**HỌC VIỆN CÔNG NGHỆ BƯU CHÍNH VIỄN THÔNG**

**CƠ SỞ TẠI THÀNH PHỒ HỒ CHÍ MINH**

**KHOA CÔNG NGHỆ THÔNG TIN**

**A picture containing text

Description automatically generated**

**ĐỒ ÁN HỌC PHẦN**

TÊN HỌC PHẦN:   
**KỸ THUẬT LẬP TRÌNH ỨNG DỤNG ĐA PHƯƠNG TIỆN   
(Multimedia Application Programming Techniques)**

MÃ SỐ LỚP HP: **MUL14146\_01**

Tên đề tài: **LẬP TRÌNH PHÁT TRIỂN HỆ THỐNG QUẢN LÝ VÀ XỬ LÍ MULTIMEDIA** **GIỐNG/ LOÀI BƯỚM DÙNG CV2 VỚI PYTHON**

Họ tên sinh viên: **Phan Huế Anh**

**Mã số sinh viên: N17DCPT002**

**Lớp: D17CQPU01-N**

**Ngày nộp: 02/12/2021**

**Ký tên:**

**TP.HCM, ngày tháng năm 2021**

**HỌC VIỆN CÔNG NGHỆ BƯU CHÍNH VIỄN THÔNG**

**CƠ SỞ TẠI THÀNH PHỒ HỒ CHÍ MINH**

**KHOA CÔNG NGHỆ THÔNG TIN**

**ĐỀ TÀI ĐỒ ÁN HỌC PHẦN**

**Giảng viên giảng dạy: VÕ XUÂN THỂ**

TÊN HỌC PHẦN: **KỸ THUẬT LẬP TRÌNH ỨNG DỤNG ĐA PHƯƠNG TIỆN   
 (Multimedia Application Programming Techniques)**

MÃ SỐ LỚP HP: **MUL14146\_01**

Tên đề tài: **LẬP TRÌNH PHÁT TRIỂN HỆ THỐNG QUẢN LÝ VÀ XỬ LÍ MULTIMEDIA** **GIỐNG/ LOÀI BƯỚM DÙNG CV2 VỚI PYTHON**

Họ tên sinh viên: **Phan Huế Anh**

**Mã số sinh viên: N17DCPT002 [07]**

**Lớp: D17CQPU01-N**

**Tên sản phẩm đề tài: 07.PhanHueAnh.DAHP.MP\_Butterflys**

**Công cụ sử dụng (phiên bản): Spyder 5.1.5 (Anaconda3)**

**Thư viên (phiên bản): cv2 4.5.3**

**Ngôn ngữ lập trình sử dụng: Python 3.8.3 64-bit | Qt 5.9.7 | PyQt5 5.9.2**

**Tập dữ liệu thực nghiệm:**

. Hình ảnh: <https://farm7.static.flickr.com/6179/6218417201_acac5de168_o.jpg>

<https://live.staticflickr.com/6204/6162182582_4684c675fe_b.jpg>

<https://i.pinimg.com/originals/27/b4/b5/27b4b53e73907d976d07264971a2ad91.jpg>

. Video Clips: <https://www.youtube.com/watch?v=h_glLV23Z1M>

. Audio: cach\_nap\_file.mp3, chon\_chuc\_nang.mp3, chon\_chuc\_nangEn.mp3, Kichthuocanh.mp3,…

**Nhận xét của giảng viên:**

**………………………………………………………..**

**Điểm đánh giá:………….(…………………)**

**Ngày……./……../2021**

**Giảng viên: Ký tên**

**TP.HCM, ngày tháng năm 2021**

# LỜI CẢM ƠN

Trước tiên với tình cảm sâu sắc và chân thành nhất, em xin bày tỏ lòng biết ơn đối với thầy cô và bạn bè ở trường Học viện Công nghệ Bưu chính Viễn thông đã đồng hành và giúp đỡ em trong suốt quá trình học tập. Trong thời gian từ khi bắt đầu học tập tại trường đến nay em đã nhận nhận được nhiều kiến thức mới lạ, sự quan tâm và lời khuyên bổ ích cho sự phát triển tương lai sau này.

Đặc biệt em xin chân thành cảm ơn thầy Võ Xuân Thể đã hỗ trợ rất tận tình trong suốt thời gian học tập và thực hiện đồ án để kết thúc học phần. Tuy sản phẩm còn nhiều thiếu sót nhưng nhờ có sự hướng dẫn của thầy nên quá trình thực hiện có nhiều suôn sẻ và hoàn thiện tốt đẹp.

Bước đầu thực hiện đồ án không tránh khỏi những thiếu sót, em rất mong nhận được những ý kiến đóng góp quý báu để kiến thức và năng lực của em trong lĩnh vực này được hoàn thiện.

Một lần nữa, em xin cảm ơn thầy, người đã trực tiếp giúp đỡ, quan tâm, hướng dẫn em hoàn thành tốt bài báo cáo này trong thời gian qua.

Em xin chân thành cảm ơn!

# DANH MỤC CÁC TỪ VIẾT TẮT

|  |  |
| --- | --- |
|  | |
| PYTHON | Ngôn ngữ lập trình Python |
| RGB | Red, green, blue |
|  |  |
| Spyder(anaconda 3) | Công cụ lập trình |

# MỤC LỤC

[LỜI CẢM ƠN 3](#_Toc89444466)

[DANH MỤC CÁC TỪ VIẾT TẮT 4](#_Toc89444467)

[MỤC LỤC 5](#_Toc89444468)

[PHIẾU GIAO NHIỆM VỤ ĐỀ TÀI ĐỒ ÁN HỌC PHẦN 7](#_Toc89444469)

[Chương 1: GIỚI THIỆU VỀ ĐỀ TÀI 9](#_Toc89444470)

[**1.1.** **Tổng quan về đề tài** 9](#_Toc89444471)

[**1.2.** **Nội dung chuyên môn chính của đề tài** 9](#_Toc89444472)

[**1.3.** **Bố cục của báo cáo** 9](#_Toc89444473)

[Chương 2: CƠ SỞ LÝ THUYẾT VỀ KỸ THUẬT LẬP TRÌNH ỨNG DỤNG ĐA PHƯƠNG TIỆN 11](#_Toc89444474)

[**2.1.** **Tổng quan về kỹ thuật multimedia (kỹ thuật đa phương tiện)** 11](#_Toc89444475)

[2.1.1. Kỹ thuật âm thanh (giọng nói) 11](#_Toc89444476)

[2.1.2. Kỹ thuật hình ảnh (image) 14](#_Toc89444477)

[2.1.3. Kỹ thuật video clips 16](#_Toc89444478)

[**2.2.** **Công cụ và các hệ thống thư viện hỗ trợ lập trình multimedia** 18](#_Toc89444479)

[2.2.1. Phần mềm công cụ sử dụng chính trong học phần 18](#_Toc89444480)

[2.2.2. Các thư viện lập trình chính được sử dụng trong học phần 19](#_Toc89444481)

[2.2.3. Tập các Multimedia mẫu được sử dụng trong đề tài của Học phần 19](#_Toc89444482)

[Chương 3: GIỚI THIỆU CÁC THƯ VIỆN MULTIMEDIA SỬ DỤNG 20](#_Toc89444483)

[**3.1.** **Nạp các thư viện** 20](#_Toc89444484)

[**3.2.** **Giới thiệu các thư viện được dùng** 20](#_Toc89444485)

[Chương 4: GIỚI THIỆU SẢN PHẨM 21](#_Toc89444486)

[**4.1.** **Mô tả sản phẩm** 21](#_Toc89444487)

[**4.3** **Giới thiệu các chức năng:** 21](#_Toc89444488)

[Chương 5: CHỌN NGÔN NGỮ VÀ CÁCH SỬ DỤNG 22](#_Toc89444489)

[**5.1.** **Chọn ngôn ngữ** 22](#_Toc89444490)

[5.2. Chọn cách sử dụng 22](#_Toc89444491)

[Chương 6: XỬ LÝ ÂM THANH VÀ GIỌNG NÓI 23](#_Toc89444492)

[6.1 Xử lý âm thanh 23](#_Toc89444493)

[6.2 Xử lý giọng nói 23](#_Toc89444494)

[Chương 7: NẠP HÌNH ẢNH VÀ VIVDEO TỪ FILE HOẶC NẠP VIDEO TỪ CAMERA , KẾT XUẤT CAMERA 26](#_Toc89444495)

[**7.1 Nạp hình ảnh từ file** 26](#_Toc89444496)

[7.2 Nạp video 26](#_Toc89444497)

[7.3.1 : Giới thiệu 30](#_Toc89444498)

[7.3.2: Code xử lý 30](#_Toc89444499)

[Chương 8: CÁC CHỨC NĂNG XỬ LÝ ẢNH 32](#_Toc89444500)

[**8.1** **Chức năng chính:** 32](#_Toc89444501)

[8.1.1 Giới thiệu: 32](#_Toc89444502)

[Chương 9: KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN CỦA ĐỀ TÀI 40](#_Toc89444503)

[**9.1** **Kết luận** 40](#_Toc89444504)

[9.1.1Những kết quả đạt được 40](#_Toc89444505)

[9.1.2Hạn chế 40](#_Toc89444506)

[**9.2** **Hướng phát triển** 40](#_Toc89444507)

[9.2.1 Hướng khắc phục các hạn chế 40](#_Toc89444508)

[9.2.2Hướng mở rộng đề tài 40](#_Toc89444509)

[DANH MỤC TÀI LIỆU THAM KHẢO 41](#_Toc89444510)

**HỌC VIỆN CÔNG NGHỆ BƯU CHÍNH VIỄN THÔNG**

**CƠ SỞ TẠI THÀNH PHỒ HỒ CHÍ MINH**

**KHOA CÔNG NGHỆ THÔNG TIN**

# PHIẾU GIAO NHIỆM VỤ ĐỀ TÀI ĐỒ ÁN HỌC PHẦN

|  |  |
| --- | --- |
| Giảng viên giảng dạy: **VÕ XUÂN THỂ** | |
| Tên học phần: **KỸ THUẬT LẬP TRÌNH ỨNG DỤNG ĐA PHƯƠNG TIỆN** Mã số lớp HP: **MUL14146\_01** | |
| Tên đề tài: **LẬP TRÌNH PHÁT TRIỂN HỆ THỐNG QUẢN LÝ VÀ XỬ LÍ MULTIMEDIA** **GIỐNG/ LOÀI BƯỚM DÙNG CV2 VỚI PYTHON.** | |
| Sinh viên thực hiện:: **Phan Huế Anh, N17DCPT002** | |
| Thời gian thực hiện:: **29/09/2021**  đến **10/11/2021** | |
| **Yêu cầu của đề tài**  **Lý thuyết:** Vận dụng kiến thức KỸ THUẬT LẬP TRÌNH ỨNG DỤNG ĐA PHƯƠNG TIỆN vào việc phát triển một hệ thống ứng dụng quản lý đa phương tiện dùng trong một lĩnh vực chuyên môn thực tiễn, minh họa với ngôn ngữ lập trình PYTHON dùng công cụ Spyder (Anaconda3) với một số thư viện hỗ trợ lập trình ứng dụng đa phương tiện, như: cv2, os, … tham khảo tập multimedia mẫu tại danh mục tài liệu tham khảo mục 2.  **Thực hành: Lập trình phát triển hệ thống quản lý và xử lí multimedia** **giống/ loài bướm dùng cv2 với python.**  gồm:  Media Input: có 2 trường hợp  + Nạp các file video hoặc hình ảnh về các loài bướm (có cả âm thanh)  + Quay video hoặc chụp trực tiếp các loài bướm từ camera của máy tính hoặc nạp từ file  Media Ouput: kết xuất thông tin loài bướm đó dưới dạng giọng nói, text và hình ảnh chính thức về loài bướm đó( hình ảnh rõ nhất về loài bướm đó được trích khung ảnh từ video hoặc các xử lý ảnh cần thiết từ đầu vào, ví dụ: ảnh màu -> ảnh trắng đen  Các xử lý cần thiết:   * Có thể cho phép người sử dụng ra lệnh bằng thao tác bằng giọng nói: lựa chọn phương pháp nạp media đầu vào hoặc chọn media đầu ra là dạng gì: cho phép người sử dụng lựa chọn ngôn ngữ sử dụng ít nhất 2 ngôn ngữ tiếng Anh và tiếng Việt cả dạng text và giọng nói * Cho phép trích 1 phần hình ảnh của loài bướm từ ảnh gốc: Cho phép trích 1 phần hình ảnh của loài bướm đó từ ảnh gốc…(việc xác định vị trí khung ảnh con có thể dùng chuột xác định trực tiếp trên hình thay vì nhập tọa độ) ): bao gồm cả việc nhận diện đặc trưng ảnh từ video hoặc khung thời gian của khung ảnh được trích xuất ra (nhận diện ảnh tốt nhất của loài bướm đó) * Giao diện có thể sử dụng GUI hoặc dạng text * Nguồn media đầu vào:   + Thư viện chính được dùng: cv2 #thư viện Vomputer Vision version 2  + Thiết kế và phát triển giao diện GUI dùng thư viện: Tkinter  + Xử lý trợ lý âm thanh (Voice Assistant)  speech\_recognition: thư viện để thực hiện nhân dạng giọng nói  playsound: thư viện dùng để phát âm thanh  + Xử lý ảnh: cv2  + Xử lý Video: cv2  + Các thư viện hỗ trợ khác:  . os # Thư viện os của MS. Windows = Lập thư mục & lưu các files Media  . numpy #Thư viện Mummeric Python: Lập ma trận giữ chỗ đề ghép các Images và Videos | |
| **GIẢNG VIÊN** | **Ngày 29 tháng 09 năm 2021**  **SV Thực hiện** |

# GIỚI THIỆU VỀ ĐỀ TÀI

## **Tổng quan về đề tài**

Tìm hiểu về kỹ thuật đa phương tiện (bao gồm các kỹ thuật về âm thanh, giọng nói, hình ảnh, video clip,..) đồng thời dựa trên công cụ lập trình Spyder (Anaconda3) với ngôn ngữ lập PYTHON, sử dụng các thư viện hỗ trợ kỹ thuật lập trình ứng dụng đa phương tiện (Multimedia Program), như: cv2, PlaySound, speech\_reconition,… để phát triển ứng dụng (App) dạng Winforms (tức là WinApp) cho phép quản lý và xử lý các tập tin multimedia sử dụng trong lập trình phát triển hệ thống quản lý và xử lí multimedia giống/ loài bướm dùng cv2 với python.

Tập dữ liệu thực nghiệm chính được tham khảo (thu thập) từ internet và một số files multimedia mẫu:

.Videos: buom.mp4, Bướm đêm lớn nhất thế giới (Attacus atlas) Bướm khế.webm, Thế giới các loài bướm - butterfly world.webm

. Images: buom\_dom\_xanh.jpg, buom\_khe.jpg, buom\_trang.jpg, chai\_xanh.jpg, duoi\_kiem.jpg, duoi\_la\_xanh.jpg, ho\_van.jpg, la\_kho.jpg, phan\_cam.jpg

. Audio: chon\_chuc\_nang.mp3, KichthuocanhEn.mp3,…

## **Nội dung chuyên môn chính của đề tài**

+ Thư viện: cv2 với ngôn ngữ Python

+ Giải thuật gì: Xử lý media dùng thư viện với ngôn ngữ lập trình PYTHON

+ Sản phẩm đề tài: Tên sản phẩm: **07.PhanHueAnh.DAHP.MP\_Butterflys**

ỨNG DỤNG QUẢN LÝ & XỬ LÝ MULTIMEDIA: LẬP TRÌNH BỞI PHAN HUẾ ANH SINH VIÊN HỌC VIỆN CÔNG NGHỆ BƯU CHÍNH VIỄN THÔNG TP.HCM

+ Nhập Media từ 2 nguồn: Files có sẵn + trực tiếp từ MICrophone, CAMera.

## **Bố cục của báo cáo**

Chương 1: Giới thiệu tổng quan về đề tài lập trình Multimedia quản lý và xử lý các đối tượng đa phương tiện liên quan đến Lập trình phát triển hệ thống quản lý và xử lí multimedia nhận diện giống/ loài bướm dùng cv2 với python.

Chương 2: Cơ sở lý thuyết của học phần: gồm những lý thuyết cơ bản về

Chương 3: Giới thiệu các thư viện multimedia sử dụng

Chương 4: Giới thiệu sản phẩm

Chương 5: Chọn ngôn ngữ và cách sử dụng

Chương 6: Xử lý âm thanh và giọng nói

Chương 7: Nạp hình ảnh và video từ file hoặc nạp video từ camera

Chương 8: Các chức năng xử lý ảnh

Chương 9: Kết xuất video

Chương 10: Kết luận và hướng phát triển của đề tài

# CƠ SỞ LÝ THUYẾT VỀ KỸ THUẬT LẬP TRÌNH ỨNG DỤNG ĐA PHƯƠNG TIỆN

## **Tổng quan về kỹ thuật multimedia (kỹ thuật đa phương tiện)**

### Kỹ thuật âm thanh (giọng nói)

#### Âm thanh

+ Là các [dao động](https://vi.wikipedia.org/wiki/Dao_%C4%91%E1%BB%99ng) [cơ học](https://vi.wikipedia.org/wiki/C%C6%A1_h%E1%BB%8Dc) (biến đổi vị trí qua lại) của các [phân tử](https://vi.wikipedia.org/wiki/Ph%C3%A2n_t%E1%BB%AD), [nguyên tử](https://vi.wikipedia.org/wiki/Nguy%C3%AAn_t%E1%BB%AD) hay các hạt làm nên [vật chất](https://vi.wikipedia.org/wiki/V%E1%BA%ADt_ch%E1%BA%A5t) và lan truyền trong vật chất như các [sóng](https://vi.wikipedia.org/wiki/S%C3%B3ng). Âm thanh, giống như nhiều sóng, được đặc trưng bởi [tần số](https://vi.wikipedia.org/wiki/T%E1%BA%A7n_s%E1%BB%91), [bước sóng](https://vi.wikipedia.org/wiki/B%C6%B0%E1%BB%9Bc_s%C3%B3ng), [chu kỳ](https://vi.wikipedia.org/wiki/Chu_k%E1%BB%B3), [biên độ](https://vi.wikipedia.org/wiki/Bi%C3%AAn_%C4%91%E1%BB%99) và [vận tốc](https://vi.wikipedia.org/wiki/V%E1%BA%ADn_t%E1%BB%91c) lan truyền ([tốc độ âm thanh](https://vi.wikipedia.org/wiki/T%E1%BB%91c_%C4%91%E1%BB%99_%C3%A2m_thanh)).

+ Đối với [thính giác](https://vi.wikipedia.org/wiki/Th%C3%ADnh_gi%C3%A1c) của [người](https://vi.wikipedia.org/wiki/Ng%C6%B0%E1%BB%9Di), âm thanh thường là sự dao động, trong dải tần số từ khoảng 16 [Hz](https://vi.wikipedia.org/wiki/Hz) đến khoảng 20 000 [Hz](https://vi.wikipedia.org/wiki/Hertz)

#### Các đặc tính cơ bản của âm thanh

+ Cao độ của âm thanh ≈ Tần số âm thanh (sound frequency [Hz]):   
 Âm trầm (bass), Âm trung (mid), Âm cao (treble)

+ Cường độ âm thanh (sound intensity) [deciBel = dB = 1/10 B || Bel = B]

+ Trường độ của âm thanh = Băng thông (Độ trễ || Bandwidth) [Hz]

+ Âm sắc (Timbre) (theo đối tượng phát ra âm).

NGOÀI RA, CÒN CÓ:

+ Thanh áp (áp suất âm thanh, ký hiệu p [miliBar = mB, Bar])

+ Công suất âm thanh P = psv [Watt] { s : thiết diện, v: vận tốc âm thanh, p: thanh áp}

+ Độ xuất hiện đột ngột (transients)

+ Độ rung (vibrato)

+ Độ thay đổi (envelope modulation).

CHÚ Ý:

+ Tone (tông) của âm thanh [“Giọng” của âm thanh] = trường độ + cường độ + cao độ + âm sắc

VD: Tông Nam, Tông Nữ trong Karaoke

#### Các đặc tính cơ bản của file lưu trữ âm thanh (Digital)

+ Sample (giá trị biên độ tần số lấy mẫu)

+ Sample Rate (Số lần lấy mẫu/s) [T]=> Sample Frequency (tần số lấy mẫu) [f]; VD: 44100 Hz

+ BitDepth (số lượng bit số trong mẫu được lấy, còn gọi là độ “nét”: resolution); VD: 16 bits, 24 bits

+ Channel (số kênh tín hiệu số được tách ra); VD 2 kênh

Đây là cơ sở xác định âm thanh Mono (âm thanh “phẳng” = đơn kênh = 1 channel) hay   
 Stereo (âm thanh “nổi” = đa kênh); có nhiều dạng

2D = 2 channel = kênh “trái” và kênh “phải” = 2 chiều

3D (3 Dimensions = 3 chiều) = 3 kênh, . . .

Ghi âm lời nói nên Mono, nhưng Âm nhạc thì nên Stereo, 2D, 3D, Surround,…

+ Bit-Rate (số lượng bit số của âm thanh/s) [kbps (Kilobits per second)] ;  
 VD: Một phút nhạc 128 kbps có dung lượng khoảng 1 MB và   
 bản nhạc 320 kbps thì chắc chắn sẽ hay hơn bản nhạc 128 kbps.

#### Các dạng đuôi (phần mở rộng || Extensions) file âm thanh

Trên Internet công cụ online (offline=Installed) cho phép chuyển kiểu file audio . => .

.mp3 (MPEG-1 audio Player 3 || Motion Pictures Expert Group 1 Layer 3)

= nén có mất thông tin Kich thức file rất bé, nhưng âm thanh không còn tốt như ban đầu (do nén âm)

.wma (Windows Media Audio) = File âm thanh đa phương tiện của hệ điều hành Windows

.wav (Waveform Audio File Format)

= File âm thanh nguyên bản (dạng sóng)= trung thực, kích thước lớn

.mid (Musical Instrument Digital) || midi (Musical Instrument Digital Interface) file

= File âm thanh kỹ thuật số của các nhạc cụ (lưu thông tin của file âm thanh)

.m4a MPEG-4 Compressed Audio File

.flac (Free Lossless Audio Codec) = File âm thanh mã hóa không mất mát

.alac / m4a file (Apple Lossless Audio Code) = Files âm thanh mã hóa không mất mát của Apple

.aac = .ac (Advanced Audio Coding) = File âm thanh mã hóa cải tiến

.ogg (file âm thanh sử dụng phương pháp nén Ogg Vorbis) = files âm thanh nén theo giải thuật OGG

.aif Audio Interchange File Format Sound File

.aiff Audio Interchange File Format

**NGOÀI RA, CÒN NHIỀU EXTENSION KHÁC, NHƯ**

.mod: amiga MODules = File âm thanh điều chế tín hiệu theo hệ điều hành AmigaOS

.it: Impulse Tracker Music Module File

.s3d: Scream Tracker 3

.xm: Fast Tracker 2

.ftm FamiTracker Module File

.amr: Adaptive Multi-rate Compressed Audio File

.msv Mystars! Saved View File

.mus Finale Notation File

.nra Nero Audio-cd Compilation File

.odm Overdrive Media Console Media Control File

.pcm Pulse Code Modulation File

.pk Audition Peak File

.pls Audio Playlist File

.ram Real Audio Metafile

.rbs Rebirth Song File

#### Các dạng nén file âm thanh thông dụng dựa vào BitRate

+ CBR (Constant BitRate) = Dùng giá trị BitRate cố đinh (chọn), đơn giản, ít hiệu quả

+ ABR (Average BitRate) = Dùng giá trị bình quân của BitRate, phức tạp hơn, hiệu quả hơn

+ VBR (Variable BitRate) = Dùng các giá trị BitRate khác biệt cho từng đoạn nén, rất phức tạp, hiệu quả rất cao

#### Một số dạng xử lý chất lượng âm thanh

+ Xử lý tạp âm (Noise Processing) và lọc nhiễu (Noise Filter) dùng “bộ lọc”

+ Cộng hưởng âm= Cộng hưởng qua tần số của âm (cùng tần số f sẽ cộng hưởng âm)

= Khuếch đại âm thanh

+ Trích lọc âm và tách âm

+ Hòa âm (Kết hợp các dạng âm thanh) – phối khí (phối hợp các nhạc cụ)

+ Âm thanh Hi-Fi (Chất lượng cao) và Hi-End

+ Hợp âm (chord) trong âm nhạc: Đồ, Rê, Mi, . . .= kết hợp nhiều âm thanh (âm thanh phát ra từ nhiều dây đàn)

#### Một số kỹ thuật thu thập âm thanh (Thu âm)

Chương trình phần mềm, có thể:

+ Ghi âm

+ Nạp files âm thanh

#### Các xử lý cơ bản với đặc tính âm thanh kỹ thuật số

+ Xử lý đặc tính âm thanh:

. Cao độ (Bass, treble),

. tăng/giảm dB, (khuếch đại âm thanh)

. chuyển âm sắc,

. chuyển tông/Tone..

### Kỹ thuật hình ảnh (image)

#### Hình ảnh

.Hình ảnh có thể có hai [chiều](https://vi.wikipedia.org/wiki/Chi%E1%BB%81u), như thể hiện trên tranh vẽ trên mặt phẳng, hoặc ba chiều, như thể hiện trên tác phẩm [điêu khắc](https://vi.wikipedia.org/wiki/%C4%90i%C3%AAu_kh%E1%BA%AFc) hoặc [hologram](https://vi.wikipedia.org/wiki/Hologram). Hình ảnh có thể được ghi lại bằng thiết bị [quang học](https://vi.wikipedia.org/wiki/Quang_h%E1%BB%8Dc) – như [máy ảnh](https://vi.wikipedia.org/wiki/M%C3%A1y_%E1%BA%A3nh), [gương](https://vi.wikipedia.org/wiki/G%C6%B0%C6%A1ng), [thấu kính](https://vi.wikipedia.org/wiki/Th%E1%BA%A5u_k%C3%ADnh), [kính viễn vọng](https://vi.wikipedia.org/wiki/K%C3%ADnh_vi%E1%BB%85n_v%E1%BB%8Dng), [kính hiển vi](https://vi.wikipedia.org/wiki/K%C3%ADnh_hi%E1%BB%83n_vi) do con người tạo ra, hoặc bởi các cơ chế tự nhiên, như [mắt người](https://vi.wikipedia.org/wiki/M%E1%BA%AFt_ng%C6%B0%E1%BB%9Di) hay mặt nước..

.Pixel: là một điểm vật lý trong một [hình ảnh raster](https://vi.wikipedia.org/wiki/%C4%90%E1%BB%93_h%E1%BB%8Da_raster), hoặc một khối màu rất nhỏ và là đơn vị cơ bản nhất để tạo nên một bức [ảnh kỹ thuật số](https://vi.wikipedia.org/wiki/%E1%BA%A2nh).

. Độ phân giải (Resolution): chỉ lượng thông tin được chứa đựng trong 1 tập tin ảnh hiển thị trên màn hình. Hiểu một cách đơn giản đó chính là số lượng điểm ảnh chứa trên 1 màn hình hiển thị

. Aspect Ratio: là tỷ lệ khung hình được được xác định bởi kích thước của cảm biến của máy ảnh, vd: **tỷ lệ 3: 2, tỷ lệ 4:3**

#### Tiến trình xử lý ảnh

. Tiến trình xử lỷ ảnh: Nạp -> tiền xử lý - … bộ lọc -> Blend màu -> Focus ->..

#### Histogram( *Biểu đồ tần suất)*

* Là một dạng biểu đồ cột cho thấy bằng hình ảnh sự thay đổi, biến động của một tập hợp các dữ liệu theo những hình dạng nhất định.
* Ý nghĩa :
  + Để đánh giá các chỉ tiêu chất lượng cần thu thập rất nhiều dữ liệu khác nhau và các dữ liệu luôn biến động. Nếu nhìn vào những số liệu thu được một cách ngẫu nhiên đó sẽ rất khó đánh giá hết ý nghĩa của những thông tin mà chúng đem lại và rất khó nhận dạng biến động của chúng.
  + Để có thể phân tích, đánh giá tình hình chất lượng từ những dữ liệu thu thập được, đưa ra những kết luận chính xác, người ta tập hợp, phân loại, sắp xếp lại chúng biểu diễn sự phân bố dưới những dạng biểu đồ cột khác nhau theo đặc điểm của các dữ liệu thu được.

#### Hệ màu (không gian màu) RGB

* Sử dụng [mô hình bổ sung](https://vi.wikipedia.org/w/index.php?title=M%C3%B4_h%C3%ACnh_b%E1%BB%95_sung&action=edit&redlink=1) trong đó ánh sáng [đỏ](https://vi.wikipedia.org/wiki/%C4%90%E1%BB%8F), [xanh lục](https://vi.wikipedia.org/wiki/Xanh_l%C3%A1_c%C3%A2y) và [xanh lam](https://vi.wikipedia.org/wiki/Xanh_lam) được tổ hợp với nhau theo nhiều phương thức khác nhau để tạo thành các màu khác. Từ viết tắt RGB trong tiếng Anh có nghĩa là đỏ (red), xanh lục (green) và xanh lam (blue), là ba màu gốc trong các mô hình ánh sáng bổ sung.

#### Một số thao tác xử lý ảnh thông dụng

. Rotate

. Co \_ dãn

. Cắt ảnh theo khung (frame)

. Chuyển xám (3D -> 2D): mục tiêu

#### Nhận diện biên

Trong hình ảnh, thường tồn tại các thành phần như: vùng trơn, góc / cạnh và nhiễu. Cạnh trong ảnh mang đặc trưng quan trọng, thường là thuộc đối tượng trong ảnh (object). Do đó, để phát hiện cạnh trong ảnh, giải thuật Canny là một trong những giải thuật phổ biến / nổi tiếng nhất trong Xử lý ảnh.

Giải thuật phát hiện cạnh Canny gồm 4 bước chính sau:

* **Giảm nhiễu**
* **Tính Gradient và hướng gradient**
* **Non-maximum Suppression**(viết tắt NMS)
* **Lọc ngưỡng**

Ứng dụng: Nhận diện vân tay, nhận khuôn mặt,…

#### Ảnh màu (color = 3 D) và Gray (Ảnh trắng đen = 2D)

* Ảnh màu:
  + Ảnh màu theo lý thuyết của Thomas là ảnh tổ hợp từ 3 màu cơ bản: đỏ (R), lục (G), lơ (B) và thường thu nhận trên các dải băng tần khác nhau. Với ảnh màu, cách biểu diễn cũng tương tự như với ảnh đen trắng, chỉ khác là các số tại mỗi phần tử của ma trận biểu diễn cho ba màu riêng rẽ gồm: đỏ (red), lục (green) và lam (blue). Để biểu diễn cho một điểm ảnh màu cần 24 bit. 24 bit này được chia thành ba khoảng 8 bit. Mỗi màu cũng phân thành *L*cấp màu khác nhau (thường *L=256*). Mỗi khoảng này biểu diễn cho cường độ sáng của một trong các màu chính.
  + Do đó, để lưu trữ ảnh màu người ta có thể lưu trữ từng màu riêng biệt, mỗi màu lưu trữ như một ảnh đa cấp xám. Do đó, không gian nhớ dành cho một ảnh màu lớn gấp 3 lần một ảnh đa cấp xám cùng kích cỡ.
* Ảnh Gray:
  + Ảnh đen trắng chỉ bao gồm 2 màu: màu đen và màu trắng. Người ta phân mức đen trắng đó thành L mức Nếu sử dụng số bit B=8 bít để mã hóa mức đen trắng (hay mức xám) thì  L được xác định :  
    L=2B(trong ví dụ của ta L=28= 256 mức)
  + Nếu L bằng 2, B=1, nghĩa là chỉ có 2 mức: mức 0 và mức 1, còn gọi là ảnh nhị phân. Mức 1 ứng với màu sáng, còn mức 0 ứng với màu tối. Nếu L lớn hơn 2 ta có ảnh đa cấp xám.
  + Ảnh nhị phân khá đơn giản, các phần tử ảnh có thể coi như các phần tử logic. Ứng dụng chính của nó được dùng theo tính logic để phân biệt đối tượng ảnh với nền hay để phân biệt điểm biên với điểm khác.

### Kỹ thuật video clips

#### Video Clips & Flash

. Video là phương tiện [điện tử](https://vi.wikipedia.org/wiki/%C4%90i%E1%BB%87n_t%E1%BB%AD_h%E1%BB%8Dc) để ghi, sao chép, phát lại, [phát sóng](https://vi.wikipedia.org/wiki/Ph%C3%A1t_s%C3%B3ng) và hiển [thị](https://vi.wikipedia.org/wiki/H%C3%ACnh_%E1%BA%A3nh) [hình ảnh](https://vi.wikipedia.org/wiki/H%C3%ACnh_%E1%BA%A3nh) [chuyển động](https://vi.wikipedia.org/wiki/%C4%90i%E1%BB%87n_%E1%BA%A3nh) được lưu trữ trong các [phương tiện](https://vi.wikipedia.org/wiki/Media_(truy%E1%BB%81n_th%C3%B4ng)).. Nguyên lý Video = số lương khung hình (frames/s)

. Flash: Flash dùng kỹ thuật đồ họa vector và đồ họa điểm (raster graphics). Kỹ thuật đồ họa này cho phép bạn có thể phóng to và thu nhỏ hình, chữ không bị vỡ hạt.

#### Video Analog và Video kỹ thuật số (Digital)

* Video Analog:
  + Video tương tự là tín hiệu video được biểu diễn bằng một hoặc nhiều [tín hiệu tương tự](https://en.wikipedia.org/wiki/Analog_signal) . Tín hiệu video màu tương tự bao gồm [độ chói](https://en.wikipedia.org/wiki/Luma_(video)) , độ sáng (Y) và [độ sắc](https://en.wikipedia.org/wiki/Chrominance) (C). Như trường hợp này, khi được kết hợp thành một kênh, trong số những kênh khác với [NTSC](https://en.wikipedia.org/wiki/NTSC) , [PAL](https://en.wikipedia.org/wiki/PAL) và [SECAM,](https://en.wikipedia.org/wiki/SECAM) nó được gọi là [video tổng hợp](https://en.wikipedia.org/wiki/Composite_video) . Video tương tự có thể được truyền trong các kênh riêng biệt, như ở hai định dạng [video](https://en.wikipedia.org/wiki/Component_video)[S-Video](https://en.wikipedia.org/wiki/S-Video) (YC) kênh và [video thành phần](https://en.wikipedia.org/wiki/Component_video) đa kênh .
  + Video analog được sử dụng trong cả ứng dụng [sản xuất truyền hình](https://en.wikipedia.org/wiki/Television_production) tiêu dùng và chuyên nghiệp .
* Video kỹ thuật số:
  + Video kỹ thuật số (DV) là video được ghi lại và lưu trữ ở định dạng kỹ thuật số dưới dạng số một và số không, chứ không phải là một loạt ảnh tĩnh được chụp trong phim. Thông tin được xử lý và lưu trữ dưới dạng chuỗi dữ liệu kỹ thuật số để máy tính thao tác dễ dàng, nhưng video vẫn được trình chiếu cho người xem thông qua màn hình ở dạng analog. Video kỹ thuật số bao gồm một loạt hình ảnh bitmap (BMP) trực giao được hiển thị liên tiếp nhanh chóng liên tục với các tần số phổ biến là 15, 24, 30 và 60 khung hình / giây (FPS); DV càng có nhiều khung hình, thì càng chụp hoặc hiển thị nhiều chi tiết chuyển động.
  + [Các](https://en.wikipedia.org/wiki/Digital_video) định dạng tín hiệu [video kỹ thuật số](https://en.wikipedia.org/wiki/Digital_video) đã được chấp nhận, bao gồm [giao diện kỹ thuật số nối tiếp](https://en.wikipedia.org/wiki/Serial_digital_interface) (SDI), [Giao diện hình ảnh kỹ thuật số](https://en.wikipedia.org/wiki/Digital_Visual_Interface) (DVI), [Giao diện đa phương tiện độ nét cao](https://en.wikipedia.org/wiki/HDMI) (HDMI) và Giao diện [DisplayPort](https://en.wikipedia.org/wiki/DisplayPort) .

#### Video nén

+ Video chiếm rất nhiều dung lượng - chỉ số lượng thay đổi tùy thuộc vào định dạng video, độ phân giải và số khung hình trên giây bạn chọn. Quay video HD 1080p không nén chiếm khoảng 10,5GB không gian mỗi phút của video. Nếu bạn sử dụng điện thoại thông minh để quay video, cảnh quay 1080p sẽ có tốc độ 130MB mỗi phút, trong khi video 4K chiếm 375 MB không gian cho mỗi bộ phim phút. "Nén" chỉ có nghĩa là thông tin được đóng gói vào một không gian nhỏ hơn.

+ Nén mất dữ liệu nghĩa là tệp nén có ít dữ liệu hơn tệp gốc. Trong một số trường hợp, điều này chuyển thành các tệp chất lượng thấp hơn, bởi vì thông tin đã bị "mất". Tuy nhiên, bạn có thể mất một lượng dữ liệu tương đối lớn trước khi bắt đầu nhận thấy sự khác biệt. Nén mất dữ liệu bù đắp cho sự mất chất lượng bằng cách tạo ra các tệp tương đối nhỏ.

+ Nén không bị mất là chính xác những gì nó giống như âm thanh, nén mà không có thông tin nào bị mất. Đây không phải là hữu ích như nén mất dữ liệu vì các tệp thường kết thúc bằng cùng kích thước như trước khi nén. Điều này có vẻ vô nghĩa, vì việc giảm kích thước tệp là mục tiêu chính của nén. Tuy nhiên, nếu kích thước tệp không phải là vấn đề, việc sử dụng nén không mất kết quả sẽ mang lại hình ảnh chất lượng hoàn hảo.

.mp4, .avi, .wmv, flv

#### Frames

* Frame rate hay còn được hiểu là tốc độ khung hình. Đây chính là tần số xuất hiện những khung hình riêng lẻ mà máy ảnh chụp trong khoảng thời gian 1 giây.
* Frame có đơn vị là FPS (viết ngắn gọn của Frames – per – second). Chẳng hạn như video của bạn xuất hiện 25 khung hình mỗi giây thì video đó là 25 fps.

Ứng dụng:

. Nhận diện đối tượng di động trong Video

#### Slide Windows

. Nhận diện đặc trưng của Video

+ Window size (kernel size): kích thước cửa sổtrượt= là số lẻ, tức 3, 5, 7, ...

+ Padding: sốpixel mở rộng thêm vào ảnh đầu vô.

+ Stride: khoảng cách lần trượt.

+ Dilation: khoảng cách của mỗi pixel trên cửa sổ

## **Công cụ và các hệ thống thư viện hỗ trợ lập trình multimedia**

### Phần mềm công cụ sử dụng chính trong học phần

\* Công cụ chính: Spyder (Anaconda3)

\* Ngôn ngữ lập chính: Python

\* Thư viện: cv2

NGOÀI RA: có thể dùng Pythong trên Microsoft Visual Studio = .NET  
 (2005, [2007]2008, 2010, [2012]2013, [2014]2015, 2017, 2019)

*Tên khác : .NET == DOT NET (thiết lập Environment Python)*

### Các thư viện lập trình chính được sử dụng trong học phần

#### Xử lý âm thanh và giọng nói

Thư viện speech\_reconition: thư viện nhận dạng giọng nói

Playsound: thư viện hỗ trợ phát âm thanh

#### Xử lý hình ảnh

Thư viện cv2: xử lý ảnh

#### Xử lý Video

Thư viện cv2: xử lý video

### Tập các Multimedia mẫu được sử dụng trong đề tài của Học phần

<http://www.vncreatures.net/e_showpic.php?vietname=B%C6%B0%E1%BB%9Bm%20chai%20xanh&latinname=Graphium%20sarpedon&author=Ph%C3%B9ng%20M%E1%BB%B9%20Trung&imagepath=./pictures/insect/381_4>

<https://i.vietgiaitri.com/2020/1/21/nhung-thien-tai-nguy-trang-trong-the-gioi-dong-vat-p1-3da181.jpg>

<https://kenh14cdn.com/txccccccccccccDqHcybwkI06dmgnY/Image/2012/06/120628CLbuom09_d40da.JPG>

<https://i.pinimg.com/originals/27/b4/b5/27b4b53e73907d976d07264971a2ad91.jpg>

<https://photo-cms-kienthuc.zadn.vn/zoomh/800/uploaded/ctvkhoahoc/2016_11_15/top-nhung-loai-buom-dep-nhat-viet-nam.jpg>

<https://farm7.static.flickr.com/6179/6218417201_acac5de168_o.jpg>

<https://live.staticflickr.com/6204/6162182582_4684c675fe_b.jpg>

<https://www.youtube.com/watch?v=h_glLV23Z1M>

http://mobiclip.vn/899093/xem-video/buom-la-kho

<https://www.youtube.com/watch?v=tSNlI7bPjQY>

https://www.youtube.com/watch?v=0Qu\_CHol44c

# Chương 3: GIỚI THIỆU CÁC THƯ VIỆN MULTIMEDIA SỬ DỤNG

## **Nạp các thư viện**

import cv2 # thư viện hỗ trợ xử lí ảnh và video

import speech\_recognition as sr # thư viện nhận dạng giọng nói

import os # thư viện để thao tác với tệp và thư mục

import numpy as np # thư viện làm việc với mảng

import datetime # thư viện để làm việc với thời gian

import playsound # thư viện hỗ trợ phát âm thanh

## **Giới thiệu các thư viện được dùng**

+ Audio và Speech + speech\_recognition: có chức năng là nhận dạng giọng nói để chuyển âm thanh thành văn bản. Âm thanh được đọc vào microphone của máy tính sau đó được xử lý và sẽ được nhận dạng ở ngôn ngữ tiếng việt để chuyển thành dạng văn bản.

+ playsound: phát âm thanh từ văn bản cho trước.

+ Images Processing + CV2 (OpenCV) là viết tắt của Open Source Computer Vision, tạm dịch là Thị giác máy tính mã nguồn mở. Đúng như tên gọi mục đích chính là nhằm tạo ra các chức năng thị giác máy tính, giám sát các chức năng tổng thể tập trung vào thị giác máy tính tức thì. Vì OpenCV không có tài liệu thích hợp, cho nên theo nhiều nhà phát triển, nó là một trong những thư viện khó học nhất. Tuy nhiên, nó vẫn cung cấp đầy đủ khả năng để giúp bạn học thị giác máy tính dễ hơn. Hỗ trợ xử lý hình ảnh và video. Video Clip Processing + CV2 (OpenCV) Các thư viện hỗ trợ

+ numpy: thư viện được sử dụng làm việc với mảng

+ os # Thư viện os của MS. Windows = Lập thư mục & lưu các files Media + numpy # Thư viện Numeric Python: Lập ma trận giữ chỗ để ghép các Images và Videos.

+ datetime:  là một module, được thiết kế dựa trên hướng đối tượng (object oriented) để làm việc với ngày tháng và thời gian trong **Python**

# Chương 4: GIỚI THIỆU SẢN PHẨM

## **Mô tả sản phẩm**

Tên sản phẩm: Lập trình phát triển hệ thống quản lý và xử lí multimedia giống/ loài bướm dùng cv2 với python

Ứng dụng quản lý và xử lí hình ảnh, video được nạp từ file có sẵn và camera máy tính đang chạy chương trình, thực hiện các chức năng chỉnh sửa hình ảnh.

Xử lý các file âm thanh và giọng nói để thực hiện lựa chọn các chức năng trong chương trình.

**07.PhanHueAnh.DAHP.MP\_Butterflys**

**ỨNG DỤNG QUẢN LÝ & XỬ LÝ MULTIMEDIA: LẬP TRÌNH BỞI PHAN HUẾ ANH HỌC VIÊN CÔNG NGHỆ BƯU CHÍNH VIỄN THÔNG TP.HCM**

**Quản lý và xử lý Multimedia giống/ loài bướm dùng cv2 với python.**

Thực hiện bởi: Phan Huế Anh

Ngôn ngữ lập trình: **PYTHON**

Công cụ: Spyder (anaconda 3)

Thư viện chính: cv2, speech\_reconition, os , numpy

* 1. **Tập dữ liệu được tham khảo (thu thập) từ:**

Mục lục [2]

## **Giới thiệu các chức năng:**

* Chọn ngôn ngữ: tiếng Anh hoặc tiếng Việt.
* Chọn cách sử dụng: nhập hoặc giọng nói.
* Nạp hình ảnh và video từ file, camera
* Các chức năng chỉnh sửa :
  + Chuyển ảnh xám
  + Lấy kích thước ảnh
  + Cắt ảnh
  + Xoay ảnh
  + Lấy giá trị màu tại vị trí
  + Thay đổi kích thước
  + Làm mờ ảnh
  + Chuyển ảnh chất lượng cao
* Xuất video

# Chương 5: CHỌN NGÔN NGỮ VÀ CÁCH SỬ DỤNG

## **Chọn ngôn ngữ**

**5.1.1: Giới thiệu**

Chọn ngôn ngữ tiếng Anh hoặc tiếng Việt để sử dụng trong suốt chương trình.

Yêu cầu:

- Nạp đầu vào lựa chọn ngôn ngữ

**5.1.2: Code xử lý**

print("""

          Chọn ngôn ngữ:

              1. Tiếng Việt

              2. Tiếng Anh

        """)

    lang = input("nhập lựa chọn: ") #Chọn ngôn ngữ sử dụng

* + 1. **Thực nghiệm**

Chọn ngôn ngữ:

1. Tiếng Việt

2. Tiếng Anh

Nhập lựa chọn(vd: 1): 1

# Chọn cách sử dụng

* + 1. **Giới thiệu**

Chọn cách sử dụng bằng giọng nói hoặc cách nhập từ bàn phím

Yêu cầu:

* Nạp đầu vào cách sử dụng bằng cách nhập hoặc sử dụng giọng nói
  + 1. **Code xử lý**

if lang == "1":

        print("""

          1. Sử dụng giọng nói

          2. Sử dụng bằng cách nhập

          """)

        playsound.playsound("voice/voice\_or\_nhap.mp3")

        chon = input("Chọn: ")

if lang == "2":

        print("""

          1. Voice

          2. Enter

          """)

        playsound.playsound("voice/voice\_or\_nhapEn.mp3")

        chon = input("Input: ")

* + 1. **Thực nghiệm**

1. Nạp file hình ảnh

2. Nạp từ camera

3. Nạp video

Chọn: 1

# 

# Chương 6: XỬ LÝ ÂM THANH VÀ GIỌNG NÓI

# Xử lý âm thanh

* + 1. **Giới thiệu**

Phát đoạn âm thanh phù hợp từ các file âm thanh có sẵn trong thư mục

Yêu cầu:

* Phát giọng nói hướng dẫn sử dụng
  + 1. **Code xử lý**

playsound.playsound("voice/stt\_file.mp3") #phát âm thanh

playsound.playsound("voice/stt\_fileEn.mp3")

playsound.playsound("voice/xuat\_video.mp3")

playsound.playsound("voice/voice\_or\_nhap.mp3")

# Xử lý giọng nói

* + 1. **Giới thiệu**

Chuyển giọng nói sang chữ, số để thực hiện các chức năng cần thiết

Yêu cầu:

* Lấy được giọng nói tiếng Anh hoặc tiếng Việt phù hợp để điều khiển chương trình
  + 1. **Code xử lý**

# hàm xử lí giọng nói lấy số

def voice():

    y=True

    while y == True:

        print("Hãy nói số thứ tự muốn mở(ví dụ: số 0)")

        playsound.playsound("voice/stt\_file.mp3") #phát âm thanh

        r = sr.Recognizer() # khởi tạo

        with sr.Microphone() as source:

            print("Adjusting noise ")

            r.adjust\_for\_ambient\_noise(source, duration=1)

            audio\_data = r.record(source, duration=5)

            print("Kết quả nhận diện...")

                # chuyển giọng nói sang chữ

            try:

                query = r.recognize\_google(audio\_data,language="vi")

            except:

                continue

            print(query)

            query=query.strip('số ') # cắt bỏ chữ thừa

            try:

                query=int(query)

                break

            except:

                continue

    return query

# voice xử lý tiếng anh

def voiceEnglish():

    y=True

    while y == True:

        print("Please say the number you want to open(ex: number 1)")

        playsound.playsound("voice/stt\_fileEn.mp3")

        r = sr.Recognizer() # khởi tạo

        with sr.Microphone() as source:

            print("Adjusting noise ")

            r.adjust\_for\_ambient\_noise(source, duration=1)

            audio\_data = r.record(source, duration=5)

            print("Kết quả nhận diện...")

                # chuyển giọng nói sang chữ

            try:

                query = r.recognize\_google(audio\_data,language="en")

            except:

                continue

            print(query)

            query=query.strip('number ') # cắt bỏ chữ thừa

            try:

                query=int(query)

                break

            except:

                continue

    return query

# hàm xử lý giọng nói lấy chữ

def voicexl():

    r = sr.Recognizer()

    with sr.Microphone() as source:

         r.adjust\_for\_ambient\_noise(source, duration=1)

         print("Hãy nói chức năng bạn muốn")

         playsound.playsound("voice/noichucnang.mp3")

         audio\_data = r.record(source, duration=5)

try:

             text = r.recognize\_google(audio\_data,language="vi")#nhận giọng nói với language là tiếng việt

         except:

             text = "Bạn chọn chưa đúng bạn chọn lại đi!"

             playsound.playsound("voice/banchonchuadung.mp3")

         print("Bạn đã chọn: {}".format(text))      #in ra kết quả người dùng đã chọn

    return text

* + 1. **Thực nghiệm**

Adjusting noise

Kết quả nhận diện...

Bạn chọn Số 1

# Chương 7: NẠP HÌNH ẢNH VÀ VIVDEO TỪ FILE HOẶC NẠP VIDEO TỪ CAMERA , KẾT XUẤT CAMERA

## **7.1 Nạp hình ảnh từ file**

* + 1. **Giới thiệu**

Nạp file hình ảnh từ thư mục ảnh được tạo sẵn

Yêu cầu:

* Nạp số thứ tự file trong danh sách file xuất hiện trên màn hình.
  + 1. **Code xử lý**

def getImg(c):

    myList = os.listdir(path) #lấy danh sách tên file trong thư mục

            # in list images

    if lang == "1": # chọn ngôn ngữ

        print("Danh sách file hình ảnh: ")

    else: # chọn ngôn ngữ

         print("List Images: ")

    for i in myList:

        print("{} : {}".format(myList.index(i),i)) # in danh sách các file trong thư mục

    if c == "voice":

        if lang == "1":

            query = voice() # gọi hàm xử lý giọng nói tiếng Việt

        else:

            query = voiceEnglish() # gọi hàm xử lý giọng nói tiếng Anh

        img\_path = os.path.join(path,myList[query]) # lưu tên file được chọn vào biến

    elif c == "nhap":

        if lang == "1":

            playsound.playsound(("voice/nhap\_stt\_file.mp3"))# phát âm thanh

            nhap = int(input("Nhập số thứ tự file: "))

        if lang == "2":

            playsound.playsound(("voice/nhap\_stt\_fileEn.mp3"))# phát âm thanh

            nhap = int(input("Enter number: "))

        img\_path = os.path.join(path,myList[nhap])# lưu tên file được chọn vào biến

    return img\_path

* + 1. Thực nghiệm

Danh sách file hình ảnh:

0 : buom\_dom\_xanh.jpg

1 : buom\_khe.jpg

2 : buom\_trang.jpg

3 : chai\_xanh.jpg

4 : duoi\_kiem.jpg

5 : duoi\_la\_xanh.jpg

6 : ho\_van.jpg

7 : la\_kho.jpg

8 : phan\_cam.jpg

Nhập số thứ tự file(vd: 0): 1

# Nạp video

* + 1. **Nạp từ file**
       1. **Giới thiệu:**

Nạp file video từ thư mục video có sẵn.

Yêu cầu: Nhập số thứ tự file video trong danh sách file trên màn hình

* + - 1. **Code xử lý:**

def video(c,video\_path):

    if video\_path == "cam":

        video\_path = 0

        if lang == "1":

            input\_name = input("Nhập tên folder: ")

        else:

            input\_name = input("Folder name: ")

    else:

        if lang == "1": #ngôn ngữ tiếng Việt

            print('''

                      Danh sách các file:

                      ''')

        else:#ngôn ngữ tiếng Anh

            print('''

                      List Video:

                      ''')

        myList = os.listdir(path\_video) #lưu danh sách file video vào biến

        for i in myList:

            print("{} : {}".format(myList.index(i),i))# in danh sách file video

        if c == "voice":

            query = voice()  #gọi hàm xử lý giọng nói

        elif c == "nhap":

            if lang == "1":

                playsound.playsound(("voice/nhap\_stt\_file.mp3"))# phát âm thanh

                query = int(input("Nhập số thứ tự file: "))

            else:

                playsound.playsound(("voice/nhap\_stt\_fileEn.mp3"))

                query = int(input("Enter file number: "))

        if lang == "1":

                playsound.playsound(("voice/ten\_folder.mp3"))

                input\_name = input("Nhập tên folder: ")

        else:

                playsound.playsound(("voice/ten\_folderEn.mp3"))

                input\_name = input("Enter folder name: ")

        video\_path = "{}/{}".format(path\_video,myList[query])

    global folder

    folder = input\_name

    os.makedirs(folder, exist\_ok=True)

    cap = cv2.VideoCapture(video\_path)

    count = 0       #biến đếm số khung hình -> bắt đầu từ số 0

    while cap.isOpened():       #trong khi Video clip đang còn phá

        ret, frame = cap.read()     #chụp ra một khung hình: khung chụp được lưu vào biến frame; ret = vị trí tiếp theo của Video (sau khung hình vừa chụp)

        cv2.imshow('Khung Hinh', frame)     #Hiển thỉ khung hình vừa chụp (ở trên: trong biến frame)

        count = count + 1   # tăng chỉ count lên 1 để chuẩn bị lưu khung hình kế tiếp

        if cv2.waitKey(10) & 0xFF == ord('p'):      # kết thúc khi đạt số lượng frame người dùng chọn

            cv2.waitKey()

            if lang == "1":

                playsound.playsound(("voice/luu\_hinh.mp3"))

                print("Bạn có muốn lưu hình này không?(y/n) ")

                ans = input("Nhập: ")

            else:

                playsound.playsound(("voice/luu\_hinhEn.mp3"))

                print("Do you want to save this image?(y/n) ")

                ans = input("Input: ")

            if ans == "y":

                ts = datetime.datetime.now()

                cv2.imwrite(os.path.join(folder,ts.strftime("%Y-%m-%d\_%H-%M-%S")+".jpg") ,frame)

                filename = folder+"/"+ts.strftime("%Y-%m-%d\_%H-%M-%S")+".jpg"

                XuLiAnh(c,filename,folder)

            elif ans == "n":

                continue

        elif cv2.waitKey(10) & 0xFF == ord('q'):

            break

    cap.release()       #Giải phóng biến đối tượng Video cap

    cv2.destroyAllWindows()

    return folder

* + - 1. **Thực nghiệm**

Danh sách các file:

0 : buom.mp4

1 : Bướm đêm lớn nhất thế giới (Attacus atlas) Bướm khế.webm

2 : Thế giới các loài bướm - butterfly world.webm

Nhập số thứ tự file: 0



* + 1. **Nạp từ Camera** 
       1. **Giới thiệu:**

Nạp video từ camera của máy tính đang chạy chương trình

Yêu cầu: Nạp đầu vào là từ camera máy tính đang chạy chương trình, xuất hiện video từ camera

* + - 1. **Code xử lý:**

def video(c,video\_path):

    if video\_path == "cam":

        video\_path = 0

        if lang == "1":

            input\_name = input("Nhập tên folder: ")

        else:

            input\_name = input("Folder name: ")

    else:

        if lang == "1": #ngôn ngữ tiếng Việt

            print('''

                      Danh sách các file:

                      ''')

        else:#ngôn ngữ tiếng Anh

            print('''

                      List Video:

                      ''')

        myList = os.listdir(path\_video) #lưu danh sách file video vào biến

        for i in myList:

            print("{} : {}".format(myList.index(i),i))# in danh sách file video

        if c == "voice":

            query = voice()  #gọi hàm xử lý giọng nói

        elif c == "nhap":

            if lang == "1":

                playsound.playsound(("voice/nhap\_stt\_file.mp3"))# phát âm thanh

                query = int(input("Nhập số thứ tự file: "))

            else:

                playsound.playsound(("voice/nhap\_stt\_fileEn.mp3"))

                query = int(input("Enter file number: "))

        if lang == "1":

                playsound.playsound(("voice/ten\_folder.mp3"))

                input\_name = input("Nhập tên folder: ")

        else:

                playsound.playsound(("voice/ten\_folderEn.mp3"))

                input\_name = input("Enter folder name: ")

        video\_path = "{}/{}".format(path\_video,myList[query])

    global folder

    folder = input\_name

    os.makedirs(folder, exist\_ok=True)

    cap = cv2.VideoCapture(video\_path)

    count = 0       #biến đếm số khung hình -> bắt đầu từ số 0

    while cap.isOpened():       #trong khi Video clip đang còn phá

        ret, frame = cap.read()     #chụp ra một khung hình: khung chụp được lưu vào biến frame; ret = vị trí tiếp theo của Video (sau khung hình vừa chụp)

        cv2.imshow('Khung Hinh', frame)     #Hiển thỉ khung hình vừa chụp (ở trên: trong biến frame)

        count = count + 1   # tăng chỉ count lên 1 để chuẩn bị lưu khung hình kế tiếp

        if cv2.waitKey(10) & 0xFF == ord('p'):      # kết thúc khi đạt số lượng frame người dùng chọn

            cv2.waitKey()

            if lang == "1":

                playsound.playsound(("voice/luu\_hinh.mp3"))

                print("Bạn có muốn lưu hình này không?(y/n) ")

                ans = input("Nhập: ")

            else:

                playsound.playsound(("voice/luu\_hinhEn.mp3"))

                print("Do you want to save this image?(y/n) ")

                ans = input("Input: ")

            if ans == "y":

                ts = datetime.datetime.now()

                cv2.imwrite(os.path.join(folder,ts.strftime("%Y-%m-%d\_%H-%M-%S")+".jpg") ,frame)

                filename = folder+"/"+ts.strftime("%Y-%m-%d\_%H-%M-%S")+".jpg"

                XuLiAnh(c,filename,folder)

            elif ans == "n":

                continue

        elif cv2.waitKey(10) & 0xFF == ord('q'):

            break

    cap.release()       #Giải phóng biến đối tượng Video cap

    cv2.destroyAllWindows()

    return folder

* + - 1. **Thực nghiệm**



* 1. **Kết xuất video:**

# 7.3.1 : Giới thiệu

Xuất các file hình ảnh đã chỉnh sửa thành video

Yêu cầu: Xuất video dạng . mp4 là kết hợp các hình ảnh đã xử lý

# 7.3.2: Code xử lý

def ghepVideo(folder):

    if lang == "1":

        playsound.playsound('voice/nhap\_video\_name.mp3')

        videoout = input("Nhập tên video(vd: video.mp4): ")

    else:

        playsound.playsound('voice/nhap\_video\_nameEn.mp3')

        videoout = input("Enter Video name(ex: video.mp4): ")

    images\_out = []

    filename = os.listdir(folder)

    for name in filename:

        name = folder+"/"+name

        img = cv2.imread(name)

        height, width, layers = img.shape

        size = (width,height)

        images\_out.append(img)

    #ghép video

    out = cv2.VideoWriter(videoout,cv2.VideoWriter\_fourcc(\*'DIVX'), 1,size)

    for i in range(len(images\_out)):

        out.write(images\_out[i])

    out.release()

* + 1. **Thực nghiệm**

Bạn có muốn xuất video ko?(y/n) y

Nhập tên video(vd: video.mp4): video.mp4

# Chương 8: CÁC CHỨC NĂNG XỬ LÝ ẢNH

## **Chức năng chính:**

# Giới thiệu:

Hiển thị các chức năng của chương trình như: chuyển ảnh xám, lấy kích thước ảnh, cắt ảnh bằng chuột, xoay ảnh, lấy giá trị màu tại 1 vị trí bất kỳ, thay đổi kích thước ảnh, làm mờ ảnh, chuyển ảnh chất lượng cao.

Yêu cầu: Xuất hiện danh sách các chức năng xử lý ảnh trên màn hình theo ngôn ngữ đã lựa chọn ban đầu

* + 1. **Code xử lý:**

#menu chức năng xử lý ảnh

def MenuXuLiAnh():

    print("""

                 1. Chuyển ảnh xám

                 2. Lấy kích thước ảnh

                 3. Cắt ảnh

                 4. Xoay ảnh

                 5. Lấy giá trị màu tại vị trí

                 6. Thay đổi kích thước

                 7. Làm mờ ảnh

                 8. Chuyển ảnh chất lượng cao

                 9. Thoát """)

def MenuXuLiAnhEnglish():

    print("""

                 1. Gray Image

                 2. Get size Image

                 3. Crop Image

                 4. Rotate Image

                 5. Get value color

                 6. Resize Image

                 7. Blur Image

                 8. High-quality image transfer

                 9. Exit """)

* + 1. Thực nghiệm

1. Chuyển ảnh xám

2. Lấy kích thước ảnh

3. Cắt ảnh

4. Xoay ảnh

5. Lấy giá trị màu tại vị trí

6. Thay đổi kích thước

7. Làm mờ ảnh

8. Chuyển ảnh chất lượng cao

9. Thoát

Nhập số lựa chọn: 1

* 1. **Chuyển ảnh xám:**
     1. **Giới thiệu**

Chuyển ảnh màu (3D) thành ảnh xám (2D)

Yêu cầu:

* Lấy được hình ảnh đầu vào
* Chuyển hình ảnh màu sang ảnh xám
* Lưu hình ảnh vào thu mục đã tạo
  + 1. **Code xử lý**

def gray(file, folder):

    img = cv2.imread(file)  # đọc file hình ảnh

    img\_gray = cv2.cvtColor(img,cv2.COLOR\_BGR2GRAY) # chuyển sang ảnh xám

    ts = datetime.datetime.now() # lấy thời gian hiện tại

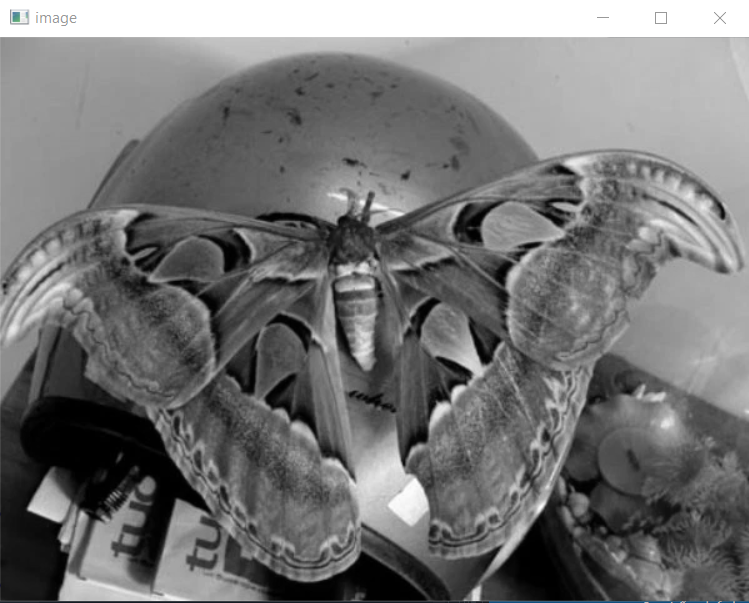
    cv2.imwrite(os.path.join(folder,ts.strftime("%Y-%m-%d\_%H-%M-%S")+".jpg") ,img\_gray) # lưu hình ảnh vào thư mục

    cv2.imshow('image',img\_gray) # show hình ảnh

    cv2.waitKey(0)

    cv2.destroyAllWindows()

* + 1. **Thực nghiệm**



* 1. **Lấy kích thươc ảnh**
     1. **Giới thiệu**

Lấy kích thước chiều dài, chiều cao, chiều sau của ảnh

Yêu cầu:

* Lấy được hình ảnh đầu vào
* Lấy kích thước ảnh và xuất ra màn hình
  + 1. **Code xử lý**

#Lấy kích thước ảnh

def getSize(file):

    img = cv2.imread(file) # đọc file hình ảnh

    (h, w, d) = img.shape # lấy kích thước hình ảnh màu lưu vào 3 biến h: chiều cao, w: chiều rộng, d: chiều sâu

    if lang == "1":

        playsound.playsound(("voice/Kichthuocanh.mp3"))# phát âm thanh

        print ("Kích thước hình ảnh là: ")

    else:

        playsound.playsound(("voice/KichthuocanhEn.mp3"))# phát âm thanh

        print ("Image size is: ")

    print("width={}, height={}, depth={}".format(w, h, d)) # in ra màn hình chiều rộng, chiều cao, chiều sâu ảnh

* + 1. **Thực nghiệm**

Kích thước hình ảnh là:

width=600, height=451, depth=3

* 1. **Cắt ảnh**
     1. **Giới thiệu**

Sử dụng con chuột máy tính để cắt ảnh

Yêu cầu:

* Lấy được hình ảnh đầu vào
* Sử dụng con trỏ chuột để xác định vị trí muốn cắt ảnh, phần xác định vị trí có đường biên màu xanh
* In ra hình ảnh đã cắt
* Có thể thực hiện cắt lại nếu muốn thay đổi
* Lưu hình ảnh vào thu mục đã tạo
* Nhấn q để đóng ảnh góc
  + 1. **Code xử lý**

cropping = False

x\_start, y\_start, x\_end, y\_end = 0, 0, 0, 0

# Cắt ảnh

def crop(img\_path):

    image = cv2.imread(img\_path) #đọc hình ảnh

    global oriImage

    oriImage = image.copy() # Copy hình ảnh

    print("ori",oriImage)

    cv2.namedWindow("image")

    cv2.setMouseCallback("image", mouse\_crop)

    while True:

        i = image.copy()

        if not cropping:

            cv2.imshow("image", image)

        elif cropping:

            cv2.rectangle(i, (x\_start, y\_start), (x\_end, y\_end), (255, 0, 0), 2) # vẽ đường cắt ảnh

            cv2.imshow("image", i)

        cv2.waitKey(1)

        if cv2.waitKey(10) & 0xFF == ord('q'):

            break

    # close all open windows

    cv2.destroyAllWindows()

def mouse\_crop(event, x, y, flags, param):

    # tham chiếu đến các biên toàn cục

    global x\_start, y\_start, x\_end, y\_end, cropping

    #  nếu nút chuột trái đã XUỐNG, bắt đầu GHI

    # (x, y)  tọa độ và chỉ ra rằng việc cắt xén đang được thực hiện

    if event == cv2.EVENT\_LBUTTONDOWN:

        x\_start, y\_start, x\_end, y\_end = x, y, x, y

        cropping = True

    # Mouse is Moving

    elif event == cv2.EVENT\_MOUSEMOVE:

        if cropping == True:

            x\_end, y\_end = x, y

    # if the left mouse button was released

    elif event == cv2.EVENT\_LBUTTONUP:

        # record the ending (x, y) coordinates

        x\_end, y\_end = x, y

        cropping = False # cropping is finished

        refPoint = [(x\_start, y\_start), (x\_end, y\_end)]

        if len(refPoint) == 2: #when two points were found

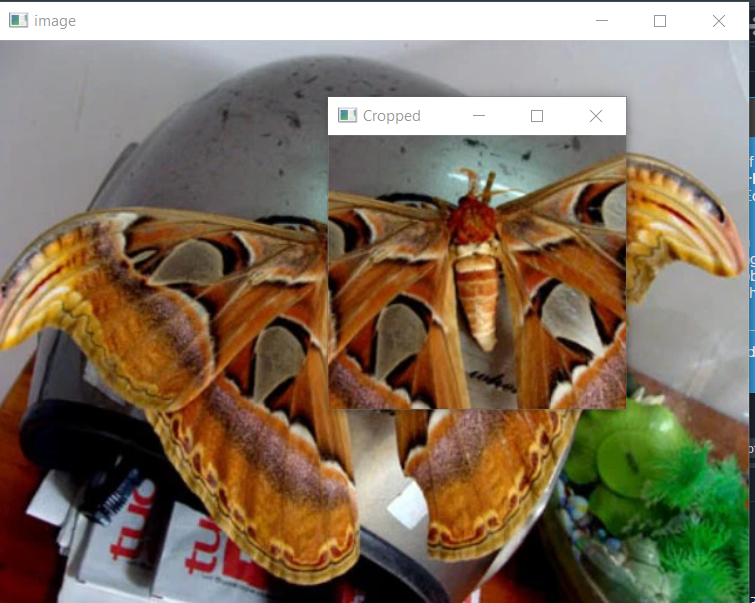
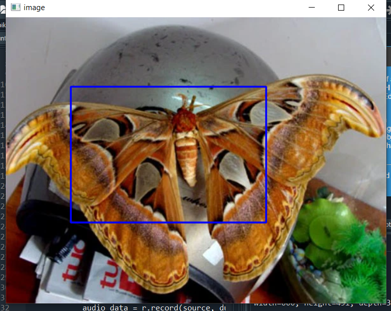
            roi = oriImage[refPoint[0][1]:refPoint[1][1], refPoint[0][0]:refPoint[1][0]]

            cv2.imshow("Cropped", roi)

            ts = datetime.datetime.now()

            cv2.imwrite(os.path.join(folder,ts.strftime("%Y-%m-%d\_%H-%M-%S")+".jpg") ,roi)

* + 1. **Thực nghiệm**



* 1. **Xoay ảnh**
     1. **Giới thiệu**

Xoay hình ảnh với số góc người dùng tự chọn

Yêu cầu:

* Lấy được hình ảnh đầu vào
* Nhập số góc muốn thay đổi
* In hình ảnh
* Lưu hình ảnh vào thu mục đã tạo
  + 1. **Code xử lý**

def rotate(file,xoay,folder):

    img = cv2.imread(file) # đọc file hình ảnh

    (h, w, d) = img.shape # lấy kích thước hình ảnh màu lưu vào 3 biến h: chiều cao, w: chiều rộng, d: chiều sâu

    center = (w // 2, h // 2) # tính tọa độ trung tâm của ảnh

    M = cv2.getRotationMatrix2D(center, xoay, 1.0) #ma trận biến đổi M sẽ được sử dụng để quay một hình ảnh.

    rotated = cv2.warpAffine(img, M, (w, h)) # xoay ảnh

    cv2.imshow("image rotate", rotated)# show hình ảnh

    ts = datetime.datetime.now() # lấy thời gian hiện tại

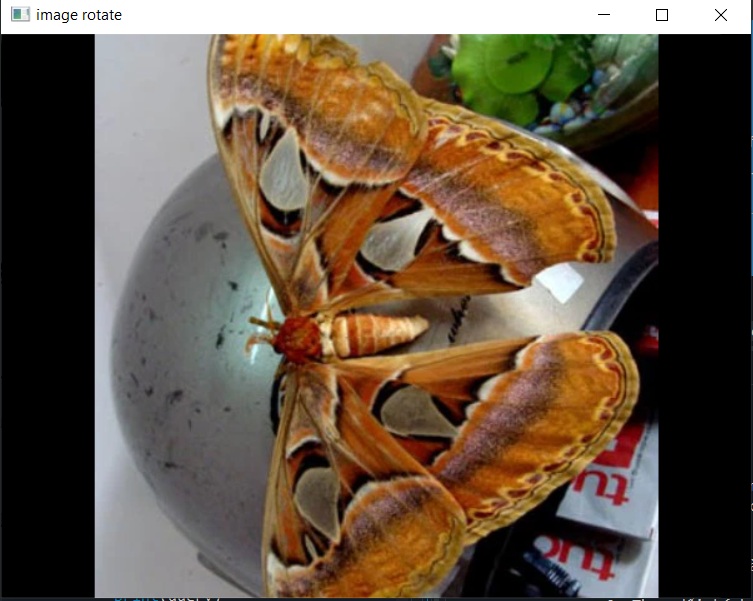
    cv2.imwrite(os.path.join(folder,ts.strftime("%Y-%m-%d\_%H-%M-%S")+".jpg") ,rotated) # lưu hình ảnh vào thư mục

    cv2.waitKey(0)

    cv2.destroyAllWindows()

* + 1. **Thực nghiệm**

bạn muốn xoay bao nhiêu độ: 90



* 1. **Lấy giá trị màu tại vị trí**
     1. **Giới thiệu**

Lấy giá trị màu tại vị trí tọa độ mà người dùng muốn lấy

Yêu cầu:

* Lấy được hình ảnh
* Nhập các giá trị vị trí theo tọa độ x, y
* In giá trị màu ra màn hình
  + 1. **Code xử lý**

def getColor(img\_path):

    img = cv2.imread(img\_path)

    if lang == "1":

        playsound.playsound('voice/toadoX.mp3') #Phát âm thanh

        x = int(input("Nhập tọa độ x: "))

        playsound.playsound('voice/toadoY.mp3')#Phát âm thanh

        y = int(input("Nhập tọa độ y: "))

    else:

        playsound.playsound('voice/toadoXEn.mp3') #Phát âm thanh

        x = int(input("Enter Coordinate X: "))

        playsound.playsound('voice/toadoYEn.mp3')#Phát âm thanh

        y = int(input("Enter Coordinate Y: "))

    (B, G, R) = img[x, y] # Xác định màu

    print("R={}, G={}, B={}".format(R, G, B))

* + 1. **Thực nghiệm**

Nhập tọa độ x: 100

Nhập tọa độ y: 100

R=106, G=104, B=105

* 1. **Thay đổi kích thước**
     1. **Giới thiệu**

Thay đổi kích thước thước theo tỉ lệ ảnh

Yêu cầu:

* Lấy được hình ảnh, kích thước đã nạp ban đầu
* Lấy được giá trị kích thước
* Hiển thị hình ảnh sau khi đã thay đổi kích thước
* Lưu hình ảnh vào thu mục đã tạo
  + 1. **Code xử lý**

def resize(file,size,folder):

    img = cv2.imread(file) # đọc file hình ảnh

    (h, w, d) = img.shape # lấy kích thước ảnh ban đầu

    r = float(size) / w

    dim = (size, int(h \* r))

    resized = cv2.resize(img, dim) #chuyển đổi ảnh theo size đã chọn

    cv2.imshow("image resize", resized)

    ts = datetime.datetime.now() # lấy thời gian hiện tại

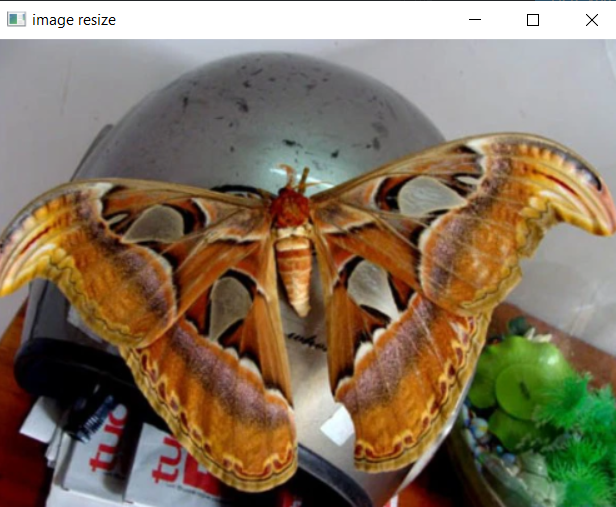
    cv2.imwrite(os.path.join(folder,ts.strftime("%Y-%m-%d\_%H-%M-%S")+".jpg") ,resized)# lưu hình ảnh vào thư mục

    cv2.waitKey(0)

    cv2.destroyAllWindows()

* + 1. **Thực nghiệm**

Nhập tỉ lệ muốn thay đổi: 500



* 1. **Chuyển ảnh chất lượng cao** 
     1. **Giới thiệu**

Chuyển hình ảnh sang chất lượng cao

Yêu cầu:

* Lấy được hình ảnh đầu vào
* In ra hình ảnh đã dược xử lý
* Lưu hình ảnh vào thu mục đã tạo
  + 1. **Code xử lý**

def apply\_sliding\_window(img, kernel, padding = 0, stride = 1): #padding = số pixel mở rộng ảnh input

                                                            #Stride: khoảng cách trượt.

                                                            #kernel: kích thước cửa sổ trượt

    h, w = img.shape[:2]            # lấy chiều cao & chiều rộng ảnh

    img\_p= np.zeros([h+2\*padding, w+2\*padding]) # Numpy = lập ma trận trống zero = chiều cao|| rộng + 2 lần pixel mở rộng

    img\_p[padding:padding+h, padding:padding+w] = img # gán ảnh vào khung ma trận nêu trên

    kernel = np.array(kernel) # lập cửa sổ trượt

    assert len(kernel.shape) == 2 and kernel.shape[0] == kernel.shape[1]    #  assert =kiểm tra đàm bảo (tương tự if) square kernel = số chiều = 2 và 2 chiều bằng nhau

    assert kernel.shape[0] % 2 != 0     # kernel size is odd number = số chiều cao là lè (đương nhiên chiều w cũng lẻ ) = khung vuông 2 chiều có kích thước là số lẻ

    k\_size = kernel.shape[0] # chiều cao của khung trượt

    k\_half = int(k\_size/2)   # nửa chiều cao của khung trượt

        y\_pos = [v for idx, v in enumerate(list(range(k\_half, h-k\_half))) if idx % stride == 0] # tập vị trí y (dọc)

    x\_pos = [v for idx, v in enumerate(list(range(k\_half, w-k\_half))) if idx % stride == 0] # tập vị trí x (ngang)

    new\_img = np.zeros([len(y\_pos), len(x\_pos)]) #lập ma trận khung trồng (chuần bị điền ảnh vào khung trượt)

    for new\_y, y in enumerate(y\_pos): #chạy vị trí y trong tập vị trí y đã xác định ở trên

        for new\_x, x in enumerate(x\_pos):  #chạyvị trí x trong tập vị trí x đã xác định ở trên

            if k\_half == 0: #bắt đầu điền ảnh từ vị trí 1/2 ảnh đầu (đã xác định ở trên)

                pixel\_val = img\_p[y, x] \* kernel # element-wise multiply = nhân -> mở rộng phần ảnh

            else:

                pixel\_val = np.sum(img\_p[y-k\_half:y-k\_half+k\_size, x-k\_half:x-k\_half+k\_size] \* kernel)  # mở rộng = tích vô hướng 2 vector

            new\_img[new\_y, new\_x] = pixel\_val   # gán vị trí ảnh phù hợp vào vị mới => chuẩn trượt tiếp

    return new\_img

def imgPro(img\_path):

    img = cv2.imread(img\_path) # đọc hình ảnh

    pro = apply\_sliding\_window\_on\_3\_channels(img,kernel=[[1]], padding=0, stride=2)

    cv2.imshow("Hinh lighten\_blur",pro)

    ts = datetime.datetime.now()# lấy thời gian hiện tại

    cv2.imwrite(os.path.join(folder,ts.strftime("%Y-%m-%d\_%H-%M-%S")+".jpg") ,pro)# lưu hình ảnh vào thư mục

    cv2.waitKey(0)

    cv2.destroyAllWindows()

* + 1. **Thực ngiệm**



* 1. **Làm mờ ảnh:**
     1. **Giới thiệu**

Chuyển hình ảnh ban đầu sang ảnh mờ

Yêu cầu:

* Lấy được hình ảnh đầu vào
* In ra hình ảnh đã được xử lý
* Lưu hình ảnh vào thu mục đã tạo
  + 1. **Code xử lý**

def blur(img\_path, folder):

    img = cv2.imread(img\_path) # đọc hình ảnh

    lighten\_blur\_img = apply\_sliding\_window\_on\_3\_channels(img, kernel=[[0.33, 0.33, 0.33], [0.33, 0.33, 0.33], [0.33, 0.33, 0.33]], padding=1, stride=1) #Chuyển ảnh mờ

    cv2.imshow("Hinh lighten\_blur",lighten\_blur\_img) #show hih2 ảnh

    ts = datetime.datetime.now()# lấy thời gian hiện tại

    cv2.imwrite(os.path.join(folder,ts.strftime("%Y-%m-%d\_%H-%M-%S")+".jpg") ,lighten\_blur\_img)# lưu hình ảnh vào thư mục

    cv2.waitKey(0)

    cv2.destroyAllWindows()

* + 1. **Thực nghiệm**



# Chương 9: KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN CỦA ĐỀ TÀI

## **Kết luận**

### 9.1.1Những kết quả đạt được

Đề tài đã:

+ Tìm hiểu về kỹ thuật multimedia

+ Thiết kế và lập trình được sản phẩm minh họa của đề tài dùng PYTHON trên công cụ Spyder (Anaconda3) với thư viện cv2, bao gồm:

* Sử dụng giọng nói
* Sử dụng bằng cách nhập
* Các xử lí ảnh cơ bản: chuyển ảnh xám, lấy kích thước ảnh, cắt ảnh, xoay ảnh, lấy giá trị màu tại vị trí, thay đổi kích thước, làm mờ ảnh, chuyển ảnh chất lượng cao
* Nạp hình ảnh từ file , nạp video từ file và camera

### 9.1.2 Hạn chế

Do thời gian hạn chế, nên App sản phẩm của đề tài còn một số hạn chế sau:

+ Phân chức năng nhận diện các giống loài bướm

+ Chưa bắt hết lỗi khi thực hiện nhập

## **Hướng phát triển**

### 9.2.1 Hướng khắc phục các hạn chế

Sẽ tiếp tục thực hiện :

+ Tìm hiểu cách xử lý nhận diện giống loài bướm

+ Hoàn thành bắt tất cả các lỗi khi nhập

### 9.2.2 Hướng mở rộng đề tài

Sẽ nghiên cứu và phát triển app sản phẩm trên Python dùng các thư viện mở.

Thiết kế phần giao diện ứng dụng

# DANH MỤC TÀI LIỆU THAM KHẢO

1. Võ Xuân Thể (2021), *Bài giảng học phần KỸ THUẬT LẬP TRÌNH ỨNG DỤNG ĐA PHƯƠNG TIỆN*, Học viện Công nghệ Bưu chính – Viễn thông, cơ sở TP.HCM (lưu hành nội bộ).
2. Nguồn media đầu vào:

+ hình ảnh:

Bướm chai xanh

<http://www.vncreatures.net/e_showpic.php?vietname=B%C6%B0%E1%BB%9Bm%20chai%20xanh&latinname=Graphium%20sarpedon&author=Ph%C3%B9ng%20M%E1%BB%B9%20Trung&imagepath=./pictures/insect/381_4>

Bướm lá khô

<https://i.vietgiaitri.com/2020/1/21/nhung-thien-tai-nguy-trang-trong-the-gioi-dong-vat-p1-3da181.jpg>

Bướm khế:

<https://kenh14cdn.com/txccccccccccccDqHcybwkI06dmgnY/Image/2012/06/120628CLbuom09_d40da.JPG>

Bướm đuôi kiếm:

<https://i.pinimg.com/originals/27/b4/b5/27b4b53e73907d976d07264971a2ad91.jpg>

Bướm đuôi lá xanh:

<https://photo-cms-kienthuc.zadn.vn/zoomh/800/uploaded/ctvkhoahoc/2016_11_15/top-nhung-loai-buom-dep-nhat-viet-nam.jpg>

Bướm hổ vằn:;

<https://farm7.static.flickr.com/6179/6218417201_acac5de168_o.jpg>

Bướm Phấn cam:

<https://live.staticflickr.com/6204/6162182582_4684c675fe_b.jpg>

+ video:

<https://www.youtube.com/watch?v=h_glLV23Z1M>

http://mobiclip.vn/899093/xem-video/buom-la-kho

<https://www.youtube.com/watch?v=tSNlI7bPjQY>

https://www.youtube.com/watch?v=0Qu\_CHol44c