BÁO CÁO GIỮA KỲ MÔN XỬ LÝ ẢNH

Đề tài: Tìm kiếm các vật thể trong ảnh mẫu

Họ và Tên: Bùi Ngọc Khánh
Mã sinh viên: 22022551

I. Mục tiêu và hướng tiếp cận

Mục tiêu:
Bài toán nhằm phát hiện và định vị chính xác vị trí của các đối tượng được chỉ định trong ảnh đầu vào.

Hướng tiếp cận:
Để giải quyết bài toán, tôi đã áp dụng kỹ thuật **template matching** (khớp mẫu), một phương pháp cơ bản trong xử lý ảnh nhằm tìm kiếm các vùng trong ảnh gốc có độ tương đồng cao với ảnh mẫu. Phương pháp này hoạt động bằng cách trượt ảnh mẫu qua ảnh gốc và tính toán độ tương đồng tại mỗi vị trí.

Tuy nhiên, các đối tượng trong ảnh gốc có thể xuất hiện ở nhiều kích thước khác nhau. Vì vậy, tôi đã thay đổi kích thước các đối tượng theo nhiều nhiều tỷ lệ khác nhau trước khi thực hiện khớp mẫu, giúp phát hiện các đối tượng ở các kích thước đa dạng.

II. Các bước thực hiện và giải thích kỹ thuật

1. Chuẩn bị dữ liệu

- Đọc ảnh gốc từ file full.jpg bằng hàm cv2.imread.
- Định nghĩa danh sách các đối tượng cần tìm (items) và tạo danh sách đường dẫn đến các ảnh mẫu tương ứng (ví dụ: data/balloon.jpg, data/bear.jpg, v.v.) thông qua list comprehension:

2. Thiết lập tham số

- Tạo mảng scales chứa 25 tỷ lệ từ 0.4 đến 1.2 bằng np.linspace(0.4, 1.2, 25). Điều này cho phép thay đổi kích thước ảnh mẫu để phát hiện đối tượng ở các kích thước khác nhau.
- Đặt ngưỡng threshold = 0.7 để lọc các kết quả khớp mẫu, chỉ giữ lại các vùng có độ tương đồng lớn hơn hoặc bằng 0.7 (giá trị nằm trong khoảng [0, 1], với 1 là khớp hoàn hảo).
- Đặt mask_value = 240, có thể dùng để xử lý nền hoặc các vùng không quan tâm của đối tượng.

3. Áp dụng template matching với multi-scale

Xây dựng hàm find_items để thực hiện quá trình phát hiện đối tượng:

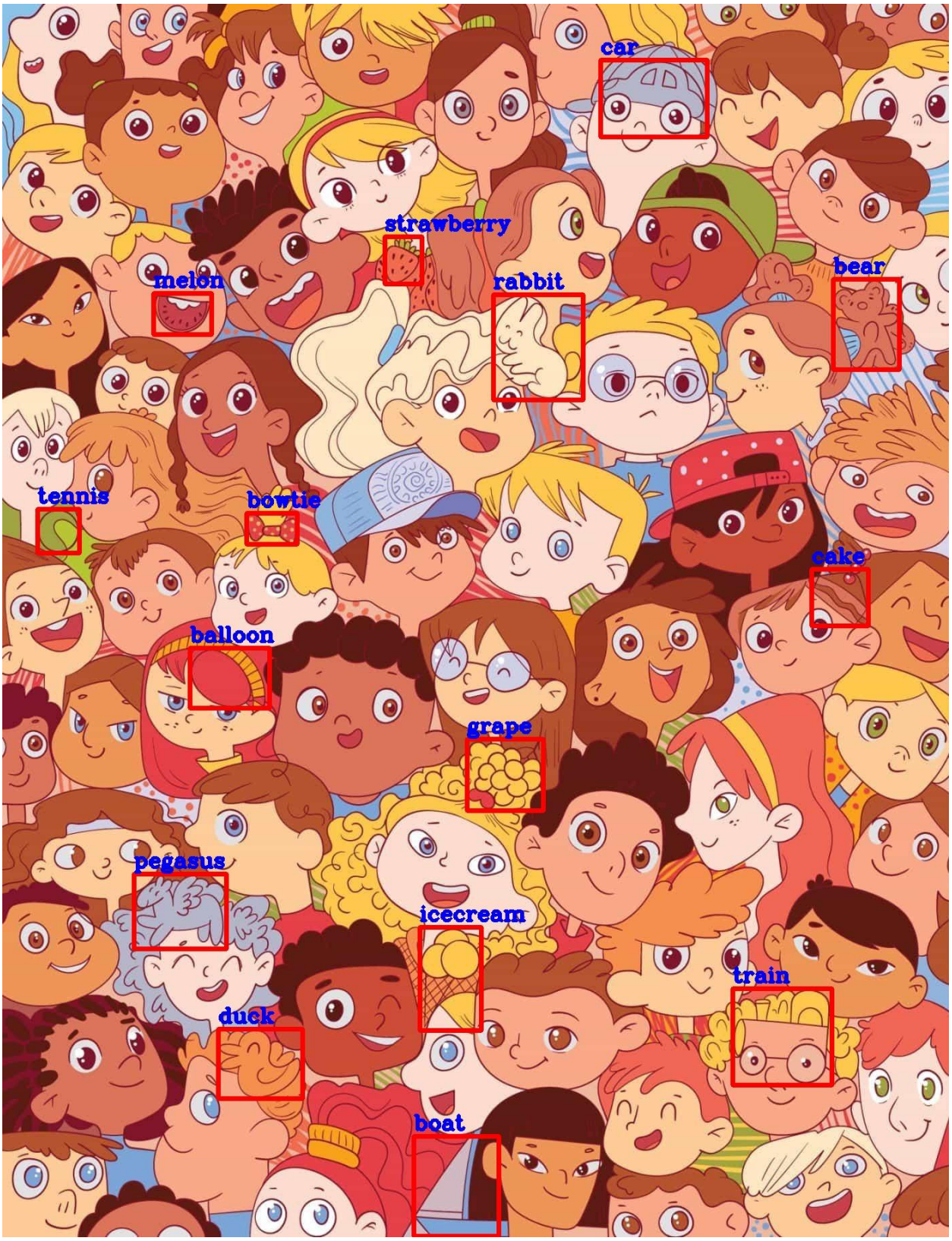
- Đọc từng ảnh mẫu từ path_items.
- Thay đổi kích thước ảnh mẫu theo các tỷ lệ trong scales.
- Sử dụng hàm cv2.matchTemplate để tính toán độ tương đồng giữa ảnh mẫu và ảnh gốc ở mỗi tỷ lệ. Phương pháp cụ thể có thể là TM_CCOEFF_NORMED (độ tương đồng chuẩn hóa).
- Lọc các vị trí có độ tương đồng vượt ngưỡng threshold.
- Tìm ra và lấy vị trí có độ tương đồng cao nhất.
- Trả về danh sách các bounding box (tọa độ và kích thước) của các đối tượng được phát hiện cùng tên đối tượng tương ứng.

4. Hiển thị và lưu kết quả

- Vẽ các bounding box lên ảnh gốc tại các vị trí phát hiện được (giả định thực hiện trong find_items).
- Lưu ảnh kết quả vào file output.jpg.
- Trả về danh sách các đối tượng được tìm thấy dưới dạng các tuple, mỗi tuple chứa tọa độ (x, y, w, h) và tên đối tượng.

III. Kết quả

Kết quả phát hiện đối tượng:
Phương pháp template matching với multi-scale đã thành công trong việc định vị tất cả 15 đối tượng trong ảnh.


car
strawberry
melon
rabbit
bear
tennis
bowtie
cake
balloon
grape
pegasus
icecream
train
duck
boat

IV. Các cải tiến đã thực hiện

Vấn đề tiềm ẩn	Cải tiến
Đối tượng có kích thước khác nhau	Sử dụng multi-scale template matching với 25 tỷ lệ từ 0.4 đến 1.2 để phát hiện đối tượng ở các kích thước khác nhau.
Độ tương đồng không đủ cao	Đặt ngưỡng threshold = 0.7 để chỉ giữ lại các kết quả có độ tương đồng cao.
Nền phức tạp hoặc nhiễu	Sử dụng mask_value = 240 (giả định) để xử lý nền hoặc các vùng không quan tâm.
Hiệu suất với nhiều đối tượng	Tối ưu hóa bằng cách sử dụng NumPy và OpenCV cho các phép toán mảng và hình ảnh.

V. Kết luận

Phương pháp template matching với multi-scale đã được áp dụng hiệu quả để phát hiện và định vị 15 đối tượng trong ảnh full.jpg. Kỹ thuật này cho phép tìm kiếm các đối tượng ở các kích thước khác nhau bằng cách thay đổi tỷ lệ ảnh mẫu. Kết quả cho thấy tất cả các đối tượng trong danh sách đã được định vị chính xác với tọa độ và kích thước bounding box tương ứng.

Ưu điểm của phương pháp này là sự đơn giản và không cần huấn luyện mô hình. Tuy nhiên, nó có thể kém hiệu quả trong các trường hợp phức tạp hơn (ví dụ: đối tượng bị xoay, che khuất, hoặc thay đổi ánh sáng). Để cải thiện, có thể xem xét tích hợp các phương pháp học sâu như YOLO hoặc SSD trong các nghiên cứu tiếp theo.

Báo cáo này đã trình bày đầy đủ quá trình thực hiện, kết quả đạt được và các gợi ý cải tiến, đáp ứng yêu cầu của bài toán xử lý ảnh.

Link Github: link