



TRƯỜNG ĐẠI HỌC VINH
VINH UNIVERSITY
Nơi tạo dựng tương lai cho tuổi trẻ



Chương 1. Tổng quan về thị giác máy tính và OpenCV


ThS. Nguyễn Thị Minh Tâm
Email: tamntm@vinhuni.edu.vn



Đại học Vinh
Viện Kỹ thuật Công nghệ

ĐẠI HỌC VINH - 2025

1



Tổng quan về thị giác máy tính

- 1. Khái niệm**
2. Các lĩnh vực nghiên cứu của thị giác máy tính
3. Ứng dụng của thị giác máy tính
4. Thư viện OpenCV

NGUYỄN THỊ MINH TÂM - VIỆN KTCN - ĐH VINH

2



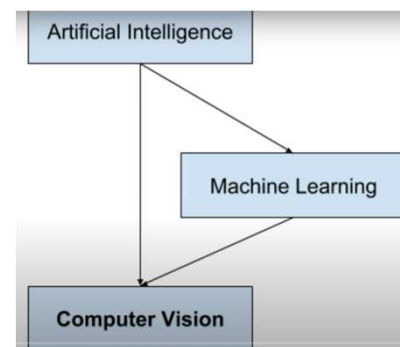
Khái niệm

▪ Thị giác máy tính (computer vision) là một lĩnh vực bao gồm các phương pháp thu nhận, xử lý ảnh kỹ thuật số, phân tích và nhận dạng các hình ảnh và, nói chung là dữ liệu đa chiều từ thế giới thực để cho ra các thông tin số hoặc biểu tượng, ví dụ trong các dạng quyết định

▪ Dữ liệu hình ảnh có thể nhiều dạng:

- chuỗi video, các cảnh từ đa camera,
- dữ liệu đa chiều từ máy quét y học.

▪ Thị giác máy tính còn là một môn học kỹ thuật, trong đó tìm kiếm việc áp dụng các mô hình và các lý thuyết cho việc xây dựng các hệ thống thị giác máy tính.



Computer Vision

Computer vision is an interdisciplinary scientific field that deals with how computers can gain high-level understanding from digital images or videos.

From the perspective of engineering, it seeks to understand and automate tasks that the human visual system can do

Thị giác máy tính (Computer Vision) là lĩnh vực nghiên cứu giúp máy tính có khả năng “nhìn” và “hiểu” hình ảnh/video tương tự như con người.



Lịch sử phát triển của Thị giác máy tính

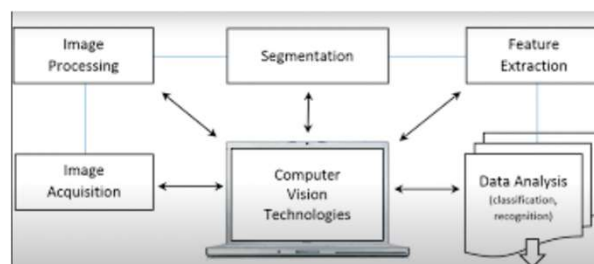
- 1960s: Nghiên cứu xử lý ảnh cơ bản.
- 1980s: Xuất hiện các thuật toán phát hiện biên, phân vùng ảnh.
- 2000s: Phát triển các thuật toán phát hiện/nhận dạng đối tượng truyền thống.
- 2012 đến nay: Bùng nổ Deep Learning, CNN, YOLO.

5



Các lĩnh vực nghiên cứu của thị giác máy tính

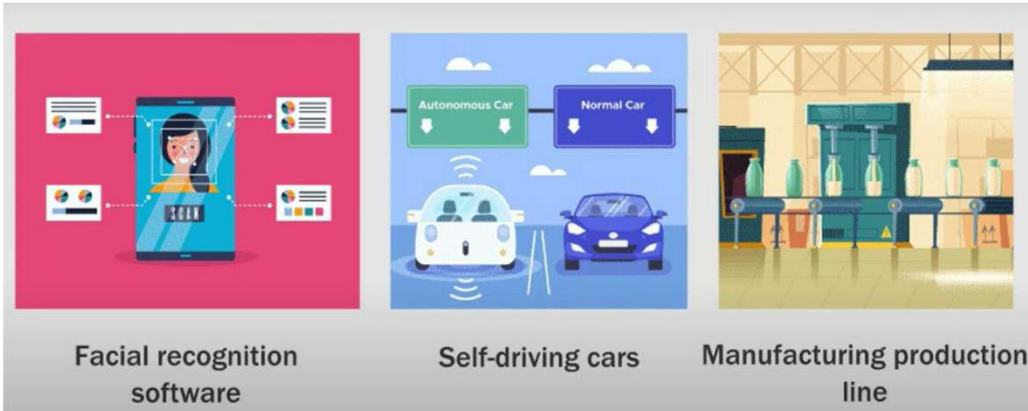
- Thị giác máy tính bao gồm các lĩnh vực sau:
 - Xử lý hình ảnh: Phát triển các thuật toán xử lý ảnh như tăng/giảm chất lượng ảnh, lọc nhiễu...
 - Nhận diện mẫu: Giải thích các kỹ thuật khác nhau để phân loại mẫu.
 - Quang trắc: Liên quan đến việc thu thập các số đo chính xác từ hình ảnh



6



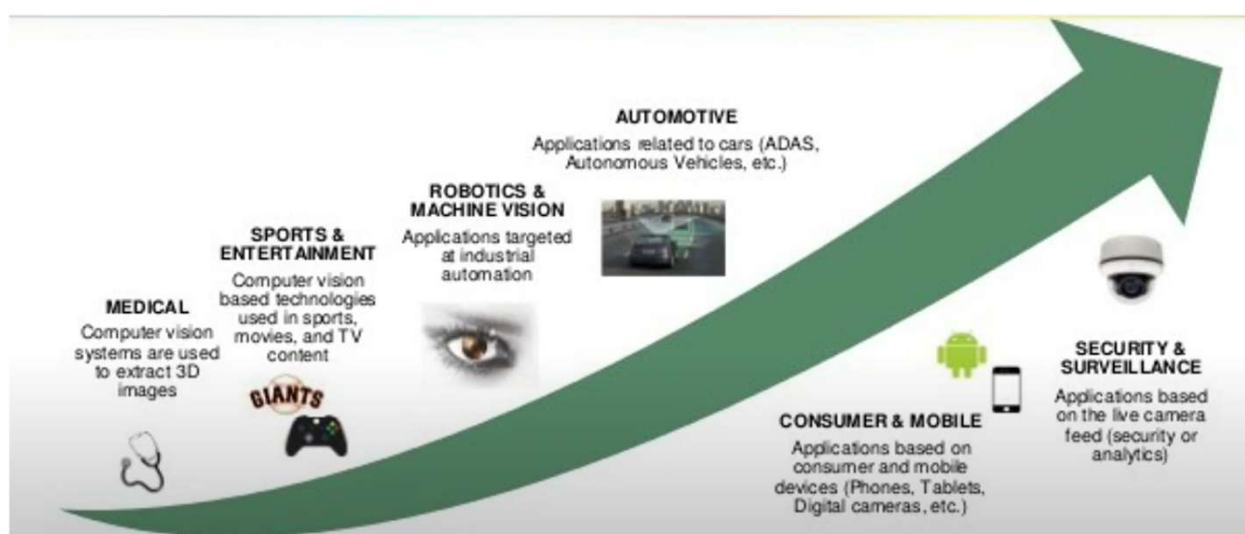
Ứng dụng của thị giác máy tính



7



Ứng dụng của thị giác máy tính



8



Ứng dụng của thị giác máy tính

- Xe tự hành (Autonomous Vehicles)
- Giám sát an ninh (CCTV)
- Y tế (chẩn đoán hình ảnh)
- Công nghiệp (kiểm tra lỗi sản phẩm)
- Nông nghiệp (đếm cây, phát hiện sâu bệnh)

9



Ứng dụng của thị giác máy tính

- Trong y học, các thuật toán xử lý ảnh cho phép biến đổi hình ảnh được tạo ra từ nguồn bức xạ X-ray hay nguồn bức xạ siêu âm hình ảnh quang học trên bề mặt phim x-quang hoặc trực tiếp lên bề mặt màn hình hiển thị. Hình ảnh các cơ quan chức năng của con người được xử lý tiếp nâng cao độ tương phản, lọc, tách các phần cần thiết hay tạo ra hình ảnh không gian 3 chiều (siêu âm ba chiều). Mới nhất hiện nay, thị giác máy tính tạo ra ứng dụng giúp hỗ trợ thị giác cho người mù...
- Trong lĩnh vực địa chất, kỹ thuật làm nổi đường biên và khôi phục hình ảnh giúp nâng cao chất lượng vệ tinh tạo ra bản đồ địa hình 3D với độ chính xác cao.
- Trong khí tượng học, ảnh thông qua vệ tinh, đưa ra các thông tin Trái Đất trên một vùng rộng lớn, giúp cho việc dự báo thời tiết chính xác hơn.

10



Ứng dụng của thị giác máy tính

- Trong lĩnh vực hình sự và bảo mật, kiểm soát truy cập, trong quá trình xử lý nhận dạng vân tay, khuôn mặt hay nhận dạng biển số xe giúp phát hiện nhanh đối tượng nghi vấn, cũng như nâng cao quá trình bảo mật, giám sát hay nhận dạng mục tiêu trong lĩnh vực quân sự.
- Không chỉ thế, thị giác máy tính ứng dụng trong việc điều khiển tiến trình như các robot trong công nghiệp, hay các thiết bị, xe tự hành hay trong các quá trình kiểm tra trong môi trường công nghiệp, ...
- Ngoài ra, thị giác máy tính còn có vai trò tương tác (đóng vai trò làm đầu vào cho thiết bị trong quá trình tương tác giữa người và máy)

11



Các bước chính trong một hệ thống thị giác

- Thu nhận ảnh/video.
- Tiền xử lý (lọc nhiễu, chuẩn hóa).
- Trích xuất đặc trưng.
- Nhận dạng và ra quyết định.

12



Các thư viện mã nguồn mở

▪ Chúng ta có thể tận dụng những thành quả của cộng đồng mã nguồn mở để áp dụng cho ứng dụng thị giác máy tính của mình. Hiện nay trên thế giới có rất nhiều thư viện mã nguồn mở cho ta sử dụng, như:

- Intel OpenCV
- CMCVision
- ImLib 3D
- ImLab...



Thư viện OpenCV

▪ OpenCV là một thư viện mã nguồn mở hàng đầu cho thị giác máy tính (computer vision), xử lý ảnh và máy học, và các tính năng tăng tốc GPU trong hoạt động thời gian thực





Thư viện OpenCV

OpenCV is an image processing library created by Intel and later supported by Willow Garage and now maintained by Itseez. opencv is available on Mac, Windows, Linux. Works in C, C++, and Python. It support cross-platform integration and it is flexible. It is Open Source and free. opencv is easy to use and install.

25



Thư viện OpenCV

- OpenCV được phát hành theo giấy phép BSD, do đó nó hoàn toàn miễn phí cho cả học thuật và thương mại, được sử dụng trên khắp thế giới
- Có các interface C++, C, Python, Java và hỗ trợ Windows, Linux, Mac OS, iOS và Android.
- OpenCV được thiết kế để tính toán hiệu quả và với sự tập trung nhiều vào các ứng dụng thời gian thực.
- Phạm vi sử dụng từ nghệ thuật tương tác, cho đến lĩnh vực khai thác mỏ, bản đồ trên web hoặc công nghệ robot.

26



Cài đặt

IDE (Integrated Development Environment)

Editor to write Code

Most of the IDE contain
PYTHON AS INTERNAL PACKAGE

**Choose IDE as
per your
compatibility**

27



Cài đặt



**Anaconda Navigator contains
so many libraries.**

28



Cài đặt

- Anaconda là một nền tảng phân phối ngôn ngữ lập trình python miễn phí, phục vụ cho tính toán khoa học và phát triển trí tuệ nhân tạo. Anaconda bao gồm ngôn ngữ lập trình python cộng với các thư viện và gói package cần thiết thường dùng, với mục đích quản lý và triển khai môi trường lập trình python dễ dàng hơn.
- Thông thường, để sử dụng Python, sau khi cài đặt Python, chúng ta cần phải cài đặt thêm các thư viện và các gói package cần thiết để chạy chương trình. Ví dụ như để phát triển AI bằng python, chúng ta cần cài đặt thêm Tensorflow, matplotlib, Open CV chẳng hạn. Tuy nhiên, nếu bạn cài đặt anaconda python, không những python mà các thư viện và gói package thường dùng cũng sẽ được cài đặt luôn theo đó.

29



Cài đặt



30



Cài đặt Anaconda

- Cài anaconda python individual edition
- <https://www.anaconda.com/products/individual>

Windows	MacOS	Linux
Python 3.9 64-Bit Graphical Installer (510 MB) 32-Bit Graphical Installer (404 MB)	Python 3.9 64-Bit Graphical Installer (515 MB) 64-Bit Command Line Installer (508 MB)	Python 3.9 64-Bit (x86) Installer (581 MB) 64-Bit (Power8 and Power9) Installer (255 MB) 64-Bit (AWS Graviton2 / ARM64) Installer (488 MB) 64-bit (Linux on IBM Z & LinuxONE) Installer (342 MB)

- Cài OpenCV:
 - Vào command prompt, gõ
 - `pip install opencv-python`
- Mở spyder, gõ `import cv2`
- Chạy chương trình để test

31



Cài đặt OpenCV

- Cài OpenCV:
- Vào command prompt, gõ
- `pip install opencv-python`

- Mở spyder, gõ `import cv2`
- Chạy chương trình để test



```
(base) C:\Users\LENOVO>pip install opencv-python
Collecting opencv-python
  Downloading opencv_python-4.5.5.62-cp36-abi3-win_amd64.whl (35.4 MB)
    | 35.4 MB 3.3 MB/s
Requirement already satisfied: numpy>=1.14.5 in g:\anaconda\lib\site-packages (from opencv-python) (1.20.3)
Installing collected packages: opencv-python
Successfully installed opencv-python-4.5.5.62
```

32



Khái niệm cơ bản về ảnh số

- Ảnh dưới đây là ảnh đen-trắng hay còn gọi là ảnh xám (Grayscale)



- Khi phóng to hơn nữa có thể thấy hình ảnh bị méo mó và thấy một số ô vuông nhỏ trên hình ảnh này → Những ô nhỏ này được gọi là **Pixel (điểm ảnh)**



33



- Máy tính lưu trữ hình ảnh dưới dạng số
- Mỗi pixel này được biểu thị bằng một giá trị số, và những con số này được gọi là **Giá trị Pixel**.
- Các giá trị pixel này biểu thị cường độ của các pixel.
- Đối với ảnh xám, chúng ta có giá trị pixel từ 0 đến 255.
- Giá trị 0: màu đen, giá trị 255: màu trắng

0	2	15	0	0	11	10	0	0	0	0	9	9	0	0	0
0	0	0	4	60	157	236	255	255	177	95	61	32	0	0	29
0	10	16	119	238	255	244	245	243	250	249	255	222	103	10	0
0	14	170	255	255	244	254	255	253	245	255	249	253	251	124	1
2	98	255	228	255	251	254	211	141	116	122	215	251	238	255	49
13	217	243	255	155	33	226	52	2	0	10	13	232	255	255	36
16	229	252	254	49	12	0	0	7	7	0	70	237	252	235	62
6	141	245	255	212	25	11	9	3	0	115	236	243	255	137	0
0	87	252	250	248	215	60	0	1	121	252	255	248	144	6	0
0	13	113	255	255	245	255	182	181	248	252	242	208	36	0	19
1	0	5	117	251	255	241	255	247	255	241	162	17	0	7	0
0	0	0	4	58	251	255	246	254	253	255	120	11	0	1	0
0	0	4	97	255	255	255	248	252	255	244	255	182	10	0	4
0	22	206	252	246	251	241	100	24	118	255	245	255	194	9	0
0	111	255	242	255	158	24	0	0	6	39	255	232	230	56	0
0	218	251	250	137	7	11	0	0	0	2	62	255	250	125	3
0	173	255	255	101	9	20	0	13	3	13	182	251	245	61	0
0	107	251	241	255	230	98	55	19	118	217	248	253	255	52	4
0	18	146	250	255	247	255	255	255	249	255	240	255	125	0	5
0	0	23	113	215	255	250	248	255	255	248	248	118	14	12	0
0	0	6	1	0	52	153	233	255	252	147	37	0	0	4	1
0	0	5	5	0	0	0	0	0	14	1	0	6	6	0	0

34



- Mọi hình ảnh trong máy tính đều được lưu dưới dạng này, trong đó bạn có một ma trận số và ma trận này còn được gọi là kênh (channel)

```

0 2 15 0 0 11 10 0 0 0 0 9 9 0 0 0
0 0 0 4 60 157 236 255 255 177 95 61 32 0 0 29
0 10 16 119 238 255 244 245 243 250 249 255 222 103 10 0
0 14 170 255 255 244 254 255 253 245 255 249 253 251 124 1
2 98 255 228 255 251 254 211 141 116 122 215 251 238 255 49
13 217 243 255 155 33 226 52 2 0 10 13 232 255 255 36
16 229 252 254 49 12 0 0 7 7 0 70 237 252 235 62
6 141 245 255 212 25 11 9 3 0 115 236 243 255 137 0
0 87 252 250 248 215 60 0 1 121 252 255 248 144 6 0
0 13 113 255 255 245 255 182 181 248 252 242 208 36 0 19
1 0 5 117 251 255 241 255 247 255 241 162 17 0 7 0
0 0 0 4 58 251 255 246 254 253 255 120 11 0 1 0
0 0 4 97 255 255 255 248 252 255 244 255 182 10 0 4
0 22 206 252 246 251 241 100 24 113 255 245 255 194 9 0
0 111 255 242 255 158 24 0 0 6 39 255 232 230 56 0
0 218 251 250 137 7 11 0 0 0 2 62 255 250 125 3
0 173 255 255 101 9 20 0 13 3 13 182 251 245 61 0
0 107 251 241 255 230 98 55 19 118 217 248 253 255 52 4
0 18 146 250 255 247 255 255 255 249 255 240 255 129 0 5
0 0 23 113 215 255 250 248 255 255 248 248 118 14 12 0
0 0 6 1 0 52 153 233 255 252 147 37 0 0 4 1
0 0 5 5 0 0 0 0 0 14 1 0 6 6 0 0

```



Ảnh và ma trận ảnh

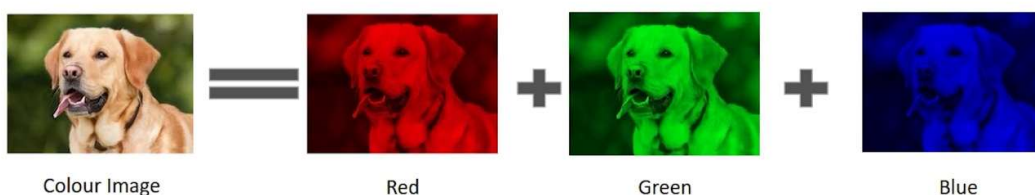
Khái niệm cơ bản

- Hình ảnh được lưu trữ trong máy tính dưới dạng ma trận số được gọi là giá trị pixel.
- Các giá trị pixel này biểu thị cường độ của từng pixel.
- Trong ảnh thang độ xám, giá trị pixel 0 biểu thị màu đen và 255 biểu thị màu trắng.
- Kênh là một ma trận các giá trị pixel.
- Ảnh xám chỉ có một kênh.



Lưu trữ ảnh màu trong máy tính

- Các màu đều có thể được tạo ra từ ba màu cơ bản: **Đỏ, Lục và Lam (Red, Green, Blue)**.
- Mỗi hình ảnh màu là một sự kết hợp của ba màu, hay 3 kênh - Đỏ, Lục và Lam: ảnh RGB



37



Ma trận biểu diễn ảnh màu

- Trong một ảnh màu, số lượng ma trận hoặc số lượng kênh sẽ nhiều hơn.
- Ta có 3 ma trận:
 - 1 ma trận cho màu đỏ, được gọi là kênh Red
 - 1 ma trận cho màu xanh lá cây, được gọi là kênh Green
 - 1 ma trận cho màu xanh lam, được gọi là kênh Blue
- Mỗi giá trị của từng ma trận có thuộc khoảng từ 0 đến 255, trong đó mỗi số biểu thị cường độ của điểm ảnh. Hay nói cách khác, các giá trị này biểu thị các sắc thái khác nhau của màu đỏ, lục và lam.
- Tất cả các kênh hoặc ma trận này chồng lên nhau để tạo thành hình dạng của hình ảnh khi được tải vào máy tính.
- Máy tính sẽ đọc hình ảnh này như sau: $M \times N \times 3$ (M: chiều cao, N: chiều rộng, 3: số kênh)



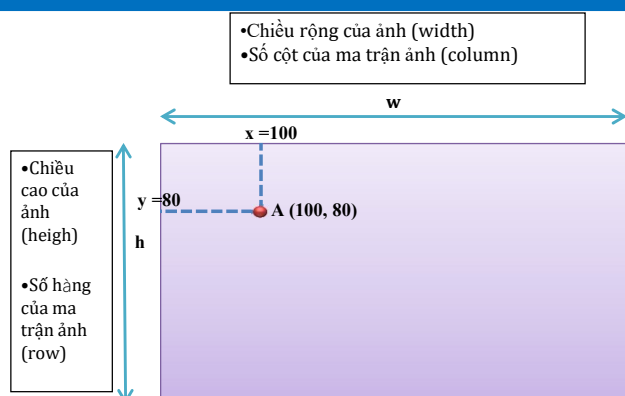
38



39



Giá trị điểm ảnh



- Ảnh (image): chiều rộng (w), chiều cao (h), độ sâu (d) - ảnh màu $d=3$, ảnh xám và đen trắng $d=2$
- Ma trận ảnh (img): số hàng ($=h$), số cột ($=w$), số chiều ($=d$)
- 1 điểm ảnh (pixel): A(100, 80) có màu đỏ: $x = 100, y = 80$ tương đương với 1 giá trị trong ma trận $\text{img}[80,100] = (0,0,255)$

$\text{img}[0,0] = 231$
 $\text{img}[0,1] = 221$
 $\text{img}[1,0] = 12$
 $\text{img}[2,3] = ?$
 $\text{img}[4,3] = ?$
 $\text{img}[3,4] = ?$

$\text{img} = \begin{bmatrix} 231 & 221 & 220 & 120 & 211 & 235 & 255 & 255 & 220 & 220 & 220 \\ 12 & 0 & 0 & 100 & \dots & \dots & \dots & \dots & 220 & 80 & 25 \\ 34 & 10 & 12 & 120 & \dots & \dots & \dots & \dots & 212 & 90 & 30 \\ 34 & 20 & 35 & 170 & \dots & \dots & \dots & \dots & 245 & 67 & 42 \\ 35 & 0 & 0 & 180 & \dots & \dots & \dots & \dots & 124 & 77 & 70 \end{bmatrix}$

40



Đọc hiện ảnh

- Đọc ảnh: `cv2.imread(filename, flags)`
 - flags: `cv2.IMREAD_UNCHANGED` or `-1`
`cv2.IMREAD_GRAYSCALE` or `0`
`cv2.IMREAD_COLOR` or `1`
- Hiện ảnh: `cv2.imshow(window_name, image)`

`import cv2`

```
img = cv2.imread("đường dẫn") # "C:\\Picture\\Apple.jpg"
cv2.imshow("Picture Apple",img)
cv2.waitKey(0)
cv2.destroyAllWindows()
```

46



Đọc hiện ảnh

- **Đọc ảnh:** `img = cv2.imread('đường dẫn ảnh')`

Đọc ảnh xám: `img = cv2.imread('đường dẫn',0)`

VD: `img=cv2.imread('C:/Pic/anh1.jpg')` → xược phải chỉ cần 1 xược

`img=cv2.imread('C:\\Pic\\anh1.jpg')` → xược trái cần 2 xược

Hoặc `img=cv2.imread(r'C:\\Pic\\anh1.jpg')` → xược trái 1 xược phải có thêm chữ `r` ở trước dấu nháy

- **Hiện ảnh:** `cv2.imshow('Tên cửa sổ', img)`
- **`cv2.waitKey(0):`** dừng chờ ấn phím; số `0`: dừng chờ đến khi nào ấn phím, `cv2.waitKey(10)`: chờ 10ms
- **`cv2.destroyAllWindows():`** đóng cửa sổ

47



Ví dụ đọc hiện ảnh

```
import cv2
img = cv2.imread("earth.jpg")
cv2.imshow("Demo", img)
cv2.waitKey(0)
cv2.destroyAllWindows()
```

48



Lưu ảnh

- `cv2.imwrite(filename, image)`
- Ví dụ:
 - `cv2.imwrite('output.jpg',img)`

49



Lấy thông tin của ảnh

- Kích thước ảnh:

- Ảnh màu: `h, w, d = img.shape[]`

Nếu chỉ lấy chiều cao, chiều rộng: `h, w = img.shape[:2]`

- Ảnh xám: `h, w = img.shape` (chỉ có 2 giá trị, không có giá trị `d` vì ảnh xám chỉ 1 kênh màu)

- Thông thường ta chỉ lấy chiều cao, chiều rộng, vì vậy thường dùng câu lệnh:

`h, w = img.shape[:2]`

Câu lệnh này đúng cho cả ảnh màu và ảnh xám

- Lấy giá trị màu tại 1 điểm ảnh: `img[100,120]` → Lấy giá trị màu tại ô ở hàng 100, cột 120

50



Đọc video

- Video là chuỗi các ảnh liên tiếp

```
import cv2
```

```
cap = cv2.VideoCapture(r"D:\Video\video1.mp4")
```

```
while True:
```

```
    ret, frame = cap.read()
```

```
    cv2.imshow('Xem video', frame)
```

```
    if cv2.waitKey(10) == ord('q'):
```

```
        break
```

```
cap.release()
```

```
cv2.destroyAllWindows()
```

51



Các thuộc tính của video

- `fps = cap.get(5)` #CAP_PROP_FPS frame per second
- `print('Số hình trong 1 giây:', fps)`
- `frame_count = cap.get(7)` #CAP_PROP_FRAME_COUNT
- `print('Số khung hình:', frame_count)`

52



Đọc video từ webcam

- `cap= cv2.VideoCapture(0)`
- Nếu có nhiều camera, thay đổi 0 thành 1, 2,...

53



Hệ màu HSV

- Hệ màu HSV dùng để xử lý màu rất thuận tiện
- Mỗi màu sẽ có 1 giá trị Hue $[0;360]$, còn Value và Saturation chỉ thị độ sáng và độ bão hoà của màu. Hệ màu này được ứng dụng để theo dõi đối tượng theo màu
- Hệ màu HSB tương tự HSV nhưng Value được thay bằng Brightness

56



Chuyển đổi hệ màu

- Trong xử lý ảnh, việc chuyển đổi ảnh màu sang ảnh xám là công việc vô cùng phổ biến.
- Ảnh màu thực chất chỉ là tập hợp của những ma trận số có cùng kích thước. Khi muốn xử lý thông tin trên ảnh, sẽ dễ dàng hơn nếu ta chỉ xử lý dữ liệu trên một ma trận số thay vì nhiều ma trận số. Việc biến đổi ảnh màu về ảnh số (Grayscale converting) xuất hiện vì mục đích trên - biến đổi thông tin ảnh về một ma trận số hai chiều duy nhất.
- Ảnh đen trắng thường được ứng dụng trong bài toán phân vùng ảnh (Image segmentation). Ví dụ, bằng cách biến đổi ảnh về nhị phân, có thể loại bỏ các thông tin khung cảnh xung quanh và chỉ giữ lại thông tin cần thiết trên hình.

57



Hàm chuyển đổi màu ảnh

- Đổi ảnh màu sang ảnh xám:

```
image = cv2.cvtColor(src, cv2.COLOR_BGR2GRAY)
```

- Đổi màu BGR sang HSV:

```
image = cv2.cvtColor(src, cv2.COLOR_BGR2HSV)
```

58



Thư viện Numpy

- Numpy (Numeric Python): là một thư viện toán học phổ biến và mạnh mẽ của Python. Cho phép làm việc hiệu quả với ma trận và mảng, đặc biệt là dữ liệu ma trận và mảng lớn với tốc độ xử lý nhanh hơn nhiều lần khi chỉ sử dụng “core Python” đơn thuần.

- Cài đặt thư viện Numpy: Mở Command Prompt và gõ lệnh: `pip install numpy`

- Khai báo thư viện: `import numpy as np`

- Khởi tạo mảng một chiều với kiểu dữ liệu các phần tử là Integer

```
– arr = np.array([1,3,4,5,6], dtype = int)
```

- Khởi tạo mảng hai chiều

```
– arr1 = np.array([(4,5,6), (1,2,3)], dtype = int)
```

- Tạo mảng hai chiều các phần tử 0 với kích thước 3x4

```
– np.zeros((3,4), dtype = int)
```

60



Lấy kích thước ảnh

Kiểm tra ảnh đọc vào là ảnh màu hay ảnh xám

```
import cv2
```

```
img = cv2.imread("G:\\Codes\\rgb.jpg",1)
```

```
try:
```

```
    (h, w, d) = img.shape
```

```
    print("width={}, height={}, depth={}".format(w, h, d))
```

```
    print('anh màu')
```

```
except:
```

```
    print('anh xám')
```

```
    (h, w) = img.shape
```

```
    print("width={}, height={}".format(w, h))
```

61



Một số hàm thuộc tính ảnh

- Tách màu ảnh: `b,g,r = cv2.split(img)`
- Gộp màu ảnh: `img = cv2.merge((b,g,r))`

62



Bài tập

1. Đọc ảnh màu, chuyển sang ảnh xám

Hiện các thuộc tính của ảnh: kích thước ảnh

Lưu ảnh

2. Đọc và hiện video, điều chỉnh tốc độ

Hiện các tham số của video: số khung hình trên giây, tổng khung hình, tốc độ phát

3. Đọc nhiều ảnh: đọc và hiện tất cả các ảnh có trong 1 thư mục

4. Đọc chuỗi ảnh: đọc và hiện tất cả các ảnh có trong 1 thư mục có cùng tên, chỉ khác số thứ tự. Ví dụ: Trong thư mục picture có các ảnh cat01.jpg, cat02.jpg, cat03.jpg, dog01.jpg, dog02.jpg. Chỉ đọc các ảnh có tên đầu là 'cat'



Tham khảo

- <https://www.geeksforgeeks.org/opencv-python-tutorial/?ref=lbp>
- https://docs.opencv.org/3.4/d2/d96/tutorial_py_table_of_contents_imgproc.html

Thank you!



TRƯỜNG ĐẠI HỌC VINH
VINH UNIVERSITY

Nơi tạo dựng tương lai cho tuổi trẻ

