



### ***Individual Lab: Requirement 3***

*Môn: Truy vấn thông tin thị giác*

Họ tên: Phan Đặng Anh Khôi

MSSV: 21127325

Lớp 21TGMT

**Giảng viên phụ trách**

Phạm Minh Hoàng

Nguyễn Trọng Việt

Võ Hoài Việt



## Contents

<b>I.</b>	<b>Mức độ hoàn thành .....</b>	<b>3</b>
<b>II.</b>	<b>Hệ thống truy vấn ảnh .....</b>	<b>3</b>
1.	Về truy vấn.....	3
2.	Hệ thống truy vấn ảnh.....	3
<b>III.</b>	<b>Về dữ liệu được dùng cho Lab này.....</b>	<b>4</b>
<b>IV.</b>	<b>Về tổ chức code.....</b>	<b>4</b>
1.	libs.h: Bao gồm các thư viện sử dụng.....	4
2.	database.h và database.cpp .....	4
3.	globalFeature.h và globalFeature.cpp .....	4
4.	localFeature.h và localFeature.cpp.....	4
5.	utils.h và utils.cpp .....	4
6.	utils2.h và utils2.cpp .....	4
7.	localFeatureKmeans.cpp.....	5
<b>V.</b>	<b>Yêu cầu 1 – Xây dựng hệ thống truy vấn sử dụng các đặc trưng.....</b>	<b>5</b>
<b>i.</b>	<b>Sử dụng đặc trưng Histogram cho hệ thống truy vấn ảnh.....</b>	<b>5</b>
1.	Histogram của ảnh .....	5
2.	Truy vấn .....	5
<b>ii.</b>	<b>Sử dụng đặc trưng SIFT cho hệ thống truy vấn ảnh .....</b>	<b>5</b>
1.	SIFT .....	5
2.	Truy vấn .....	6
<b>iii.</b>	<b>Sử dụng đặc trưng ORB cho hệ thống truy vấn ảnh .....</b>	<b>6</b>
1.	ORB .....	6
2.	Truy vấn .....	7
<b>VI.</b>	<b>Nhận xét cho Yêu cầu 1 .....</b>	<b>7</b>
<b>VII.</b>	<b>Về Yêu cầu 2 – Cải tiến chất lượng truy vấn.....</b>	<b>8</b>
<b>VIII.</b>	<b>Yêu cầu 3 – Cải tiến tốc độ truy vấn .....</b>	<b>8</b>
<b>i.</b>	<b>Kmeans là gì và thực hiện Kmeans lên descriptors của SIFT .....</b>	<b>8</b>
1.	Về Kmeans [4] .....	8
2.	Bag of words.....	8
<b>IX.</b>	<b>Nhận xét cho Yêu cầu 3 .....</b>	<b>9</b>
<b>X.</b>	<b>Kết quả thực nghiệm MAP .....</b>	<b>9</b>
<b>XI.</b>	<b>Reference .....</b>	<b>10</b>



## I. Mức độ hoàn thành

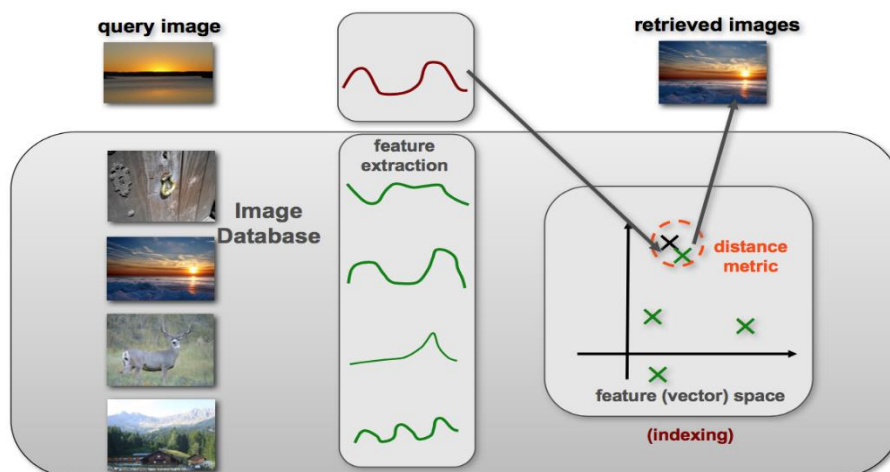
Yêu cầu 1 – Xây dựng hệ thống truy vấn sử dụng các đặc trưng:	Hoàn thành
1/ Histogram	100%
2/ Color Correlogram	0%
3/ SIFT	100%
4/ ORB	100%
Yêu cầu 2 – Cải tiến chất lượng truy vấn	100%
Yêu cầu 3 – Cải tiến tốc độ truy vấn	
1/ SIFT	100%
2/ ORB	100%

## II. Hệ thống truy vấn ảnh

### 1. Về truy vấn

- Truy vấn là việc tìm kiếm, sắp xếp dữ liệu – là một nhiệm vụ cơ bản trong lĩnh vực về dữ liệu.
- Hằng ngày chúng ta đều sử dụng Google để “hỏi”, các kết quả trả về cho câu hỏi của ta chính là kết quả truy vấn.
- Việc truy vấn đó sử dụng từ khóa (hay còn gọi là content-based retrieval). Đối với ảnh cũng thế, chúng ta đưa ảnh vào hệ thống và hệ thống trả về cho ta các ảnh giống nhất với nó.

### 2. Hệ thống truy vấn ảnh



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

- Để xây dựng một hệ thống truy vấn, ta cần đi qua 3 công đoạn chính: Chuẩn bị dữ liệu (1), Rút trích đặc trưng (2), Truy vấn (3).
  - (1) Chuẩn bị dữ liệu: “Có dữ liệu là có tất cả”
  - (2) Rút trích đặc trưng: Khi ta đã có CSDL ảnh, bước tiếp theo là rút trích đặc trưng của tập ảnh và lưu lại CSDL đặc trưng đó để phục vụ cho việc truy vấn.
  - (3) Truy vấn: thực hiện đưa ảnh truy vấn (query) vào hệ thống để rút trích đặc trưng cho ảnh truy vấn và thực hiện so sánh trên CSDL đặc trưng để từ đó sắp xếp các kết quả “tìm kiếm” theo độ tương đồng.

### III. Về dữ liệu được dùng cho Lab này

- Dữ liệu được sử dụng là folder [images] và file [dataset.csv] được tải từ [0]
- Thực hiện khám phá dữ liệu, phát hiện sai sót và sửa lại file csv cũng như folder ảnh.
- Các chỉnh sửa đã được lưu vào file jupyter notebook đính kèm [checkDataRetrieval.ipynb].
- File csv được sửa lại được đặt tên [dataset\_modified.csv] (trong data).

### IV. Về tổ chức code

1. libs.h: Bao gồm các thư viện sử dụng
2. database.h và database.cpp
  - Bao gồm các phương thức lưu CSDL đặc trưng từ folder ảnh và đọc đặc trưng từ file.
3. globalFeature.h và globalFeature.cpp
  - Bao gồm factory cho đặc trưng toàn cục (chỉ gồm Histogram).
  - Xử lý truy vấn.
4. localFeature.h và localFeature.cpp
  - Bao gồm factory cho đặc trưng cục bộ (gồm SIFT và ORB).
  - Xử lý truy vấn.
5. utils.h và utils.cpp
  - Các hàm tiện ích hỗ trợ tính toán và xử lý khác.
6. utils2.h và utils2.cpp

- Hàm xử lý truy vấn theo flow: gồm hàm truy vấn Global, Local, LocalKmeans.

#### 7. localFeatureKmeans.cpp

- Các hàm phục vụ truy vấn đặc trưng cục bộ (SIFT, ORB) sử dụng Bag of Words.
- Bao gồm phương thức tạo codebook (hay từ điển) – là các centers sau khi thực hiện kmeans; Map descriptor thành histogram (thể hiện các descriptors bằng tần số); Tạo histogram từ codebook; Truy vấn sử dụng Bag of Words.

## V. Yêu cầu 1 – Xây dựng hệ thống truy vấn sử dụng các đặc trưng

### i. Sử dụng đặc trưng Histogram cho hệ thống truy vấn ảnh

#### 1. Histogram của ảnh

- Là một đặc trưng toàn cục.
- Thể hiện tần số của số lần xuất hiện độ sáng của ảnh.

#### 2. Truy vấn

- Độ đo được sử dụng:  $\text{sum}(\min(\text{histogram1}, \text{histogram2}))$ , là độ đo tổng min của tất cả bin màu. Nếu độ đo ra 0 thì hai histogram hoàn toàn khác, và càng lớn thì càng giống nhau.
- Công đoạn: tạo CSDL đặc trưng cho histogram (1), đọc CSDL đặc trưng và tải lên hệ thống (2), truy vấn (3), hiển thị (4).

(1) Đặc trưng được rút là một Ma trận (cv::Mat) kết hợp từ việc tính toán và chuẩn hóa trên ba kênh màu Blue, Green và Red [1]. Tạo CSDL từ việc tính đặc trưng kết hợp mở file [dataset\_modified.csv] để lấy ra cột Picture Name, Label, tạo thêm cột URL và Ma trận [1 x 768] (1 cột, 768 hàng) sau đó thực hiện chuyển đổi nó thành dạng chuỗi để lưu vào.

(2) Sau khi có CSDL đặc trưng, thực hiện load và tạo thành `vector<myRecord>`. Với `myRecord` là một struct có 4 thuộc tính Picture Name, Label, URL và Ma trận descriptor.

(3) Sử dụng vector có được từ (2) để truy vấn với độ đo ở trên.

(4) Hiển thị ảnh.

### ii. Sử dụng đặc trưng SIFT cho hệ thống truy vấn ảnh

#### 1. SIFT

- Là một đặc trưng cục bộ.

- Các bước của SIFT [2]:
  - Build scale-space: xây dựng không gian tỉ lệ. Sử dụng Difference of Gaussian.
  - Keypoint Localization: định vị các điểm chính (là điểm cực trị trong không gian trên).
  - Orientation Assignment: gán hướng cho các điểm chính.
  - Keypoint Descriptor: mô tả điểm chính bằng vector đặc trưng. Các pixel xung quanh điểm chính được dùng để tạo Descriptor. Cụ thể chia lưới 16x16 quanh điểm chính và tiếp tục được chia thành một khối 4x4. Sau đó tính Histogram of Gradients cho mỗi khối => tạo ra 128 bin giá trị đại diện cho vector đặc trưng.
- SIFT bền vững với thay đổi trong tỷ lệ, xoay, ánh sáng của ảnh.

## 2. Truy vấn

- Độ đo: sử dụng FLANN và thực hiện match để tìm ra số điểm giống và tính khoảng cách (Euclidean). Khoảng cách Euclidean càng nhỏ thì ảnh càng giống và ngược lại.
- Công đoạn: giống với truy vấn ở Histogram, chỉ có thêm bước trung gian trước khi load tạo record.
  - (1) Đặc trưng được rút là descriptors (cv::Mat) sử dụng sift->detectAndCompute. Đồng thời lưu lại đường dẫn URL thành 1 file txt với mỗi descriptors của ảnh được rút ra → vector<pair<string, Mat>>.
  - (2) Sử dụng vector cặp ở trên kết hợp với file [dataset\_modified.csv] để lấy ra đủ 4 thuộc tính tạo thành record
  - (3) Thực hiện truy vấn ảnh bằng cách so khớp sử dụng độ đo trên và tính khoảng cách của các match.
  - (4) Hiển thị ảnh

## iii. Sử dụng đặc trưng ORB cho hệ thống truy vấn ảnh

### 1. ORB

- Là một đặc trưng cục bộ.
- Các bước của ORB [3]:
  - Phát hiện đặc trưng bằng FAST.
  - Lọc đặc trưng tốt sử dụng Harris Corner Measurement.
  - Tính toán gradient và hướng của đặc trưng.

- Sử dụng BRIEF làm descriptor.
- Xoay BRIEF theo hướng đặc trưng.
- ORB bền vững với thay đổi trong tỷ lệ, xoay, ánh sáng nhưng thấp hơn SIFT.

## 2. Truy vấn

- Độ đo: sử dụng Brute Force Matcher với khoảng cách Hamming (hợp với mô tả đặc trưng nhị phân). Cũng giống với Euclidean
- Công đoạn: giống với công đoạn của SIFT.

## VI. Nhận xét cho Yêu cầu 1

- Histogram:
  - Tạo ra file CSV từ CSDL ảnh và file CSV: 116s
  - Đọc file từ file CSV vừa tạo và lưu vào records: 2.7s
  - Truy vấn Histogram: 1 ảnh xấp xỉ 120ms
  - Thực hiện lưu AP vào file: 1 ảnh xấp xỉ 480ms

**=> Thời gian chạy rất nhanh. Nhưng độ chính xác kém nhất.**

- SIFT [144 x 256]:
  - Tạo CSDL đặc trưng: 186s.
  - Đọc CSDL đặc trưng: 220ms.
  - Truy vấn = tính MAP (cho 1 ảnh): ~20s

**=> Thời gian chạy rất chậm. Nhưng độ chính xác tốt nhất.**

- ORB [144 x 256]:
  - Tạo CSDL đặc trưng: 160s
  - Đọc CSDL đặc trưng: 88ms
  - Truy vấn = tính MAP (cho 1 ảnh): ~7s

**=> Thời gian chạy trung bình với độ chính xác ổn.**

- Có sự thay đổi ở truy vấn giữa SIFT và ORB là do SIFT sử dụng descriptors dạng Float còn ORB sử dụng unsigned char (uchar) – vector nhị phân.
- Nếu xét về độ hiệu quả thì ORB là một lựa chọn tốt và còn miễn phí trong một dự án thay vì tính phí như SIFT.



## VII. Về Yêu cầu 2 – Cải tiến chất lượng truy vấn

- Ở requirement 1, chúng ta đã thực hiện tìm ra đặc trưng có độ chính xác tốt nhất là SIFT và đối với requirement 2 là “tìm hiểu thêm đặc trưng khác” – là không cần thiết bởi descriptors mà SIFT lưu trữ khá nhiều – vector 128 chiều mô tả gradient của hình ảnh xung quanh mỗi điểm keypoints. Ngoài ra, SIFT bất biến với tỉ lệ, xoay và ánh sáng. Hơn thế nữa độ chính xác của SIFT cũng khá cao nên lần này chỉ thực hiện nâng cao tốc độ truy vấn sử dụng kmeans.

## VIII. Yêu cầu 3 – Cải tiến tốc độ truy vấn

### - Hướng tiếp cận: Bag Of Words

#### i. Kmeans là gì và thực hiện Kmeans lên descriptors của SIFT

##### 1. Về Kmeans [4]

- Về cơ bản K-means liên quan đến trung bình với số K. Thực chất, K-means là một thuật toán dùng để phân cụm thành K điểm dữ liệu gọi là trung tâm (centroids).
- Nhắc về phân cụm thì đây là một bài toán học không giám sát, nghĩa là đầu vào không nhãn (unlabeled dataset), mục tiêu là tìm ra các nhóm dựa vào sự tương đồng giữa các điểm dữ liệu.
- Về thuật toán K-means:
  - $X = \{x_1, x_2, \dots, x_n\}$ , K là số lượng nhóm cần phân cụm ( $K < n$ ).
  - Bước 1: Khởi tạo: chọn ngẫu nhiên K điểm từ tập dữ liệu X làm trung tâm  $\{c_1, c_2, \dots, c_k\}$ .
  - Bước 2: Với mỗi xi trong X, tính khoảng cách từ nó đến tất cả các trung tâm  $c_j$  và gán xi cho nhóm có trung tâm gần nhất.
  - Bước 3: Tính lại trung tâm của mỗi nhóm bằng cách lấy trung bình.
  - Bước 4: Lặp lại 2, 3 cho đến khi không có thay đổi đáng kể trong việc cập nhật/ phân cụm.

##### 2. Bag of words

- BoW (Bag-of-Words, hay còn gọi là Bag-of-Features – BoF) là một kỹ thuật chuyển đổi một văn bản sang thành một tập hợp từ (gọi là từ điển) không xét đến trật tự.
- Đối với ảnh, kỹ thuật này bắt nguồn từ nhận dạng texture trong ảnh (đếm số lần xuất hiện của texture đó).

- Quy trình BoW:
  - Trích xuất đặc trưng quan trọng trong ảnh: sử dụng SIFT, ORB
  - Sử dụng các đặc trưng đó để xây dựng vector “từ điển” (codebook/visual vocabulary) và mỗi phần tử của codebook là vector “từ vựng trực quan” (codevector/visual word): sử dụng Kmeans.
  - Thể hiện ảnh bằng tần số của “từ vựng trực quan”: Mỗi hình ảnh được biểu diễn dưới dạng histogram của các từ trực quan (hay còn gọi là vector hóa).

## IX. Nhận xét cho Yêu cầu 3

**Truy vấn Kmeans [Số cụm = 100; Số lần lặp để train Kmeans: 50 x 1; Size: 144 x 256]**

- SIFT:
  - Làm codebook: 617s
  - Đọc codebook tạo histograms: 13s
  - Truy vấn = tính MAP (cho 1 ảnh): ~180ms
- ORB:
  - Làm codebook: 250s
  - Đọc codebook tạo histograms: 1.7s
  - Truy vấn = tính MAP (cho 1 ảnh): ~170ms

**=> Thời gian truy vấn khi sử dụng mô hình BoW đã giảm đáng kể.**

## X. Kết quả thực nghiệm MAP

- Các kết quả thí nghiệm ở requirement đã được lưu vào folder Result.
- Thực hiện kết quả truy vấn cho ảnh SIFT tốn khá nhiều thời gian nhưng có thể thấy với 50 ảnh đầu tiên thì SIFT vẫn hơn ORB (với ~5% dữ liệu thì ta có thể nói rằng kết quả đặc trưng SIFT cao hơn ORB).
- Ngoài ra khi thực hiện kmeans lên SIFT và ORB, ta vẫn có thể thấy SIFT luôn có độ đo MAP cao hơn.

Đặc trưng/Kq trả về	N=3	N=5	N=11	N=21	Trung bình
Histogram (1118 ảnh)	13.87%	14.63%	14.97%	14.43%	14.47%
SIFT (50 ảnh đầu)	64.33%	64.44%	60.97%	53.88%	60.91%

SIFT BoW (1118 ảnh)	29.99%	30.47%	29.17%	27.10%	29.18%
ORB (50 ảnh đầu)	63.00%	62.80%	59.32%	53.28%	59.60%
ORB (1118 ảnh)	69.60%	68.91%	65.57%	61.81%	66.47%
ORB BoW (1118 ảnh)	23.99%	25.10%	24.74%	23.19%	24.25%

- Tốc độ đã có nhưng độ chính xác đã giảm đi khá nhiều, có thể là do kmeans chưa tìm ra được các center tốt hoặc chọn số lượng cụm chưa tối ưu.

## XI. Reference

- Các buổi học lý thuyết môn [Truy Vấn Thông Tin Thị Giác] từ đầu năm HKIII (khóa 2023-2024).
- Các bài Lab01, Lab02, Lab03, Requirement1 - Individual Lab, Requirement2 – Individual Lab [Truy Vấn Thông Tin Thị Giác]

[0] [TMBuD Dataset](#)

[1] [OpenCV: Histograms](#)

[2] [What is SIFT? \(educative.io\)](#)

[3] [Introduction to ORB \(Oriented FAST and Rotated BRIEF\) | by Deepanshu Tyagi | Medium](#)

[4] [OpenCV: Clustering](#)