

**Trường Đại học Khoa học Tự nhiên**  
***Khoa Công nghệ Thông tin***



**Project: Food Image Retrieval**

Nhóm 3

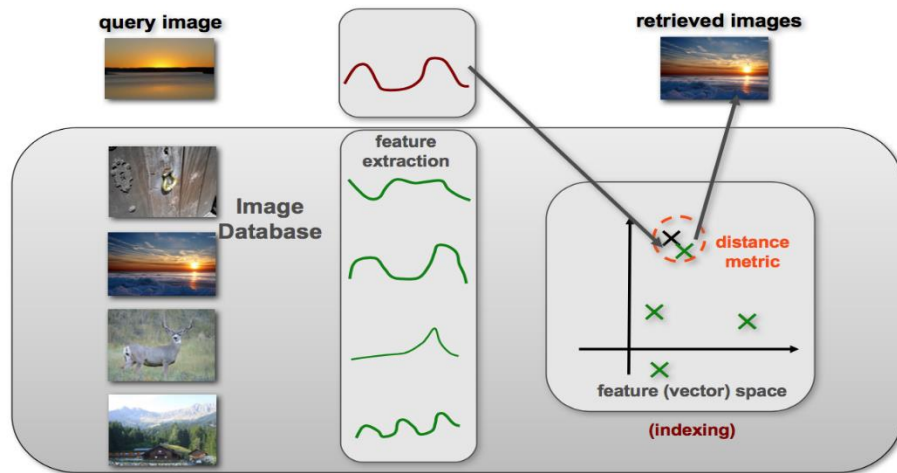
Tổng Gia Huy – 21127307

Phan Đăng Anh Khôi – 21127325

## Contents

<b>I.</b>	<b>Sơ đồ chung của bài toán truy vấn</b>	<b>2</b>
<b>II.</b>	<b>Related Work</b>	<b>2</b>
<b>1/</b>	<b>[Food image classification and image retrieval based on visual features and machine learning] - [Phân loại hình ảnh và truy xuất hình ảnh dựa trên đặc trưng hình ảnh và học máy]</b>	<b>2</b>
i.	Tóm Tắt	2
ii.	Related Work của bài báo	2
a)	Phương pháp phân loại và truy vấn hình ảnh dựa trên đặc điểm thị giác của hình ảnh	2
b)	Phương pháp phân loại và truy vấn hình ảnh dựa kết hợp học máy và đặc điểm thị giác hình ảnh	4
c)	Phân loại hình ảnh thực phẩm và các phương pháp truy xuất hình ảnh dựa trên đặc trưng hình ảnh và học máy	5
iii.	Kết quả	6
<b>2/</b>	<b>[Food Image Retrieval with Gray Level Co-Occurrence Matrix Texture Feature and CIE L*a*b* Color Moments Feature] [Truy xuất ảnh thức ăn sử dụng đặc trưng kết cấu Gray Level Co-Occurrence Matrix và mô men màu CIE L*a*b*]</b>	<b>7</b>
i.	Tóm Tắt	7
ii.	Dữ liệu	7
iii.	Phương Pháp	8
a)	Tiền xử lý	9
b)	Phân đoạn	9
iv.	Phương pháp - Gray Level Co-Occurrence Matrix	9
v.	Phương pháp - Đặc Trưng màu CIE L*a*b	10
vi.	Phương pháp - Độ Đo Khoảng Cách	10
vii.	Kết quả	11
<b>III.</b>	<b>Giải pháp của nhóm</b>	<b>12</b>
<b>IV.</b>	<b>Các kết quả truy vấn</b>	<b>13</b>
<b>V.</b>	<b>MAP của mô hình</b>	<b>21</b>
<b>VI.</b>	<b>Cải tiến của nhóm</b>	<b>21</b>

## I. Sơ đồ chung của bài toán truy vấn



Hình ảnh hệ thống truy vấn đơn giản (Nguồn: <https://github.com/pochih/CBIR/blob/img/CBIR.png>)

- Ở đây nhóm cũng theo sơ đồ này theo các bước sau:

- 1/ Chuẩn bị dữ liệu.
- 2/ Rút trích đặc trưng từ dữ liệu.
- 3/ Truy vấn.

## II. Related Work

### 1/ [Food image classification and image retrieval based on visual features and machine learning] - [Phân loại hình ảnh và truy xuất hình ảnh dựa trên đặc trưng hình ảnh và học máy]

#### i. Tóm Tắt

- Phương pháp truy vấn và phân loại ảnh truyền thống dựa trên từ khóa không đáp ứng nhu cầu hằng ngày của con người => phương pháp nhận dạng ảnh dựa trên nội dung trong ảnh ra đời.
- Bài báo này đề xuất nghiên cứu về phương pháp phân loại hình ảnh thực phẩm và truy xuất hình ảnh dựa trên các đặc trưng thị giác và học máy - phương pháp truy xuất và phân loại hình ảnh thực phẩm dựa trên mạng Faster R-CNN.
- Tập dữ liệu là Dish-233 gồm 233 món ăn và 49.168 hình ảnh.

#### ii. Related Work của bài báo

##### a) Phương pháp phân loại và truy vấn hình ảnh dựa trên đặc điểm thị giác của hình ảnh

##### (1) Phương pháp phân loại dựa trên các đặc điểm thị giác trực tiếp của ảnh

- Đặc điểm màu sắc (RGB):  $F = r[R] + g[G] + b[B]$ . Ta cũng đã biết mô hình RGB trên máy không thể ánh xạ hết các màu sắc trong tự nhiên  $\Rightarrow$  Đổi sang hệ tọa độ HSV.

$$H = \begin{cases} \text{undefined} & \max = \min \\ 60^\circ \times \frac{g-b}{\max-\min} + 0^\circ & \max = r, g \geq b \\ 60^\circ \times \frac{g-b}{\max-\min} + 360^\circ & \max = r, g < b \\ 60^\circ \times \frac{b-r}{\max-\min} + 120^\circ & \max = g \\ 60^\circ \times \frac{r-g}{\max-\min} + 240^\circ & \max = b, \end{cases}$$

$$S = \begin{cases} 0 & \max = 0 \\ 1 - \frac{\min}{\max} & \max \neq 0, \end{cases} \quad V = \max.$$

- Đặc điểm hình dạng (shape, texture): đôi khi màu sắc thôi là chưa đủ và ta phải cần đặc điểm hình dạng của vật thể trong ảnh – contour (đường viền) và edge (biên cạnh).
  - + Nhắc về cạnh, định nghĩa nó là sự thay đổi đột ngột trong màu sắc. Đối với đường viền thì nó cũng được lấy từ cạnh nhưng nó khác biệt với cạnh ở chỗ tạo thành một boundary xung quanh vật thể, giúp xác định ranh giới giữa vật và nền.
  - + Sobel, Canny, Laplacian là bước đầu tiên trong việc rút trích đặc trưng. Tiếp theo, để rút trích đặc trưng hình dạng, phương pháp phổ biến được sử dụng là sử dụng góc (angle) để đếm số cạnh và biểu diễn tần suất (tạo ra histogram) tạo ra descriptor để từ đó lấy thông tin đặc trưng.

## (2) Phương pháp truy vấn dựa vào Hash

- Khi nhắc đến truy vấn, người ta thường tìm các điểm gần nhất với chỉ mục trong tập dữ liệu.
- Phương pháp cấu trúc cây để tìm chỉ mục: giải quyết được truy vấn nhưng gặp phải vấn đề khi dữ liệu tăng thì cấu trúc cây tăng lên theo cấp số nhân.  $\Rightarrow$  Sử dụng các loại hash: spectral hash (SH), position-sensitive hash (LSH),...
- Thuật toán truy vấn sử dụng spectral hash: Giả sử có  $m$  mẫu, mỗi mẫu kích thước  $n$  chiều, tính giá trị trung bình của mỗi đặc trưng.

$$u_j = \sum_{i=1}^m x_j^i / m$$

với  $i$  là số thứ tự của mẫu,  $j$  là chiều của vector mẫu, và  $x_{ij}$  là giá trị của đặc trưng ở chiều thứ nhất của mẫu thứ nhất.

**b) Phương pháp phân loại và truy vấn hình ảnh dựa kết hợp học máy và đặc điểm thị giác hình ảnh.**

**(1) BOW - Sử dụng vector đặc trưng gồm từ vựng để đại diện cho văn bản bỏ qua trình tự.**

1/ Xây dựng không gian tỉ lệ: Miêu tả tính không thay đổi theo tỉ lệ của hình ảnh.

$$L(x, y, \sigma) = G(x, y, \sigma) \times (x, y)$$

2/ Phát hiện điểm cực trị trong không gian DOG.

3/ Lựa chọn điểm đặc trưng: Xác định vị trí và tỉ lệ của điểm đặc trưng, loại bỏ điểm ảnh có độ cong cục bộ không đối xứng.

4/ Gán vector đặc trưng 128 chiều: Tính giá trị magnitude và hướng gradient tại mỗi điểm đặc trưng:

$$m(x, y) = \sqrt{(L(x+1, y) - L(x-1, y))^2 + (L(x, y+1) - L(x, y-1))^2}$$

$$\theta(x, y) = \text{atan2}(L(x, y+1) - L(x, y-1), L(x+1, y) - L(x-1, y))$$

Sau đó sử dụng thuật toán phân cụm như k-means hoặc SVM để giảm bớt dữ liệu và độ phức tạp tính toán.

5/ Sử dụng BOW lên ảnh.

**(2) Thuật toán truy vấn hình ảnh sau BOW**

1/ Nhập hình ảnh cần tìm và cơ sở dữ liệu.

2/ Sử dụng mô hình đã huấn luyện imagenet-vgg-very-deep-16.mat, thuật toán CNN để trích xuất đặc trưng của hình ảnh.

3/ Tính toán độ tương đồng giữa hình ảnh tìm kiếm và hình ảnh cơ sở dữ liệu bằng khoảng cách Euclid hoặc góc.

4/ Sắp xếp hình ảnh theo kích thước tích vô hướng, chọn N hình ảnh đầu tiên.

5/ Phân loại N hình ảnh này dựa trên mô hình BOW.

6/ Loại bỏ các hình ảnh thuộc loại không đồng nhất, giữ lại hình ảnh thuộc loại lớn.

7/ Cung cấp kết quả tìm kiếm cuối cùng cho người dùng.

**c) Phân loại hình ảnh thực phẩm và các phương pháp truy xuất hình ảnh dựa trên đặc trưng hình ảnh và học máy**

**(1) Mạng CNN**

- Đã được giới thiệu ở môn Thị giác máy tính gồm các đặc điểm giúp cho mạng trở nên vượt trội khi làm việc với ảnh.
- Local Connectivity - CNN chỉ kết nối nơ-ron trong một khu vực cục bộ của ảnh đầu vào;
- Shared Weights - trong các lớp tích chập của CNN, các bộ lọc được dùng để trích xuất đặc trưng từ ảnh được chia sẻ trọng số (tức các vị trí trong ảnh);
- Spatial or Temporal Sub-sampling - sau mỗi lớp tích chập là lớp subsampling giúp giảm kích thước đầu ra bằng cách giảm sự biểu diễn của các đặc trưng.

**(2) Cấu trúc mạng CNN**

- Gồm 3 lớp, convolution, subsampling và dense.

**(3) Trích xuất đặc trưng hình ảnh vùng miền dựa trên Faster R-CNN**

- So với các phương pháp thị giác truyền thống, CNN có thể trích xuất thông tin ngữ nghĩa giàu hơn.
- Bài báo đã tận dụng đầy đủ thuật toán Faster R-CNN để trích xuất các khu vực của hình ảnh thực phẩm, rút trích đặc trưng của các vùng thực phẩm và áp dụng chúng vào các nhiệm vụ phân loại và truy xuất hình ảnh thực phẩm.
- Để trích xuất hiệu quả các đặc trưng hình ảnh thực phẩm, 2 bước sau đây được dùng là: Fine-tune FasterR-CNN (tinh chỉnh mạng FasterR-CNN), Trích xuất đặc trưng CNN dựa trên các khu vực phát hiện thực phẩm.  
1/ Lựa chọn các danh mục thực phẩm đã được hiệu chỉnh từ thư viện visual gene. Sau đó, tinh chỉnh lại FasterR-CNN và cuối cùng thu được các vùng ứng viên của mỗi hình ảnh thực phẩm.  
2/ Sau khi tinh chỉnh FasterR-CNN, mô hình đó được dùng để thu các vùng ứng viên của hình ảnh thực phẩm và điểm số của mỗi ứng viên.
- Đối với các vùng ứng viên có điểm số cao cho mỗi hình ảnh, các đặc trưng của lớp FC7 được trích xuất bằng mạng AlexNet dựa trên các tọa độ của các khung ứng viên, và sau đó các đặc trưng của các vùng có điểm số cao hơn được nối tiếp để thu được biểu diễn đặc trưng của hình ảnh cuối cùng.

- Khi input một hình ảnh query, các đặc trưng thị giác của hình ảnh thực phẩm được trích xuất dựa trên Faster R-CNN đã được điều chỉnh và CNN. Sau đó phép tính độ tương đồng được thực hiện với cơ sở dữ liệu truy vấn để trả lại kết quả truy vấn.

### iii. Kết quả

Method	Accuracy rate
CNN-G	0.356
CNN-G-F	0.704
Faster-R-CNNG	0.748
Article method	0.754

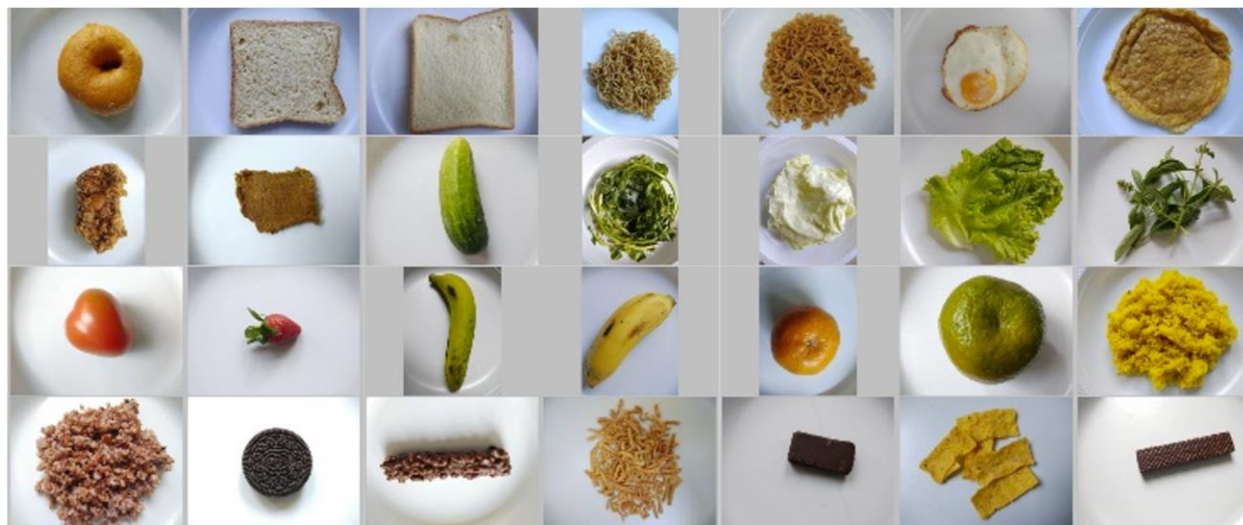
*Độ chính xác của các phương pháp được đề cập trong bài báo*

## 2/ [Food Image Retrieval with Gray Level Co-Occurrence Matrix Texture Feature and CIE L\*a\*b\* Color Moments Feature] [Truy xuất ảnh thức ăn sử dụng đặc trưng kết cấu Gray Level Co-Occurrence Matrix và mô men màu CIE L\*a\*b\*]

### i. Tóm Tắt

- Truy vấn văn bản không thể đáp ứng được nhu cầu của người dùng khi họ cần tìm thông tin và công thức của món ăn bằng hình ảnh. Do vậy, một hệ thống truy vấn ảnh chuyên dụng dựa vào nội dung là cần thiết cho việc tìm kiếm thực phẩm.
- Bài báo này đề xuất tác vụ truy xuất ảnh thức ăn mà sau đó có thể được ghép với công thức tương ứng của nó. Các đặc trưng được sử dụng trong tác vụ này là kết cấu (Gray-Level Co-occurrence Matrix) và mô men màu (CIE L\*a\*b\*).
- Với 1303 mẫu huấn luyện và 31 mẫu kiểm tra, hệ thống có thể đạt được kết quả tốt tới **97.6% Mean Average Precision** trong tập 10. Các độ đo khoảng cách Euclid, Manhattan, Minkowski, Canberra cũng được dùng để đánh giá độ hiệu quả của truy vấn.

### ii. Dữ liệu



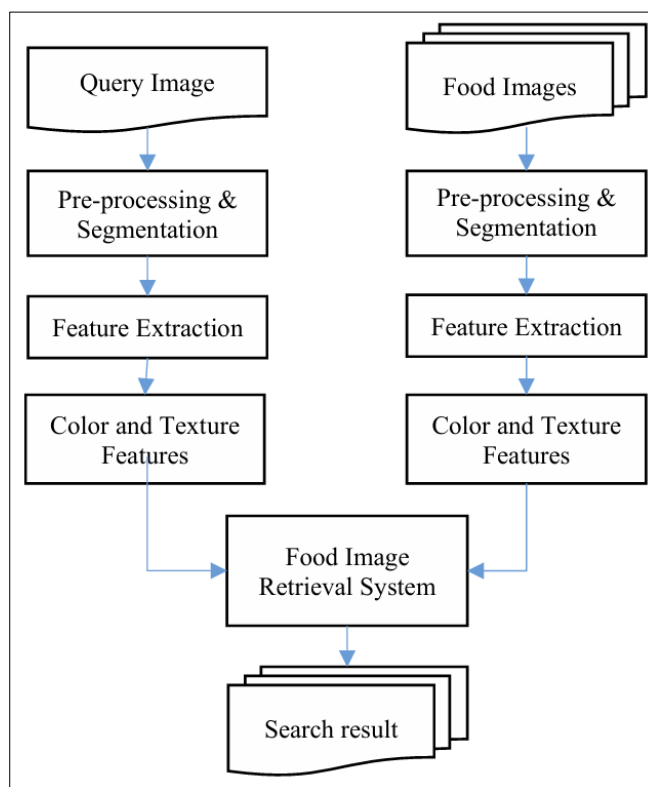


Code	Class	Image Count
001	Donut	71
002	Wheat bread	71
003	White bread	17
004	Instant fried noodles	17
005	Flat noodles	17
006	Sunny side egg	71
007	Omellete	17
008	Fried chicken	17
009	Rendang (Indonesian beef simmered with coconut milk)	53
...	...	...
030	Milo nuggets	35
031	Monde's Genji Pie	17

Gồm 1334 ảnh và 32 lớp.

Được chụp ở nhiều góc độ, cao độ, và trạng thái trước và sau khi ăn.

### iii. Phương Pháp



*Sơ đồ chung của hệ thống*

**a) Tiền xử lý:**

- Ảnh được chuyển sang ảnh xám (để trích xuất đặc trưng kết cấu) và khử nhiễu bằng bộ lọc trung vị.

**b) Phân đoạn:**

- Mask được tạo ra bằng cách:
  - Sử dụng Otsu Thresholding để chuyển các pixel thấp hơn ngưỡng T thành màu đen và ngược lại thành trắng.
  - Thực hiện phép đóng lên mask để khử các điểm nhiễu.
- Xử dụng phép AND lên mask và từng kênh màu của ảnh gốc để cho ra ảnh được phân đoạn.

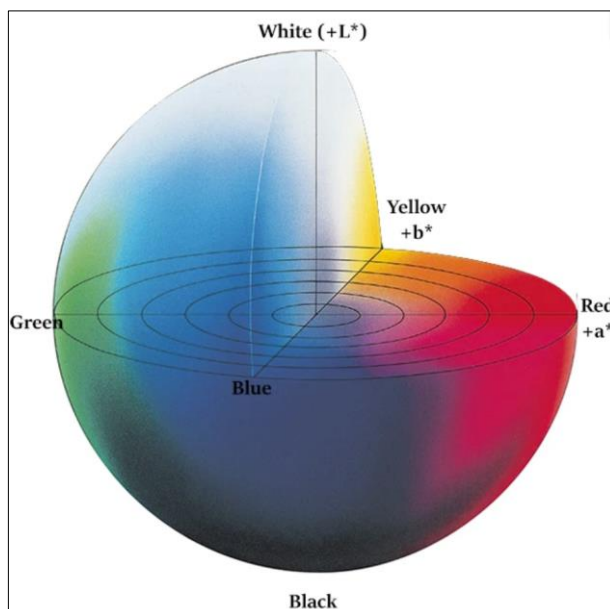
**iv. Phương pháp - Gray Level Co-Occurrence Matrix**

- Gray Level Co-Occurrence Matrix (GLCM) là phương pháp để trích xuất đặc trưng thống kê kết cấu.
- GLCM được xây dựng dựa vào giá trị độ xám của các pixel tại vùng đang xét và tần suất xuất hiện của các cặp pixel. Các cặp pixel được xét với khoảng cách d và hướng góc  $\theta$  (bài báo này sử dụng các góc 0, 45, 90, và 135 độ).
- Sau khi xây dựng GLCM, các giá trị đặc trưng thống kê được tính theo công thức:

$Contrast = \sum_{i=0}^{G-1} \sum_{j=0}^{G-1} (i - j)^2 P(i, j) \quad (2)$	$Dissimilarity = \sum_{i=0}^{G-1}  i - j  P(i, j) \quad (5)$
$Energy = \sum_{j=0}^{G-1} \sum_{i=0}^{G-1} (P(i, j))^2 \quad (3)$	$Homogeneity = \sum_{i=0}^{G-1} \sum_{j=0}^{G-1} \frac{P(i, j)}{1 +  i - j } \quad (6)$
$Entropy = - \sum_{j=0}^{G-1} \sum_{i=0}^{G-1} P(i, j) \log(P(i, j)) \quad (4)$	$Correlation = \frac{\sum_{i=0}^{G-1} \sum_{j=0}^{G-1} (i - \mu_y)(j - \mu_x) P(i, j)}{\sigma_x \sigma_y} \quad (7)$

**v. Phương pháp - Đặc Trưng màu CIE L\*a\*b**

- Các đặc trưng màu trong bài báo được trích xuất từ không gian màu CIE L\*a\*b.
- Các đặc trưng được sử dụng là: trung bình, độ lệch chuẩn, độ xiên, và độ nhọn.



**vi. Phương pháp - Độ Đo Khoảng Cách**

- Bài báo sử dụng nhiều độ đo để kiểm tra độ hiệu quả của hệ thống khi thực hiện truy vấn. Các độ đo sử dụng bao gồm các khoảng cách: Euclidean, Manhattan, Minkowski, và Canberra.

$$D_{EUCLIDEAN}(v_1, v_2) = \sqrt{\sum_{k=1}^N (v_{1k} - v_{2k})^2} \quad (8)$$

$$D_{MANHATTAN}(v_1, v_2) = \sum_{k=1}^N |v_{1k} - v_{2k}| \quad (9)$$

$$D_{MINKOWSKI}(v_1, v_2) = \sqrt[p]{\sum_{k=1}^N |v_{1k} - v_{2k}|^p} \quad (10)$$

$$D_{CANBERRA}(v_1, v_2) = \sum_{k=1}^N \frac{|v_{1k} - v_{2k}|}{|v_{1k} + v_{2k}|} \quad (11)$$

**vii. Kết quả**

<b>Proximity Measure</b>	<b>GLCM</b>	<b>Color Moments</b>	<b>GLCM+Color Moments</b>
Euclidean	0.808	0.848	0.896
Manhattan	0.792	0.825	0.858
Minkowski	0.819	0.870	<b>0.910</b>
Canberra	0.788	0.827	0.865

*Thống kê kết quả mean average precision với các đặc trưng và độ đo khoảng cách tương ứng*

<b>Top-K rank</b>	<b>Proximity Measure</b>			
	<i>Euclidean</i>	<i>Manhattan</i>	<i>Minkowski</i>	<i>Canberra</i>
10	0.954	0.930	<b>0.976</b>	0.927
20	0.896	0.858	<b>0.910</b>	0.865
30	0.843	0.799	<b>0.839</b>	0.779
40	0.788	0.768	<b>0.802</b>	0.754
50	0.762	0.743	<b>0.770</b>	0.722
60	0.740	0.713	<b>0.747</b>	0.705
70	0.716	0.689	<b>0.718</b>	0.685
80	0.694	0.663	<b>0.696</b>	0.665
90	<b>0.684</b>	0.643	0.679	0.651
100	<b>0.675</b>	0.635	0.667	0.638


*Thống kê mean average precision với các độ đo khoảng cách và thứ hạng kết quả trả về*

### III. Giải pháp của nhóm


- Sử dụng mạng CNN với bộ trọng số đã được huấn luyện sẵn (pretrained weight) là “imagenet” để rút trích đặc trưng và truy vấn.
- “imagenet” là bộ dữ liệu gồm hơn 14 triệu ảnh với hơn 20000 danh mục khác nhau.
- Nhóm đã thực nghiệm thử nghiệm trích xuất đặc trưng trên hai mô hình là VGG-16 và Resnet tạo ra 2 file đặc trưng .h5.
- Bằng cách loại bỏ các lớp cuối cùng (như Dense) và giữ lại các lớp convolutional và pooling.
- Dữ liệu sử dụng để trích xuất: [\[food-11 Image Classification Dataset\]](#)
- Tập dữ liệu gồm 11 class và mỗi class gồm 900 ảnh, với tổng cộng 9900 ảnh (training).
- Sau khi trích xuất đặc trưng thì ta có được file.h5 lưu trữ các vector đặc trưng.
- Sử dụng tích vô hướng để tính độ tương đồng:  $np.dot(queryVec, feats.T)$ . Với queryVec là một vector đặc trưng – mảng 1 chiều có kích thước (n,) và feats (n, m) với m là số ảnh và n là chiều dài của mỗi vector đặc trưng.
- Tích vô hướng càng cao thì độ tương đồng càng giống nhau. Sắp xếp từ thấp đến cao sử dụng  $rank\_ID = np.argsort(scores)[::-1]$
- Tạo và lưu figure.

IV. Các kết quả truy vấn  
Apple Pie


Resnet




Query Image




R1 french\_fries




R2 french\_fries




R3 french\_fries




R4 apple\_pie




R5 french\_fries




R6 french\_fries




R7 apple\_pie



R8 hot\_dog



R9 hamburger



R10 hot\_dog

VGG16



Query Image



R1 french\_fries



R2 french\_fries



R3 apple\_pie



R4 hot\_dog



R5 hot\_dog



R6 french\_fries



R7 apple\_pie



R8 french\_fries



R9 french\_fries



R10 french\_fries

## Burger

# Resnet



Query Image



R1 hamburger



R2 hamburger



R3 hamburger



R4 hamburger



R5 hamburger



R6 hamburger



R7 hamburger



R8 hamburger



R9 hamburger



R10 hamburger

# VGG16



Query Image



R1 hamburger



R2 hamburger



R3 hamburger



R4 hamburger



R5 hamburger



R6 hamburger



R7 hamburger



R8 hamburger



R9 hamburger



R10 hamburger



## Cheesecake

# Resnet



Query Image



R1 cheesecake



R2 cheesecake



R3 cheesecake



R4 cheesecake



R5 hamburger



R6 apple\_pie



R7 cheesecake



R8 hot\_dog



R9 omelette



R10 apple\_pie

# VGG16



Query Image



R1 omelette



R2 hamburger



R3 omelette



R4 fried\_rice



R5 chicken\_curry



R6 omelette



R7 fried\_rice



R8 fried\_rice



R9 pizza



R10 apple\_pie



## Hotdog

# Resnet



Query Image



R1 sushi



R2 sushi



R3 hot\_dog



R4 hot\_dog



R5 sushi



R6 cheesecake



R7 hamburger



R8 hot\_dog



R9 pizza



R10 sushi

# VGG16



Query Image



R1 hot\_dog



R2 hot\_dog



R3 hot\_dog



R4 hot\_dog



R5 hamburger



R6 hamburger



R7 hot\_dog



R8 hot\_dog



R9 hot\_dog

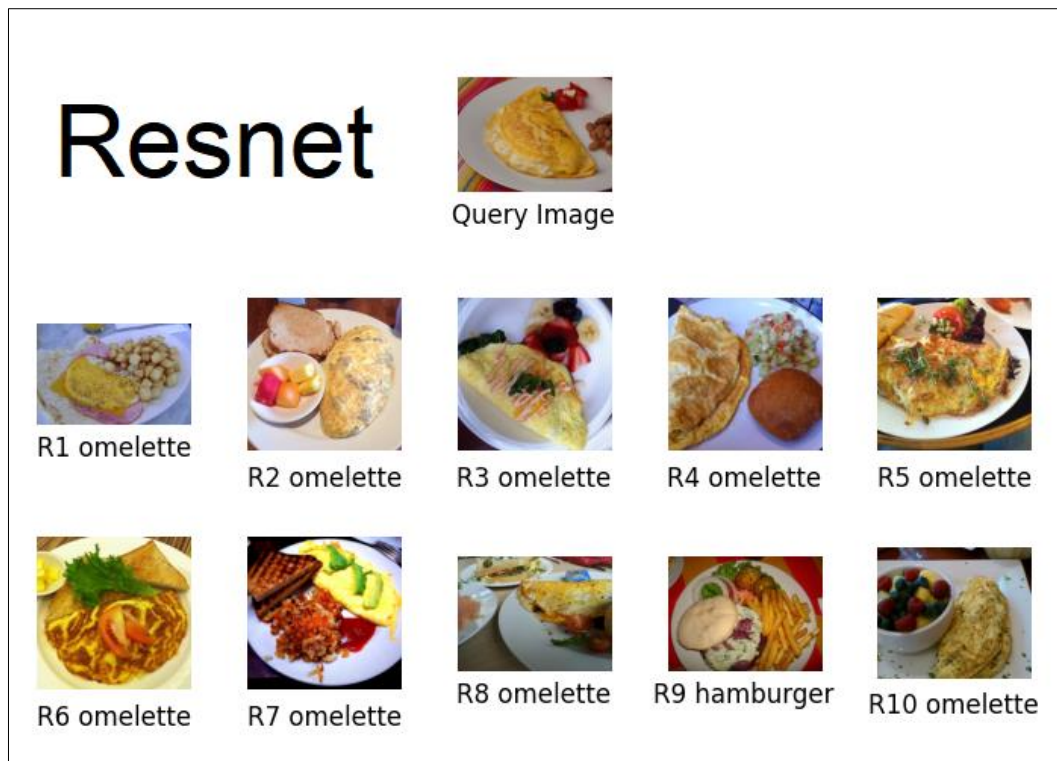


R10 hot\_dog

## Ice cream



## Omelette





## Pizza

# Resnet



Query Image



R1 pizza



R2 omelette



R3 apple\_pie



R4 pizza



R5 pizza



R6 pizza



R7 omelette



R8 cheesecake



R9 pizza



R10 chicken\_curry

# VGG16



Query Image



R1 pizza



R2 omelette



R3 pizza



R4 pizza



R5 pizza



R6 omelette



R7 pizza



R8 pizza



R9 omelette



R10 pizza

## Sushi

# Resnet



Query Image



R1 sushi



R2 sushi



R3 sushi



R4 omelette



R5 sushi



R6 cheesecake



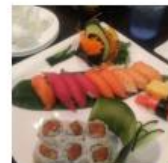
R7 sushi



R8 omelette



R9 ice\_cream



R10 sushi

# VGG16



Query Image



R1 sushi



R2 sushi



R3 ice\_cream



R4 sushi



R5 sushi



R6 fried\_rice



R7 ice\_cream



R8 sushi



R9 apple\_pie



R10 sushi

## **V. MAP của mô hình**

### **VGG-16**

- k=3: 0.8060
- k=5: 0.7006
- k=11: 0.4647
- k=21: 0.0700

### **ResNet**

- k=3: 0.8589
- k=5: 0.7810
- k=11: 0.5953
- k=21: 0.1267

⇒ Mô hình ResNet có kết quả tốt hơn có thể là nhờ độ phức tạp của ResNet để rút được nhiều vector đặc trưng tốt hơn, từ đó nâng cao hiệu quả của mô hình.

## **VI. Cải tiến của nhóm**

- Chưa có, nhưng nhóm đã nghĩ tới việc finetuning lại mô hình sử dụng bộ trọng số imagenet trước khi rút trích đặc trưng có thể sẽ tăng độ chính xác cho việc truy vấn.