốt đối với những ảnh bị nhiễu random như nhiễu Gaussian hoặc nhiễu Uniform.	Khi khử nhiễu, luôn phải cần sự tham gia trợ giúp của tất cả pixel trong khu vực đang duyệt. Quá trình làm mờ nhiễu cũng đồng thời làm mờ ảnh vì bộ lọc trung bình có xu hướng cân bằng cường độ sáng các điểm ảnh.

7. Chứng minh đạo hàm

$$\begin{aligned} \int_x (x, y) &= \int (x, y) - \int (x+1, y) \\ \int_y (x, y) &= \int (x, y) - \int (x, y-1) \\ \frac{\partial f}{\partial x}(x, y) &= \lim_{h \rightarrow 0} \frac{\int (x+h, y) - \int (x, y)}{h} = \frac{\int (x+1, y) - \int (x, y)}{1} \end{aligned}$$

Tuần 5 Giải đáp thắc mắc về Project (không có BTLN) (20/11/2021)

Tuần 6 (27/11/2021)

1. Xác định độ phức tạp tính toán của 2 thuật toán Fourier $f * h$ và $F * H$ bằng cách lập bảng so sánh

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

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

$f * h$	$F * H$
$f(x, y) * h(x, y) = \frac{1}{MN} \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} f(x-m, y-n) h(m, n)$ <p>Độ phức tạp của phương pháp lọc ảnh trong miền không gian: $O(n^2 * m^2)$</p>	<p>Bước 1: Fourier(f) = $F(u, v)$ $O(n^4)$ giảm còn $O(n^2 \log 2 n)$: FFT</p> <p>Bước 2: Fourier(h) = $H(u, v)$: $O(n^2)$</p> <p>Bước 3: $F(u, v) \cdot H(u, v)$: $O(n^2)$</p> <p>Bước 4: Fourier⁻¹{F.H}: $O(n^2 \log 2 n)$</p> <p>Độ phức tạp của phương pháp lọc ảnh trong miền tần số: $O(n^2 \log 2 n)$</p>

Kết luận: Phương pháp nào nhanh hơn sẽ phụ thuộc vào giá trị của m và n.

2. Cái thứ 2 dùng để làm gì Discrete Fourier Transform

Sử dụng để lọc ảnh trong miền tần số.

3. Tầm quan trọng của định lý trong 2 tác vụ làm trơn ảnh và phát hiện biên cạnh 7.2.2 Convolution (2 tác vụ dùng định lý này ở chỗ nào)

- Ứng dụng hữu ích 1: Sử dụng không gian tần số để hiểu tác dụng của bộ lọc

- Ví dụ: Biến đổi Fourier của một Gaussian là một Gaussian

- Do đó: làm giảm tần số cao

Ứng dụng hữu ích 2: Tính toán hiệu quả - Biến đổi Fourier nhanh (FFT) mất thời gian $O(n \log n)$

- Do đó, có thể thực hiện tích chập trong thời gian $O(n \log n + m \log m)$

- Tăng hiệu quả lớn nhất cho các bộ lọc lớn

4.
$$f(t) = \begin{cases} 0, & t < 0 \\ e^{-at}, & t \geq 0, a > 0 \end{cases}$$

a) $F(w)$

$$\begin{aligned} F(w) &= \frac{1}{2\pi} \int_0^{\infty} e^{-iwt} dt = \frac{e^{-(a+iw)t} \Big|_0^{\infty}}{-2\pi(a+iw)} = \frac{1 - \lim_{t \rightarrow \infty} e^{-(a+iw)t}}{2\pi(a+iw)} = \frac{1 - \lim_{t \rightarrow \infty} |e^{-(a+iw)t}|}{2\pi(a+iw)} \\ &= \frac{1 - \lim_{t \rightarrow \infty} |e^{-at} \cdot e^{iwt}|}{2\pi(a+iw)} = \frac{1 - \lim_{t \rightarrow \infty} |e^{-at}|}{2\pi(a+iw)} = \frac{1 - 0}{2\pi(a+iw)} = \frac{1}{2\pi(a+iw)} \end{aligned}$$

b) $f(t)$

$$f(t) = \frac{1}{2\pi} \int_{-\infty}^{\infty} \frac{e^{iwt}}{a+iw} dw$$

5. Giữ cái nào bỏ cái nào?

Để cực tiểu sai số $E(||x - \hat{x}||^2)$

$$= \sum_{i=m}^{n-1} a_i^T \cdot E[xx^T] \cdot a_i = \sum_{i=m}^{n-1} a_i^T \cdot \lambda_i \cdot a_i = \sum_{i=m}^{n-1} \lambda_i$$

thì ta sắp theo thứ tự giảm dần rồi

- giữ $\lambda_0 \rightarrow \lambda_{m-1}$: lớn nhất
- bỏ $\lambda_m \rightarrow \lambda_{n-1}$: bé nhất

Tuần 7 Buổi giảng bài tập trên lớp (không có BTLN)
(4/12/2021)

Tuần 8 (11/12/2021)

1. Qui nạp: Giá trị trung bình độ xám của 8 vùng Region growing

$$m(R_i^{(k)}) = \frac{1}{N(R_i^{(k)})} \cdot \sum_{(k,l) \in R_i^{(k)}} f(k,l)$$

$$\begin{aligned}
\sigma(R_i^{(k)}) &= \sqrt{\frac{1}{N(R_i^{(k)})} \cdot \sum_{(k,l) \in R_i^{(k)}} (f(k,l) - m(R_i^{(k)}))^2} \\
&\Rightarrow m(R_i^{(k+1)}) - m(R_i^{(k)}) \\
&= \left[\frac{1}{N(R_i^{(k)})} + 1 \right] [f(x,y) + N(R_i^{(k)}) * m(R_i^{(k)})] - \frac{1}{N(R_i^{(k)})} \cdot \sum_{(k,l) \in R_i^{(k)}} f(k,l) \\
&= \left[\frac{1}{N(R_i^{(k)})} + 1 \right] [f(x,y) + N(R_i^{(k)}) * \frac{1}{N(R_i^{(k)})} \cdot \sum_{(k,l) \in R_i^{(k)}} f(k,l)] \\
&\quad - \frac{1}{N(R_i^{(k)})} \cdot \sum_{(k,l) \in R_i^{(k)}} f(k,l) \\
&= \left[\frac{1}{N(R_i^{(k)})} + 1 \right] [f(x,y) * \frac{1}{N(R_i^{(k)})} * \sum_{(k,l) \in R_i^{(k)}} f(k,l) + \sum_{(k,l) \in R_i^{(k)}} f(k,l)] \\
&\quad - \frac{1}{N(R_i^{(k)})} \cdot \sum_{(k,l) \in R_i^{(k)}} f(k,l) \\
&= \frac{1}{N(R_i^{(k)})} * f(x,y) * \frac{1}{N(R_i^{(k)})} * \sum_{(k,l) \in R_i^{(k)}} f(k,l) + \frac{1}{N(R_i^{(k)})} * \sum_{(k,l) \in R_i^{(k)}} f(k,l) + f(x,y) \\
&\quad * \frac{1}{N(R_i^{(k)})} * \sum_{(k,l) \in R_i^{(k)}} f(k,l) \\
&+ \sum_{(k,l) \in R_i^{(k)}} f(k,l) - \frac{1}{N(R_i^{(k)})} \cdot \sum_{(k,l) \in R_i^{(k)}} f(k,l) \\
&= \left(\frac{1}{N^2(R_i^{(k)})} * f(x,y) + \frac{1}{N(R_i^{(k)})} + f(x,y) * \frac{1}{N(R_i^{(k)})} + 1 - \frac{1}{N(R_i^{(k)})} \right) * \sum_{(k,l) \in R_i^{(k)}} f(k,l) \\
&= [f(x,y) * \frac{1}{N(R_i^{(k)})} * (\frac{1}{N(R_i^{(k)})} + 1) + 1] * \sum_{(k,l) \in R_i^{(k)}} f(k,l) \\
&= [f(x,y) * \frac{1}{N(R_i^{(k)})} * (\frac{1}{N(R_i^{(k)})} + 1) + 1] * \sum_{(k,l) \in R_i^{(k)}} f(k,l)
\end{aligned}$$

Qui nạp:

$$m(R_i^{(k+1)}) = [\frac{1}{N(R_i^{(k)})} + 1][f(x, y) + N(R_i^{(k)}) * m(R_i^{(k)})]$$

$$\sigma(R_i^{(k+1)}) = \sqrt{\frac{1}{N(R_i^{(k)})} + 1 + N(R_i^{(k)}) * \sigma^2(R_i^{(k)}) + \frac{N(R_i^{(k)})}{N(R_i^{(k)}) + 1} (f(x, y) - m(R_i^{(k)}))^2}$$

2. Giải thuật region growing

```
import os
import cv2
import numpy as np

def region_growing(img, x, y, delta=15):
    """
    x: relative coord (0-1)
    y: relative coord (0-1)
    """
    def is_valid_px(x, y, img):
        if x>=0 and x<img.shape[1] and y>=0 and y<img.shape[0]:
            return True
        return False

    mask = np.zeros(img.shape[:2])
    y_abs = int(y * img.shape[0])
    x_abs = int(x * img.shape[1])

    loop_points = []
    loop_points.append((x_abs, y_abs))
    px_intensity = img[y_abs, x_abs, :]
    blue = px_intensity[0]
    green = px_intensity[1]
    red = px_intensity[2]

    while len(loop_points) != 0:
        (cx, cy) = loop_points.pop()
        current_intensity = img[cy, cx, :]
        b = int(current_intensity[0])
        g = int(current_intensity[1])
        r = int(current_intensity[2])

        if abs(r-red)<delta and abs(g-green)<delta and abs(b-blue)<delta:
            mask[cy, cx] = 255
            neighbors = [
                (cx-1, cy),
                (cx+1, cy),
                (cx-1, cy-1),
                (cx, cy-1),
                (cx+1, cy-1),
                (cx-1, cy+1),
                (cx, cy+1),
                (cx+1, cy+1),
            ]
```

```

        (cx+1, cy+1)
    ]
    for (nx, ny) in neighbors:
        if is_valid_px(nx, ny, img) and mask[ny, nx] == 0:
            loop_points.append((nx, ny))
    else:
        pass
    return mask

def main(img_path):
    img = cv2.imread(img_path, cv2.IMREAD_COLOR)
    KERNEL_WIDTH = KERNEL_HEIGHT = 5
    SIGMA_X = SIGMA_Y = 2
    img[:, :, 0] = cv2.GaussianBlur(img[:, :, 0], ksize=(KERNEL_WIDTH, KERNEL_HEIGHT),
    sigmaX=SIGMA_X, sigmaY=SIGMA_Y)
    img[:, :, 1] = cv2.GaussianBlur(img[:, :, 1], ksize=(KERNEL_WIDTH, KERNEL_HEIGHT),
    sigmaX=SIGMA_X, sigmaY=SIGMA_Y)
    img[:, :, 2] = cv2.GaussianBlur(img[:, :, 2], ksize=(KERNEL_WIDTH, KERNEL_HEIGHT),
    sigmaX=SIGMA_X, sigmaY=SIGMA_Y)
    cv2.imwrite('out_blur_'+os.path.basename(img_path), img)

    x = 0.5
    y = 0.6
    delta = 20
    mask = region_growing(img=img, x=x, y=y, delta=delta)
    cv2.imwrite('out_mask.jpg', mask)

    img_color = cv2.imread(img_path, cv2.IMREAD_COLOR)
    img_color[mask>0, :] = (0, 255, 255)

    cv2.circle(img_color, (int(x*img.shape[1]), int(y*img.shape[0])), radius=5, color=(0, 0,
    255), thickness=2)
    cv2.imwrite('out_seg_'+os.path.basename(img_path), img_color)

if __name__ == "__main__":
    print('Running...')
    main('jung_yun.jpg')
    print('* Follow me @ ' + "\x1b[1;%dm" % (34) + '
    https://www.facebook.com/minhng.info/' + "\x1b[0m")
    print('* Join GVGroup for discussion @ ' + "\x1b[1;%dm" % (34) + '
    https://www.facebook.com/groups/ip.gvgroup/' + "\x1b[0m")
    print('* Thank you ^^~')

```

3. Prototype K mean method và so sánh với K-means++(hint: Chọn vector đặc trưng (centroid) khởi thủy như thế nào ở step 1 Kmean?)

K-means

B1: Chia lưới ảnh

B2: Mỗi ô lưới chọn ra vector đặc trưng

B3: Chọn ra 1 vector đặc trưng và đặt làm khởi thủy

K-means	K-means++
Khởi tạo vector đặc trưng (centroid) khởi thủy = lấy ngẫu nhiên vector đặc trưng của vùng từ ảnh đã bị chia ô lưới rồi tính "loss with cost-sensitivities"/ trung bình cộng/...	Chọn ngẫu nhiên centroid đầu tiên từ các điểm dữ liệu. Đối với mỗi điểm dữ liệu, tính toán khoảng cách của nó từ trung tâm gần nhất, đã chọn trước đó

4. Điều kiện dừng của vòng lặp

Khi các vùng (cluster) đã ổn định.

$$\frac{\partial E(t)}{\partial t} = 0 \approx \frac{E(t) - E(t-1)}{t - (t-1)} = 0 \Rightarrow |E(t) - E(t-1)| < \varepsilon$$

```
stop_condition:
bool flag = true;
for (int j=1; j <= k; j++){
    if ( $w_j^k \neq w_j^{k-1}$ ) flag = false;
    Else continue;
}
if (flag == true) break;
Else update prototype; goto stop_condition;
```

5. Điều kiện kết nạp ứng viên vào cluster C_j^* và nearest w_j^* như thế nào? (Hint:Tính distance Step 2)

Tính độ dị biệt của vector đặc trưng (centroid) đang xét với mỗi vùng (cluster)

Cluster mình chọn để kết nạp phải làm cho đối số độ dị biệt của vector đặc trưng đang xét với vector đặc trưng khởi thủy là cực tiểu.

Điều kiện cần của cực tiểu:

$$\frac{\partial E(t)}{\partial t} = 0 \approx \frac{E(t) - E(t-1)}{t - (t-1)} = 0 \Rightarrow |E(t) - E(t-1)| < \varepsilon$$

```

For l in range(1,n):
minDistance=distance( $i_L$ ,  $w_l$ )
minIdx=1
For  $j^*$  in range(1,k):
  If (distance( $i_L$ ,  $w_{j^*}$ )< minDistance):
    minDistance= distance( $i_L$ ,  $w_{j^*}$ )
    minIdx =  $j^*$ 
cluster[minIdx].add( $i_L$ )

```

6. Cập nhật lại centroid cho tất cả các vùng

Trung bình cộng

$$\text{prototype khởi thủy } w_j = \sum_{\text{vector đặc trưng } i_L \text{ của } \in \text{ từng vùng } C_j \text{ đã phân đoạn}} \frac{i_L}{|C_j|}$$

Tuần 9 SEMINAR (18/12/2021)

1) Với mỗi chủ đề, nhận xét (motivation, bước tiếp cận chính để giải)

1. 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 SEEDS

2. 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. 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 tính 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. 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. Ứng dụng Seamless cloning để sao chép đối tượng trong 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:

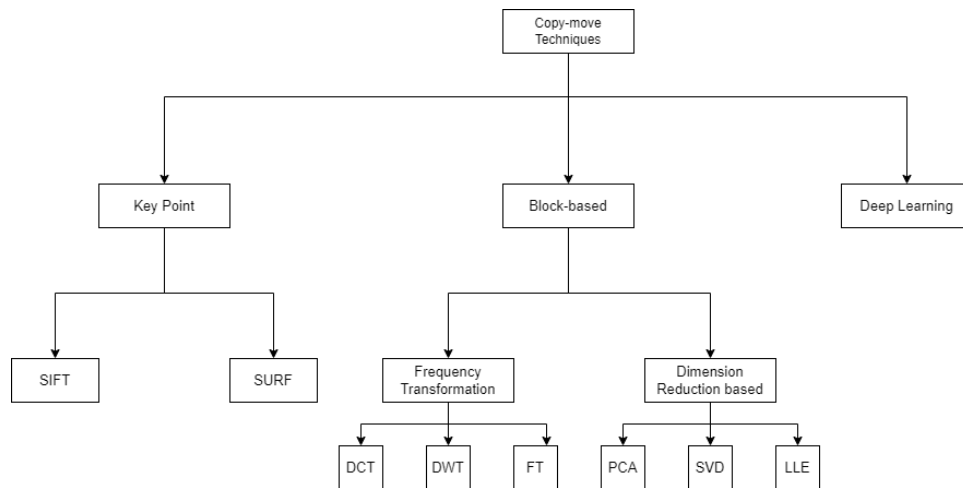
- Nhân bản liền mạch
- Che dấu đối tượng
- Đổi kết cấu của đối tượng
- Ghép hình
- b. Phương pháp
 - Chỉnh sửa ảnh bằng cách giải phương trình Poisson (dựa vào lý thuyết chỉnh sửa Poisson và sự biến thiên gradient của đối tượng và ảnh nền)

6. Ứng dụng Inpainting để sao chép đối tượng trong 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. 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. 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 - Tỉ số tín hiệu cực đại trên nhiễu
- Structural Similarity Index Measure - Chỉ mục cấu trúc tương tự

9. Nhóm Central Circus (trốn đồ án)

10. 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 khoảnh 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. 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. 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 mở 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 nơron 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 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:
 - Chụp 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.
 - Theo 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. 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. VPN: Truy vấn mặt người dựa vào thuộc tính (Facial attribute recognition)

Động lực về mặt khoa học

Trong tập dữ liệu bao gồm lớp đa số và lớp thiểu số. Trong lớp thiểu số có chứa số ít trường hợp có mức thay đổi trực quan cao. Điều này dẫn đến vấn đề mất cân bằng do các phương pháp thường cóu hướng hướng về lớp đa số, dẫn đến nó sẽ không còn đúng với lớp thiểu số.

Động lực ứng dụng:

Ứng dụng trong nhận dạng mặt người, dự đoán các thuộc tính mặt người

Phương pháp đề xuất

Large Margin Local Embedding (LMLE)

Để giải quyết vấn đề mất cân bằng, có hai giải pháp được nêu ra là data-resampling và cost-sensitive learning. LMLE nghiên cứu cả hai giải pháp thay vì chọn một trong hai.

16. 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. TMT: Không thuyết trình