ĐẠI HỌC QUỐC GIA TP HCM

TRƯỜNG ĐẠI HỌC CÔNG NGHỆ THÔNG TIN

ĐỒ ÁN MÔN HỌC NHẬN DẠNG THỊ GIÁC & ỨNG DỤNG

ĐỀ TÀI:

Cài đặt hệ thống tìm kiếm hình ảnh sử dụng BOW

GVHD: Lê Đình Duy – Nguyễn Tấn Trần Minh Khang

HVTH: Võ Tấn Mỹ - MSHV: CH1601014

Lớp: CH11-KHMT

TP HỒ CHÍ MINH – Năm 2017

Mục lục:

1. Mục tiêu

Mục tiêu của đồ án là xây dựng một ứng dụng nhỏ thực hiện chức năng tìm kiếm hình ảnh trong bộ dữ liệu Oxford Building (5K) bao gồm 5062 ảnh [1].

Input: là một ảnh từ bộ ảnh dùng để truy vấn.
Output: là một danh sách ảnh và một rank list, sắp xếp theo mức độ tương đồng.

2. Bộ dữ liệu Oxford Building (5K)

Bộ dữ liệu Oxford Building (5K) bao gồm 5062 ảnh các địa danh nổi tiếng của Oxford. Bộ sưu tập đã được chú thích theo cách thủ công để tạo ra sự thật toàn diện (*ground truth*) cho 11 địa điểm khác nhau, mỗi địa điểm được thể hiện bằng 5 truy vấn có thể xảy ra. Tổng cộng có 55 truy vấn trong thư mục *groundtruth*, ta có thể đánh giá một hệ thống truy vấn đối tượng dựa trên 55 truy vấn này.

Mỗi một ảnh trong groundtruth có thể có một trong các nhãn sau:

1. *Good* - Một hình ảnh đẹp, rõ ràng của đối tượng / tòa nhà.
2. *OK* - Có thể tìm thấy nhiều hơn 25% đặc điểm của đối tượng.
3. *Bad* - Đối tượng không hiện diện.
4. *Junk* - Có thể tìm thấy ít hơn 25% đặc điểm của đối tượng, hoặc có sự che khuất, không rõ ràng ở mức độ cao.

Mỗi một truy vấn trong groundtruth bao gồm tên của một hình ảnh và một vùng truy vấn mặc định mang tính đặc trưng của hình ảnh.

Ví dụ: file *all_souls_1_query.txt* có nội dung như sau:

oxc1_all_souls_000013 136.5 34.1 648.5 955.7

Trong đó:

all_souls_000013 là tên của hình ảnh chứa trong thư mục *oxford/images/*

136.5 34.1 648.5 955.7 là tập tọa độ của vùng (đối tượng) truy vấn mặc định tương ứng với x1,y1,x2,y2

Vì điều kiện chủ quan, báo cáo này chỉ sử dụng 845 ảnh trong số 5062 ảnh của bộ dữ liệu gốc.

55 ảnh truy vấn được đặt trong thư mục *oxford/query_images* để người dùng lựa chọn cho query.

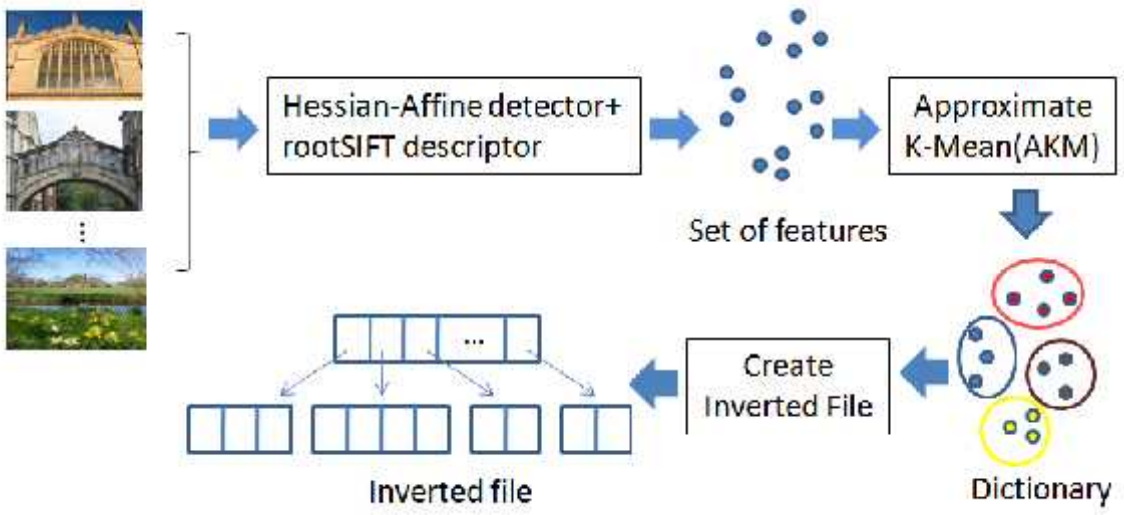
3. Các chức năng chính của chương trình

Chương trình được thực hiện qua 2 giai đoạn chính, có sử dụng code tham khảo của nvtiep [2] và chạy với giao diện Mathlab do học viên thực hiện.

Giai đoạn huấn luyện: bao gồm rút trích đặc trưng sử dụng thư viện VLFEAT [3], xây dựng Bag Of Visual Word sử dụng Approximate K-Means (AKM), tính Word-ID cho từng ảnh trong dataset và xây dựng inverted file [4].

Giai đoạn truy vấn: bao gồm phát hiện và rút trích đặc trưng từ bức ảnh truy vấn, tính Word-ID cho từng feature trong ảnh query và xây dựng ranked list. Chương trình cũng cho phép người dùng lựa chọn một vùng ảnh bất kỳ để thực hiện tìm kiếm thay vì sử dụng vùng ảnh truy vấn mặc định từ groundtruth.

4. Giai đoạn huấn luyện



Hình 1. Giai đoạn huấn luyện

Bước 1: Rút trích đặc trưng

- Sử dụng Hessian-Affine region detector để rút trích các keypoint
- Tính đặc trưng SIFT trên các keypoint
- Số chiều đặc trưng: 128
- Các đặc trưng SIFT được lưu vào file *feature.bin* trong thư mục *oxford/feat*
- Các đặc trưng cho từng ảnh dược lưu vào file *feat_info.mat* trong thư mục *oxford/feat*
- Link download file *feature.bin*: https://github.com/votanmy/DoAn/releases/download/v1.0/feature.bin

Bước 2: Xây dựng Bag Of Visual Words (dictionary)

- Sử dụng thuật toán gom cụm Approximate K-Mean (AKM)
- Số lượng cluster: 1.000.000
- Số lượng k-d tree: 8
- Số lần lặp: 5
- Bag Of Visual Words được lưu vào file *dict.mat* trong thư mục *oxford/feat*

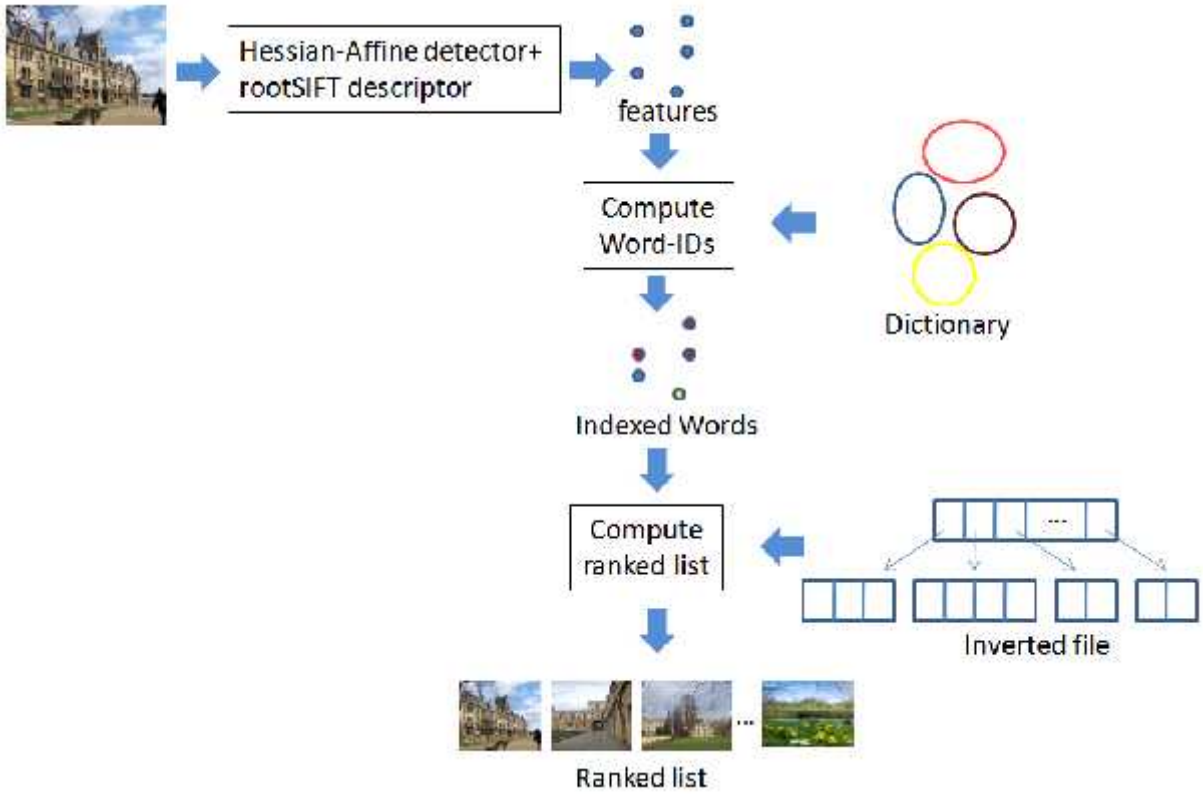
Bước 3: Tính Word-ID cho từng ảnh trong dataset

- Với mỗi word (SIFT feature) trong từng document (ảnh), ta tìm Word-ID của word dựa trên dictionary (tập Visual Words) đã xây dựng ở bước 2
- Ở bước này ta sẽ chuyển các khái niệm của xử lý ảnh sang bài toán xử lý văn bản hay ngôn ngữ tự nhiên:
    - visual words → dictionary
    - feature → word
    - index của feature → word ID
    - ảnh → documents
- Words được lưu vào file *words.mat* trong thư mục *oxford/feat*

Bước 4: Xây dựng inverted file

- Inverted file được xây dựng từ hàm *ccvInvFileInsert*
- Theo thứ tự bình thường, với mỗi document, ta sẽ biết được trong document này có các word nào
- Inverted file: với mỗi word, ta sẽ lưu danh sách những document có chứa nó
- ‘tf-idf’ weighting: các visual word xuất hiện ở nhiều class-of-document thì càng ít có vai trò để phân loại một document nên được đánh trọng số thấp hơn. Các visual word xuất hiện ở càng ít các class-of-document thì có trọng số cao hơn.

5. Giai đoạn truy vấn



Hình 2. Giai đoạn truy vấn ánh

Bước 5: phát hiện và rút trích đặc trưng SIFT sử dụng Hessian-Affine region detector (tương tự bước 1)

Bước 6: tính Word-ID cho từng feature trong ảnh query (tương tự bước 3)

Bước 7: Tính ranked list

- Xây dựng bảng Word Count để thống kê các word và tần số xuất hiện tương ứng
- Sử dụng inverted file để so sánh query document với tất cả các document trong trong inverted file → list score distance
- Sắp xếp list score theo tự giảm dần.
- File rank list các ảnh query với vùng lựa chọn mặc định của groundtruth được lưu trong thư mục *oxford/groundtruth*, tên file: *rank_list.txt*
- File rank list các ảnh query với vùng lựa chọn của người dùng được lưu trong thư mục *oxford/groundtruth*, tên file: *rank_list_cropped.txt*

Bước 8: Evaluation

- Tập ảnh truy vấn gồm 55 ảnh khoanh vùng các đối tượng truy vấn chính
- Các ảnh thuộc tập ground truth “good” và “ok” đứng ở những vị trí đầu tiên của rank_list thì độ chính xác càng cao và ngược lại.
- Độ chính xác của truy vấn với vùng lựa chọn mặc định của groundtruth được lưu trong thư mục *oxford/result*, tên file: *tên ảnh_result.txt*
- Độ chính xác của truy vấn với vùng lựa chọn của người dùng được lưu trong thư mục *oxford/cropped_result*, tên file: *tên ảnh_crop _result.txt*

6. Kết quả thực nghiệm với SIFT

Bảng 1. Độ chính xác khi truy vấn trên 55 ảnh ground truth

STT	Query image	Accuracy
1	all_souls_000013	0.280969
2	all_souls_000026	0.360845
3	all_souls_000051	0.813622
4	ashmolean_000000	0.867209
5	ashmolean_000007	0.491238
6	ashmolean_000058	0.456831
7	ashmolean_000269	0.458752
8	ashmolean_000305	0.775281
9	balliol_000051	0.706142
10	balliol_000167	0.121229
11	balliol_000187	0.623113
12	balliol_000194	0.579846
13	bodleian_000107	0.251146
14	bodleian_000108	0.601717
15	bodleian_000132	0.760799
16	bodleian_000163	0.670578
17	bodleian_000407	0.314086
18	christ_church_000179	0.770875
19	christ_church_000999	0.617393
20	christ_church_001020	0.728319
21	cornmarket_000019	0.86739
22	cornmarket_000047	0.756886
23	cornmarket_000105	0.38721
24	cornmarket_000131	0.79365
25	hertford_000015	0.578332
26	hertford_000027	0.633053
27	hertford_000063	0.847294
28	keble_000028	1
29	keble_000055	1
30	keble_000214	0.86518
31	keble_000227	0.838252
32	keble_000245	0.860285
33	magdalen_000058	0.212102
34	magdalen_000078	0.0869664
35	magdalen_000560	0.202183
36	oxford_000317	0.299114
37	oxford_000545	0.50712
38	oxford_001115	0.147363
39	oxford_001752	0.732616
40	oxford_001753	0.701039
41	oxford_002416	0.67934
42	oxford_002562	0.187912
43	oxford_002734	0.646414
44	oxford_002904	0.657217
45	oxford_002985	0.461621
46	oxford_003335	0.189586

47	oxford_003410	0.519339
48	pitt_rivers_000033	0.273323
49	pitt_rivers_000058	0.835518
50	pitt_rivers_000087	0.793332
51	pitt_rivers_000119	0.836414
52	pitt_rivers_000153	0.577519
53	radcliffe_camera_000095	0.669674
54	radcliffe_camera_000519	0.691571
55	radcliffe_camera_000523	0.736877

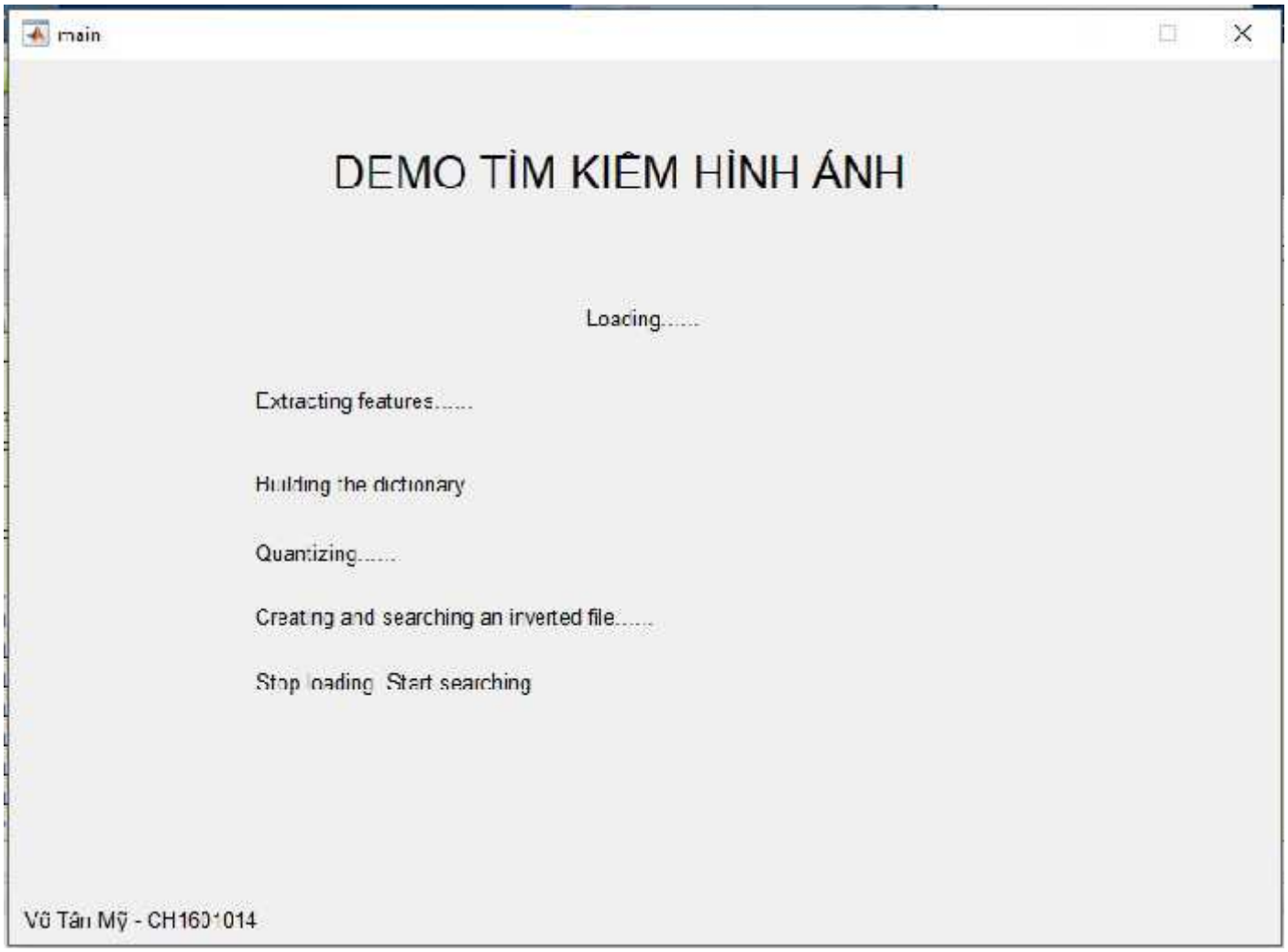
Accuracy trung bình: **0.587667**

7. Giao diện chương trình

Chương trình được viết và chạy trong môi trường có cấu hình như sau:

- Window 10
- Mathlab 2017a
- VLFEAT phiên bản 0.9.20

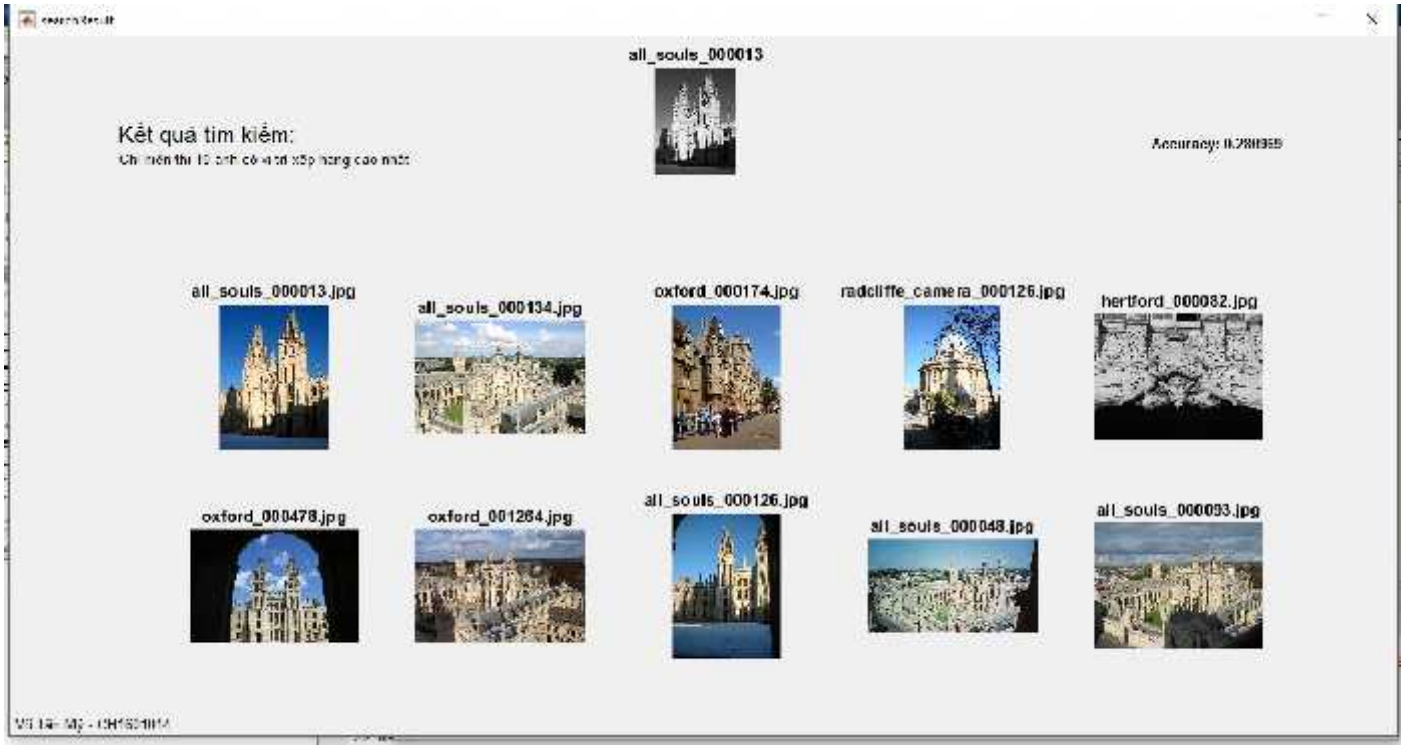
Màn hình khởi tạo khi chạy file **main.m**



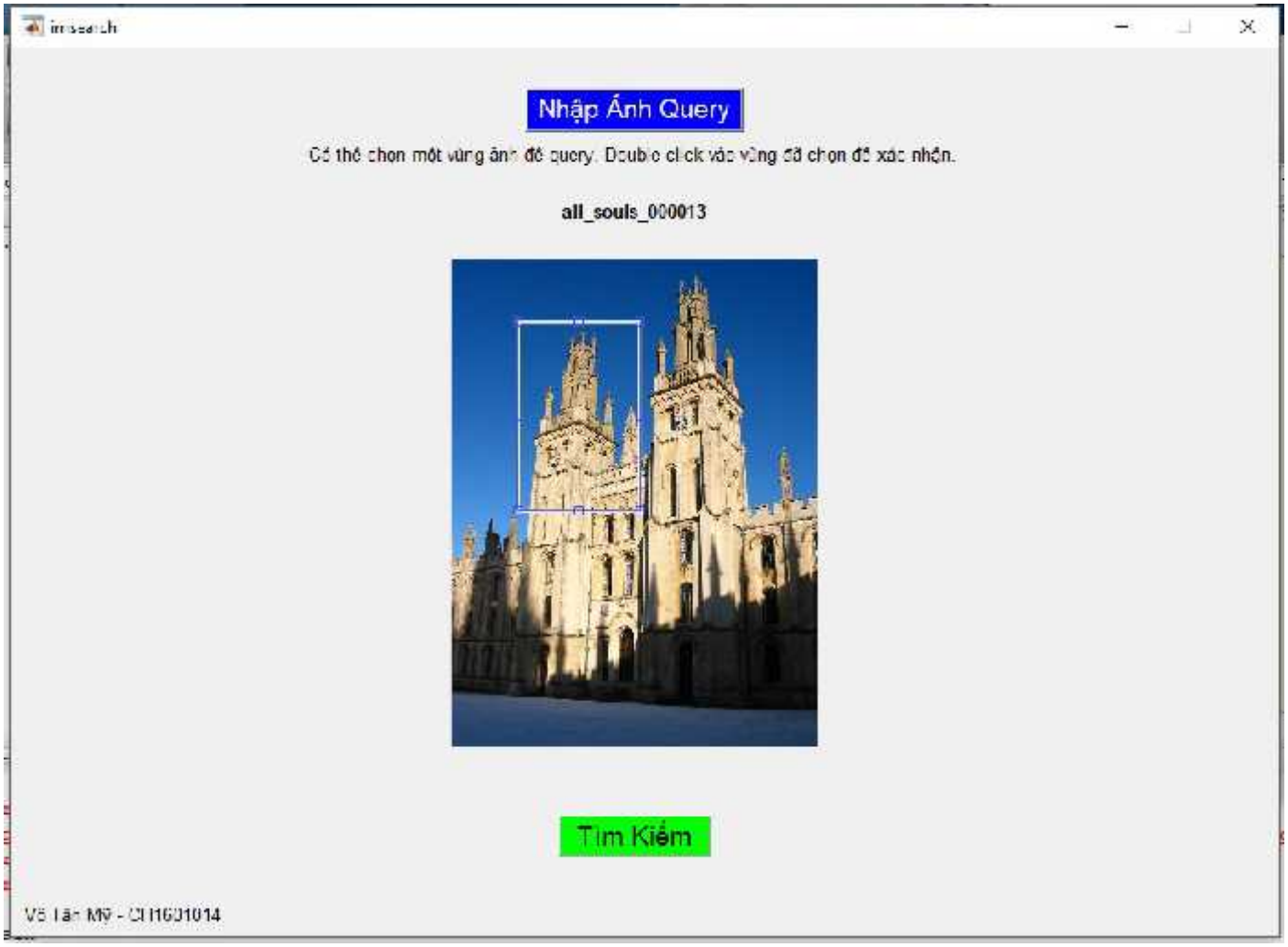
Sau khi load xong các tác vụ, chương trình sẽ lưu trữ các files quan trọng vào thư mục *oxford/feat*

Màn hình load ảnh: sau khi màn hình khởi tạo load các tác vụ xong, màn hình load ảnh sẽ xuất hiện

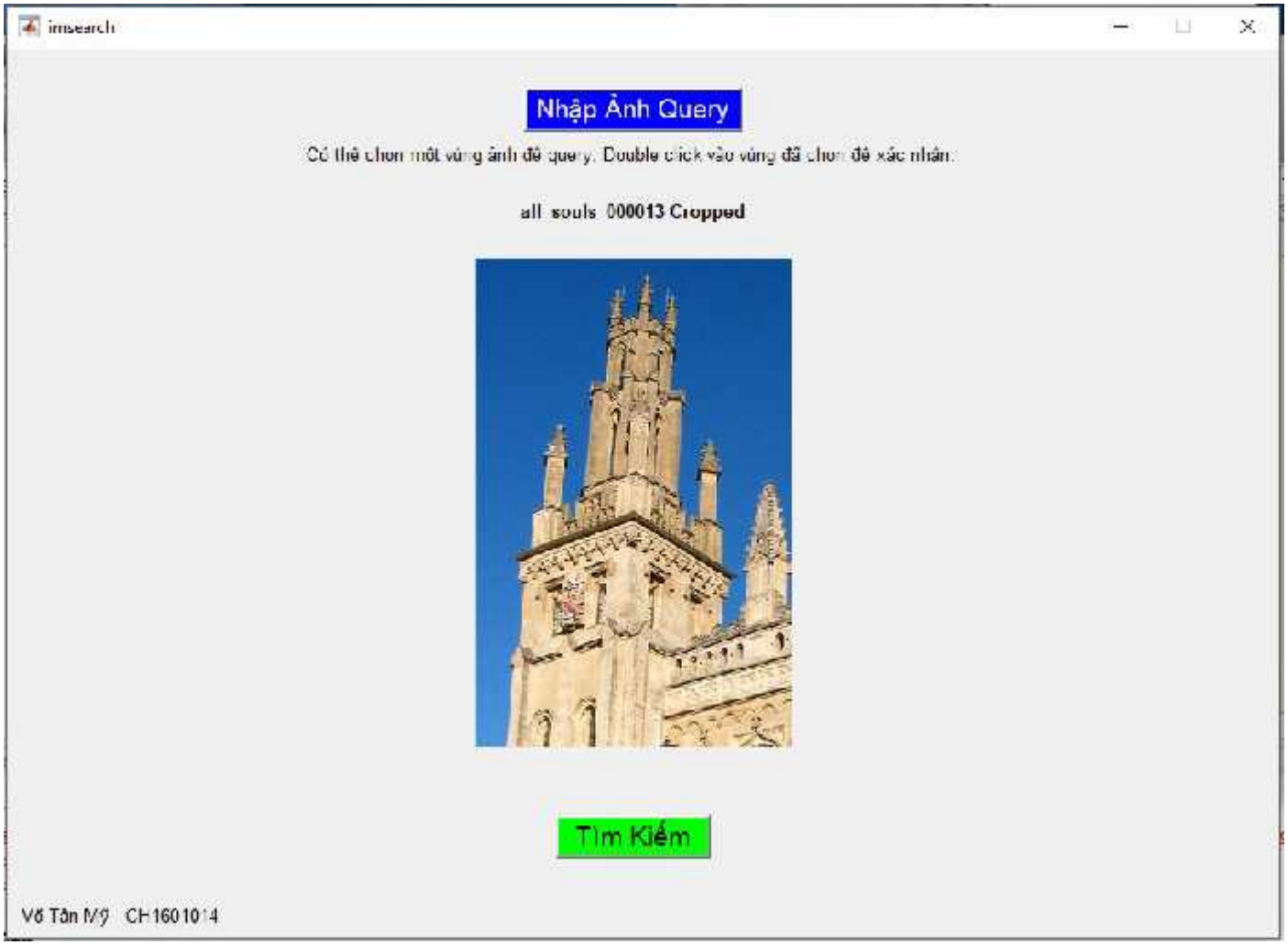
Khi nhấn nút Tìm Kiếm, hệ thống sẽ tiến hành tìm kiếm và xuất màn hình kết quả



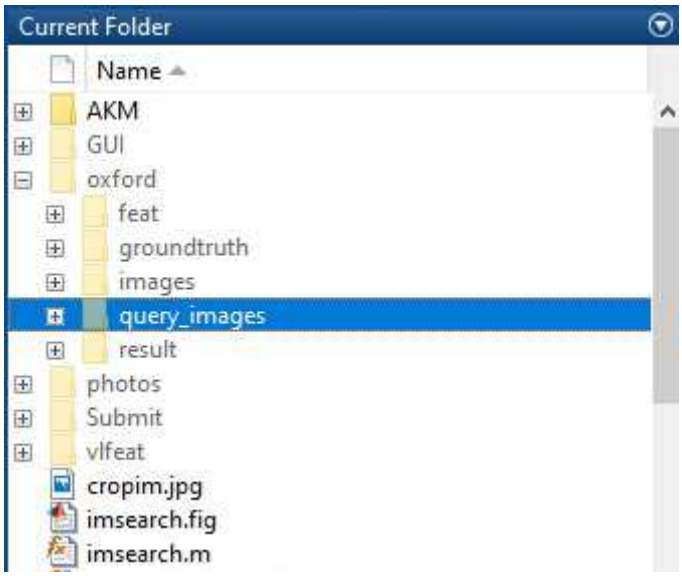
Hệ thống cũng cho phép người dùng lựa chọn một vùng trên ảnh để tìm kiếm



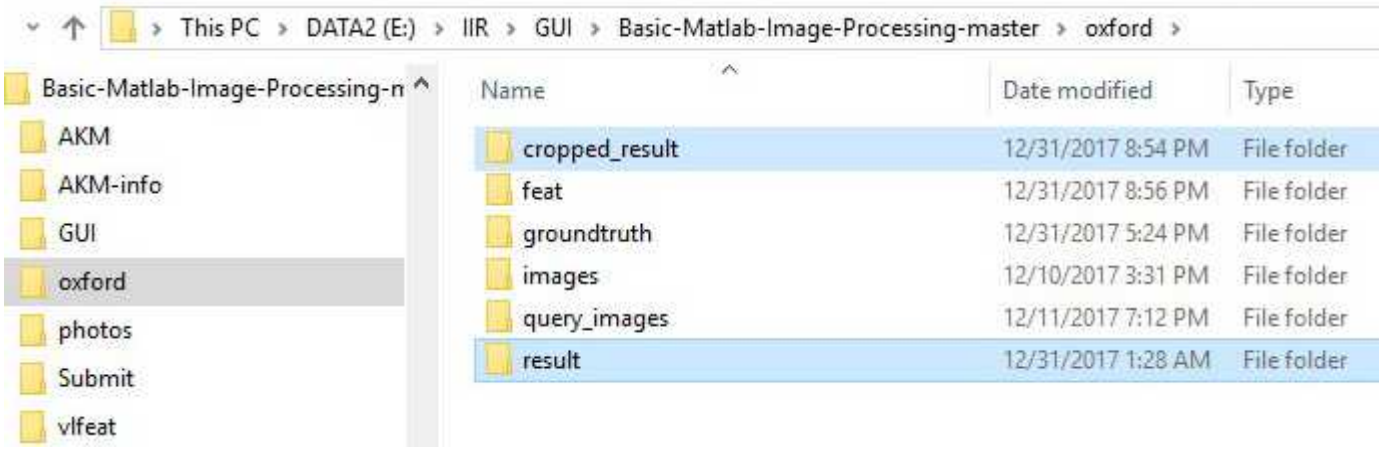
Sau khi double click vào vùng chọn, màn hình sẽ hiển thị hình ảnh đã lựa chọn



Các ảnh query được đặt trong một folder riêng để dễ dàng lựa chọn. Khi nhấn nút "Nhập Ảnh Query", hệ thống sẽ tự động trỏ tới thư mục này.



Các kết quả truy vấn được tự động lưu vào thư mục *oxford/result* và *oxford/cropped_result.* Kết quả truy vấn cho ảnh mặc định có dạng **_result.txt*. Kết quả truy vấn cho ảnh crop được lưu dưới dạng **_crop_result.txt*



8. Báo cáo mở rộng

Theo tài liệu [5], Relja Arandjelovi´c và cộng sự đã đề xuất ba phương pháp để cải thiện kết quả truy vấn.

8.1 RootSIFT

RootSIFT là phương pháp rút trích đặc trưng sử dụng căn bậc hai của nhân (*Hellinger kernel*) thay cho cách tính khoảng cách Euclidean để tính độ tương đồng giữa các SIFT descriptors.

Một vài tính chất của RootSIFT:

- Dễ dàng cài đặt, chỉ với một công thức đơn giản ta có thể chuyển SIFT sang RootSIFT

  rootsift = sqrt(sift / sum(sift))
- Không cần chỉnh sửa trên SIFT, không cần mã nguồn của SIFT, chỉ sử dụng cùng một chương trình.
- Không cần tính toán lại các SIFT descriptors đã xây dựng.
- Không yêu cầu thêm dung lượng lưu trữ.
- Có thể ứng dụng rộng rãi trong ngành thị giác máy tính.

Các lợi ích của RootSIFT:

- Cải thiện hiệu suất với các thử nghiệm khác nhau (không chỉ là vấn đề truy vấn).
- Bất cứ hệ thống nào có sử dụng SIFT đều có thể sử dụng RootSIFFT.

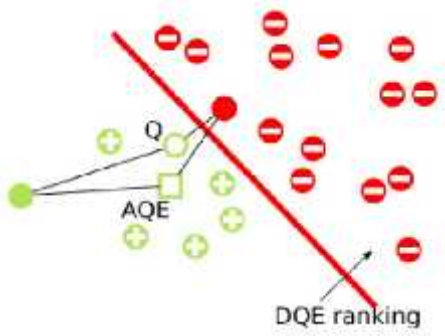
Dễ dàng cài đặt, không phát sinh chi phí tính toán và lưu trữ.

8.2 Mở rộng truy vấn có phán đoán (*Discriminative query expansion - DQE*):

Phương pháp này huấn luyện một bộ phân lớp SVM tuyến tính:

- Sử dụng truy vấn các vectors BOW mở rộng như là dữ liệu huấn luyện tích cực.
- Sử dụng các hình ảnh bị xếp hạng thấp như là dữ liệu huấn luyện thụ động.

- Xếp hạng hình ảnh theo khoảng cách của chúng đến đường ranh giới đã xác định.



Phương pháp này làm gia tăng hiệu suất đáng kể mà không phát sinh thêm chi phí

mAP on Oxford 105k:

Retrieval method	SIFT	RootSIFT
Philbin et.al. 2007: tf-idf with spatial reranking	0.581	0.642
Chum et.al. 2007: Average Query expansion (AQE)	0.726	0.756
Discriminative Query Expansion (DQE)	**0.752**	**0.781**

Các lợi ích của DQE:

- Hiệu quả cao hơn so với việc mở rộng truy vấn trung bình.
- Hiệu quả đạt được tối thiểu tương đương với mở rộng truy vấn trung bình.
- Không có tham số nào đối nghịch với nó ngoại trừ độ phức tạp trong cài đặt có tăng nhẹ.

8.3 Bổ sung đặc trưng về phía cơ sở dữ liệu (*Database-side feature augmentation - AUG*)

Các phương pháp này bao gồm:

- Bổ sung các visual words từ các hình ảnh lân cận (AUG).
- Chỉ bổ sung các visual words rõ ràng, xác định được (Spatial AUG).

Các kết quả đạt được:

Uses RootSIFT

Retrieval method	Oxford 5k	Oxford 105k
tf-idf ranking	0.683	0.581
tf-idf with spatial reranking	0.720	0.642
AUG: tf-idf ranking	0.785	0.720
AUG: tf-idf with spatial reranking	0.827	0.759

Note: idf weights are re-computed for the augmented dataset which improves performance, also our contribution

Uses RootSIFT

Retrieval method	Oxford 5k	Oxford 105k
tf-idf ranking	0.683	0.581
tf-idf with spatial reranking	0.720	0.642
AUG: tf-idf ranking	0.785	0.720
AUG: tf-idf with spatial reranking	0.827	0.759
Spatial AUG: tf-idf ranking	0.820	0.746
Spatial AUG: tf-idf with spatial reranking	**0.838**	**0.767**

Lợi ích của các phương pháp nói trên:

- Giúp gia tăng độ phủ (recall).
- Giúp gia tăng độ chính xác, tuy nhiên cũng kéo theo sự gia tăng dung lượng lưu trữ. Vì vậy, người dùng cần cân nhắc khi sử dụng các phương pháp này.

8.4 Tổ hợp các phương pháp cải tiến vào cùng một hệ thống

Việc tổ hợp này sẽ bao gồm cài đặt các phương pháp cải tiến đã đề xuất vào cùng một hệ thống truy vấn:

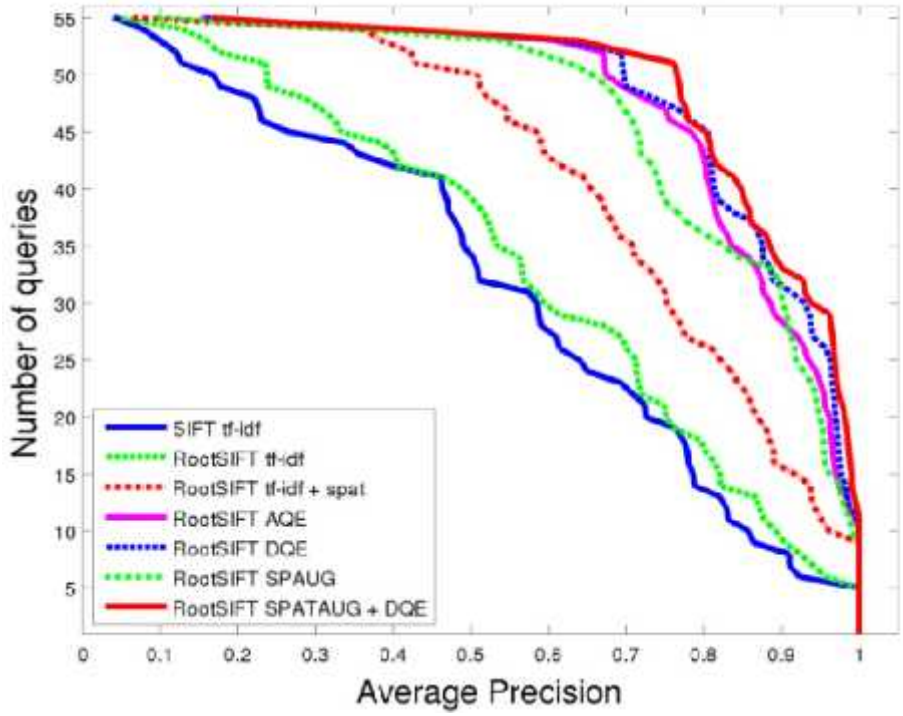
- RootSIFT
- Mở rộng truy vấn có phán đoán
- Bổ sung đặc trưng về phía cơ sở dữ liệu

Kết quả của sự tổ hợp này khá ấn tượng như sau:

Độ chính xác mAP thu được với các dataset khác nhau:

Oxford 5k	Oxford 105k	Paris 6k
0.929	**0.891**	**0.910**

Kết quả rất tiệm cận với độ phủ toàn phần (total recall) trên dataset Oxford 105k:



9. Kết luận

Báo cáo đã giới thiệu một ứng dụng giao diện Mathlab nhỏ được xây dựng để tìm kiếm hình ảnh theo phương pháp Bag Of Visual Word. Báo cáo này cũng giới thiệu ba phương pháp do Relja Arandjelovi´c và cộng sự đề xuất nhằm cải thiện hiệu suất truy vấn. Các kết quả thực nghiệm cho thấy sử dụng RootSIFT làm gia tăng đáng kể độ chính xác so với SIFT.

Riêng hai phương pháp cải tiến còn lại, vì thời gian và trình độ còn hạn chế nên học viên chưa thể thực hiện được. Hy vọng các báo cáo tiếp theo sẽ hiện thực được các kết quả nghiên cứu đáng trân trọng của Relja Arandjelovi´c và cộng sự.

10. Tài liệu tham khảo

1. Dataset Oxford Building: http://www.robots.ox.ac.uk/~vgg/data/oxbuildings/
2. Code tham khảo: https://github.com/nvtiep/Instance-Search/
3. Thư viện VLFEAT: http://www.vlfeat.org/
4. Mohamed Aly, Mario Munich, and Pietro Perona. *Indexing in Large Scale Image Collections: Scaling Properties and Benchmark*. IEEE Workshop on Applications of Computer Vision WACV, January 2011.
5. Relja Arandjelovi´c, Andrew Zisserman. *Three things everyone should know to improve object retrieval*. Department of Engineering Science, University of Oxford, University of Oxford. 2nd April 2012.

11. Các thông tin liên quan đến báo cáo

Github link: https://github.com/votanmy/DoAn

Link download file *feature.bin*:
https://github.com/votanmy/DoAn/releases/download/v1.0/feature.bin

File *feature.bin* phải được đặt trong thư mục *oxford/feat*