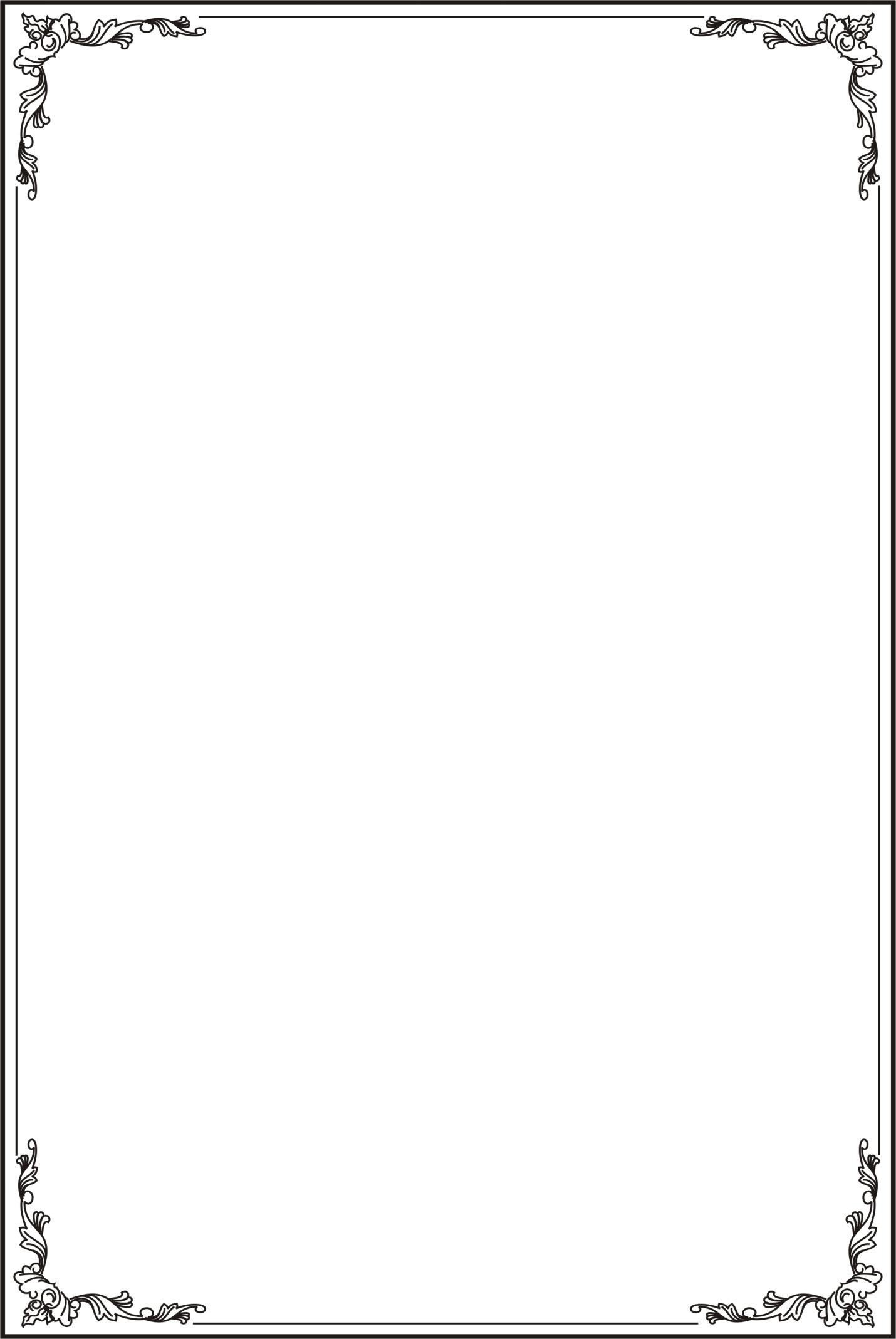
HỌC VIỆN CÔNG NGHỆ BƯU CHÍNH VIỄN THÔNG



KHOA CÔNG NGHỆ THÔNG TIN

**BÁO CÁO MÔN HỌC**

**HỆ CƠ SỞ DỮ LIỆU ĐA PHƯƠNG TIỆN**

**Giảng viên giảng dạy: Nguyễn Đình Hóa**

**Chủ đề: Xây dựng hệ CSDL lưu trữ và tìm kiếm hình ảnh đồ dùng văn phòng**

**Nhóm lớp học: 03**

**Nhóm bài tập lớn: 18**

**Thành viên:**

|  |  |
| --- | --- |
|  |  |
|  |  |
| Nguyễn Hải Phong | B20DCCN495 |

Hà Nội 2025

# MỤC LỤC

[LỜI NÓI ĐẦU 2](#_Toc198470932)

[YÊU CẦU 3](#_Toc198470933)

[PHẦN 1: XÂY DỰNG BỘ DỮ LIỆU 5](#_Toc198470934)

[1. Các nguồn dữ liệu 5](#_Toc198470935)

[2. Lưu trữ dữ liệu 5](#_Toc198470936)

[3. Miêu tả 7](#_Toc198470937)

[PHẦN 2: MỘT SỐ KỸ THUẬT XỬ LÝ VÀ NHẬN DIỆN ẢNH 9](#_Toc198470938)

[I. Các kĩ thuật xử lý ảnh (trích rút đặc trưng ảnh): 9](#_Toc198470939)

[1. Đặc trưng nội dung hình ảnh: 9](#_Toc198470940)

[II. Một số kỹ thuật nhận diện, tính toán độ tương đồng: 13](#_Toc198470941)

[1. Mục tiêu 13](#_Toc198470942)

[2. Công thức tổng hợp độ tương đồng 13](#_Toc198470943)

[3. Cách xác định trọng số ω 14](#_Toc198470944)

[PHẦN 3: HỆ THỐNG NHẬN DẠNG VÀ TÌM KIẾM ẢNH TƯƠNG ĐỒNG 16](#_Toc198470945)

[I. Sơ đồ khối và quy trình thực hiện 16](#_Toc198470946)

[1. Sơ đồ khối 16](#_Toc198470947)

[2. Quy trình thực hiện 16](#_Toc198470948)

[II. TRÍCH RÚT ĐẶC TRƯNG VÀ KỸ THUẬT TRÍCH RÚT 17](#_Toc198470949)

[1. Histogram – Phân bố mức xám toàn cục 17](#_Toc198470950)

[2. GLCM – Mô tả kết cấu cục bộ 18](#_Toc198470951)

[3. HOG – Biểu diễn cạnh và hình dạng 19](#_Toc198470952)

[4. ORB – Đặc trưng cục bộ nhị phân 20](#_Toc198470953)

[5. Tổng hợp: 20](#_Toc198470954)

[III. Lưu trữ và sử dụng các đặc trưng để tìm kiếm ảnh trong hệ thống 21](#_Toc198470955)

[1. Lưu trữ 21](#_Toc198470956)

[2. Tìm kiếm 22](#_Toc198470957)

[PHẦN 4: CÀI ĐẶT CHƯƠNG TRÌNH VÀ DEMO 23](#_Toc198470958)

[I. Module trích xuất đặc trưng ảnh 23](#_Toc198470959)

[Các hàm trích xuất đặc trưng 23](#_Toc198470960)

[II. Module tìm kiếm ảnh tương đồng với ảnh đầu vào 25](#_Toc198470961)

[1. Hàm lấy ra bộ dữ liệu ảnh đã trích xuất của hệ thống 25](#_Toc198470962)

[2. Hàm lấy ra bộ trọng số đã tính được thông qua Giải Thuật Di Truyền: 25](#_Toc198470963)

[3. Hàm tính toán độ tương đồng 26](#_Toc198470964)

[III. Kết quả kiểm nghiệm 27](#_Toc198470965)

[1. Chọn ảnh đầu vào cho hệ thống 27](#_Toc198470966)

[2. Tính toán, xử ý và hiển thị kết quả 27](#_Toc198470967)

[PHẦN 5: TÀI LIỆU THAM KHẢO 31](#_Toc198470968)

# LỜI NÓI ĐẦU

Lời đầu tiên, chúng em xin gửi lời cảm ơn đến Học viện Công nghệ Bưu chính Viễn Thông đã tạo điều kiện cho chúng em được học môn Hệ cơ sở dữ liệu đa phương tiện. Đặc biệt, chúng em xin gửi lời cảm ơn chân thành và sâu sắc nhất tới thầy Nguyễn Đình Hóa, giảng viên bộ môn đã hướng dẫn và truyền đạt những kiến thức hết sức bổ ích và quý báu trong suốt thời gian học tập vừa qua. Với vốn hiểu biết sâu rộng và kinh nghiệm nhiều năm giảng dạy cũng như làm việc trong môi trường công nghệ thông tin, thầy khiến chúng em thật sự ấn tượng, khâm phục trước những hiểu biết của thầy.

Hệ cơ sở dữ liệu đa phương tiện là một môn học thật sự rất hay và bổ ích tuy nhiên đây là môn có khối lượng kiến thức tương đối nhiều và khó có thể hiểu rõ, hiểu sâu nhanh chóng khi thời lượng học trên lớp có hạn. Mặc dù thầy đã truyền đạt nhiệt tình và tận tâm nhưng do khả năng tư duy và khả năng tiếp thu chưa đủ tốt nên trong báo cáo bài tập lớn này chúng em khó có thể tránh khỏi được những sai sót. Do đó, chúng em kính mong thầy xem xét và bổ sung giúp đỡ chúng hoàn thiện bài báo cáo này một cách đầy đủ và đúng đắn hơn. Một lần nữa, chúng em xin chân thành cảm ơn thầy !

Chúng em xin chân thành cảm ơn thầy!

# YÊU CẦU

Xây dựng hệ CSDL lưu trữ và tìm kiếm ảnh đồ vật.

1. Hãy xây dựng/sưu tầm một bộ dữ liệu ảnh gồm ít nhất 100 files ảnh đồ vật văn phòng phẩm khác nhau, các ảnh có cùng kích thước, vật trong ảnh có cùng tỉ lệ khung hình, và cùng góc chụp đứng từ trên xuống (SV tùy chọn định dạng ảnh).

2. Hãy xây dựng một bộ thuộc tính để nhận diện ảnh đồ vật từ bộ dữ liệu đã thu thập. Hãy trình bày cụ thể về lý do lựa chọn và giá trị thông tin của các thuộc tính được sử dụng.

3. Xây dựng hệ thống tìm kiếm ảnh với đầu vào là một ảnh mới về một đồ vật nào đó đã có và không có trong dữ liệu, đầu ra là 3 ảnh giống nhất, xếp thứ tự giảm dần về độ tương đồng nội dung với ảnh đầu vào.

a. Trình bày sơ đồ khối của hệ thống và quy trình thực hiện yêu cầu của đề bài.

b. Trình bày quá trình trích rút, lưu trữ và sử dụng các thuộc tính để tìm kiếm ảnh trong hệ thống.

4. Demo hệ thống và đánh giá kết quả đã đạt được.

# PHẦN 1: XÂY DỰNG BỘ DỮ LIỆU

## Các nguồn dữ liệu

* Chủ đề: hình ảnh các vật dụng văn phòng phẩm thu thập từ một bộ dữ liệu đã gán nhãn rõ ràng.
* Mô tả: Bộ dữ liệu bao gồm 1.914 hình ảnh, mỗi ảnh tương ứng với một loại đồ vật như bút, kéo, thước, hồ dán, giấy note,...
* Thông tin về ảnh
* Chuyển đổi tất cả ảnh về cùng kích thước cố định 256x256 pixels.
* Ảnh chỉ chứa 1 đồ vật văn phòng duy nhất với phông nền trắng.
* Mỗi một loại đồ vật cụ thể sẽ được lưu vào 1 thư mục riêng với tên thư mục được đặt theo format “office\_equipment name”, các ảnh đồ vật trong thư mục thì được đăt tên theo quy tắc rõ ràng sau: “name\_[number].jpg” trong đó name là tên đồ vật, number là số thứ tự.

## Lưu trữ dữ liệu

* + 1914 ảnh về đồ vật văn phòng được lưu trữ trên máy tính local.
  + Việc lưu trữ tại máy tính có tác dụng dễ dàng truy cập, tăng tốc cho quá trình xử lý video.
  + Các ảnh của mỗi đồ vật được đặt trong một thư mục riêng và được đánh số thứ tự giúp quá trình xử lý dễ dàng hơn.
  + Sau khi xử lý bộ dữ liệu, ta thu được bộ dữ liệu đặc trưng ảnh và được lưu trữ trên máy tính.
  + Có thể lưu trữ ảnh trên cloud để thuân tiện cho việc trực tuyến, nhưng vì nhóm chọn lưu trên máy tính local để thuận tiện cho quá trình triển khai và tăng tốc độ xử lý ảnh khi đọc trực tiếp từ máy tính hơn so với dùng cloud.

Dưới đây là một số hình ảnh về ảnh lưu trữ trên máy tính của nhóm

Ảnh có chứa đồ đạc, ghế

Nội dung do AI tạo ra có thể không chính xác.

*Lưu trữ ảnh ghế trên máy tính*

|  |  |
| --- | --- |
| Ảnh có chứa văn bản, máy tính xách tay, Thiết bị đầu ra, máy tính  Nội dung do AI tạo ra có thể không chính xác. | Ảnh có chứa kéo, công cụ, Thiết bị y tế  Nội dung do AI tạo ra có thể không chính xác. |

*Lưu trữ ình ảnh về laptop và kéo*

## Miêu tả

* **Số lượng ảnh**: 1.914 ảnh
* **Kích thước ảnh**: Một số ảnh đã ở kích thước chuẩn, những ảnh khác sẽ được resize về cùng kích thước trước khi trích xuất.
* **Số lượng lớp (nhãn)**: 15 loại đồ vật
* **Định dạng ảnh**: JPEG (.jpg)
* **Trạng thái gán nhãn**: Đã được gán nhãn đầy đủ, mỗi ảnh thuộc về duy nhất một loại đồ vật.
* **Phân phối lớp**: Tương đối đồng đều giữa các loại đồ vật.
* **Nhận xét**
  + - Bộ dữ liệu tập trung vào 1 chủ đề (con mèo)
    - Các hình ảnh có đối tượng duy nhất là đồ vật với phông nền màu trắng.
    - Bộ dữ liệu khá đang dạng với 15 đồ vật, với mỗi đồ vật có nhiều kiểu khác nhau.
    - Góc chụp của một số đồ vật cũng được chọn để cố gắng đa dạng góc chụp, để tăng sự đang dạng cho hệ thống nhận diện ảnh

Ví dụ về sự đa dạng về góc chụp:

|  |  |  |
| --- | --- | --- |
| Ảnh có chứa đồ đạc, ghế, văn phòng, trong nhà  Nội dung do AI tạo ra có thể không chính xác. | Ảnh có chứa đồ đạc, ghế, Tay vịn, Ghế văn phòng  Nội dung do AI tạo ra có thể không chính xác. | Ảnh có chứa đồ đạc, ghế  Nội dung do AI tạo ra có thể không chính xác. |

Ví dụ về sự đa dạng về số đồ vật văn phòng:

|  |  |  |
| --- | --- | --- |
| Ảnh có chứa đồ đạc, ghế, văn phòng, trong nhà  Nội dung do AI tạo ra có thể không chính xác. | Ảnh có chứa đèn  Nội dung do AI tạo ra có thể không chính xác. | Ảnh có chứa đồ đạc, ngăn kéo, Tủ đựng tài liệu, tủ có ngăn kéo  Nội dung do AI tạo ra có thể không chính xác. |
| Ảnh có chứa đồ điện tử, sổ tay, máy tính, máy tính xách tay  Nội dung do AI tạo ra có thể không chính xác. | Ảnh có chứa thiết kế  Nội dung do AI tạo ra có thể không chính xác. | Ảnh có chứa văn bản, Hình chữ nhật  Nội dung do AI tạo ra có thể không chính xác. |
| Ảnh có chứa văn phòng phẩm, Dụng cụ viết, Bút bi, Công cụ đánh dấu  Nội dung do AI tạo ra có thể không chính xác. | Ảnh có chứa đồ điện tử, máy in  Nội dung do AI tạo ra có thể không chính xác. |  |

# PHẦN 2: MỘT SỐ KỸ THUẬT XỬ LÝ VÀ NHẬN DIỆN ẢNH

## I. Các kĩ thuật xử lý ảnh (trích rút đặc trưng ảnh):

### 1. Đặc trưng nội dung hình ảnh:

Trong hệ thống tìm kiếm ảnh dựa trên nội dung, việc trích xuất đặc trưng ảnh là bước quan trọng để mô tả nội dung hình ảnh một cách số hóa. Đặc trưng hình ảnh là các giá trị số đại diện cho các yếu tố như màu sắc, hình dạng, kết cấu… từ đó giúp hệ thống có thể so sánh và phân biệt các ảnh khác nhau.

Một số cách lấy đặc trưng hình ảnh như sau:

1. **HOG (Histogram of Oriented Gradients):**

* **Khái niệm:**  
  HOG (Histogram of Oriented Gradients) là một phương pháp trích xuất đặc trưng được sử dụng trong thị giác máy tính và xử lý hình ảnh, tập trung vào cấu trúc và hình dạng của các đối tượng. Tương tự như các phương pháp khác như biểu đồ định hướng biên, các mô tả đặc trưng không thay đổi theo tỉ lệ, ngữ cảnh hình dạng, HOG tính toán trên một lưới ô dày đặc và chuẩn hóa sự tương phản giữa các khối để tăng cường độ chính xác. HOG chủ yếu được sử dụng để môtả hình dạng và xuất hiện của các đối tượng trong ảnh.
* **Cách hoạt động:**
* Chuẩn hóa hình ảnh trước khi xử lý.
* Tính toán gradient theo cả hướng x và y.
* Lấy phiếu bầu cùng trọng số trong các ô.
* Chuẩn hóa các histogram trên các khối (blocks) gồm nhiều cell để đạt được tính bất biến với thay đổi ánh sáng.
* Thu thập tất cả các biểu đồ cường độ gradient hướng để tạo ra vectơ đặc trưng cuối cùng.
* **Ứng dụng:**  
  HOG rất mạnh trong nhận diện hình dạng, ví dụ nhận diện người, xe cộ, vật dụng. Trong bài toán đồ vật văn phòng phẩm, HOG giúp phân biệt rõ giữa những vật có hình dạng khác nhau (ví dụ: kéo – thước – bút).
* **Ưu điểm:**
* Bất biến với thay đổi ánh sáng, vị trí, độ chính xác cao.
* Mô tả được chi tiết về hình học (cấu trúc cạnh).
* **Nhược điểm:**
* Không hiệu quả với các đối tượng nhỏ.
* Có thể bị đánh lừa bởi các đối tượng giống nhau.
* Thời gian tính toán lâu.

1. **Đặc trưng ORB (Oriented FAST and Rotated BRIEF)**

* **Khái niệm:**

ORB (Oriented FAST and Rotated BRIEF) là một phương pháp trích xuất đặc trưng cục bộ trong ảnh số, dùng để phát hiện các điểm đặc trưng (keypoints) và mô tả chúng dưới dạng vector nhị phân (binary descriptor). ORB được phát triển vào năm 2011 bởi Ethan Rublee và cộng sự nhằm cung cấp một giải pháp hiệu quả, tốc độ cao và không bị ràng buộc bản quyền như các thuật toán kinh điển trước đó như SIFT hay SURF.

* Cái tên ORB xuất phát từ việc thuật toán này kết hợp:
* FAST (Features from Accelerated Segment Test): là thuật toán phát hiện các điểm đặc trưng (góc cạnh) trong ảnh một cách nhanh chóng.
* BRIEF (Binary Robust Independent Elementary Features): là kỹ thuật mô tả đặc trưng dựa trên so sánh cường độ giữa các cặp điểm trong vùng lân cận keypoint.
* **Cách hoạt động:**
* **Phát hiện keypoints (FAST):**  
  Xác định các điểm góc bằng cách kiểm tra 16 điểm lân cận quanh mỗi pixel.
* **Gán hướng cho keypoint:**  
  Tính hướng chính của vùng xung quanh mỗi keypoint bằng moment cường độ → giúp bất biến với xoay.
* **Mô tả bằng BRIEF:**  
  So sánh độ sáng các cặp điểm gần keypoint để tạo vector nhị phân.
* **Xoay BRIEF theo hướng:**  
  Điều chỉnh các cặp điểm so sánh theo hướng keypoint → đảm bảo mô tả không đổi khi ảnh bị xoay.
* **So khớp đặc trưng:**  
  So sánh hai vector bằng **khoảng cách Hamming** – số bit khác nhau giữa hai mô tả.
* **Ưu điểm:**
* Tốc độ rất nhanh: ORB được thiết kế hướng tới hiệu năng cao.
* Đặc trưng của ORB không thay đổi khi ảnh/quang cảnh xoay hoặc phóng to, thu nhỏ ở mức độ vừa phải
* Độ gọn nhẹ và hiệu quả lưu trữ:
* Hiệu quả trong nhiều bài toán thị giác:
* **Nhược điểm:**
* Hiệu suất giảm khi số lượng điểm quá nhiều
* Độ phân biệt hạn chế trên vùng lặp và ít kết cấu
* Kém ổn định dưới nhiễu hoặc biến dạng hình học
* Không tính được đặc trưng vùng lớn:

1. **Đặc trưng kết cấu "Texture" – GLCM (Gray Level Co-occurrence Matrix)**

* **Khái niệm:**

**GLCM** (Ma trận đồng xuất hiện mức xám) là một phương pháp **phân tích kết cấu (texture)** của ảnh dựa trên thống kê phân bố cường độ điểm ảnh. Phương pháp này xem xét **mối quan hệ không gian** giữa các cặp điểm ảnh có mức xám cụ thể trong ảnh, thay vì chỉ xét từng điểm đơn lẻ. GLCM được giới thiệu bởi Haralick và cộng sự vào những năm 1970 và nhanh chóng trở thành công cụ quan trọng để trích xuất đặc trưng kết cấu trong nhiều lĩnh vực như y học, nhận dạng hoa văn và viễn thám

* **Các bước thực hiện:**
* Chuyển ảnh về thang độ xám nếu là ảnh màu. Xác định số mức xám sẽ sử dụng trong ma trận
* Chọn tham số quan hệ (độ lệch):
* Khởi tạo ma trận GLCM:
* Duyệt ảnh và đếm cặp điểm ảnh:
* Chuẩn hóa các khối
* Trích xuất đặc trưng kết cấu từ ma trận
* **Ưu điểm:**
* Thông qua các thống kê như độ tương phản, năng lượng, v.v., GLCM cung cấp thông tin liên quan trực tiếp đến cảm nhận kết cấu của con người (như độ thô ráp, độ mịn, độ đều đặn của hoa văn)
* Được chứng minh và sử dụng rộng rãi
* **Nhược điểm:**
* GLCM chỉ xét cặp điểm ảnh trong một khoảng cách cố định, do đó chỉ ghi nhận kết cấu ở quy mô hẹp (cục bộ)
* Chỉ ghi lại đặc trưng bề mặt, thiếu thông tin kết cấu cao cấp:
* Hiệu quả của GLCM gắn liền với việc chọn đúng kích thước vùng, khoảng cách, số mức xám và hướng .

1. **Histogram màu**

* **Khái niệm:**

**Histogram màu** là biểu đồ thống kê mô tả **phân bố các màu sắc** trong một ảnh số. Về cơ bản, histogram màu đếm số lượng pixel thuộc mỗi khoảng màu nhất định trong không gian màu của ảnh. Ta có thể xây dựng histogram màu trên bất kỳ không gian màu nào; phổ biến nhất là không gian màu 3 chiều như **RGB** (đỏ-lục-xanh) hoặc **HSV** (Hue-Saturation-Value)

* **Các bước thực hiện:**
* Chọn không gian màu phù hợp cho bài toán (RGB, HSV,...)
* Khởi tạo histogram: Tạo mảng đếm số pixel cho mỗi bin
* Duyệt pixel và cập nhật histogram: Lần lượt xét từng pixel của ảnh, đọc giá trị màu của pixel đó
* Hoàn thiện và chuẩn hóa (tuỳ chọn): Sau khi duyệt xong, mảng đếm chính là histogram màu của ảnh.
* Trích xuất đặc trưng kết cấu
* **Ưu điểm:**
* Phản ánh trực quan nội dung màu của ảnh:
* Không phụ thuộc định dạng ảnh hoặc phép biến đổi đơn giản
* **Nhược điểm:**
* Mất thông tin vị trí và hình dạng: Như đã đề cập, histogram màu không chứa thông tin không gian, nên hai ảnh có cùng phân bố màu sẽ được xem như nhau dù nội dung và bố cục khác nhau
* Bị ảnh hưởng bởi điều kiện ánh sáng và màu sắc tổng thể
* Không phân tách đối tượng và nền:

## II. Một số kỹ thuật nhận diện, tính toán độ tương đồng:

### 1. Mục tiêu

Mục tiêu của phần này là trình bày các kỹ thuật được sử dụng để nhận diện và so sánh nội dung hình ảnh trong hệ thống truy xuất ảnh dựa trên nội dung

Cụ thể, khi người dùng cung cấp một ảnh đầu vào, hệ thống cần:

* Trích xuất đặc trưng (feature) từ ảnh đó và các ảnh trong cơ sở dữ liệu;
* Tính toán độ tương đồng giữa ảnh truy vấn và các ảnh đã lưu;
* Xếp hạng và truy xuất những ảnh có nội dung gần giống nhất.

### 2. Công thức tổng hợp độ tương đồng

Mỗi loại đặc trưng mang ý nghĩa khác nhau (màu sắc, hình dạng, điểm keypoint…), do đó có thể **kết hợp các mức độ tương đồng riêng lẻ thành một điểm số chung**:

* Trong đó:
* **Histogram**: Độ tương đồng về màu sắc (ví dụ: cosine similarity hoặc histogram intersection).
* **HOG**: Độ tương đồng về hình dạng (ví dụ: khoảng cách Euclidean giữa các vector HOG).
* **Keypoint Matches**: Số lượng keypoints (ORB/SIFT/SURF) trùng nhau sau khi so khớp.
* **GLCM**: Độ tương đồng về kết cấu ảnh (dựa trên các thông số Haralick từ GLCM).
* **​** là trọng số biểu thị mức độ ảnh hưởng của từng đặc trưng.

### 3. Cách xác định trọng số ω

#### a) Phương pháp thủ công (Heuristic)

* Gán trọng số dựa vào hiểu biết và kinh nghiệm.
* Ví dụ: nếu hệ thống tập trung vào phân biệt vật theo hình dạng thì tăng ω2; nếu chú trọng màu thì tăng ω1.

#### b) Học trọng số từ dữ liệu (Supervised Learning)

* Sử dụng mô hình như **Logistic Regression** hoặc **Linear Regression** để học ra bộ trọng số tối ưu từ dữ liệu gán nhãn (các cặp ảnh giống nhau hoặc khác nhau).

#### c) Tối ưu hóa bằng bài toán toán học (phương pháp nhóm sử dụng)

Thay vì học trọng số ***w***  từ mô hình học máy, ta xây dựng một bài toán tối ưu hóa để tìm bộ trọng số tốt nhất.

\* Ý tưởng chính:

- Đặt hàm mục tiêu là tối đa hóa khoảng cách giữa điểm số của cặp ảnh giống nhau và cặp ảnh khác nhau.- tức là tối đa hóa khoảng cách (cặp ảnh giống, cặp ảnh khác) **(\*)**

- Sử dụng một phương pháp tối ưu hóa: **Gradient Descent, Giải thuật di t ruyền (Genetic Algorithm),** hoặc  **Particle Swarm Optimization (PSO)** để tìm trọng số w.

**Bài toán tối ưu**

Gọi là độ tương đồng giữa ảnh va , được tính như sau:

Ta cần tìm sao cho:

Nghĩa là tổng điểm của các ảnh giống nhau thì phải cao hơn tổng điểm của các của các ảnh khác nhau.(TM **(\*)**)

**Cách giải bài toàn này**

**- Dùng Gragient Descent**

* Dễ thực hiện nếu hàm số có đạo hàm
* Cập nhật trọng số theo công thức:

**- Dùng Genetic Algorithm (GA)**

* Tìm nghiệm tối ưu bằng Giải Thuật Di Truyền.
* GA mô phỏng quá trình chọn lọc tự nhiên để tìm trọng số w.

#### d) Phân tích giảm chiều / học đặc trưng mới

* Áp dụng **PCA (Principal Component Analysis)** hoặc **LDA (Linear Discriminant Analysis)** để tìm tổ hợp trọng số đặc trưng phân biệt ảnh hiệu quả nhất.

#### e) Mạng neural học tương đồng (Deep Learning)

* Xây dựng mạng **Siamese** hoặc **triplet network** để học trực tiếp hàm tính độ tương đồng giữa ảnh.

# PHẦN 3: HỆ THỐNG NHẬN DẠNG VÀ TÌM KIẾM ẢNH TƯƠNG ĐỒNG

## I. Sơ đồ khối và quy trình thực hiện

### 1. Sơ đồ khối

**Ảnh có chứa văn bản, ảnh chụp màn hình, Phông chữ, hàng

Nội dung do AI tạo ra có thể không chính xác.**

*Hình ảnh sơ đồ khối*

### 2. Quy trình thực hiện

Sơ đồ khối gồm 3 gian đoạn chính, đặc trưng bởi 3 màu:

- Giai đoạn 1: Lưu trữ đặc trưng của bộ dữ liệu ảnh đồ vật văn phòng:

* Các ảnh đều là chỉ chứa duy nhất vật thể là đồ vật và nền là màu trắng. Các ảnh được đưa về cùng kích thước 256x256.
* Trích rút đặc trưng: toàn bộ tập ảnh đồ vật văn phòng được xử lý và trích xuất ra 4 đặc trưng về màu sắc, hình dạng, kết cấu, điểm đặc trưng.
* Ảnh sau khi được trích xuất 4 đặc trưng thu được 4 vector khác nhau, ghép 4 vector này thành 1 ta thu được vector đặc trưng cho ảnh. Vector đặc trưng ảnh sẽ được lưu vào bảng cơ sở dữ liệu với 1 số trường chính sau:
* Tên ảnh
* Đường dẫn
* Loại đồ vật
* Vector đặc trưng

- Giai đoạn 2: Tìm kiếm ảnh:

* Người dùng chọn 1 ảnh đồ vật để làm đầu vào cho hệ thống, ảnh này không có trong các ảnh mà hệ thống đã lưu trữ và xử lý.
* Tiền xử lý: đưa ảnh về cùng kích cỡ 256x256 và định dạng .jpg
* Trích rút đặc trưng của ảnh đầu vào:
* Ảnh đầu vào được xử lý để trích rút ra 4 đặc trưng về màu sắc, hình dạng, kết cấu, điểm đặc trưng. Sau đó ghép 4 vector của 4 đặc trưng thành một vector đại diện cho ảnh đầu vào.

- Giai đoạn 3: Tính toán độ tương đồng và đưa ra kết quả

* Sau khi đã có đươc vector đặc trưng của ảnh đầu vào, hệ thống sẽ lọc qua toàn bộ ảnh trong cơ sở sở dữ liệu của mình để tính toán độ tương đồng đôi một của ảnh đầu vào với mỗi ảnh. Kết quả sẽ trả về 5 ảnh có điểm tương đồng cao nhất với ảnh đầu vào.

## II. TRÍCH RÚT ĐẶC TRƯNG VÀ KỸ THUẬT TRÍCH RÚT

**Tổng quan:**

Giả sử ta có **một ảnh đồ vật** (ví dụ: ảnh một cái bút nằm trên nền trắng, kích thước 256×256 pixels, ảnh màu RGB). Mục tiêu là trích xuất một vector đặc trưng duy nhất từ ảnh đó để dùng trong nhận dạng và tìm kiếm. Vector này là sự **kết hợp tuyến tính (nối lại)** của các phần:

1. Histogram (Lược đồ xám toàn cục)
2. GLCM (Phân tích kết cấu thông qua ma trận đồng xuất mức xám)
3. HOG (Cấu trúc cạnh và hình dạng)
4. ORB (Các đặc trưng cục bộ nhị phân quanh điểm nổi bật)

### 1. Histogram – Phân bố mức xám toàn cục

**Quy trình lý thuyết**:

* Bước 1: Chuyển ảnh RGB thành ảnh xám bằng cách lấy giá trị trung bình hoặc theo công thức thị giác:
* Bước 2: Ảnh giờ là ma trận 256×256 chứa các giá trị từ 0 đến 255.
* Bước 3: Chia miền giá trị (0 - 255) thành các **khoảng (bins)** (ví dụ: 16 bins → mỗi bin rộng 16 giá trị).
* Bước 4: Đếm số pixel rơi vào từng bin → tạo vector histogram chuẩn hóa:

với là số pixel trong bin *i* , N là tổng số pixel. Vector đầu ra có 16 chiều.

* **Ý nghĩa**:
* Nếu ảnh có nhiều vùng sáng (như nền trắng), histogram sẽ lệch về phía mức xám cao.
* Nếu vật thể là màu tối (ví dụ cái bút màu đen), histogram sẽ có đỉnh ở vùng tối hơn.

### 2. GLCM – Mô tả kết cấu cục bộ

**Mục tiêu**: Phân tích mối quan hệ về mức xám giữa **các pixel lân cận**, từ đó phản ánh **kết cấu** của ảnh.

**Quy trình lý thuyết**:

* Bước 1: Ảnh xám từ trên được **làm tròn xuống** thành số nguyên trong khoảng [0, 255].
* Bước 2: Ta xây dựng một ma trận 256×256, trong đó:

theo hướng cố định (thường là ngang, dọc, chéo).

* Bước 3: Chuẩn hóa GLCM bằng cách chia cho tổng số cặp → biến GLCM thành một ma trận xác suất P(i,j).
* Bước 4: Tính các chỉ số thống kê:
* **Contrast**: phản ánh sự khác biệt giữa các mức xám lân cận.
* **Dissimilarity**: tương tự contrast nhưng tuyến tính hơn.
* **Homogeneity**: cao nếu ảnh có nhiều cặp lân cận giống nhau.
* **Energy / ASM**: cao nếu ảnh có họa tiết lặp lại mạnh.  
  → Đầu ra: một vector gồm 4–5 thành phần (các thống kê).

**Ví dụ thực tế**:

* Ảnh cái bút trên nền trắng có biên rõ ràng → sẽ tạo ra các cặp mức xám đậm (bút) và nhạt (nền) → contrast và dissimilarity cao, homogeneity thấp.

### 3. HOG – Biểu diễn cạnh và hình dạng

**Mục tiêu**: Trích xuất các **cạnh, biên, hướng và cấu trúc** trong ảnh bằng cách phân tích gradient.

**Quy trình lý thuyết**:

* Bước 1: Ảnh xám được tính **gradient theo x và y** tại mỗi pixel:
* Bước 2: Với mỗi pixel:
  + Biên độ gradient:
  + Hướng gradient:
* Bước 3: Chia ảnh thành các ô nhỏ (ví dụ 8×8 pixel). Trong mỗi ô:
  + Tạo histogram gồm 9 bin tương ứng với hướng gradient (mỗi bin ~20°).
  + Mỗi hướng được cộng dồn bởi biên độ gradient tại pixel tương ứng.
* Bước 4: Tất cả các histogram từ các ô được **nối lại thành một vector lớn** → chính là đặc trưng HOG.

**Ví dụ thực tế**:

* Cái bút có hình dạng dài → gradient mạnh ở biên theo chiều dọc hoặc chéo.
* HOG sẽ ghi nhận hướng chủ đạo dọc theo trục bút → đặc trưng này mang tính định hướng hình học rất cao.

### 4. ORB – Đặc trưng cục bộ nhị phân

**Mục tiêu**: Phát hiện và mô tả các điểm đặc trưng **có ý nghĩa hình học**, thường là điểm góc, cạnh mạnh – thông qua mô tả nhị phân nhanh gọn.

**Quy trình lý thuyết**:

* Bước 1: Duyệt toàn ảnh để tìm **các điểm nổi bật** (keypoints) – thường là các điểm góc, nơi gradient biến thiên mạnh.
* Bước 2: Gán cho mỗi keypoint một hướng cụ thể dựa trên vector gradient trung bình trong vùng lân cận.
* Bước 3: Lấy **patch hình vuông nhỏ** quanh mỗi keypoint, xoay patch theo hướng đã gán.
* Bước 4: Trong mỗi patch:
  + Chọn các cặp điểm cố định.
  + So sánh giá trị xám tại hai điểm mỗi cặp: nếu thì gán 1, ngược lại 0.
  + Mỗi cặp tạo một bit → tạo vector nhị phân dài n bit (thường 256 bits).
* Bước 5: Lặp lại với nhiều keypoint → tạo danh sách các vector nhị phân → tổng hợp lại (trung bình, nối, hoặc thống kê phân bố) → tạo thành vector ORB cuối cùng.

**Ví dụ thực tế**:

* Đầu bút, mép bút, chỗ in logo... thường là điểm keypoint.
* ORB sẽ mô tả cấu trúc chi tiết cục bộ ở những vùng này, cho phép phân biệt giữa các cây bút khác nhau dù có hình dạng tương tự.

### 5. Tổng hợp

Sau khi thu được:

* Histogram: 16 chiều
* GLCM: 4–5 chiều
* HOG: vài trăm đến hàng nghìn chiều (tùy kích thước ảnh và ô)
* ORB: nhị phân dài, thường rút gọn thành vector vài trăm đến 1000 chiều

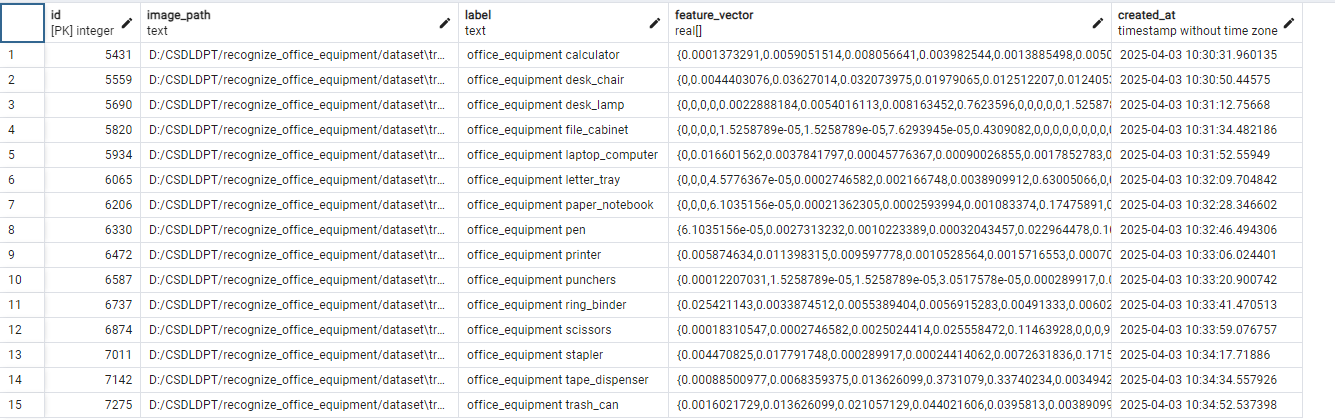
→ Ta lấy tất cả và **nối lại thành một vector đặc trưng duy nhất** đại diện cho **nội dung ảnh đầu vào**.

## III. Lưu trữ và sử dụng các đặc trưng để tìm kiếm ảnh trong hệ thống

### 1. Lưu trữ

- Dữ liệu bao gồm 1914 bản ghi được lưu trong bảng ***“image\_features”***  trong cơ sở dữ liệu đại diện cho 1914 ảnh của bộ dữ liệu, mỗi bản ghi bao gồm các trường sau:

* **image\_path**: đường dẫn tới ảnh trên máy tính, phục vụ việc hiển thị khi tìm kiếm
* **label**: tên của đồ vật (vd: computer, laptop, desk chair v.v…)
* **feature\_vector**: vector đặc trưng của ảnh



*Các hình ảnh được lưu với thông tin về vector đặc trưng trong cơ sở dữ liệu*

|  |
| --- |
| {0,0.0044403076,0.03627014,0.032073975,0.01979065,0.012512207,0.0124053955,0.0074768066,0.00355529,0.0048217773,0.006790161,0.0037841,0.0007324219,0.00010681152,3.05,  . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . .  0,0,0,0,0,0,0,0,238,233,137,46,240,233,172,144,44,179,34,121,130,164,148,187,233,173,184,143,158,27,16,25,172,250,72,11,135,19,123,17,50,76,219,243,141,18,94,123,144,72,131,143} |

*Vector đặc trưng của ảnh*

### 2. Tìm kiếm

* Các vector đặc trưng sau khi trích xuất được lưu trên PostgreSQL, phục vụ cho việc truy vấn đầy đủ và phân tích kết quả.
* Khi có một ảnh truy vấn thì với quy trình tương tự với việc tạo do bộ dữ liệu tập ảnh trong cơ sở dữ liệu ta sẽ thu được vector đặc trưng của ảnh đầu vào.
* Tiếp theo độ tương đồng giữa ảnh được tính bằng khoảng cách (Euclidean), kết hợp với trọng số đặc trưng đã được tối ưu qua *Giải Thuật Di Truyền*.

Định nghĩa

: vector đặc trưng của ảnh đầu vào

: vector đặc trưng của ảnh trong cơ sở dữ liệu

: vector trọng số

*có cùng kích thước.*

Khi đó, ta có công thức tính điểm (score) tương đồng như sau:

**Trong đó:**

* **x - y là phép trừ từng phần của 2 vector**
* là tích vô hướng, tức công thức ở trên tương đương với:
* Hệ thống trả về danh sách 5 ảnh tương đồng nhất cùng với điểm số tương đồng, cho phép người dùng đánh giá và sử dụng kết quả.

# PHẦN 4: CÀI ĐẶT CHƯƠNG TRÌNH VÀ DEMO

## I. Module trích xuất đặc trưng ảnh

### Các hàm trích xuất đặc trưng

**a. Hàm trích xuất đặc trưng Histogram màu**

A computer screen with colorful text

AI-generated content may be incorrect.

**b. Hàm trích xuất GLCM (Grey Level Co-occurrence Matrix)**

A screen shot of a computer code

AI-generated content may be incorrect.

**c. Hàm trích xuất HOG (Histogram of Oriented Gradients)**

A computer code on a black background

AI-generated content may be incorrect.

**d. Hàm trích xuất ORB (Oriented FAST and Rotated BRIEF)**

A computer code on a black background

AI-generated content may be incorrect.

**e. Hàm duyệt qua toàn bộ tập ảnh để trích xuất đặc trưng và lưu vào cơ sở dữ liệu**

**Ảnh có chứa văn bản, ảnh chụp màn hình, phần mềm

Nội dung do AI tạo ra có thể không chính xác.**

## II. Module tìm kiếm ảnh tương đồng với ảnh đầu vào

### 1. Hàm lấy ra bộ dữ liệu ảnh đã trích xuất của hệ thống

**A computer code on a black background

AI-generated content may be incorrect.**

### 2. Hàm lấy ra bộ trọng số đã tính được thông qua Giải Thuật Di Truyền:

**A screen shot of a computer code

AI-generated content may be incorrect.**

### 3. Hàm tính toán độ tương đồng

|  |
| --- |
| A screen shot of a computer program  AI-generated content may be incorrect. |
| A screen shot of a computer program  AI-generated content may be incorrect. |

## III. Kết quả kiểm nghiệm

* Yêu cầu:
* Input: Một ảnh đồ vật văn phòng
* Output: 5 ảnh có độ tương đồng cao nhất với ảnh đầu vào

### 1. Chọn ảnh đầu vào cho hệ thống

* Khi chạy chương trình hệ thông sẽ hiển thị 1 popup cho phép ta chọn một ảnh làm đầu vào cho quá trình truy vấn, tìm kiếm.

A screenshot of a computer screen

AI-generated content may be incorrect.

→ ta chọn một ảnh

### 2. Tính toán, xử ý và hiển thị kết quả

* Hệ thống sẽ tiến hành tính toán xử lý và hiển thị 5 ảnh có độ tương đồng cao nhất, lần lượt theo thứ tự từ độ tương đồng cao đến thấp nhất.

VD 1: chọn ảnh đầu vào là một cái ghế

|  |  |
| --- | --- |
| Ảnh có chứa đồ đạc, ghế  Nội dung do AI tạo ra có thể không chính xác.  Ảnh được chọn để tìm kiếm | Ảnh có chứa đồ đạc, ghế  Nội dung do AI tạo ra có thể không chính xác.  Ảnh giống nhất |
| Ảnh có chứa đồ đạc, ghế đẩu  Nội dung do AI tạo ra có thể không chính xác.  Ảnh giống thứ 2 | Ảnh có chứa đồ đạc, ghế  Nội dung do AI tạo ra có thể không chính xác.  Ảnh giống thứ 3 |
| Ảnh có chứa đồ đạc, ghế  Nội dung do AI tạo ra có thể không chính xác.  Ảnh giống thứ 4 | Ảnh có chứa đồ đạc, ghế  Nội dung do AI tạo ra có thể không chính xác.  Ảnh giống thứ 5 |

VD 2: chọn ảnh đầu vào là một cái kéo

|  |  |
| --- | --- |
| Ảnh có chứa công cụ, kéo  Nội dung do AI tạo ra có thể không chính xác.  Ảnh được chọn | Ảnh có chứa công cụ, kéo  Nội dung do AI tạo ra có thể không chính xác.  Ảnh giống nhất |
| Ảnh có chứa công cụ, kéo, văn bản  Nội dung do AI tạo ra có thể không chính xác.  Ảnh giống thứ 2 | Ảnh có chứa kéo, công cụ, Dụng cụ văn phòng  Nội dung do AI tạo ra có thể không chính xác.  Ảnh giống thứ 3 |
| Ảnh có chứa văn bản, văn phòng phẩm, bút máy, công cụ  Nội dung do AI tạo ra có thể không chính xác.  Ảnh giống thứ 4 | Ảnh có chứa văn bản, thiết kế  Nội dung do AI tạo ra có thể không chính xác.  Ảnh giống thứ 5 |

# PHẦN 5: TÀI LIỆU THAM KHẢO

|  |  |
| --- | --- |
| [1] | Gonzalez, R. C., & Woods, R. E. (2018). *Digital Image Processing* (4th Edition). Pearson. |
| [2] | Haralick, R. M., Shanmugam, K., & Dinstein, I. (1973). Textural features for image classification. *IEEE Transactions on Systems, Man, and Cybernetics*, **SMC-3**(6), 610–621. |
| [3] | Dalal, N., & Triggs, B. (2005). Histograms of Oriented Gradients for Human Detection. *IEEE Conference on Computer Vision and Pattern Recognition (CVPR).* |
| [4] | Rublee, E., Rabaud, V., Konolige, K., & Bradski, G. (2011). ORB: An efficient alternative to SIFT or SURF. *Proceedings of the IEEE International Conference on Computer Vision (ICCV).* |
| [5] | Szeliski, R. (2010). *Computer Vision: Algorithms and Applications*. Springer. |
| [6] | Ojala, T., Pietikäinen, M., & Harwood, D. (1996). A comparative study of texture measures with classification based on featured distributions. *Pattern Recognition.* |
| [7] | OpenCV.org – Image Descriptors Overview. Retrieved from <https://docs.opencv.org/> |