

ĐẠI HỌC BÁCH KHOA HÀ NỘI  
TRƯỜNG CÔNG NGHỆ THÔNG TIN VÀ TRUYỀN THÔNG

— \* —



# BÁO CÁO PROJECT I

MÔN: Nghiên cứu tốt nghiệp 1

ĐỀ TÀI: **Landmark prediction**

Giảng viên hướng dẫn: **TS. Đinh Viết Sang**

Sinh viên thực hiện:

**Nguyễn Văn Kỳ – 20194600**

Ngành: Công nghệ thông tin Việt Nhật

*Hà Nội, tháng 7 năm 2023*

# MỤC LỤC

<b>LỜI NÓI ĐẦU</b>	<b>3</b>
<b>PHÂN CÔNG CÔNG VIỆC</b>	<b>4</b>
<b>NỘI DUNG CHI TIẾT</b>	<b>5</b>
I. Khái quát về trí thông minh nhân tạo (AI)	5
II. Khái quát về thị giác máy tính	5
III. Hình ảnh, tiếp nhận và xử lý; Biểu diễn ảnh	6
IV. Fast Fourier Transform - Biến đổi Fourier	7
V. Keypoint detection	7
VI. Image segmentation	9
VII. Image Recognition	10
<b>LẬP TRÌNH</b>	<b>10</b>
A. Binary semantic segmentation using U-Net	10
B. Multiclass semantic segmentation using U-Net	12
C. Face Segmentation	14
<b>KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN</b>	<b>16</b>
<b>TÀI LIỆU THAM KHẢO</b>	<b>17</b>

## LỜI NÓI ĐẦU

Lĩnh vực Computer Vision là một trong những lĩnh vực phát triển nhanh chóng của khoa học máy tính, và nó đang góp phần quan trọng vào cuộc cách mạng công nghiệp 4.0 hiện nay. Lý do tôi quyết định chọn đề tài này là vì tôi luôn hứng thú với khả năng của máy tính trong việc nhìn và hiểu thế giới xung quanh chúng ta. Và tôi nghĩ rằng trong tương lai đây sẽ là lĩnh vực được chú trọng phát triển và được nhiều sự quan tâm. Nghiên cứu trong lĩnh vực này giúp tôi hiểu rõ hơn về cách máy tính có thể phân tích và tạo ra thông tin từ dữ liệu hình ảnh và video, từ đó ứng dụng vào nhiều lĩnh vực thực tế như xe tự hành, y tế, an ninh, và nhiều ứng dụng khác.

Mục tiêu của đề tài này là học các kiến thức liên quan tới Computer Vision và làm các bài toán nhỏ để nâng cao kiến thức và áp dụng kiến thức tôi học được vào các bài toán thực tế. Đồng thời, tôi cũng đã tập trung nghiên cứu về việc ứng dụng các mô hình Deep Learning và Convolutional Neural Networks để nâng cao hiệu quả và độ chính xác của hệ thống.

Trong quá trình nghiên cứu, tôi đã tiến hành thu thập dữ liệu từ các nguồn đáng tin cậy và xây dựng các tập dữ liệu lớn để huấn luyện mô hình. Tôi cũng đã tham khảo các công trình nghiên cứu đã xuất bản, các bài báo khoa học, và sách vở để cập nhật kiến thức về Computer Vision và Deep Learning.

---

**PHÂN CÔNG CÔNG VIỆC**

Họ và tên	Công việc thực hiện	% + Đánh giá
Nguyễn Văn Kỳ Email: <a href="mailto:ky.nv194600@sis.hust.edu.vn">ky.nv194600@sis.hust.edu.vn</a>	- Đọc các tài liệu	(50%) Hoàn thành đúng tiến độ
	- Lập trình - Viết báo cáo -	(50%) Hoàn thành đúng tiến độ

## NỘI DUNG CHI TIẾT

### A. HỌC CÁC KIẾN THỨC

#### I. Khái quát về trí thông minh nhân tạo (AI)

#### II. Khái quát về thị giác máy tính

##### 1. Khái quát về computer vision

###### a. Định nghĩa

- Định nghĩa theo IBM: Computer vision is a field of artificial intelligence (AI) that enables computers and systems to derive meaningful information from digital images, videos and other visual inputs — and take actions or make recommendations based on that information.
- Định nghĩa theo Wikipedia: thị giác máy tính là 1 lĩnh vực bao gồm các phương pháp thu nhận, xử lý ảnh kỹ thuật số, phân tích và nhận dạng các hình ảnh.
- Là lĩnh vực liên ngành:
  - + Cần có các kiến thức: quang học, vật lý
  - + Cần có kiến thức về lập trình, kiến thức về giải thuật
  - + Cần kiến thức về thần kinh: hiểu cách thức hoạt động của não bộ
  - + Kiến thức robot, cơ điện tử
  - + Xử lý ảnh: phát triển các thuật toán xử lý ảnh, tăng giảm chất lượng ảnh, lọc
  - + Nhận diện mẫu: 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  
Những thông tin có thể lấy ra từ hình ảnh.
  - + Machine learning, AI, Deeplearning

###### b. Mục tiêu

- Đào tạo máy tính có thể hiểu và diễn giải thế giới như cách con người nhìn thấy.
- Biến đổi từ thông tin hình ảnh mà máy tính nhận vào (dưới dạng số) thành ngữ nghĩa

##### 2. Các dữ liệu có thể trích xuất từ ảnh

- Thông tin ngữ nghĩa
- Thông tin dữ liệu 3D của ảnh

##### 3. Các ứng dụng của computer vision

- Thiết bị đo đạc
- Dựa vào nhiều ảnh 2D tạo ra mô hình 3D
- Trích xuất các thông tin có ích
- Các ứng dụng robot
- Tăng tính bảo mật
- Hỗ trợ trong y tế: hỗ trợ tìm các bệnh ...
- Áp dụng trong công nghiệp: phân loại các sản phẩm, phát hiện các sản phẩm lỗi
- Hỗ trợ giao thông vận tải

##### 4. Các mức độ của computer vision

- Low-level: Thu thập ảnh, xử lý ảnh
- Middle-level: trích xuất các đặc trưng
- High-level: hiểu ngữ nghĩa từ bức ảnh

**5. Một số ghi chú và so sánh với**

- Ít có tính địa phương hơn hướng nghiên cứu NLP (Natural Language Processing)

**III. Hình ảnh, tiếp nhận và xử lý; Biểu diễn ảnh****1. Quy trình tiếp nhận và tạo hình ảnh**

- Tiếp nhận các hạt photon ánh sáng chuyển sang điện tích, đo đặc điện áp, chuyển điện áp tới các cảm biến, rồi mới tại tạo ra các điểm ảnh, bằng 2 loại cảm biến CCD và CMOS
- CCD: Charge Coupled Device - Linh kiện tích điện kép, đây là cảm biến dùng để chuyển đổi hình ảnh quang học sang tín hiệu điện. Cần phân tích tổng thể, các điểm xử lý 1 màu, các màu liền cạnh bổ sung cho nhau. Ảnh mượt, chậm, kích thước lớn
- CMOS: Complementary Metal-Oxide Semiconductor (Công nghệ chế tạo mạch tích hợp) - các điểm phân tích xử lý ảnh đơn lẻ xuất ra điểm ảnh ngay lập tức). Ảnh không mịn (nét), nhanh, chip kích thước nhỏ

**2. Cách nhận diện hình ảnh, và yếu điểm khả năng nhận diện hình ảnh của mắt người**

- Não người có những hạn chế về nhận diện hình ảnh, màu sắc (Do ảnh hưởng của yếu tố tâm lý, do tác động não người)
- Bản chất thế giới tự nhiên không có màu sắc mà chỉ tồn tại ở dạng các hạt proton

**3. Cách con người xử lý thông tin màu sắc**

- Ánh sáng đi qua thấu kính tới màng mắt (có các tế bào mắt), hứng các ánh sáng và truyền thông tin tới não xử lý
- Tế bào mắt gồm 2: loại tế bào que (môi trường ít ánh sáng, chỉ cảm nhận thông tin về độ sáng - đen trắng), tế bào nón (môi trường nhiều ánh sáng, cảm nhận được thông tin về màu sắc)
- Tế bào hình nón có 3 loại (lọc tốt ở những giải màu khác nhau, RGB)
- Tại sao mắt chúng ta chỉ cảm nhận được (vùng nhìn được) các hạt có bước sóng từ 400-700 nm. Do sự thích ứng tiến hóa của con người. Mặt trời là một ngôi sao lùn. Phổ bước sóng phổ biến mà mặt trời truyền xuống trái đất nhiều nhất từ khoảng 400 - 700 nm

**4. Xử lý ảnh trong lập trình**

- Thu nhận hình ảnh: RGB đuôi định dạng JPEG, PNG, vv
- Số hóa hình ảnh: lấy mẫu, lượng tử hóa (chọn giá trị đại diện, giá trị trung bình làm đại diện), kiểu thường là kiểu kiểu uint8
- Xử lý ảnh: làm nét, vv

## 5. Không gian màu - cách biểu diễn màu sắc

- Đa phần có 3 thành phần (Có định lý: chỉ cần 3 mã màu để mã hóa toàn bộ màu sắc)
- Hệ màu cộng (chiều 3 loại màu chồng lên nhau) và hệ màu trừ (sử dụng công nghệ để chỉ có ánh sáng màu mong muốn đi qua kiểu trừ)
- Các loại: RGB (phù hợp với biểu diễn máy tính, khó cho con người), HSV (màu, đậm nhạt, sáng tối - yếu tố quan trọng nhất), LAB (dựa vào đặc điểm của mắt người)
- Tọa độ gốc của ma trận biểu diễn ảnh  $O(0, 0)$  góc trên cùng bên trái (chứ không giống gốc tọa độ đề các ở dưới cùng bên trái)

## 6. Biến đổi ảnh

- 3 kiểu: xử lý độc lập (Isolated trans), biến đổi dựa vào hàng xóm (local trans), biến đổi dựa vào toàn bộ ảnh (global trans)
- Negative: lấy phần bù  $[0, L-1]$  L ngưỡng sáng
- Log (làm sáng ảnh) và Inverse Log (làm tối ảnh)
- Gamma transformation (tương tự log) nhưng dùng để biến đổi ảnh sáng tối tốt hơn
- Biến đổi tuyến tính (từng đoạn có đạo hàm tuyến tính)

## 7. Các chỉ số

- Độ sáng của ảnh: trung bình độ sáng của tất cả các điểm ảnh
- Độ tương phản (tương phản càng lớn chúng ta càng dễ dàng nhận biết các vật thể):

## IV. Fast Fourier Transform - Biến đổi Fourier

### 1. Khái quát

- Fast Fourier Transform (FFT) là một thuật toán quan trọng và phổ biến trong xử lý ảnh. Nó cho phép chúng ta chuyển đổi một hình ảnh từ miền không gian sang miền tần số và ngược lại. Điều này rất hữu ích trong nhiều ứng dụng xử lý ảnh như lọc nhiễu, nâng cao độ tương phản, nhận dạng vật thể và nén dữ liệu ảnh. Xử lý làm rõ ảnh.

### 2. Nguyên lý hoạt động

$$X(t) = \frac{A_0}{2} + \sum_{n=1}^N (A_n \sin(\frac{2\pi nt}{P} + \phi_n)) = \sum_{n=-N}^N (C_n e^{\frac{2\pi j n t}{P}})$$

Trong đó,  $P$  là chu kỳ của tín hiệu và

$C_n$  được gọi là các hệ số Fourier của tín hiệu  $x(t)$ . Các hệ số này được tính bằng cách:

$$C_n = \frac{1}{P} \int_0^P e^{-\frac{2\pi j n t}{P}} x(t) dt$$

## V. Keypoint detection

1. Ý tưởng
  - Có 1 bộ phát hiện, trích xuất đặc trưng cục bộ - cùng 1 điểm bộ phát hiện ở nhiều ảnh khác nhau
  - Cần bộ trích xuất đặc trưng với mỗi điểm khác nhau phải khác nhau để phân biệt và có thể matching các điểm ở 2 ảnh khác nhau
2. Các bước cơ bản
  - Bước 1: Phát hiện điểm đặc trưng (interest point)
  - Bước 2: Trích xuất mô tả bất biến
3. Những câu hỏi
  - Điểm đặc trưng key point đến từ đâu trong bức ảnh ?
    - + Các điểm nằm ở góc đối tượng (corner point), image gradient phải có 2 hoặc nhiều hướng (**đạo hàm riêng theo các hướng có giá trị tuyệt đối là lớn, ít nhất theo 2 hướng**) - **giá trị độ sáng thay đổi rất lớn**
    - + Corners point là các điểm có thể lặp lại (để matching), và khác biệt

$$E(u, v) = \sum_{x, y} w(x, y) [I(x + u, y + v) - I(x, y)]^2$$

- + Từ nhu cầu tìm điểm góc, nhưng nếu tính toán tất cả các điểm thì tốn nhiều thời gian. Nên ta chỉ cần ước lượng độ dốc của điểm đó và vùng xung quanh. → Chuyển về biến đổi Taylor để giải quyết → Xác định  $\lambda_1$  và  $\lambda_2$  của elip đồng tâm. Góc nếu  $\lambda_1$  và  $\lambda_2$  lớn (độ thay đổi về đường kính của elip nhỏ)



## Classification of image points using eigenvalues of $M$

### Cornerness

$$C = \lambda_1 \lambda_2 - \alpha (\lambda_1 + \lambda_2)^2$$

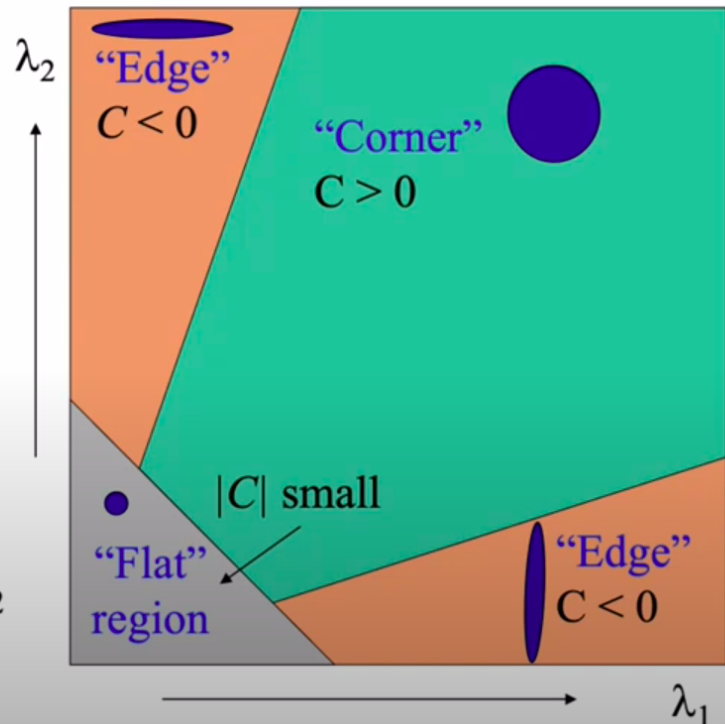
$\alpha$ : constant (0.04 to 0.06)

Remember your linear algebra:

Determinant:  $\det(A) = \prod_{i=1}^n \lambda_i = \lambda_1 \lambda_2 \cdots \lambda_n$ .

Trace:  $\text{tr}(A) = \sum_i \lambda_i$ .

$$C = \det(M) - \alpha \text{trace}(M)^2$$



#### ❖ Ghi nhớ:

- Tích của trị riêng = định thức của ma trận
- Tổng của trị riêng = vết của ma trận
- **Keypoint detection (Feature Matching - Harris corner detector), Blob detection (tuần 6)**

- Làm sao để chúng ta mô tả được các thông tin lân cận ? Làm sao để phân biệt điểm đó với các điểm khác ?
- Và làm sao để matching 2 điểm ?

## VI. Image segmentation

1. Mục đích
  - Chia ảnh thành nhiều vùng có ý nghĩa
  - Phục vụ cho bài toán GR1
  - Lấy các đặc trưng của các vật thể trong ảnh
2. Cách tiếp cận
  - Pixel-based approach
  - Dựa vào cạnh (edge-based approach): sự thay đổi đột ngột, ranh giới bị bo ở trong
  - Dựa vào sự đồng nhất của các khu vực: cùng màu, cùng kết cấu, cùng độ sáng
  - Hybrid = cách tiếp cận Cạnh + tiếp cận vùng
3. Cách tiếp cận dựa vào các Pixel-based approach
  - Dựa vào ngưỡng(thresholding) hoặc theo vùng (Clustering)
4. Phân đoạn dựa trên mô hình học máy (Machine Learning-based Segmentation)
  - Phân đoạn dựa trên mô hình học máy sử dụng các mô hình học máy như Convolutional Neural Networks (CNNs) để tự động học các đặc trưng từ dữ liệu và phân loại các vùng trong hình ảnh.

- Một trong những mô hình CNN phổ biến trong phân đoạn ảnh là U-Net. U-Net là một mạng nơ-ron tích chập được thiết kế đặc biệt để xử lý các hình ảnh y tế và được phẩm. Nó bao gồm hai phần chính là "encoder" để trích xuất thông tin đặc trưng từ hình ảnh và "decoder" để tái tạo lại hình ảnh được phân vùng. U-Net sử dụng kỹ thuật "skip connection" để kết nối các tầng trích xuất đặc trưng từ encoder đến decoder, giúp tái tạo hình ảnh chính xác hơn.

## VII. Image Recognition

1. Làm sao để từ ma trận số trở thành các ngữ nghĩa ?
  - Sử dụng học máy để tạo bộ phân lớp có khả năng học các quy luật và phân biệt các vật thể
2. a.

VIII. a

# LẬP TRÌNH

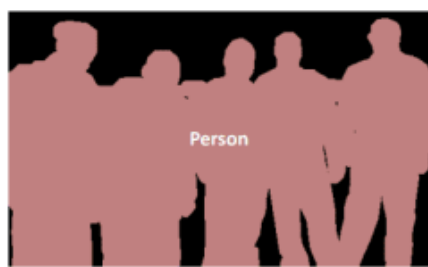
## A. Binary semantic segmentation using U-Net

Segmentation ra đời để chia ảnh thành nhiều vùng khác nhau và có thể tìm được đúng hình dạng của các đối tượng. Nhận biết được hình dạng của đối tượng cho ta nhiều tri thức để hiểu rõ và nâng cao khả năng “nhìn” của máy tính.

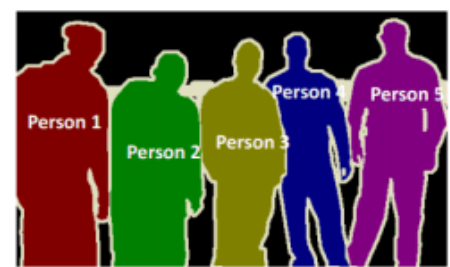
1. Phân loại bài toán Image segmentation
  - Semantic segmentation: Thực hiện segment với từng lớp khác nhau, ví dụ: tất cả người là 1 lớp, tất cả ô tô là 1 lớp.
  - Instance segmentation: Thực hiện segment với từng đối tượng trong một lớp. Ví dụ có 3 người trong ảnh thì sẽ có 3 vùng segment khác nhau cho mỗi người.



Object Detection



Semantic Segmentation



Instance Segmentation

Nguồn:

<https://towardsdatascience.com/understanding-semantic-segmentation-with-unet-6be4f42d4b47>

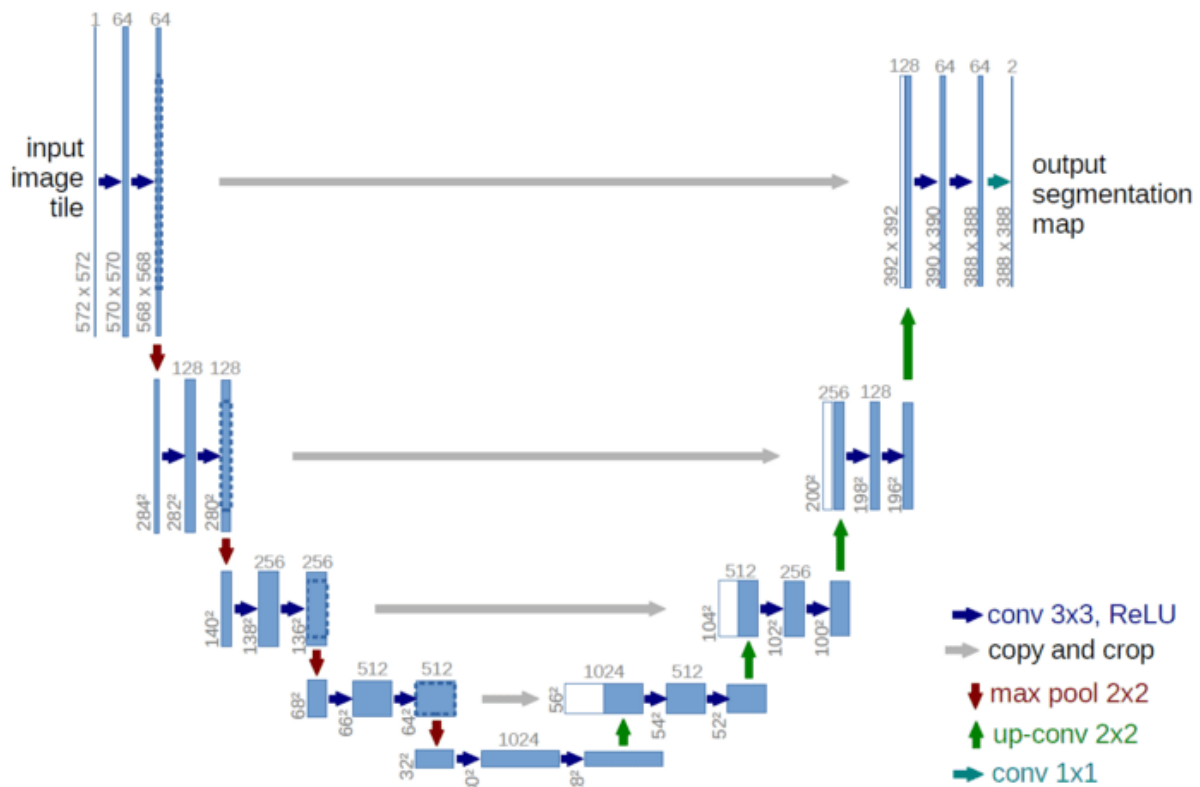
## 2. Kiến trúc mạng Unet

- Lý do ra đời: nếu dùng full convolution của CNN, với kích thước tăng rất nhỏ các tham số Weight tăng rất nhanh khiến cho việc xử lý và tính toán trở nên khó khăn. Do đó Unet ra

đời để chỉ giữ lại các đặc trưng quan trọng nhất để vẫn có thể dự đoán chính xác nhưng lại giảm các tham số giúp việc tính toán dễ dàng và nhanh hơn.

- U-Net được phát triển bởi Olaf Ronneberger. để dùng cho image segmentation trong y học. Kiến trúc có 2 phần đối xứng nhau được gọi là encoder (phần bên trái) và decoder (phần bên phải).
- Phần Encoder là ConvNet các layer sau thì width, height giảm nhưng depth tăng. Mục đích của encoder là cô đọng thông tin ngữ cảnh
- Phần Decoder có mục đích là khôi phục lại kích thước của ảnh gốc, up-conv dùng để tăng kích thước của ảnh, depth giảm. Mục đích: giải mã thông tin, xác định chính xác các đối tượng

2



**Fig. 1.** U-net architecture (example for 32x32 pixels in the lowest resolution). Each blue box corresponds to a multi-channel feature map. The number of channels is denoted on top of the box. The x-y-size is provided at the lower left edge of the box. White boxes represent copied feature maps. The arrows denote the different operations.

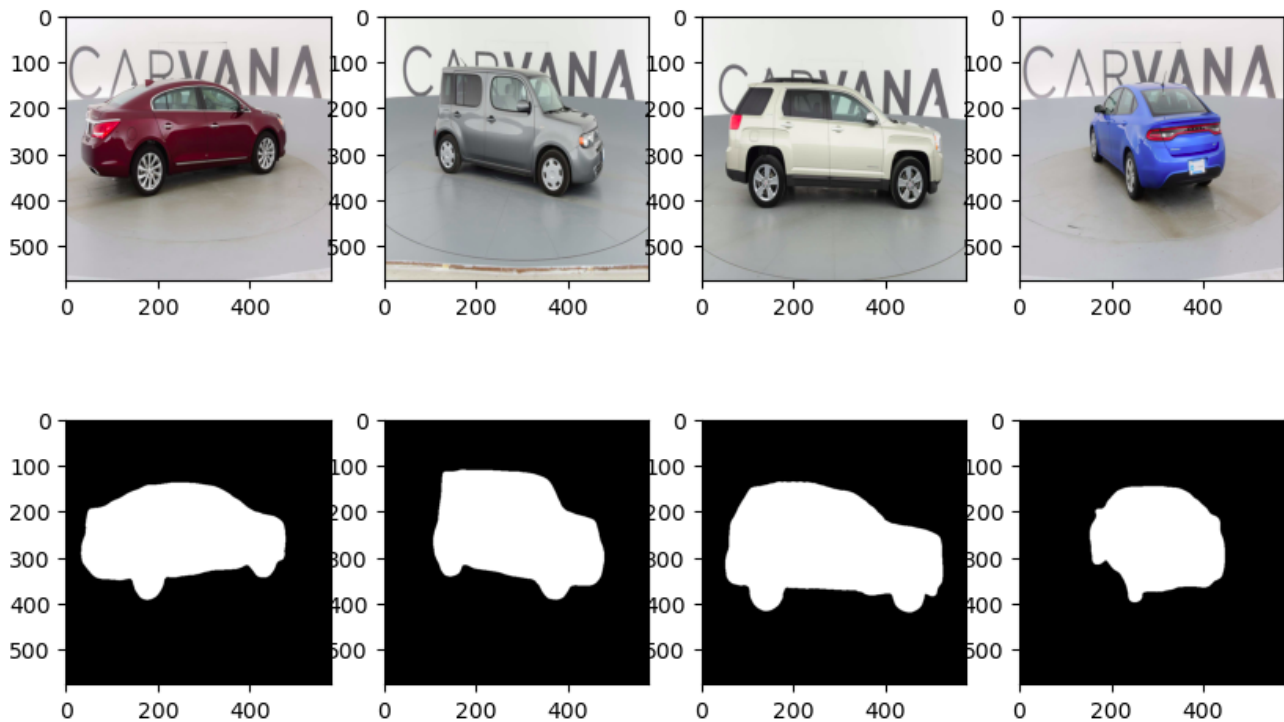
Nguồn: <https://arxiv.org/pdf/1505.04597.pdf>

### 3. Dataset: Đã nghiên cứu trên tập dữ liệu Carvana Car Image

- Nguồn kaggle: <https://www.kaggle.com/c/carvana-image-masking-challenge>

### 4. Kết quả thu được

- Sử dụng modal như trong paper gốc và hướng dẫn từ <https://pyimagesearch.com/2022/02/21/u-net-image-segmentation-in-keras/>
- Sử dụng thư viện keras để lập trình



### 5. Đánh giá

## B. Multiclass semantic segmentation using U-Net

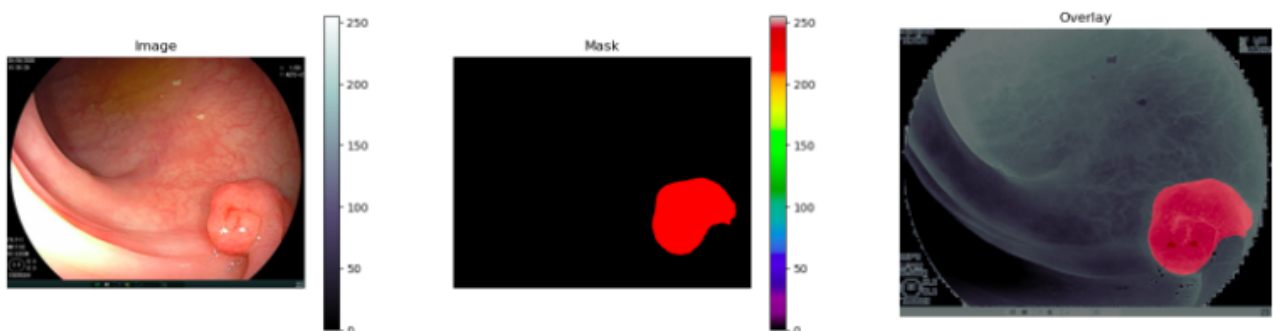
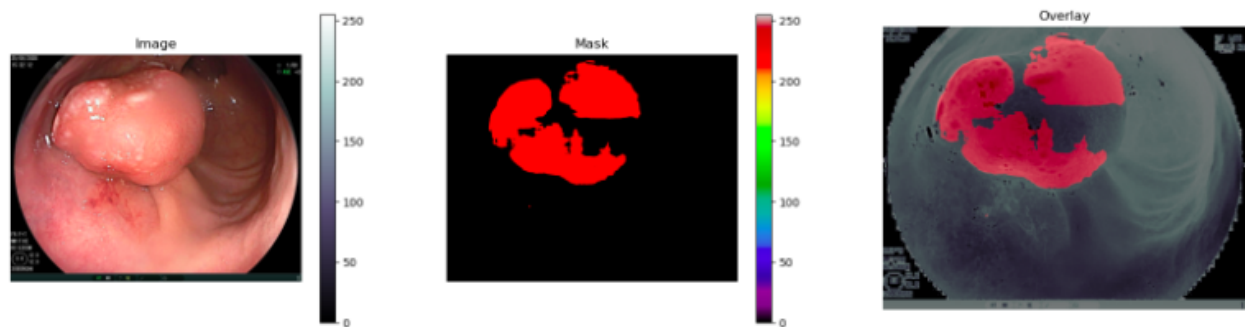
### 1. Dataset sử dụng

- <https://www.kaggle.com/competitions/bkai-igh-neopolyp>
- Một bộ dữ liệu lớn hơn có tên NeoPolyp-Large chứa khoảng 7500 hình ảnh với bốn chế độ màu khác nhau (WLI, BLI, LCI) với các chú thích chi tiết phần ác tính, lành tính và background.
- Dataset được cung cấp bởi Vingroup

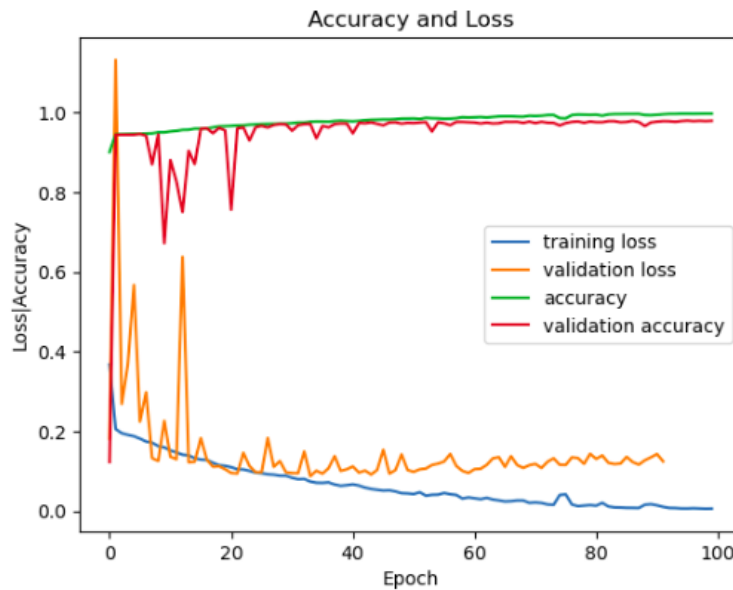
### 2. Tiền xử lý

- Ở bài toán Multiclass này khó nhất ở phần tiền xử lý và tô màu
  - Sử dụng bảng màu HUE để tăng độ ổn định khi chúng ta xử lý ảnh. Sử dụng các khoảng màu để xác định các mảng màu đỏ, xanh, background và tạo thành các mask mới giá trị tương ứng xanh: 1 - lành tính, đỏ: 2 - ác tính, 0 - background.
  - Tô màu: từ mask 0,1,2 tô màu thành mảng màu đen, xanh và đỏ để quan sát và kiểm tra xem độ chính xác của thuật toán
3. Về model sử dụng lại model Unet của bài toán Binary semantic segmentation ở trên.
- Sử dụng thêm các kĩ thuật Early Stopping của thư viện Keras, ReduceLROnPlateau và lưu lại model tốt nhất qua mỗi epoch
  - Quá trình đào tạo 100 epoch trên GPU P100 của kaggle
  - Sử dụng activation function là hàm softmax, và function loss là categorical\_crossentropy
4. Kết quả thu được

a. Kết quả dự đoán hình ảnh



b. Kết quả độ chính xác của mô hình



### c. Kết quả submit trên kaggle

56	Nguyen Van Ky		0.28440	13	14d
----	---------------	--	---------	----	-----

Your Best Entry!  
Your submission scored 0.08981, which is not an improvement of your previous score. Keep trying!

### d. Nhận xét

- Kết quả trên kaggle còn thấp, cần cải thiện tiền xử lý ảnh, sửa các lỗi có thể xảy ra trong code để tăng độ chính xác hơn nữa
- Sau 1 thời gian tương đối dài không thể tìm ra nguyên nhân đã chuyển sang chủ đề tiếp theo
- Nguyên nhân có thể do mâu thuẫn: activation function là hàm softmax, và function loss là categorical\_crossentropy

## C. Face Segmentation

### 1. Đặt vấn đề

- Phân đoạn khuôn mặt trong ảnh là một bài toán quan trọng trong lĩnh vực thị giác máy tính và trí tuệ nhân tạo.
- Việc phân đoạn khuôn mặt giúp tách riêng từng phần khác nhau của khuôn mặt từ nền ảnh và có thể ứng dụng trong nhiều lĩnh vực như nhận dạng khuôn mặt, bảo mật vv.

- Sau khi nghiên cứu kiến thức của bài toán này có thể sử dụng những kiến thức để nâng cao accuracy của bài toán Multi Class semantic segmentation ở trên, để có thể nâng cao điểm submit trên kaggle hơn nữa

## **2. Các mục nghiên cứu**

- Nghiên cứu các phương pháp và kỹ thuật phân đoạn khuôn mặt trong ảnh.
- Xây dựng một mô hình phân đoạn khuôn mặt chính xác và hiệu quả.
- Đánh giá hiệu suất

## **3. Phương pháp sử dụng mô hình Unet**

## **4. Những khó khăn gặp phải**

- Ảnh nền và các chi tiết phức tạp trong hình ảnh (ví dụ: tóc, áo, kính) có thể làm cho việc phân đoạn khuôn mặt trở nên khó khăn, đặc biệt khi các phần này gần giống với khuôn mặt.
- Ánh sáng có thể thay đổi khác nhau có thể làm việc dự đoán kém chính xác hơn

## **5. Kết quả thu được**

- Hiện tại chưa hoàn thiện mô hình, chương trình vẫn chưa chạy được nên chưa có kết quả
- Các bước tiếp theo đang tham khảo code trên github để tinh chỉnh và chỉnh sửa code của bản thân

## **6. Đánh giá**

- Cần hoàn thiện mô hình và có kết quả cụ thể để đánh giá chính xác hơn

## KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN

### 1. Kiến thức thu được:

- Các kiến thức cơ bản về Computer Vision và cách máy tính hiểu các thông tin hình, trích xuất nhận diện, dự đoán các vật thể
- Viết báo cáo dự án: xác định được các điểm thiếu sót của bản thân và định hướng để tiếp tục học tập và nghiên cứu. Trước khi hoàn thành báo cáo cần kiểm tra lỗi trình bày: font chữ, căn lề, hình ảnh, số liệu, chính tả.
- Biết và sử dụng các model như UNET

### 2. Khó khăn:

- Tiếp cận công nghệ mới về Computer Vision và không có nền tảng cơ bản trước về machine learning khiến tôi gặp nhiều khó khăn khi nghiên cứu
- Khó khăn trong việc tối và nâng cao khả năng dự đoán của mô hình

### 3. Hướng phát triển:

- Nâng cao độ chính xác của mô hình
- Học tập thêm các kiến thức chuyên sâu hơn về Deep learning và Computer vision
- Áp dụng các pretrain model để có kết quả tốt hơn



## TÀI LIỆU THAM KHẢO

- [1] Thị giác máy tính  
<https://www.youtube.com/watch?v=x21gTtVwKrU&list=PL50pDT4Q2doQYwSJ7GglOQrlx0tgle-P0>
- [2] Dive into Deep Learning  
[https://d2l.ai/chapter\\_computer-vision/index.html](https://d2l.ai/chapter_computer-vision/index.html)