



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 - 2021

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

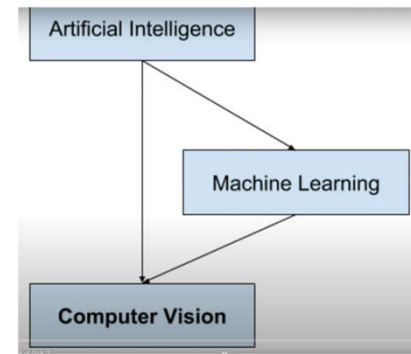
NGUYỄN THỊ MINH TÂM - VIỆN KTCN - ĐH VINH

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
- Thị giác máy tính là một môn học khoa học liên quan đến lý thuyết đằng sau các hệ thống nhân tạo có trích xuất các thông tin từ các hình ảnh. Dữ liệu hình ảnh có thể nhiều dạng, chẳng hạn như chuỗi video, các cảnh từ đa camera, hay 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.



3



Computer Vision

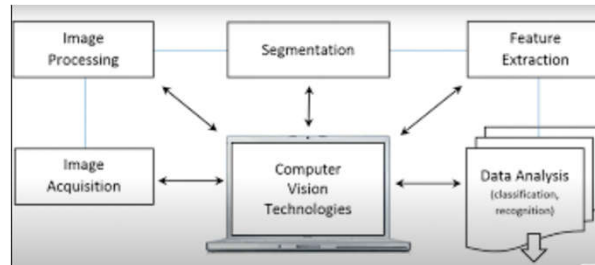
Computer vision is an interdisciplinary scientific field that deals with how computers can gain high-level understanding from digital images or videos. From the perspective of engineering, it seeks to understand and automate tasks that the human visual system can do

4

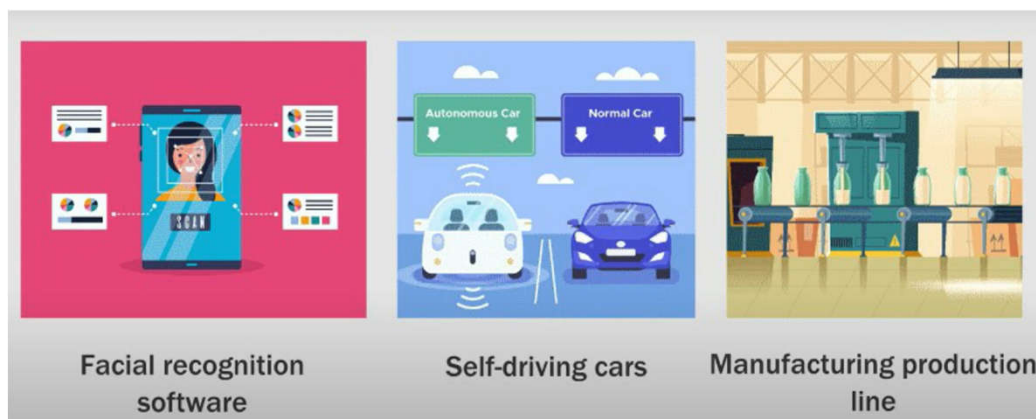


Các lĩnh vực nghiên cứu của thị giác máy tính

- 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



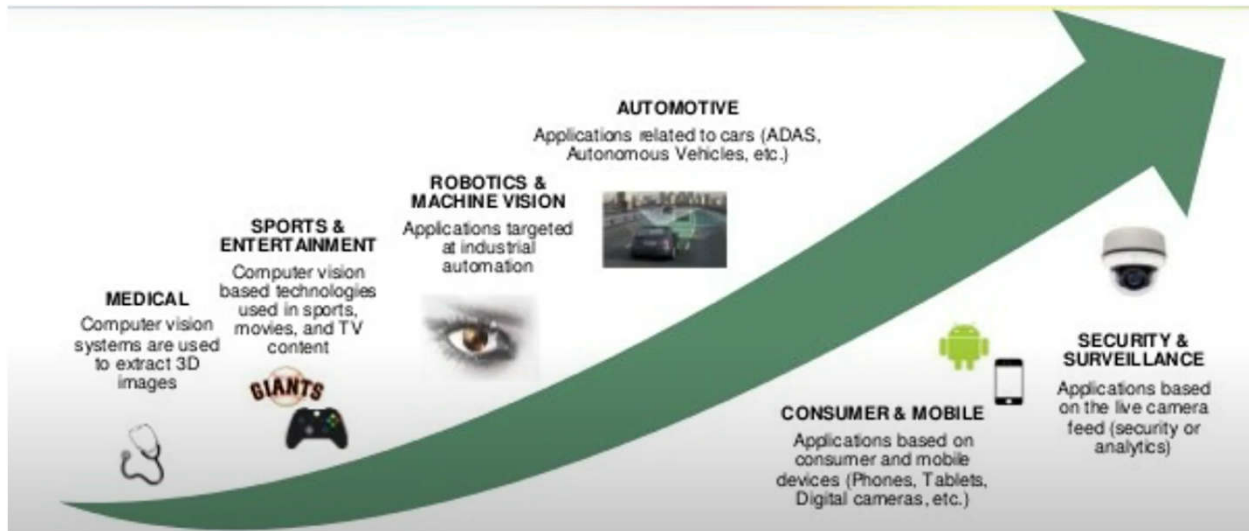
5



6



Ứng dụng của thị giác máy tính



7



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.

8



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)

9



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
 - ImaLab....

10



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



11



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.

12



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.

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.

13



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**

14



Cài đặt



**Anaconda Navigator contains
so many libraries.**

15



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.

16



Cài đặt



17



Cài đặt Anaconda

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

Windows	MacOS	Linux
Python 3.9	Python 3.9	Python 3.9
64-Bit Graphical Installer (510 MB)	64-Bit Graphical Installer (515 MB)	64-Bit (x86) Installer (581 MB)
32-Bit Graphical Installer (404 MB)	64-Bit Command Line Installer (508 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 (242 MB)

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

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

18



Cài đặt OpenCV

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



CMD.exe Prompt

0.1.1

Run a cmd.exe terminal with your current environment from Navigator activated

Launch

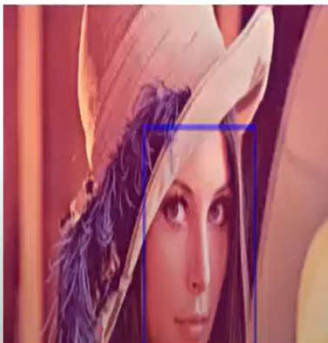
```
(base) C:\Users\LENOVO>pip install opencv-python
Collecting opencv-python
  Downloading opencv_python-4.5.5.62-cp36-abi3-win_amd64.whl (35.4 MB)
    | 35.4 MB 3.3 MB/s
Requirement already satisfied: numpy>=1.14.5 in g:\anaconda\lib\site-packages (from opencv-python) (1.20.3)
Installing collected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

19



Ảnh và ma trận

- Trong OpenCV, sử dụng mảng 2 chiều để lưu trữ ảnh
- Thư viện Numpy dùng để xử lý mảng



$i \rightarrow$

62	79	23	119	120	105	4	0
10	10	9	62	12	78	34	0
10	58	197	46	46	0	0	48
176	135	5	188	191	68	0	49
2	1	1	29	26	37	0	77
0	89	144	147	187	102	62	208
255	252	0	166	123	62	0	31
166	63	127	17	1	0	99	30

$j \downarrow$

20



Thư viện Numpy

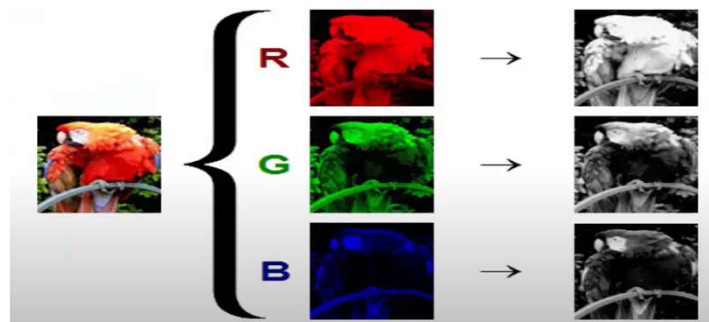
- Numpy (Numeric Python): là một thư viện toán học phổ biến và mạnh mẽ của Python. Cho phép làm việc hiệu quả với ma trận và mảng, đặc biệt là dữ liệu ma trận và mảng lớn với tốc độ xử lý nhanh hơn nhiều lần khi chỉ sử dụng “core Python” đơn thuần.
- Cài đặt thư viện Numpy: Mở Command Prompt và gõ lệnh: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

21



Màu sắc của ảnh

- Màu được tổ hợp từ 3 màu RGB (Red, Green, Blue)
- Trong OpenCV, thứ tự màu là BGR (Blue, Green, Red)



22



Màu sắc của ảnh

- Ảnh xám: 1 kênh màu (0-255)
- Ảnh màu: 3 kênh màu



SINGLE CHANNEL

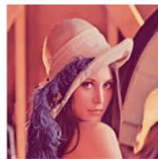


3- CHANNEL

23



Màu sắc của ảnh



$$Pixel = \begin{cases} R \\ G \\ B \end{cases}$$

$$Pixel = 0.2989 * R + 0.5870 * G + 0.1140 * B$$



$$Pixel = \begin{cases} 0, \text{nếu } Pixel < T \\ 1, \text{nếu } Pixel \geq T \end{cases}$$



24



Bài tập

- Bài tập 1: Viết chương trình chuyển ảnh màu sang ảnh xám, ảnh đen trắng (ảnh nhị phân)

25



Đọc hiện ảnh

- Đọc ảnh: `cv2.imread(filename)`
- Đọc ảnh xám: `cv2.imread(filename, 0)`
- Hiện ảnh: `cv2.imshow(window_name, image)`
- Ví dụ:


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

26



```
import cv2
cv2.namedWindow("output", cv2.WINDOW_AUTOSIZE)
im = cv2.imread("earth.jpg")
imS = cv2.resize(im, (960, 540))
cv2.imshow("output", imS)
cv2.waitKey(0)
```

27



Lưu ảnh

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

28



Đọc video

```
import cv2
cap = cv2.VideoCapture(r'đường dẫn')
while True:
    ret, frame = cap.read()
    cv2.imshow('Xem video', frame)
    if cv2.waitKey(20) == ord('q'): break
cap.release()
cv2.destroyAllWindows()
```

29



Một số thuộc tính của video:

```
print('Số hình trong 1 giây:', cap.get(5))
print('Tổng số khung hình:', cap.get(7))
```

30



Đọc video

```
import cv2
vid_capture =
cv2.VideoCapture('video_demo.mp4')
if (vid_capture.isOpened() == False):
    print("Lỗi mở file")
else:
    fps = vid_capture.get(5)
    #CAP_PROP_FPS frame per second
    print('Số hình trong 1 giây:', fps)
    frame_count = vid_capture.get(7)
    #CAP_PROP_FRAME_COUNT
    print('Số khung hình:', frame_count)
```

```
while(vid_capture.isOpened()):
    ret, frame= vid_capture.read()
    if ret == True:
        cv2.imshow('Frame', frame)
        key = cv2.waitKey(20)
        if key == ord('q'):
            break
    else:
        break
vid_capture.release()
cv2.destroyAllWindows()
```

31



Đọc chuỗi ảnh

- Sử dụng ký hiệu %nd
- Ví dụ:

```
vid_capture = cv2.VideoCapture('Resources/Image_sequence/Cars%04d.jpg')
```

- Cars%04d.jpg: trong đó %04d chỉ ra quy ước đặt tên theo dãy gồm bốn chữ số (ví dụ: Cars0001.jpg, Cars0002.jpg, Cars0003.jpg, ...).
- Nếu “Race_Cars_%02d.jpg” thì sẽ tìm kiếm các tệp có dạng: (Race_Cars_01.jpg, Race_Cars_02.jpg, Race_Cars_03.jpg, ...).

32



Đọc video từ webcam

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

33



Ghi Video

- `VideoWriter(filename, apiPreference, fourcc, fps, frameSize[, isColor])`
- `apiPreference`: Mã định danh phụ trợ API
- `fourcc`: Mã 4 ký tự của codec, được sử dụng để nén khung hình
- AVI: ('M','J','P','G')
- MP4: (*'XVID')
- `fps`: Tốc độ khung hình của luồng video đã tạo
- `frame_size`: Kích thước của khung video
- `isColor`: Nếu không phải là 0, bộ mã hóa sẽ mong đợi và mã hóa các khung hình màu. Nếu không, nó sẽ hoạt động với các khung hình xám.

```
output = cv2.VideoWriter('Resources/output_video_from_file.avi',
                        cv2.VideoWriter_fourcc('M','J','P','G'), 20, frame_size)
```

34



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

36



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.

37



Hàm chuyển đổi màu ảnh

- Đổi sang ảnh xám:

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

Convert

- Đổi màu BGR sang HSV:

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

38



Một số hàm thuộc tính ảnh

- Kích thước ảnh: `print(img.shape)`
- Nếu ảnh xám, trả về bộ gồm số hàng và số cột.
- Nếu ảnh màu, trả về bộ 3 gồm chiều cao, chiều rộng, số kênh màu (height*width*channels)
- VD:


```
img = cv2.imread('example.jpg')
(h, w, d) = img.shape
print("width={}, height={}, depth={}".format(w, h, d))
```

 - Kết quả: width=580, height=326, depth=3
- Nếu chỉ lấy chiều cao, chiều rộng: `h,w = img.shape[:2]`
- Tổng số điểm ảnh: `print(img.size)`

40



Ví dụ

Viết chương trình đọc ảnh màu và thực hiện:

- Chuyển ảnh màu sang ảnh xám. Hiện ảnh màu và ảnh xám.
- In ra chiều rộng và chiều cao của ảnh

41



Lấy giá trị màu ở một điểm ảnh

```
(B, G, R) = img[50, 50]
print('Blue ', B, 'Green ', G, 'Red ', R)
#Hoặc print('B = {}, G = {}, R = {}'.format(B, G, R))
output: B=96, G=100, R=111
```

43



Cắt vùng ảnh

▪ Trong OpenCV không có hàm cắt ảnh. Việc cắt ảnh là lấy một phần của ma trận ảnh

▪ Ví dụ:

```
import cv2
img = cv2.imread(r'C:/Documents/anhhoa.jpg')
crop = img[30:300, 50:450]
cv2.imshow('anh goc',img)
cv2.imshow('anh crop',crop)
cv2.waitKey(0)
cv2.destroyAllWindows()
```

44



Bài tập

- 1. Đọc ảnh màu, chuyển sang ảnh xám
- Hiện các thuộc tính của ảnh: kích thước ảnh
- Cắt 1 phần ảnh (dùng ma trận: lấy 1 số hàng và cột của ma trận ảnh gốc)
- 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 chuỗi ảnh: lưu các file ảnh với tên theo định dạng để đọc khoảng 10 ảnh.

45



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

46

Thank you!



**TRƯỜNG ĐẠI HỌC VINH
VINH UNIVERSITY**

Nơi tạo dựng tương lai cho tuổi trẻ



48