BÁO CÁO ĐỒ ÁN

TRUY VẤN THÔNG TIN THỊ GIÁC

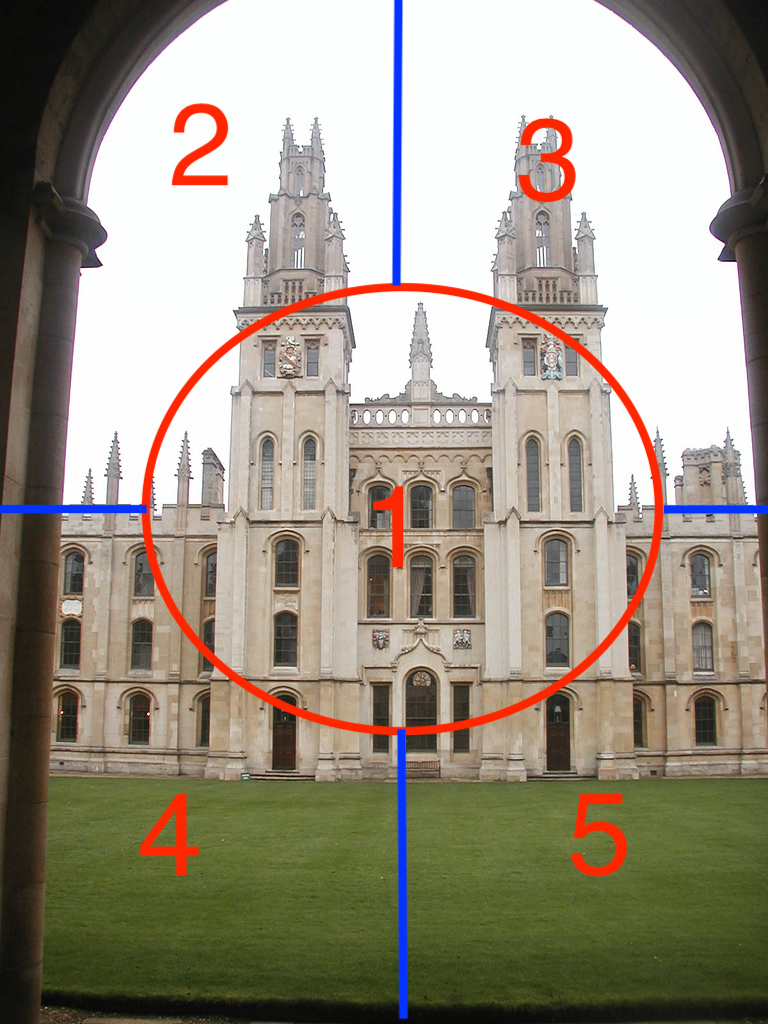
Đề tài : Truy vấn hình ảnh

Học viên : Lê Duy Thành -CH1701016

# Phương pháp và kỹ thuật

## Phương pháp

* Phần mềm truy vấn hình ảnh này sẽ tìm kiếm hình ảnh dựa trên nội dung của bức ảnh, cụ thể là histogram của bức ảnh. Tuy nhiên, điểm khác biệt ở chỗ, sẽ không dùng histogram của toàn bức ảnh để so sánh khi tìm kiếm mà bức ảnh sẽ được chia làm 5 vùng và thực hiện tìm kiếm bằng cách so sánh histogram của 5 vùng này. 5 vùng này bao gồm : góc trên bên trái, góc trên bên phải, góc dưới bên trái, góc dưới bên phải và vùng trung tâm. 5 vùng ảnh trên sẽ được tính histogram riêng cho mỗi vùng và được lưu thành 1 vector.

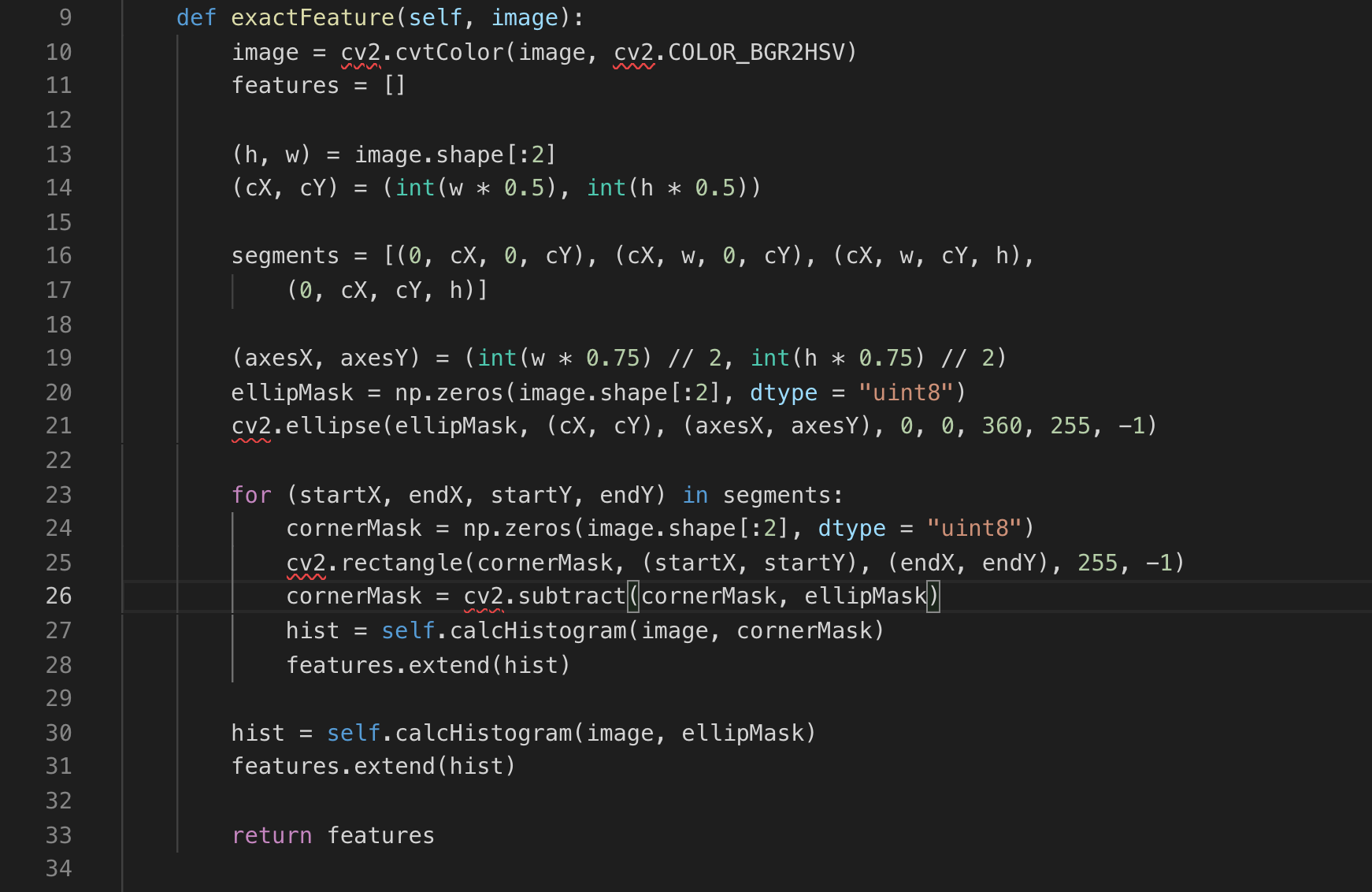


* Sử dụng HSV histogram(Hue, Saturation, Value) thay vì sử dụng histogram thông thường. Như vậy ta sẽ biểu diễn mỗi thành thành 1 vector với số nhiều là tích các bin của HSV( H x S x V). Các giá trị bins của HSV được xem như 1 giá trị tuỳ biến(turning) phụ thuộc vào nhu cầu hoặc số lượng data. Gía trị bins của HSV càng lớn thì so sánh càng chi tiết nhưng lại tốn nhiều tài nguyên, bỏ qua cái toàn cục và ngược lại. Vậy tại sao lại sử dụng HSV chứ không phải sử dụng các histogram khác. Thứ nhất, dataset đa số là ảnh màu mà màu sắc được xem như 1 nội dung của bức ảnh và việc so sánh màu sắc của bức ảnh cũng có thể đem đến 1 kết quả tốt, vậy nên sử dụng histogram ảnh xám có vẻ không phù hợp. Vậy RGB histogram, đa số chúng ta sẽ quen thuộc với RGB hơn là HSV tuy nhiên HSV lại được đánh giá là phù hợp với việc phân tích ảnh nhiều hơn
* So sánh 2 ảnh là tính toán khoảng cách giữa 2 vector biểu diễn cho 2 ảnh, khoảng cách càng ngắn thì 2 ảnh càng giống nhau. Có rất nhiều phương pháp tính khoảng cách giữa 2 vector như : Euclidean, Cosine , chi-square. Trong đó, chi-square được xem là phương pháp tính khoảng cách tốt nhất đối với histogram. Do vậy, trong project này sử dụng phương pháp chi-square để tính khoảng cách giữa 2 vector histogram

## Kỹ thuật

Phần mềm nãy sẽ được xây dựng qua 4 bước cơ bản sau đây:

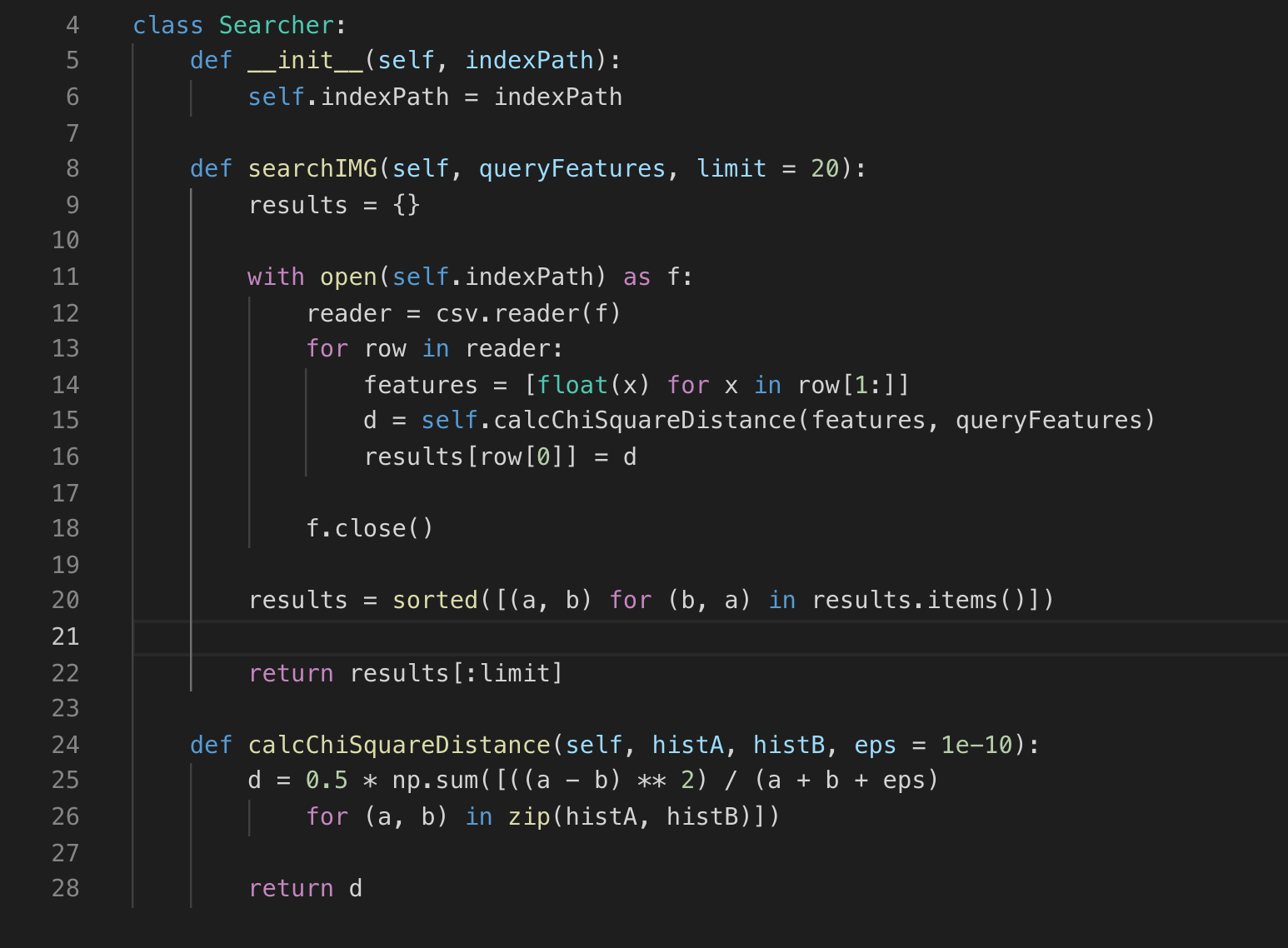
* Xây dựng hàm rút trích đặc trưng : Đặc trưng ở đây chính 5 HSV histogram của 5 vùng ảnh đã được xác định ở phần trên. Trước hết là chia mỗi bức ảnh là thánh 5 vùng tuỳ thuộc vào kích thước ảnh. Sau khi phân chia thì tiến hành tính HSV histogram của 5 vùng trên sử dụng openCV. Như vậy với mỗi ảnh, ta sẽ có 1 vector với số chiều là (H x S x V) x 5. Mỗi vector này sẽ được map với ID của mỗi ảnh và lưu vào file .csv để sử dụng cho việc so sánh sau này.



* Xử lý dataset : sau khi đã có hàm rút trích đặc trừng, ta tiến hành xử lý đối với dataset. Tuỳ thuộc vào độ lớn của dataset hoặc như cầu ta sẽ tuỳ biến các giá trị bins của HSV, bins càng lớn thì vector càng nhiều chiều, histogram càng chi tiết.



* Xây dựng hàm so sánh : như đã đề cập ở trên, project sử dụng chi-square làm phương pháp tính khoảng cách 2 vector. Khoảng cách càng nhỏ thì 2 ảnh càng giống nhau



* Xây dựng giao diện người dùng: project sử dụng Flask để xây dựng giao diện người dùng trên nền web

1. Run project

* Copy dataset vào folder dataset của project
* Cd đến thử mục chưa source và run python app.py trong command line
* Truy cập <http://127.0.0.1:5000/> trên browser

# Đánh giá

Mặc dù tuỳ chình nhiều thông số bins khác nhau cho HSV histogram nhưng kết quả thu được không được tốt. Nếu bins quá lớn dẫn đến số chiều của vector đặc trưng cũng tăng theo làm tăng dung lượng file dataset và thời gian truy vấn cũng tăng theo. Có vẻ việc sử dụng histogram trong truy vấn ảnh đem lại kết quả không cao đối với những hình ảnh có bố cục, màu sắc tương đối giống nhau.

# Tài liệu tham khảo

* <https://docs.opencv.org/3.0-beta/doc/py_tutorials/py_tutorials.html>
* <https://www.hindawi.com/journals/mpe/2015/352849/>
* <https://www.pyimagesearch.com>
* <https://docs.opencv.org/2.4/doc/tutorials/imgproc/histograms/histogram_comparison/histogram_comparison.html>
* <https://www.researchgate.net/post/What_is_chi-squared_distance_I_need_help_with_the_source_code>