

MỤC LỤC

1. Mục tiêu đề tài:	3
2. Nội dung thực hiện:	3
2.1. Tìm kiếm hình ảnh với Bag of Visual Words	3
2.2. Quy trình của hệ thống tìm kiếm	4
2.3. Đánh giá việc tìm kiếm hình ảnh	5
2.4. Một vài tùy chọn tìm kiếm được lựa chọn trong đồ án:	6
2.4.1. Các bước tìm kiếm hình ảnh dựa sử dụng bộ các đặc trưng được tùy biến (sử dụng tùy biến màu sắc):	6
2.4.2. Tìm kiếm trong hình ảnh sử dụng hình ảnh truy vấn	10
2.4.3. Tìm kiếm bộ hình ảnh cho đối tượng cụ thể sử dụng ROIs	10
3. Tổng kết:	13
3.1. Kết quả thực hiện:	13
3.2. Hạn chế:	15
3.3. Hướng phát triển	15
4. Tham khảo	16

DANH MỤC HÌNH ẢNH

Hình 2.1. Quy trình của hệ thống tìm kiếm	4
Hình 2.2. Tập dữ liệu hình ảnh các loài hoa.....	7
Hình 2.3. Tập các hình ảnh tương tự	9
Hình 2.4. Ảnh chứa đối tượng cần tìm kiếm	11
Hình 2.5. Kết quả tìm kiếm	12
Hình 3.1. Kết quả tìm kiếm theo vùng màu	13
Hình 3.2. Kết quả tìm kiếm theo vùng màu	14
Hình 3.3. Kết quả tìm kiếm hình ảnh chứa vùng ảnh đã chọn	14
Hình 3.4. Kết quả tìm kiếm theo đối tượng đặc biệt sử dụng ROIs	15

1. Mục tiêu đề tài:

Tên đề tài:

XÂY DỰNG ỨNG DỤNG TÌM KIẾM HÌNH ẢNH ĐƠN GIẢN SỬ DỤNG BAG OF WORDS

Mục tiêu vận dụng kiến thức đã học để Xây dựng ứng dụng tìm kiếm hình ảnh:

- Đơn giản, có giao diện
- Ứng dụng được Bag of Words vào việc hỗ trợ tìm kiếm hình ảnh.
- Cho phép tìm kiếm hình ảnh với một vài tùy chọn khác nhau.

Ngoài ra, việc xây dựng ứng dụng này với mong muốn sẽ tạo điều kiện cho em làm quen với việc lập trình xử lý ảnh, có kiến thức phục vụ việc học các môn về xử lý ảnh tiếp theo cũng như khóa luận tốt nghiệp sắp tới.

2. Nội dung thực hiện:

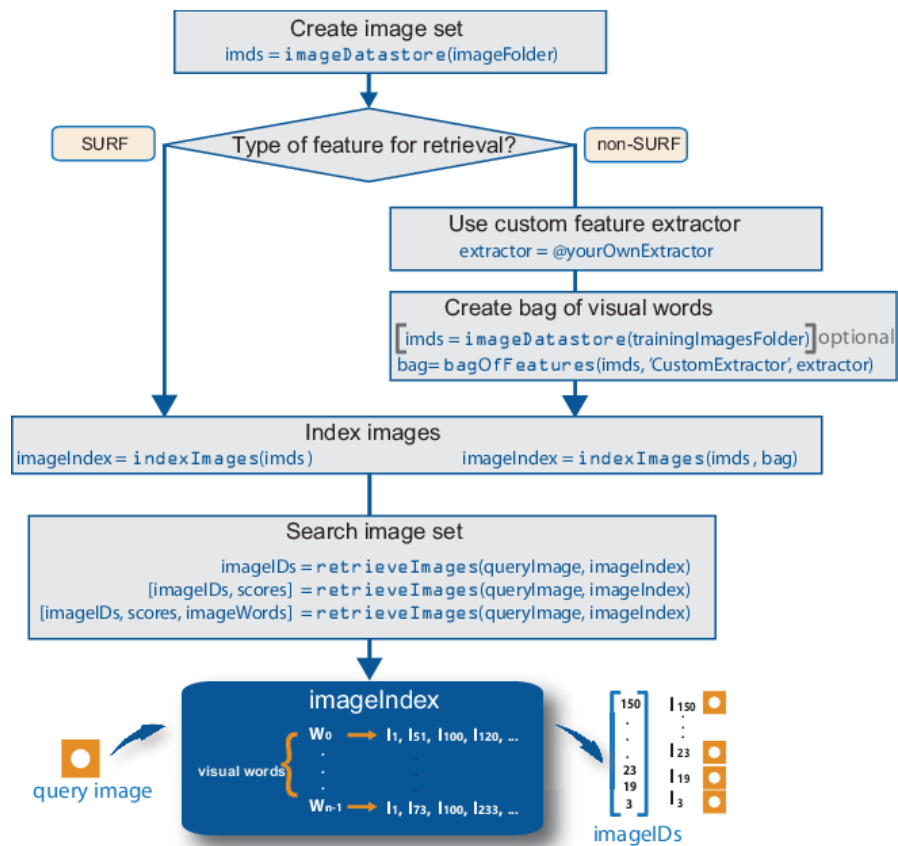
Để xây dựng ứng dụng thì ngoài việc tìm hiểu các thức xây dựng ứng dụng có giao diện trên Matlab thì em cần phải tìm hiểu thêm các kiến thức về Bag of Words và các hỗ trợ liên quan.

2.1. Tìm kiếm hình ảnh với Bag of Visual Words

Các hàm của Computer Vision System Toolbox hỗ trợ tìm kiếm bằng hình ảnh, còn được gọi là hệ thống tìm kiếm hình ảnh dựa trên nội dung (content-based image retrieval - CBIR). Các hệ thống CBIR được sử dụng để tìm các hình ảnh tương đồng với hình ảnh truy vấn từ một tập hợp các hình ảnh. Việc áp dụng các loại hệ thống này có thể được tìm thấy trong nhiều lĩnh vực như tìm kiếm sản phẩm dựa trên web, giám sát và nhận dạng địa điểm trực quan. Đầu tiên, hệ thống sẽ tìm kiếm một bộ sưu tập hình ảnh để tìm ra những hình ảnh tương tự như hình ảnh truy vấn.

Hệ thống truy xuất sử dụng một túi các từ ngữ trực quan (bag of visual words), một bộ sưu tập các mô tả hình ảnh, để đại diện cho tập dữ liệu hình ảnh. Hình ảnh được lập chỉ mục để tạo ra một ánh xạ các từ trực quan (visual words). Chỉ mục lập bản đồ mỗi từ trực quan (visual words) đến các lần xuất hiện của chúng trong tập hình ảnh. So sánh giữa hình ảnh truy vấn và chỉ mục cung cấp hình ảnh tương tự với hình ảnh truy vấn. Bằng cách sử dụng quy trình làm việc của hệ thống CBIR, ta có thể đánh giá độ chính xác của một tập kết quả tìm kiếm hình ảnh đã biết.

2.2. Quy trình của hệ thống tìm kiếm



Hình 2.1. Quy trình của hệ thống tìm kiếm

- **Tạo bộ ảnh đại diện cho các đặc trưng hình ảnh để tìm kiếm.** Sử dụng `imageDatastore` để lưu trữ dữ liệu hình ảnh. Sử dụng một số lượng lớn các hình ảnh đại diện cho các quan điểm khác nhau của đối tượng. Một số lượng lớn và đa dạng hình ảnh giúp đào tạo các bag of visual words và tăng tính chính xác của việc tìm kiếm hình ảnh.
- **Loại đặc trưng.** Hàm `indexImages` tạo ra các visual word sử dụng các đặc trưng được tăng tốc mạnh mẽ (SURF). Đối với các loại đặc trưng khác, ta có thể sử dụng một trình trích xuất tùy chỉnh (custom extractor), và sau đó sử dụng các tính năng `bagOfFeatures` để tạo ra các visual words. Có thể sử dụng `imgSet` ban đầu hoặc một bộ sưu tập hình ảnh khác cho tập huấn luyện. Để sử dụng một bộ sưu tập khác, ta tạo ra các visual words trước khi tạo chỉ mục hình ảnh, sử dụng hàm `bagOfFeatures`. Ưu điểm của việc sử dụng cùng một bộ hình ảnh là visual vocabulary được thiết kế riêng cho bộ tìm kiếm. Bất lợi

của cách tiếp cận này là hệ thống tìm kiếm phải dùng lại visual vocabulary để sử dụng trên một tập hợp các hình ảnh khác nhau đáng kể. Với một bộ độc lập, visual vocabulary sẽ có thể xử lý tốt hơn các hình ảnh mới bổ sung vào chỉ mục tìm kiếm.

- **Lập chỉ mục các hình ảnh.** Hàm `indexImages` tạo ra một chỉ mục tìm kiếm để ánh xạ các visual words với các lần xuất hiện của chúng trong bộ sưu tập hình ảnh. Khi tạo ra các visual words bằng cách sử dụng bộ sưu tập độc lập hoặc tập hợp con, các bag như một đối số đầu vào cho `indexImages`. Nếu không tạo ra một bag of visual words độc lập, thì hàm sẽ tạo ra bag dựa trên toàn bộ bộ sưu tập đầu vào `imgSet`. Có thể trực tiếp thêm và xóa hình ảnh từ chỉ mục hình ảnh bằng cách sử dụng các phương pháp `addImages` và `removeImages`.
- **Tìm kiếm từ bộ dữ liệu cho những hình ảnh tương tự.** Sử dụng hàm `retrieveImages` để tìm kiếm bộ ảnh những hình ảnh tương tự với hình ảnh truy vấn. Sử dụng thuộc tính `NumResults` để kiểm soát số kết quả. Ví dụ: để trả lại 10 hình ảnh tương tự hàng đầu, hãy đặt thuộc tính `ROI` để sử dụng một vùng nhỏ hơn của hình ảnh truy vấn. Một vùng nhỏ hơn rất hữu ích cho việc cô lập một đối tượng cụ thể trong một hình ảnh mà bạn muốn tìm kiếm.

2.3. Đánh giá việc tìm kiếm hình ảnh

Sử dụng hàm `evalImageRetrieval` để đánh giá việc tìm kiếm hình ảnh bằng cách sử dụng một hình ảnh truy vấn với một bộ kết quả đã biết. Nếu kết quả không phải là những gì bạn mong đợi, bạn có thể sửa đổi hoặc tăng cường các đặc trưng của bag of words. Kiểm tra kiểu của đặc trưng tìm kiếm. Loại đặc trưng được sử dụng để truy xuất phụ thuộc vào loại hình ảnh trong bộ sưu tập. Ví dụ: nếu bạn đang tìm kiếm một bộ sưu tập hình ảnh được tạo thành từ các cảnh, chẳng hạn như các bãi biển, thành phố, hoặc xa lộ, hãy sử dụng một đặc trưng hình ảnh toàn cục. Đặc trưng hình ảnh toàn cục, chẳng hạn như một biểu đồ màu, nắm bắt các yếu tố chính của toàn cảnh. Để tìm các đối tượng cụ thể trong bộ sưu tập hình ảnh, hãy sử dụng các đặc trưng hình ảnh cục bộ được trích ra xung quanh các điểm chính đối tượng.

2.4. Một vài tùy chọn tìm kiếm được lựa chọn trong đồ án:

2.4.1. Các bước tìm kiếm hình ảnh dựa sử dụng bộ các đặc trưng được tùy biến (sử dụng tùy biến màu sắc):

Computer Vision System Toolbox cung cấp một framework bag-of-features có thể tùy chỉnh để cài đặt một hệ thống tìm kiếm hình ảnh. Phương pháp bao gồm các bước:

❖ Bước 1 – Lựa chọn các đặc trưng hình ảnh (Image Features) cho việc tìm kiếm

Loại đặc trưng được sử dụng để truy xuất phụ thuộc vào loại hình ảnh trong bộ sưu tập. Ví dụ: nếu tìm kiếm bộ sưu tập ảnh có cảnh (bãi biển, thành phố, đường cao tốc), tốt hơn là sử dụng đặc trưng hình ảnh toàn cục, chẳng hạn như biểu đồ màu ghi lại nội dung màu của toàn cảnh. Tuy nhiên, nếu mục đích là để tìm các đối tượng cụ thể trong bộ sưu tập hình ảnh, thì các đặc trưng hình ảnh cục bộ được trích ra xung quanh các điểm chính của đối tượng là một sự lựa chọn tốt hơn.

Sử dụng đặc trưng tìm kiếm các hoa tương tự trong tập dữ liệu bằng cách sử dụng thông tin màu sắc trong hình ảnh truy vấn. Các hình ảnh trong bộ dữ liệu chứa một loại hoa trong mỗi hình ảnh. Do đó, một đặc trưng hình ảnh đơn giản dựa trên cách bố trí không gian của màu sắc là một nơi tốt để bắt đầu tìm kiếm.



Hình 2.2. Tập dữ liệu hình ảnh các loài hoa

Hàm sau mô tả thuật toán được sử dụng để trích xuất các điểm đặc trưng màu từ một hình ảnh nhất định. Chức năng này sẽ được sử dụng như một trích xuất đặc trưng tùy chỉnh của `bagOfFeatures` để trích xuất các đặc trưng màu sắc.

```
function [features, metrics] = exampleBagOfFeaturesColorExtractor(I)

[~,~,P] = size(I);

isColorImage = P == 3;

if isColorImage

    Ilab = rgb2lab(I);

    Ilab = imresize(Ilab, 1/16);

    [Mr,Nr,~] = size(Ilab);

    colorFeatures = reshape(Ilab, Mr*Nr, []);
```

```

rowNorm = sqrt(sum(colorFeatures.^2,2));

colorFeatures = bsxfun(@rdivide, colorFeatures, rowNorm + eps);

xnorm = linspace(-0.5, 0.5, Nr);

ynorm = linspace(-0.5, 0.5, Mr);

[x, y] = meshgrid(xnorm, ynorm);

features = [colorFeatures y(:) x(:)];

metrics = var(colorFeatures(:,1:3),0,2);

else

    features = zeros(0,5);

    metrics = zeros(0,1);

end

```

❖ Bước 2 – Tạo một Bag of Features

Với kiểu đặc trưng được xác định, bước tiếp theo là học visual vocabulary trong bagOfFeatures sử dụng một tập hợp các hình ảnh đào tạo. Đoạn mã dưới đây chọn một tập con ngẫu nhiên của các hình ảnh từ bộ dữ liệu để huấn luyện và sau đó đào tạo bagOfFeatures bằng tùy chọn 'CustomExtractor'.

```

trainingSet = splitEachLabel(flowerImageSet, 0.4, 'randomized');

colorBag = bagOfFeatures(trainingSet, ...

    'CustomExtractor', @exampleBagOfFeaturesColorExtractor, ...

    'VocabularySize', 10000);

```

❖ Bước 3 – Lập chỉ mục các hình ảnh

Bây giờ khi bagOfFeatures được tạo ra, toàn bộ bộ ảnh hoa có thể được lập chỉ mục để tìm kiếm. Thủ tục lập chỉ mục trích xuất các điểm đặc trưng từ mỗi bức ảnh sử dụng chức năng trích xuất tùy chỉnh từ bước 1. Các điểm đặc trưng được trích xuất được mã hoá thành một biểu đồ visual word và được thêm vào chỉ mục hình ảnh.


```
flowerImageIndex = indexImages(flowerImageSet, colorBag,
'SaveFeatureLocations', false);
```

❖ Bước 4 – Tìm kiếm những hình ảnh tương tự

Bước cuối cùng là sử dụng hàm `retrieveImages` để tìm kiếm những hình ảnh tương tự.

```
queryImage = readimage(flowerImageSet, 502);

figure

imshow(queryImage)

[imageIDs, scores] = retrieveImages(queryImage, flowerImageIndex);

helperDisplayImageMontage(flowerImageSet.Files(imageIDs))
```



Hình 2.3. Tập các hình ảnh tương tự

2.4.2. Tìm kiếm trong hình ảnh sử dụng hình ảnh truy vấn

Với tập dữ liệu Hoa ban đầu, ta lập chỉ mục mới cho tập hình ảnh

```
imageIndex = indexImages(flowerImageSet);
```

Tìm hình ảnh truy vấn phù hợp

```
imageIDs = retrieveImages(queryImage,imageIndex);
```

Hiển thị hình ảnh truy vấn và kết quả phù hợp nhất

```
helperDisplayImageMontage(handles.flowerImageSet.Files(imageIDs(1)));
```

2.4.3. Tìm kiếm bộ hình ảnh cho đối tượng cụ thể sử dụng ROIs

Tìm kiếm trong một tập hợp hình ảnh cho một đối tượng sử dụng vùng quan tâm (ROI) cho hình ảnh truy vấn.

Xác định một bộ hình ảnh để tìm kiếm.

```
imageFiles = ...  
{'elephant.jpg', 'cameraman.tif', ...  
 'peppers.png', 'saturn.png',...  
 'pears.png',    'stapleRemover.jpg', ...  
 'football.jpg', 'mandi.tif',...  
 'kids.tif',     'liftingbody.png', ...  
 'office_5.jpg', 'gantrycrane.png',...  
 'moon.tif',     'circuit.tif', ...  
 'tape.png',     'coins.png'};
```

```
imds = imageDatastore(imageFiles);
```

Tạo chỉ mục tìm kiếm.

```
imageIndex = indexImages(imds);
```

Chỉ định một hình ảnh truy vấn và ROI. ROI phác thảo đối tượng, một con voi, để tìm kiếm.

```
queryImage = imread('clutteredDesk.jpg');  
queryROI = [130 175 330 365];  
  
figure  
imshow(queryImage)  
  
rectangle('Position',queryROI,'EdgeColor','yellow')
```



Hình 2.4. Ảnh chứa đối tượng cần tìm kiếm

Có thể sử dụng chức năng `imrect` để chọn tương tác ROI. Ví dụ, `queryROI = getPosition(imrect)`

Tìm hình ảnh có chứa đối tượng.

```
imageIDs = retrieveImages(queryImage,imageIndex,'ROI',queryROI)
```

Hiển kết quả phù hợp nhất.

```
bestMatch = imageIDs(1);
```

figure

```
imshow(imageIndex.ImageLocation{bestMatch})
```



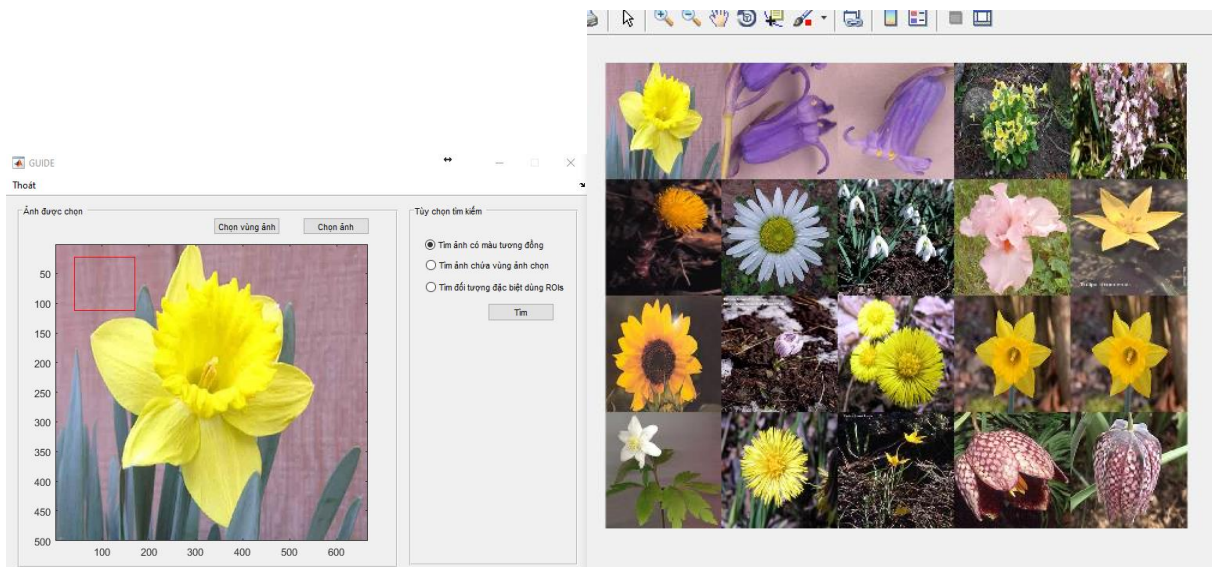
Hình 2.5. Kết quả tìm kiếm

3. Tổng kết:

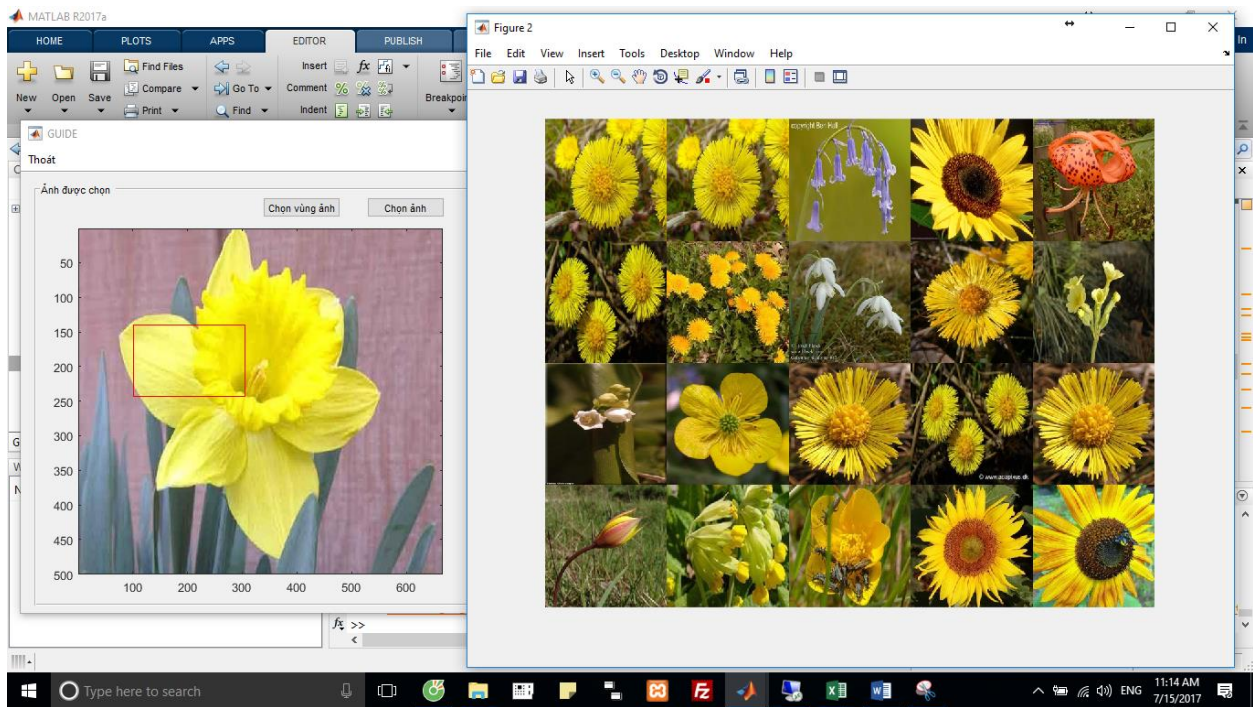
3.1. Kết quả thực hiện:

Đạt được mục tiêu ban đầu đặt ra:

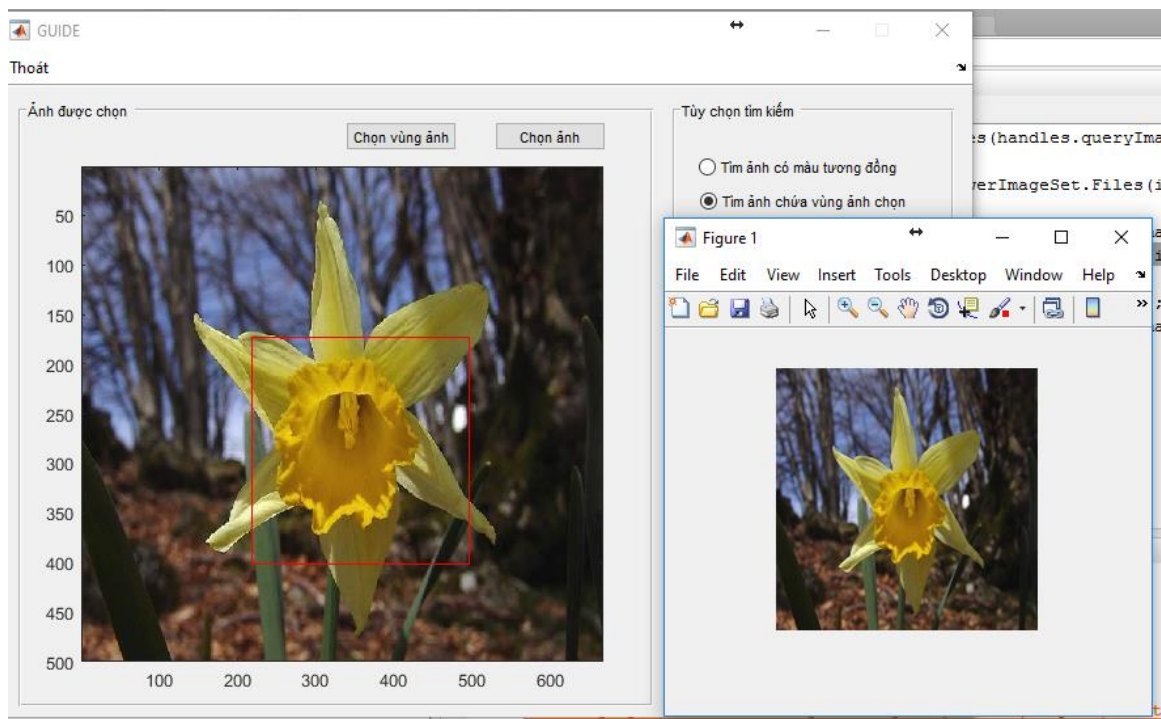
- Xây dựng ứng dụng tìm kiếm hình ảnh đơn giản, có giao diện
- Ứng dụng được Bag of Words vào việc hỗ trợ tìm kiếm hình ảnh.
- Cho phép tìm kiếm hình ảnh với một vài tùy chọn khác nhau như:
 - o Tìm kiếm hình ảnh theo màu
 - o Tìm kiếm hình ảnh theo hình
 - o Xác định được đối tượng đang chọn
 - o Cho phép chọn vùng tìm kiếm



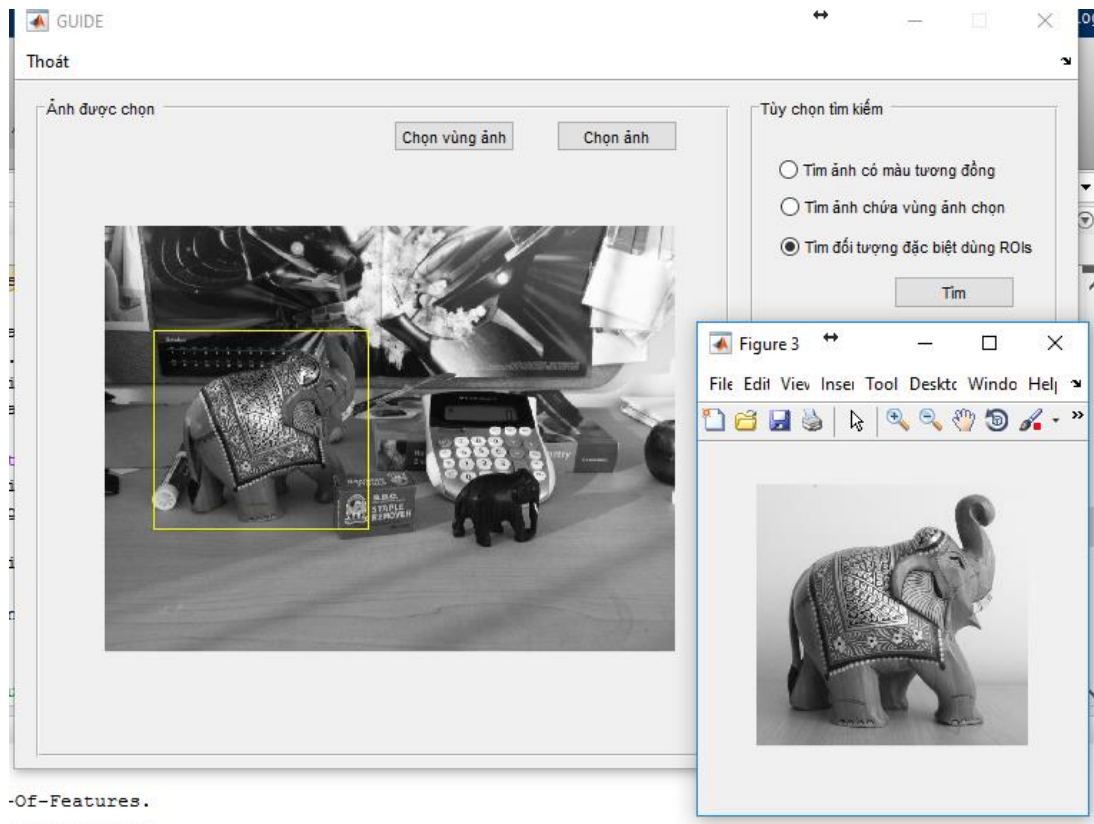
Hình 3.1. Kết quả tìm kiếm theo vùng màu



Hình 3.2. Kết quả tìm kiếm theo vùng màu



Hình 3.3. Kết quả tìm kiếm hình ảnh chứa vùng ảnh đã chọn



Hình 3.4. Kết quả tìm kiếm theo đối tượng đặc biệt sử dụng ROIs

3.2. Hạn chế:

- Chỉ sử dụng được trên tập dữ liệu có sẵn
- Kết quả tìm kiếm với độ chính xác chưa cao
- Vẫn còn hạn chế về tùy chọn tìm kiếm

3.3. Hướng phát triển

Mở rộng chức năng, cho phép người dùng tự chọn dữ liệu huấn luyện cũng như cài đặt các tùy chọn huấn luyện (tập các thông số đầu vào để huấn luyện).

4. Tham khảo

- [1] L. Đ. Duy and N. T. T. M. Khang, Slide Nhận dạng thị giác và ứng dụng, HCM, 2017.
- [2] T. MathWorks, "Create a Custom Feature Extractor," 2017. [Online]. Available: <http://www.mathworks.com/help/vision/ug/create-a-custom-feature-extractor.html>. [Accessed 5 July 2017].
- [3] T. MathWorks, "Image Retrieval Using Customized Bag of Features," 2016. [Online]. Available: <https://www.mathworks.com/help/vision/examples/image-retrieval-using-customized-bag-of-features.html>. [Accessed 24 June 2017].
- [4] T. MathWorks, "Search image set for similar image," 2016. [Online]. Available: <https://www.mathworks.com/help/vision/ref/retrieveimages.html#buo0f28-1>. [Accessed 7 July 2017].
- [5] T. MathWorks, "Save and Load Parts of Variables in MAT-Files," 2016. [Online]. Available: https://www.mathworks.com/help/matlab/import_export/load-parts-of-variables-from-mat-files.html. [Accessed 2 July 2017].
- [6] T. MathWorks, "Create a Custom Feature Extractor," 2017. [Online]. Available: <https://www.mathworks.com/help/vision/ug/create-a-custom-feature-extractor.html>. [Accessed 22 June 2017].
- [7] T. MathWorks, "Specify rectangle with mouse," 2017. [Online]. Available: <https://www.mathworks.com/help/images/ref/getrect.html>. [Accessed 3 July 2017].