

BÁO CÁO

Đếm số lượng mẫu trong hình

Quy trình

1. **Tiền xử lý:**
 - Chuyển sang ảnh xám
 - Tạo Binary Mask
2. **Chia kênh RGB:** Để xử lý hiệu quả hơn, ảnh và mẫu được tách thành ba kênh màu R, G, B. Khớp mẫu được thực hiện riêng biệt cho từng kênh màu.
3. **Matching:** Sử dụng `cv2.matchTemplate` với `cv2.TM_SQDIFF_NORMED` để tính toán mức độ tương đồng trên từng kênh màu (R,G,B). Sau đó, kết quả được kết hợp lại thành các bounding box bằng cách tính trung bình.
4. **Loại bỏ các kết quả trùng lặp:** Tính toán IoU (Intersection over Union) để gộp các bounding box lặp nhau.
5. **Trực quan hóa kết quả**

Kết quả

Đếm được số lượng mẫu trong ảnh, tuy nhiên còn nhiều hạn chế.

Hạn chế

Các mẫu từ vật thể có sự biến đổi khác nhau hình dáng, màu sắc và kích thước. Ví dụ trong bài đếm số thỏ, khi chọn mẫu là tai thỏ, có sự biến đổi khác nhau giữa các con thỏ trong hình: kích thước và hướng dẫn đến việc tìm kiếm sẽ gặp khó khăn.

Tìm mẫu trong hình

Quy trình xử lý ảnh:

1. **Chuyển đổi ảnh sang ảnh xám:** Cả hình ảnh đầu vào và mẫu đều được chuyển sang thang độ xám để đơn giản hóa xử lý và giảm tải tính toán. Thông tin màu không cần thiết cho khớp mẫu, vì vậy bước này giúp tập trung vào các cấu trúc hình ảnh.
2. **Tạo binary mask:** Sử dụng ngưỡng là 240. **Binary mask** làm nổi bật các khu vực không phải là nền trong mẫu, giúp thuật toán bỏ qua các vùng nền trong quá trình khớp mẫu.

3. **Thay đổi kích thước mẫu:** Mẫu được thay đổi kích thước theo tỷ lệ trong khoảng **scale_min=0.5** đến **scale_max=1.0**, với **scale_steps=50** nhằm đảm bảo có thể phát hiện mẫu ở các kích thước khác.
4. **Match:** Sử dụng `cv2.matchTemplate` của OpenCV với phương pháp `TM_SQDIFF_NORMED`
5. **Trực quan hóa kết quả**

Kết quả

Thuật toán phát hiện các đối tượng trong hình ảnh.