

BTVN-03: ĐẶC TRƯNG CỤC BỘ

Thị giác máy tính – 19TGMT



Giáo viên:

- TS. Trần Thái Sơn
- ThS. Nguyễn Trọng Việt
- ThS. Võ Hoài Việt

Sinh viên:

- Chung Kim Khánh (19127644)

Mục lục

I.	THÔNG TIN.....	2
II.	BẢNG ĐÁNH GIÁ	2
III.	THUẬT TOÁN VÀ ẢNH KẾT QUẢ.....	2
IV.	HƯỚNG DẪN SỬ DỤNG	43
V.	NGUỒN THAM KHẢO.....	43

I. Thông tin

MSSV	HỌ TÊN	EMAIL	SĐT
19127644	Chung Kim Khánh	ckkhanh19@clc.fitus.edu.vn	0779930393

II. Bảng đánh giá

STT	TÊN KẾT QUẢ	TỈ LỆ ĐIỂM	MÚC ĐỘ HOÀN THÀNH
1	<ul style="list-style-type: none"> Trình bày mã nguồn Code trong sáng, rõ ràng Chú thích đầy đủ Đặt tên biến, tên hàm 	30%	100%
2	Thực hiện đủ yêu cầu chức năng	40%	100%
3	Báo cáo	30%	100%

III. Thuật toán và ảnh kết quả

1. Harris

Harris keypoint detector được Harris và Stephens giới thiệu vào năm 1988 trong bài báo *A Combined Corner and Edge Detox*. Đây là một trong những thuật toán xác định corners phổ biến nhất mà bạn sẽ gặp rất nhiều trong thế giới Computer Vision [2].

Về mặt ý tưởng, thuật toán Harris sẽ tìm kiếm sự thay đổi lớn về cường độ xám theo các hướng khác nhau bằng cách dùng một cửa sổ nhỏ để làm nhiệm vụ rà soát và phát hiện những điểm được định nghĩa là “góc” [4].

Chúng ta có thể sử dụng function cornerHarris trong thư viện OpenCv, nhưng để hiểu bản chất thuật toán hơn chúng ta sẽ implement lại từ đầu thuật toán [3].

- Đầu tiên, chuyển từ ảnh RGB thành ảnh xám sau đó xoá nhiễu bằng mặt nạ Gauss
- Thực hiện đạo hàm ảnh theo x, y bằng Sobel, Ix, Iy
- Tính toán độ tương quan giữa các đạo hàm:
 - $A = Ix^2$
 - $B = Ix * Iy$
 - $C = Iy^2$

Có 1 điểm đặc biệt ở đây là khi nhân ma trận, chúng ta sẽ không thực hiện phép nhân ma trận thông thường mà nhân các điểm tương ứng của ma trận này với ma trận kia: Ví dụ: $A[i][j] = Ix[i][j] * Ix[i][j]$, sau đó áp dụng mặt nạ Gauss lên các ma trận A, B, C thu được _A, _B, _C

4. Tính corner response:

$$R = \det(M) - k * \text{Trace}(M)$$

Với: $\det(M) = _A * _B - _C^2$, $\text{Trace}(M) = _A + _B$, k là giá trị thực nghiệm $[0,04 - 0,06]$

5. Threshhold và lấy giá trị max ở mỗi vùng. Ở bước này sau khi threshhold, chúng ta sẽ lấy 1 cửa sổ (3×3) trượt qua từng pixel của R và lấy giá trị max ở mỗi vùng.

Sau đây là kết quả của thuật toán tự viết:



BTVN-03: ĐẶC TRƯNG CỤC BỘ



2. Blob

Blob là những điểm nhỏ, nhóm điểm. Hãy tưởng tượng một tờ giấy trắng bị vẩy mực lên, thì khi đó mỗi vết mực tạo thành một blob. Một trong những thiết bị phát hiện đốm màu đầu tiên và cũng là phổ biến nhất dựa trên tiếng Laplacian của người Gaussian (LoG) [5].





3. DoG (Difference of Gaussian)

Trong tài liệu về thị giác máy tính, cách tiếp cận này được gọi là sự khác biệt của cách tiếp cận Gaussians (DoG). Tuy nhiên, bên cạnh những kỹ thuật nhỏ, toán tử này về bản chất tương tự như Laplacian và có thể được xem như là một phép gần đúng của toán tử Laplacian. Theo cách tương tự như đối với máy dò đốm màu Laplace, các đốm màu có thể được phát hiện từ cực trị không gian quy mô của sự khác biệt của Gaussian — xem (Lindeberg 2012, 2015) để biết mối quan hệ rõ ràng giữa toán tử sai biệt của Gaussian và toán tử Laplacian chuẩn hóa theo tỷ lệ. Ví dụ, cách tiếp cận này được sử dụng trong thuật toán biến đổi đối tượng bất biến theo tỷ lệ (SIFT) — xem Lowe (2004) [5].

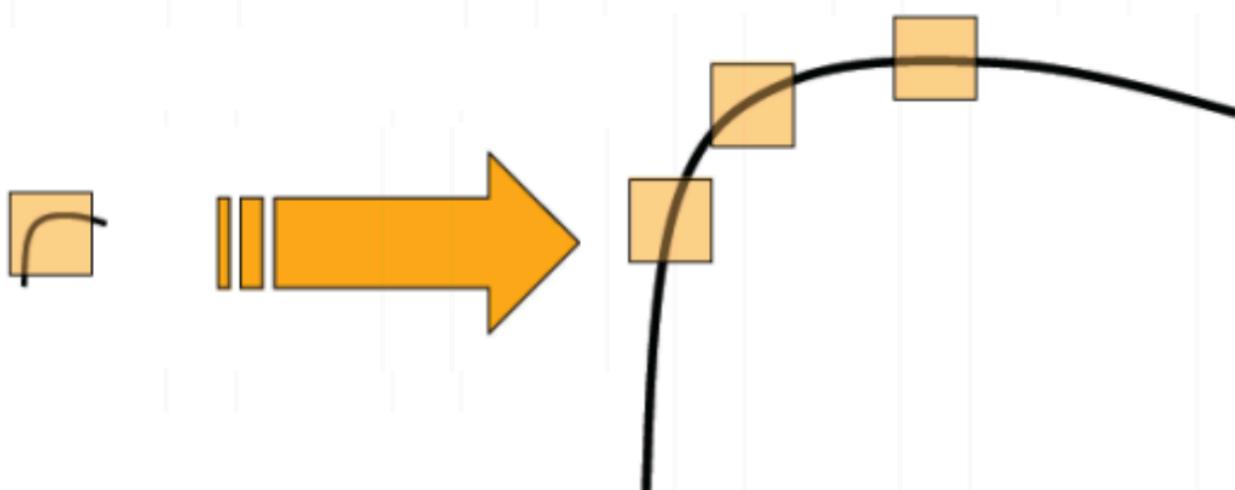
BTVN-03: ĐẶC TRƯNG CỤC BỘ





4. SIFT (Scale-Invariant Feature Transform)

Trong quá trình nghiên cứu so sánh đối chiếu các điểm đặc trưng của ảnh số, người ta thường gặp những vấn đề có liên quan đến việc các điểm đặc trưng tương đồng giữa các ảnh không có được sự so sánh đối chiếu chính xác nếu như các điểm đó không có cùng tỉ lệ. Điều này có nghĩa là nếu ta cố gắng tìm ra các điểm đặc trưng tương đồng từ hai hình ảnh được chụp lại từ hai góc độ khác nhau của một khung hình, ta sẽ không có được kết quả tốt nhất [4].



Hình 1. Mô tả sự không hoàn thiện của thuật toán Harris

Trong hình trên, ta dễ dàng nhận ra ở bên trái là một góc đã được thuật toán tìm kiếm góc Harris phát hiện. Thế nhưng nếu như ta xem xét góc này với một tỉ lệ lớn hơn nhiều thì đó không còn là một góc theo thuật toán Harris nữa, nhưng trên thực tế, đó hoàn toàn là một góc.

Để giải quyết vấn đề này, khái niệm “Điểm đặc trưng cục bộ bất biến SIFT” (Scale-Invariant Feature Transform) đã được David G. Lowe đưa ra hồi năm 2004 trong “Distinctive Image Features from Scale-Invariant Keypoints”.

Phương pháp trích chọn điểm đặc trưng cục bộ bất biến SIFT gồm các bước:

- Phát hiện các điểm cực trị Scale-Space: Tìm những khu vực chứa những điểm đặc trưng tiềm năng có tính chất không thay đổi dưới các phép phóng và xoay ảnh.
- Định vị các điểm nổi bật: Một hàm kiểm tra sẽ được đưa ra để quyết định xem các điểm nổi bật tiềm năng có được lựa chọn hay không.
- Xác định hướng cho các điểm nổi bật: Mỗi điểm nổi bật sẽ được gán cho một hoặc nhiều hướng dựa trên hướng gradient của ảnh. Mọi phép toán xử lý ở các bước sau này sẽ được thực hiện trên những dữ liệu ảnh mà đã được biến đổi tương đối so với hướng đã gán, kích cỡ và vị trí của mỗi điểm đặc trưng. Nhờ đó, tạo ra một sự bất biến trong các phép xử lý này.
- Mô tả các điểm nổi bật: Các hướng gradient cục bộ được đo trong ảnh có kích cỡ cụ thể nào đó trong vùng lân cận với mỗi điểm đặc trưng. Sau đó, chúng sẽ được biểu diễn thành một dạng mà cho phép mô tả các tầng quan trọng của quá trình bóp méo hình dạng cục bộ và sự thay đổi về độ sáng.

Sau đây là thử nghiệm tìm điểm đặc trưng của 2 ảnh bằng phương pháp SIFT sử dụng Harris để nhận dạng key-points.





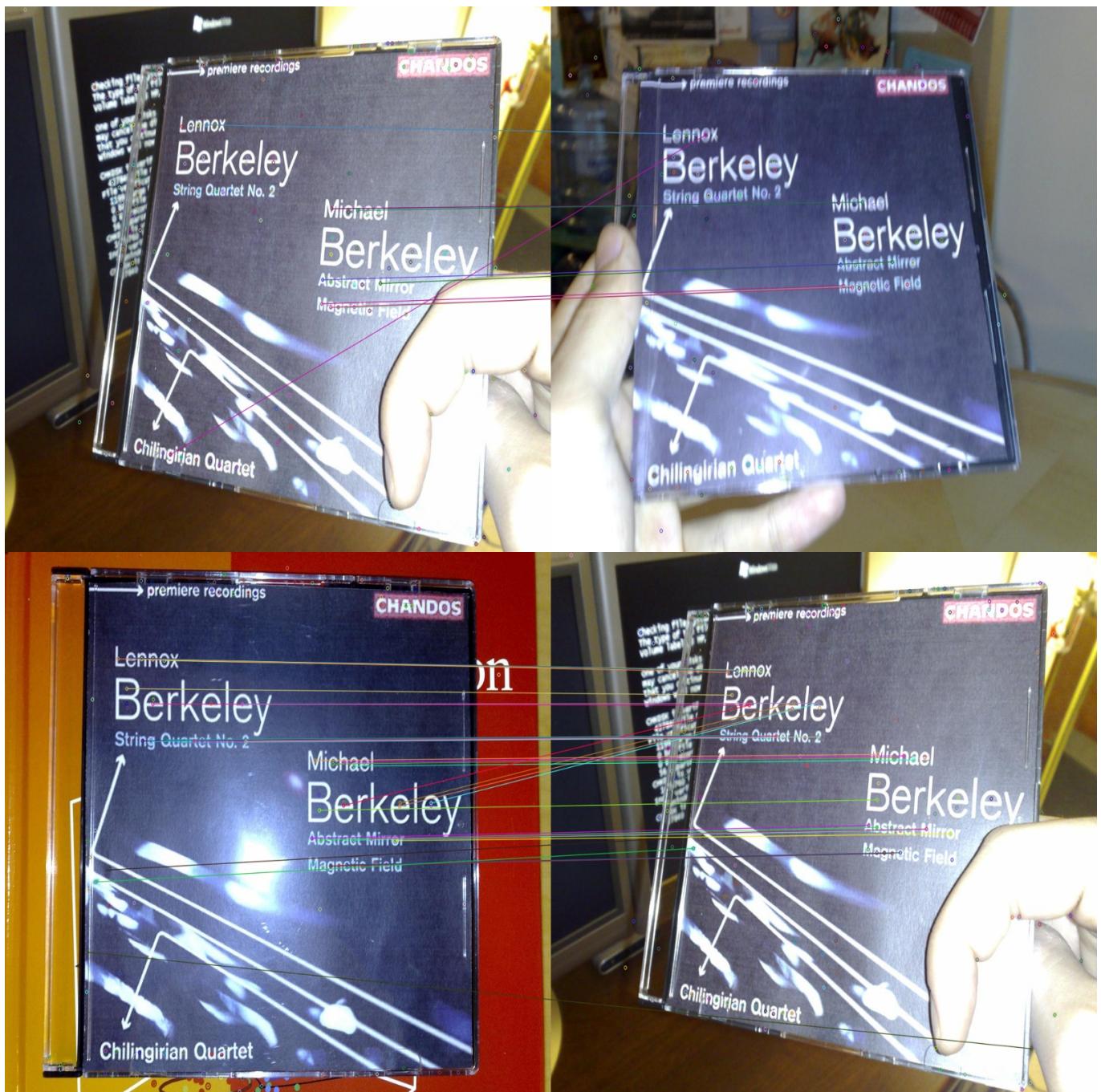




BTVN-03: ĐẶC TRƯNG CỤC BỘ



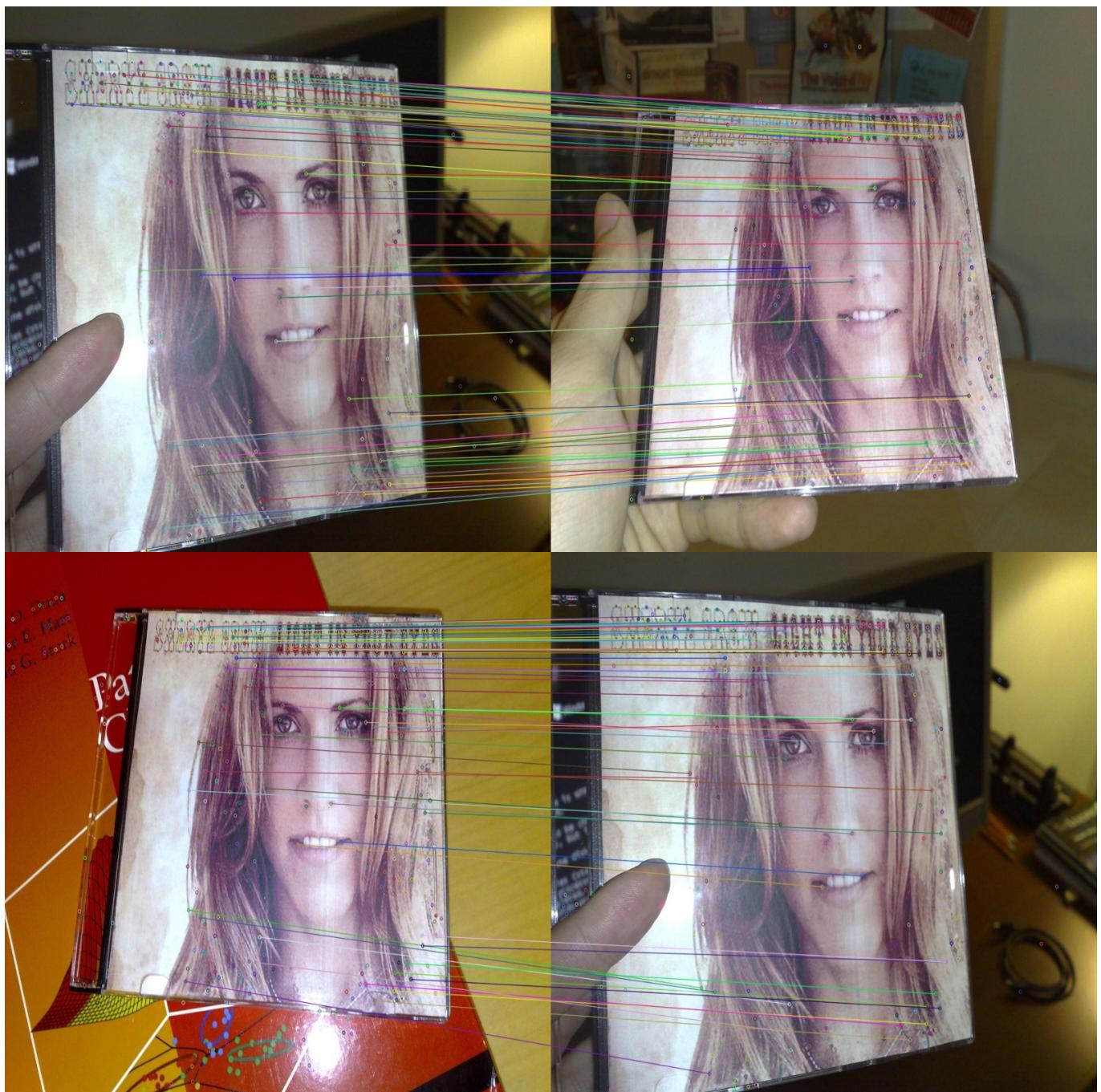














BTVN-03: ĐẶC TRƯNG CỤC BỘ

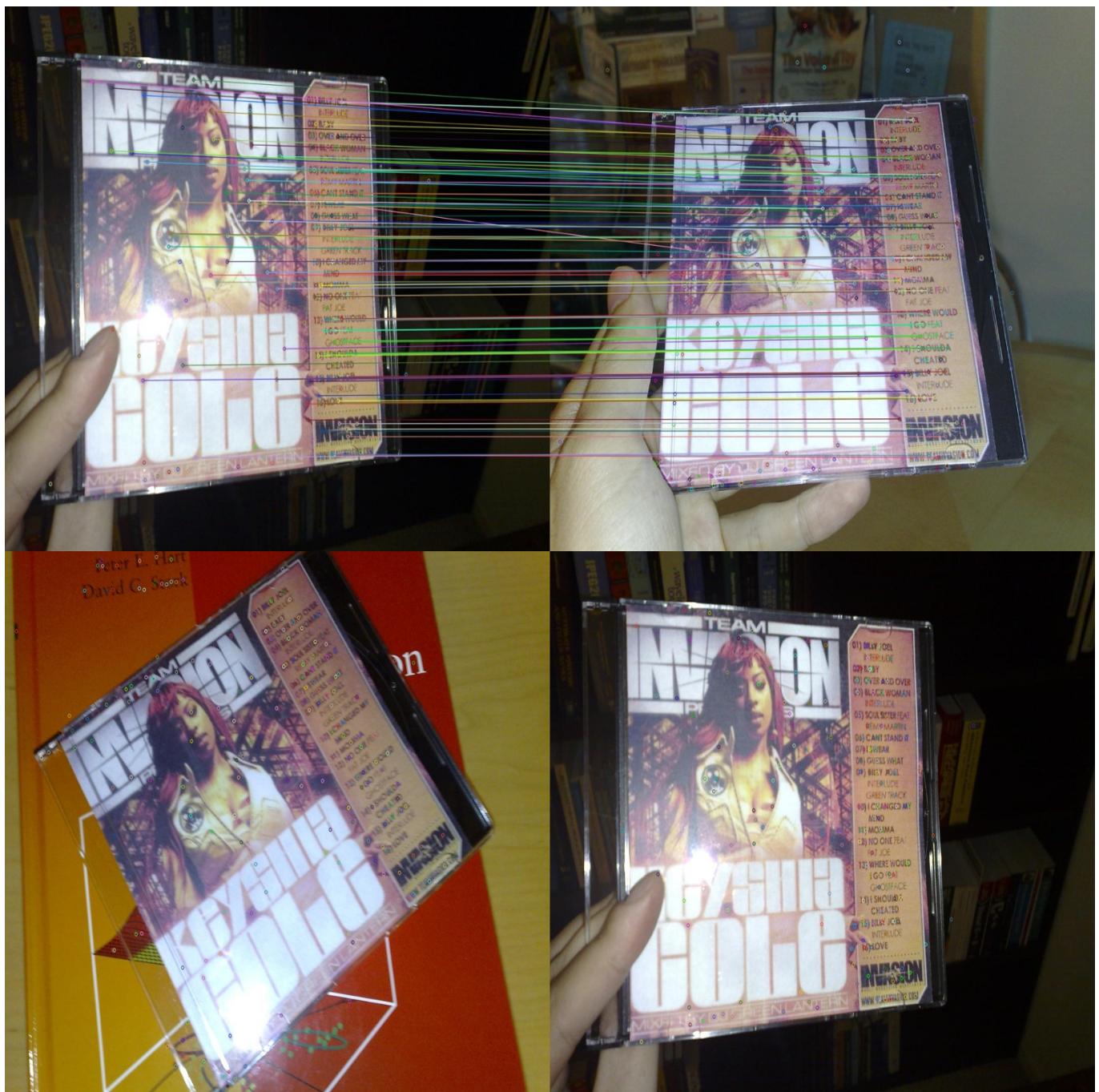




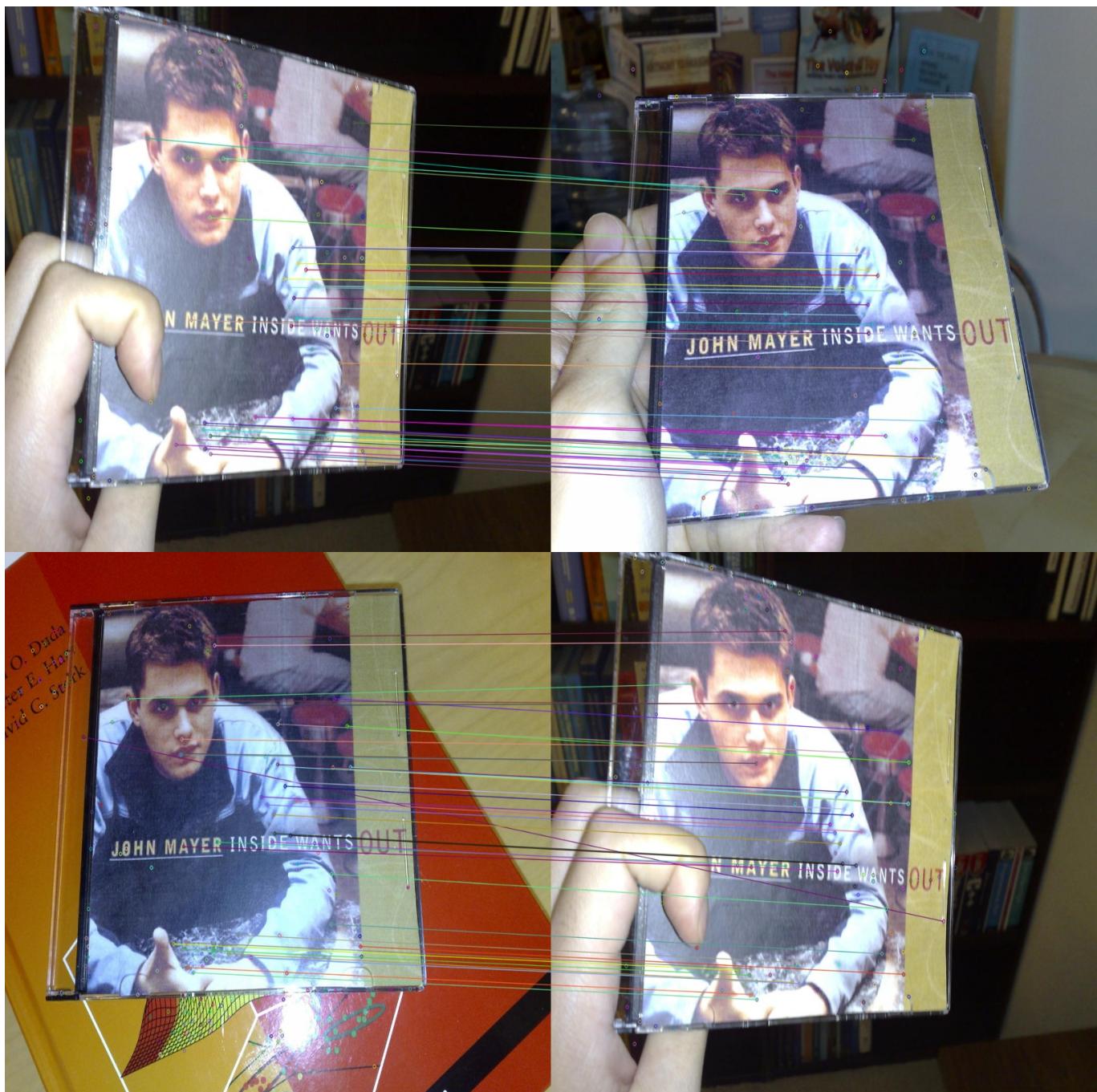












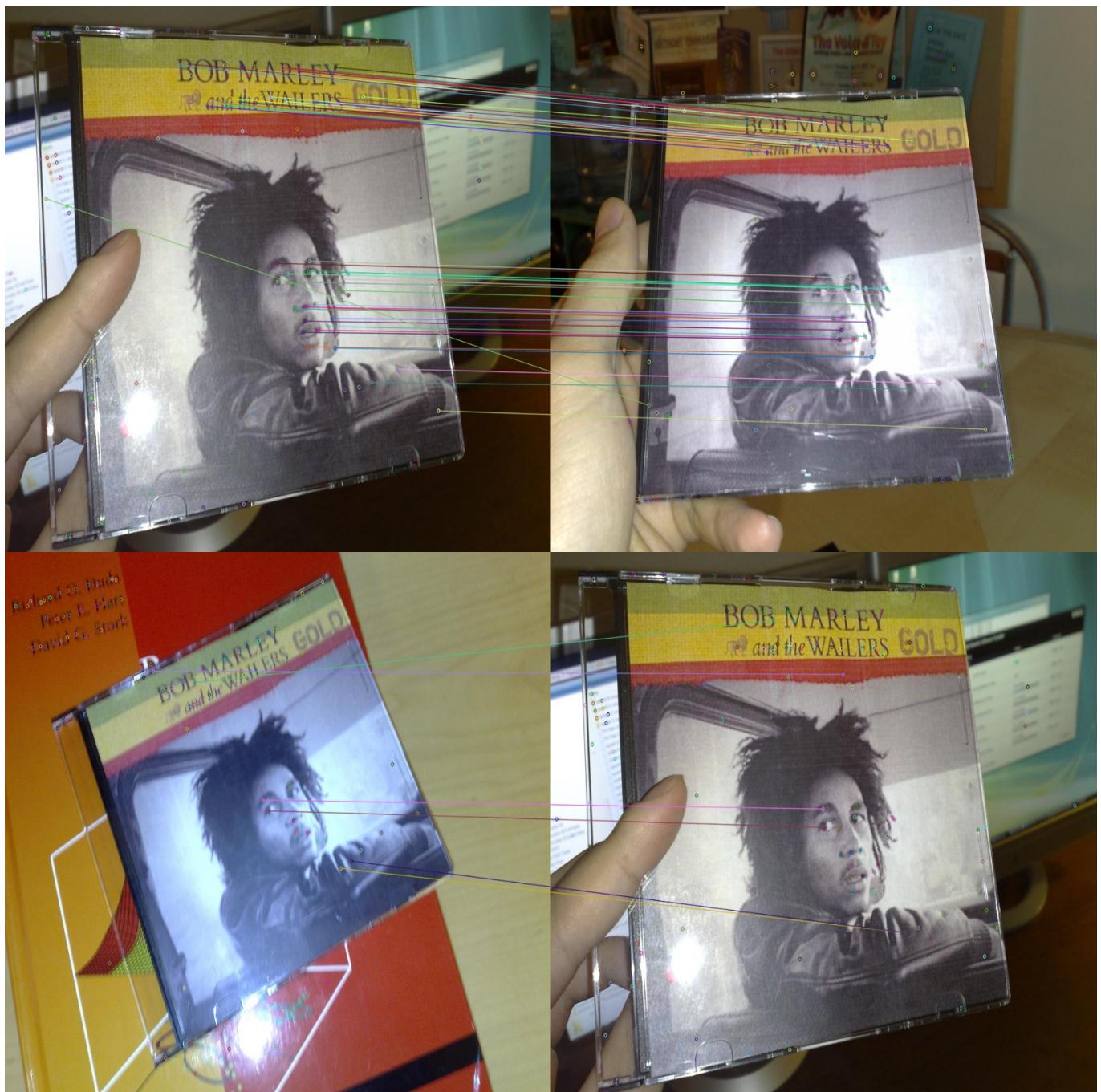






BTVN-03: ĐẶC TRƯNG CỤC BỘ











BTVN-03: ĐẶC TRƯNG CỤC BỘ











5. So sánh

Qua việc thử nghiệm và những kết quả trên, ta đưa ra được Harris là thuật toán có tốc độ chạy nhanh nhất. Tuy nhiên, thuật toán Harris có một nhược điểm là nó không là scale invariant tức là cùng một bức ảnh đó, khi ta phóng to/thu nhỏ lại thì Harris không tìm lại được các corner trong ảnh gốc. Vì vậy, ta cần sử dụng một số thuật toán khác để thỏa mãn được tính chất scale invariant [6].

Tìm điểm maxima của một hàm số nào đó mà thỏa mãn yêu cầu scale invariant & rotation invariant. Hàm số thường được sử dụng là DoG (Difference of Gaussian). Trong đó,

ta tính hàm gaussian với các kernel size khác nhau cho ảnh, rồi trừ hai Gaussian liên tiếp thì thu được các DoG.

IV. Hướng dẫn sử dụng

Bước 1: Chạy cmd tại thư mục Release

Bước 2: Nhập dòng lệnh có cú pháp

<name.exe> <path> <command-name> <parameter-1>(nếu có) <parameter-2>(nếu có)

<name.exe> là tên của chương trình, nhập 19127644_Lab03.exe

<path> là đường dẫn đến hình ảnh, ví dụ D:\demo.jpg

<command-name> là tên lệnh

- -harris : sử dụng thuật toán của Harris
- -blob : sử dụng thuật toán của Blob
- -dog : sử dụng thuật toán của DoG
- -sift : sử dụng thuật toán của SIFT

Nếu sử dụng SIFT thì sẽ có 2 files ảnh nhập vào

V. Nguồn tham khảo

[1] <https://opencv.org/>

[2] [UNIDUC](#)

[3] [VIBLO](#)

[4] [Báo cáo nghiên cứu khoa học, Khoa Điện-Điện tử, Đại học Dân Lập Duy Tân](#)

[5] [Blob Detection – Wikipedia](#)

[6] [Image Preprocessing, DoThanhBlog](#)

[7] Slide lý thuyết