

I H C QU C GIA TP HCM
TR NG I H C CÔNG NGH THÔNG TIN



ÁN MÔN H C NH N D NG TH GIÁC & NG D NG

TÀI:

Cài t h th ng tìm ki m hình nh s d ng BOW

GVHD: Lê ình Duy – Nguy n T n Tr n Minh Khang

HVTH: Võ T n M - MSHV: CH1601014

L p: CH11-KHMT

TP H CHÍ MINH – N m 2017

M c l c:

1. M c tiêu
2. B d li u Oxford Building (5K)
3. Các ch c n ng chính c a ch ng trình
4. Giai o n hu n luy n
5. Giai o n truy v n
6. K t qu th c nghi m
7. Giao di n ch ng trình
8. Báo cáo m r ng
9. K t lu n
10. Tài li u tham kh o
11. Các thông tin liên quan n báo cáo

1. Mục tiêu

Mục tiêu của bài toán là xây dựng một ứng dụng nhúng cùng với giao diện Matlab để hiển thị các kết quả tìm kiếm hình ảnh trong bộ dữ liệu Oxford Building [1].

Input: là một hình ảnh để dùng 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 dữ liệu này đã được chú thích theo cách thủ công để tạo ra sự thật toàn diện (*ground truth*) cho 11 ảnh khác nhau, mỗi ảnh có thể hiển thị bằng 5 truy vấn có thể xảy ra. Tổng cộng có 55 truy vấn trong tập *groundtruth*, ta có thể đánh giá mức độ thành công của truy vấn dựa trên 55 truy vấn này.



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

1. *Good* - Một hình ảnh đẹp, rõ ràng của một tòa nhà.
2. *OK* - Có thể tìm thấy nhiều hơn 25% của một tòa nhà.
3. *Bad* - Một hình ảnh không hiển thị địa điểm.
4. *Junk* - Có thể tìm thấy ít hơn 25% của một tòa nhà, hoặc có sự che khuất, không rõ ràng về mặt cảnh cao.

Mỗi truy vấn trong *groundtruth* bao gồm tên của một hình ảnh và một vùng truy vấn mà nó 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 tập *oxford/images/*

136.5 34.1 648.5 955.7 là tọa độ của vùng (điểm) truy vấn trong ảnh gốc với x_1, y_1, x_2, y_2



Vì điều kiện quan trọng, báo cáo này chỉ sử dụng 845 ảnh trong số 5062 ảnh của bộ dữ liệu gốc. 55 ảnh truy vấn có mặt trong tập ảnh của *oxford/query_images* để dùng làm ảnh cho query.

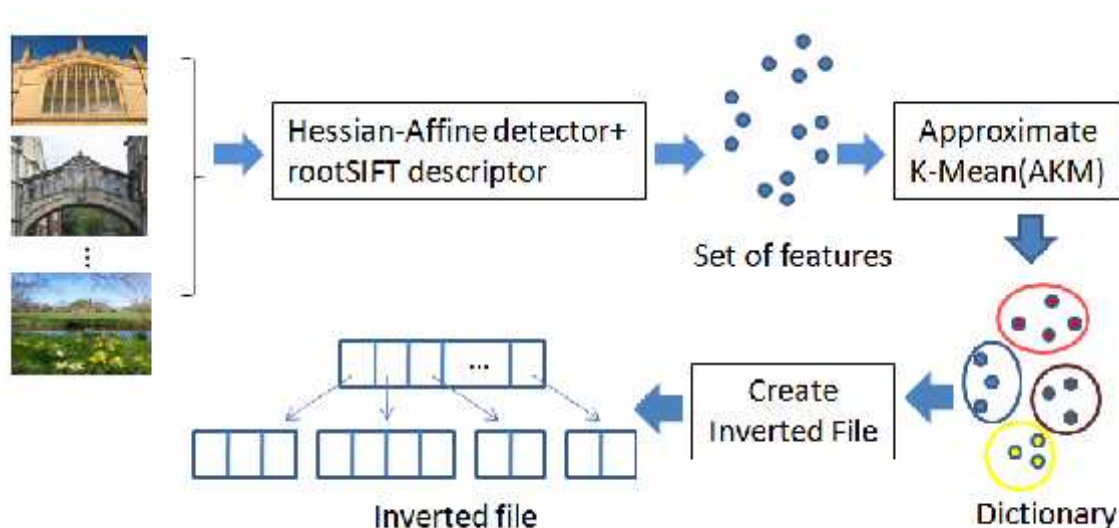
3. Các bước chính của chương trình

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

Giai đoạn huấn luyện: bao gồm rút trích đặc trưng sử dụng thuật toán VLFEAT [3], xây dựng Bag Of Visual Words 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 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 hiển thị tìm kiếm thay vì sử dụng vùng ảnh truy vấn mặc định như groundtruth.

4. Giai đoạn huấn luyện



Hình 1. Giai đoạn huấn luyện

B c 1: Rút trích c tr ng

- S d ng Hessian-Affine region detector rút trích các keypoint
- Tính c tr ng SIFT trên các keypoint
- S chi u c tr ng: 128
- Các c tr ng SIFT c l u vào file *feature.bin* trong th m c *oxford/feat*
- Các c tr ng cho t ng nh d c l u vào file *feat_info.mat* trong th m c *oxford/feat*
- Link download file *feature.bin*:
<https://github.com/votanmy/DoAn/releases/download/v1.0/feature.bin>

B c 2: Xây d ng Bag Of Visual Words (dictionary)

- S d ng thu t toán gom c m Approximate K-Mean (AKM)
- S l ng cluster: 1.000.000
- S l ng k-d tree: 8
- S l n l p: 5
- Bag Of Visual Words c l u vào file *dict.mat* trong th m c *oxford/feat*

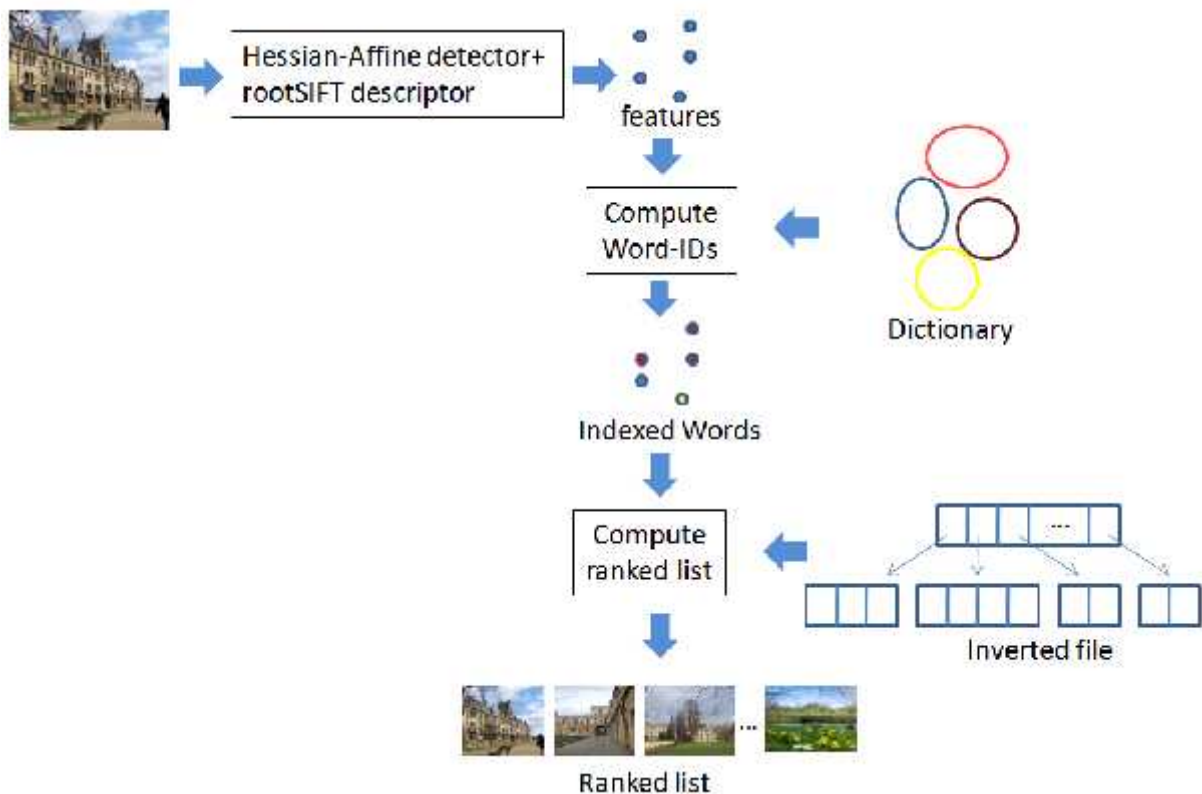
B c 3: Tính Word-ID cho t ng nh trong dataset

- V i m i word (SIFT feature) trong t ng document (nh), ta tìm Word-ID c a word đ a trên dictionary (t p Visual Words) ã xây d ng b c 2
- b c này ta s chuy n các khái ni m c a x lý nh sang bài toán x lý v n b n hay ngôn ng t nhiên:
 - o visual words → dictionary
 - o feature → word
 - o index c a feature → word ID
 - o nh → documents
- Words c l u vào file *words.mat* trong th m c *oxford/feat*

B c 4: Xây d ng inverted file

- Inverted file c xây d ng t hàm *ccvInvFileInsert*
- Theo th t bình th ng, v i m i document, ta s bi t c trong document này có các word nào
- Inverted file: v i m i word, ta s l u danh sách nh ng document có ch a nó
- ‘tf-idf’ weighting: các visual word xu t hi n nhi u class-of-document thì càng ít có vai trò phân lo i m t document nên c ánh tr ng s th p h n. Các visual word xu t hi n càng ít các class-of-document thì có tr ng s cao h n.

5. Giai đoạn truy vấn



Hình 2. Giai đoạn truy vấn ảnh

Bước 5: phát hiện và rút trích các tring 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 ghi các word và tần số xuất hiện của chúng
- Sử dụng inverted file để so sánh query document với tất cả các document trong inverted file → list score distance
- Sắp xếp list score theo thứ tự giảm dần.
- File rank list các ảnh query và vị trí của ảnh gốc trong tập dữ liệu trong thư mục *oxford/groundtruth*, tên file: *rank_list.txt*
- File rank list các ảnh query và vị trí của ảnh gốc trong tập dữ liệu trong thư mục *oxford/groundtruth*, tên file: *rank_list_cropped.txt*

Bước 8: Evaluation

- Tập ảnh truy vấn gồm 55 ảnh khoảng vùng các điểm truy vấn chính
- Các ảnh thu thập ground truth “good” và “ok” để đánh giá vị trí ưu tiên của rank_list thì chính xác càng cao và ngược lại.
- Đánh giá chính xác của truy vấn và vị trí của ảnh gốc trong tập dữ liệu trong thư mục *oxford/result*, tên file: *tên ảnh_result.txt*
- Đánh giá chính xác của truy vấn và vị trí của ảnh gốc trong tập dữ liệu trong thư mục *oxford/cropped_result*, tên file: *tên ảnh_crop_result.txt*

6. Kết quả thực nghiệm

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:

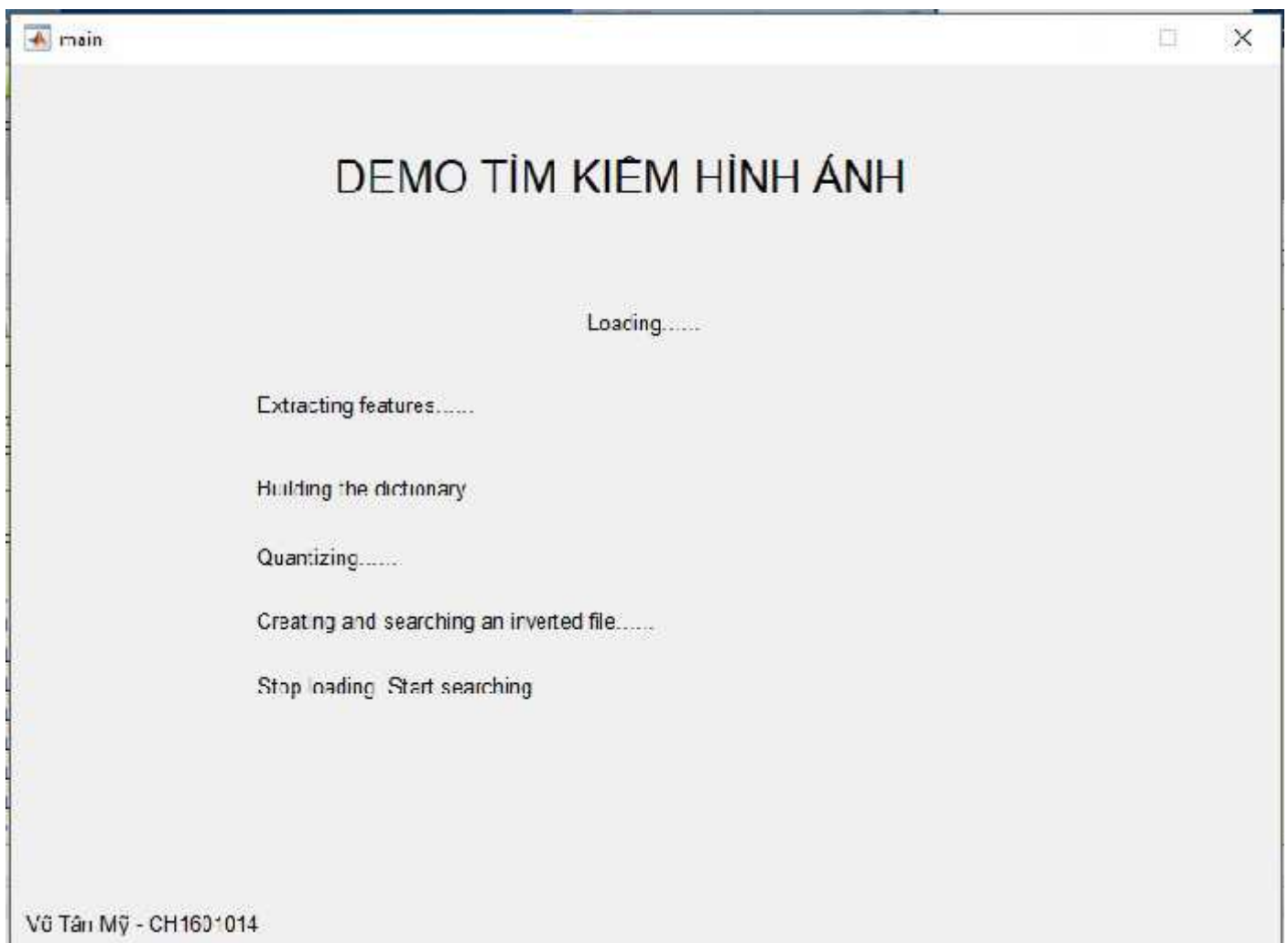
0.587667

7. Giao diện chương trình

Chương trình có vị trí và chạy trong môi trường có cấu hình như sau:

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

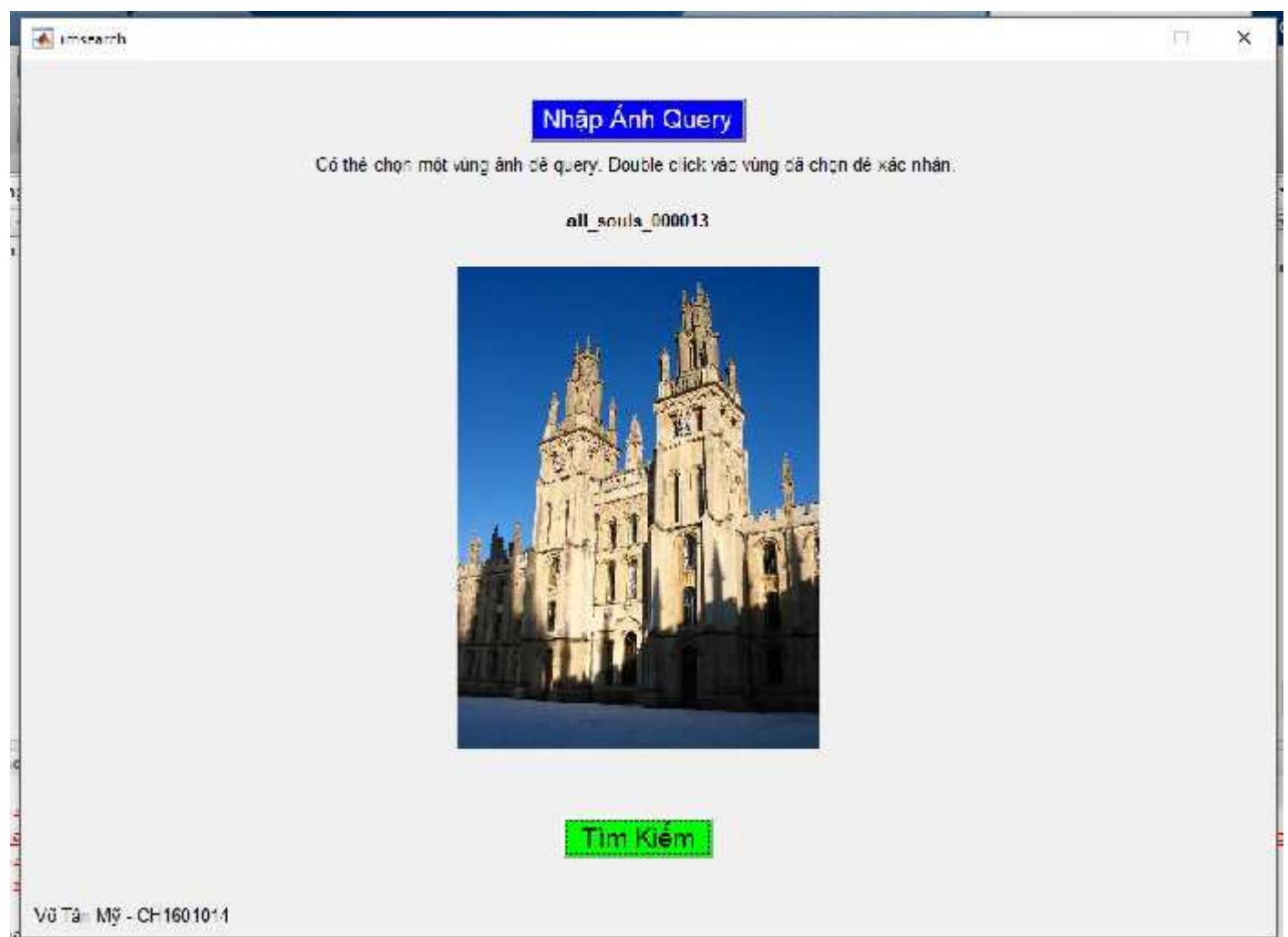
Màn hình khi tải o khi chạy file **main.m**



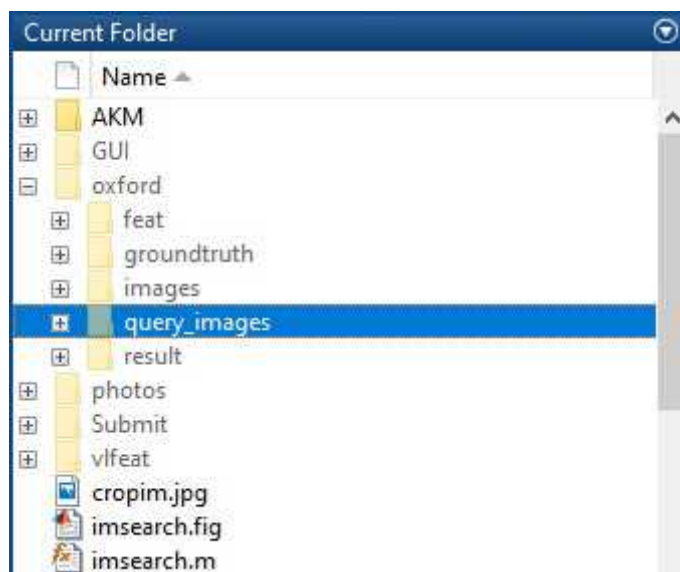
Sau khi load xong các tác v , ch ãng trình s ã l u tr ã các files quan tr ãng vào th ã m c *oxford/feat*



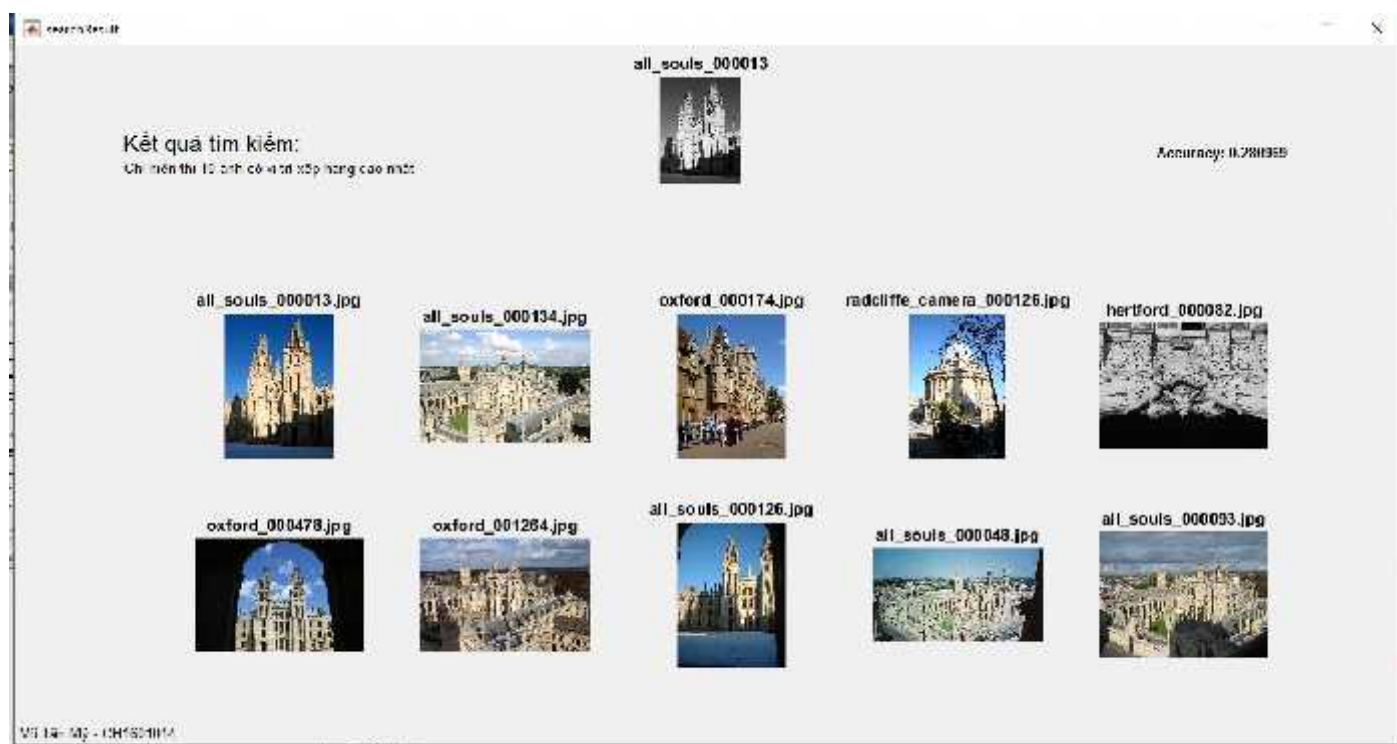
Màn hình load ãnh: sau khi màn hình kh ãi t ão load các tác v ã xong, màn hình load ãnh s ã xu t hi ãn



Khi nh ấn nút “Nh p nh Query”, h th ng s t ng tr t i th m c *query_images*. Các nh query c t trong folder riêng bi t ng i dùng d dàng l a ch n.



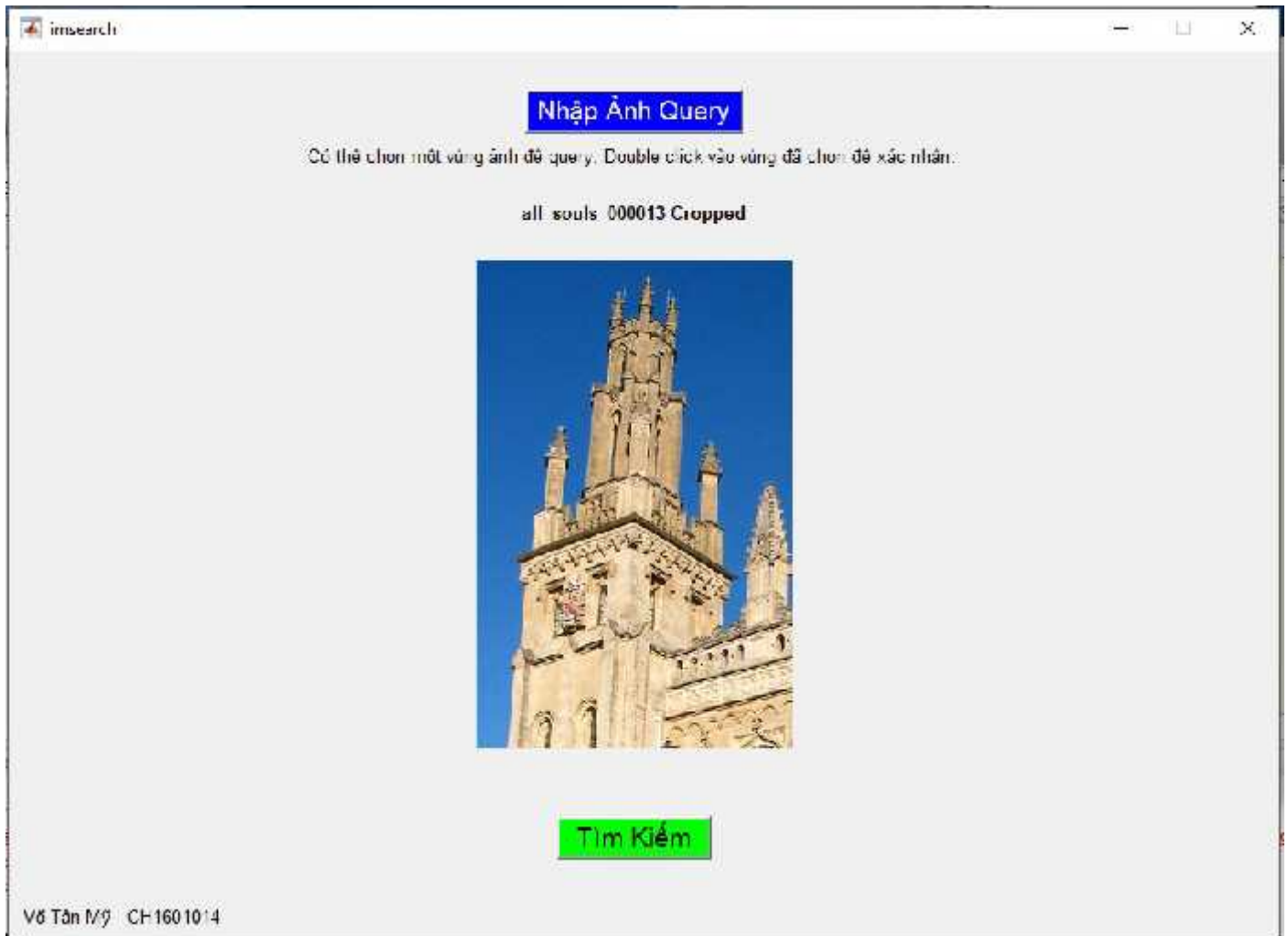
Khi nh ấn nút “Tìm Kì m”, h th ng s t i n hành tìm kì m và xu t màn hình k t qu



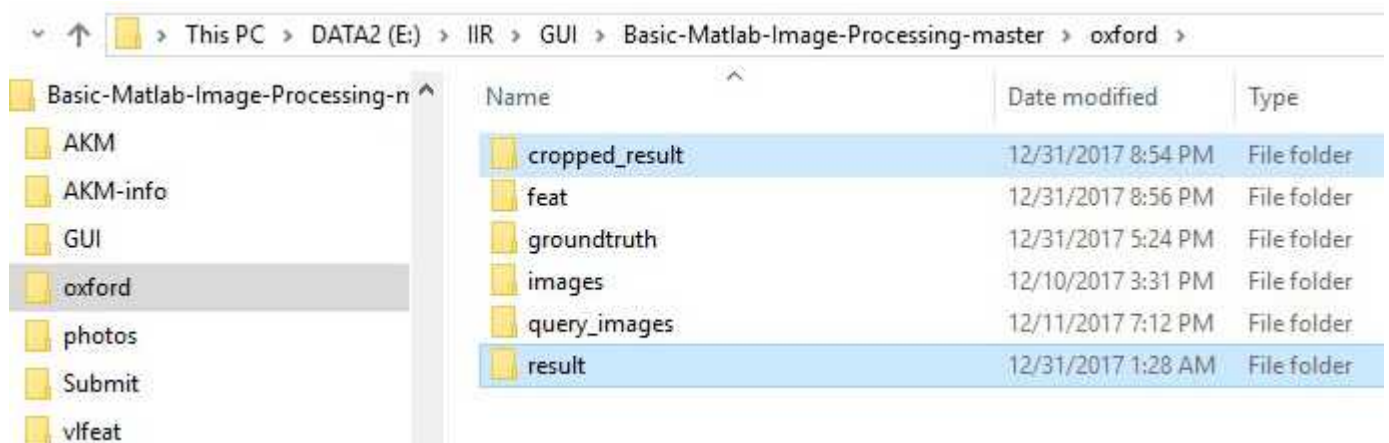
Hệ thống cũng cho phép người dùng lựa chọn một vùng trên ảnh để tìm kiếm.



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



Các kết quả truy vấn sẽ được lưu vào thư mục *oxford/result* và *oxford/cropped_result*. Kết quả truy vấn cho ảnh mà cần nhúng có đuôi **_result.txt*. Kết quả truy vấn cho ảnh crop sẽ lưu đuôi **_crop_result.txt*



8. Báo cáo mở rộng

Theo tài liệu [5], Relja Arandjelović và cộng sự đã xuất bản phương pháp cải thiện kỹ thuật truy vấn như sau:

8.1 RootSIFT

RootSIFT là phương pháp rút trích đặc trưng sử dụng các nhân hai chiều (Hellinger kernel) thay cho cách tính khoảng cách Euclidean tính trên tổng bình phương 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 chuẩn hóa trên SIFT, không cần mã ngược 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.

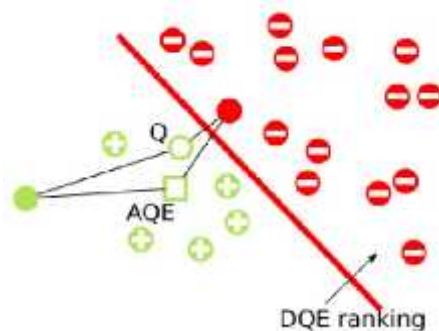
Các lợi ích của RootSIFT:

- Cải thiện hiệu suất với các thử nghiệm khác nhau (không chỉ là về truy vấn).
- Bất kể hệ thống nào có sử dụng SIFT đều có thể sử dụng RootSIFT.
- Dễ dàng cài đặt, không phát sinh chi phí tính toán và lưu trữ.

8.2 Mở rộng truy vấn có phân hoạch (Discriminative query expansion - DQE):

Phương pháp này huấn luyện một bộ phân loại SVM tùy chỉnh:

- Sử dụng truy vấn các vectors BOW mở rộng như là dữ liệu huấn luyện tích cực.
- Sử dụng các hình ảnh phản diện như là dữ liệu huấn luyện tiêu cực.
- Xây dựng hình ảnh theo khoảng cách của chúng đến 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 các mô hình truy vấn trung bình.
- Hiệu quả tốt hơn các mô hình truy vấn trung bình.
- Không có tham số riêng cho việc nó hoạt động như thế nào trong cài đặt cụ thể.

8.3 Bổ sung các truy vấn 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 (Spatial AUG).

Các kết quả:

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 Tích hợp các phương pháp cải tiến vào cùng một hệ thống

Vì thế hệ thống này sẽ bao gồm cài đặt các phương pháp cải tiến đã đưa vào cùng một hệ thống truy vấn:

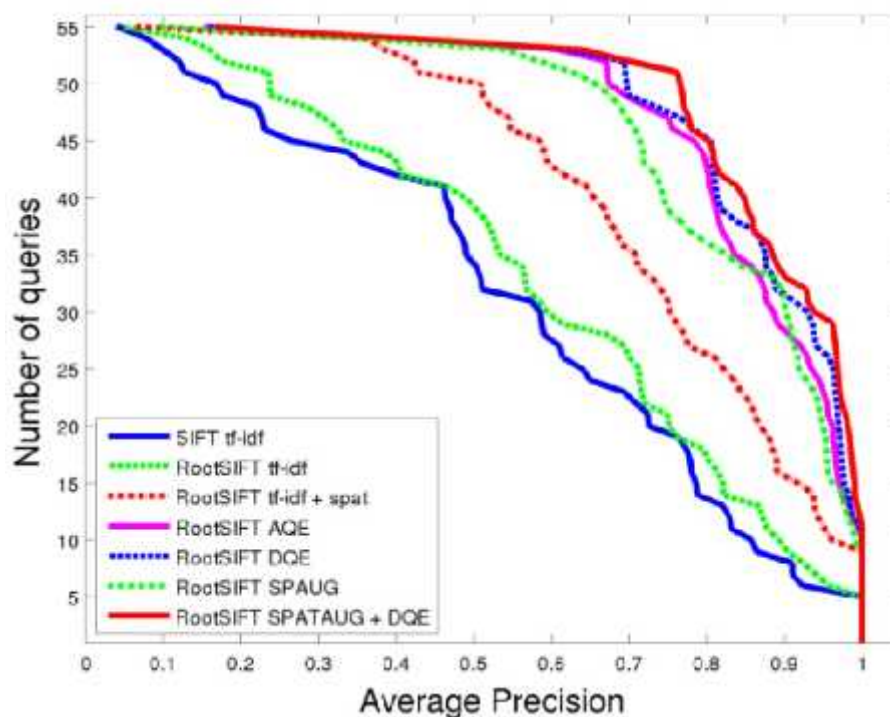
- RootSIFT
- Mô hình truy vấn có phân đoạn
- Bổ sung các truy vấn phía cơ sở dữ liệu

Kết quả của hệ thống này rất đáng kinh ngạc sau:

độ chính xác mAP thu được từ các dataset khác nhau:

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

Kết quả rút tỉm của việc phân tích toàn phần (total recall) trên dataset Oxford 105k:



9. Kết luận

Báo cáo đã giới thiệu một ứng dụng giao diện Matlab nhằm xây dựng tìm kiếm hình ảnh theo phương pháp Bag Of Visual Word. Báo cáo này cũng giới thiệu ba phương pháp do Relja Arandjelović và cộng sự đề xuất nhằm cải thiện hiệu suất truy vấn.

Vì thời gian và trình độ còn hạn chế nên các viên chức đã thực hiện các phương pháp cải thiện giới thiệu. Hy vọng các báo cáo tiếp theo sẽ hiển thị các kết quả nghiên cứu đáng trân trọng của Relja Arandjelović 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. Thử nghiệm 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 Arandjelović, Andrew Zisserman. *Three things everyone should know to improve object retrieval*. Department of Engineering Science, University of Oxford, University of Oxford. 2nd April 2012.
6. Báo cáo tham khảo: <https://github.com/biendltb/Instance-Search/blob/master/Report.docx>

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 có trong thư mục *oxford/feat*

Demmo video: <https://youtu.be/oqJil4Jh4ks>