

BÁO CÁO GIẢI THÍCH

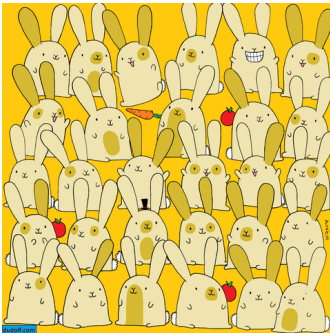
Finding & Counting Objects

Nguyễn Văn Thành - 19020627 - K64K2
Bài kiểm tra giữa kỳ môn Xử lý ảnh INT 3404_1

1. PHƯƠNG PHÁP:

- Để sử dụng ta cần 3 tham số: ảnh cần tìm đối tượng, mẫu và phương pháp tìm. Template matching có tất cả 6 phương pháp để tìm, và em sẽ dùng `cv2.TM_CCOEFF_NORMED`.

- **Ảnh gốc:**



Ở đây em sẽ tìm những chú thỏ có template như bên dưới và đếm chúng

- **Template:**



Template matching cho phép phát hiện vật thể trong ảnh đầu vào bằng cách sử dụng một ảnh tham chiếu chứa vật thể cần phát hiện. Cách thuật toán hoạt động:

- Trượt ảnh template trên ảnh đầu vào như tích chập 2D
- Ảnh template sẽ được so sánh với cửa sổ trượt tương ứng trên ảnh đầu vào bằng một công thức. Dưới đây em sẽ liệt kê 1 số phương pháp so sánh:

cv2.TM_SQDIFF	$R(x, y) = \sum_{x', y'} (T(x', y') - I(x + x', y + y'))^2$
cv2.TM_SQDIFF_NORMED	$R(x, y) = \frac{\sum_{x', y'} (T(x', y') - I(x + x', y + y'))^2}{\sqrt{\sum_{x', y'} T(x', y')^2 \cdot \sum_{x', y'} I(x + x', y + y')^2}}$
cv2.TM_CCORR	$R(x, y) = \sum_{x', y'} (T(x', y') \cdot I(x + x', y + y'))$
cv2.TM_CCORR_NORMED	$R(x, y) = \frac{\sum_{x', y'} (T(x', y') \cdot I(x + x', y + y'))}{\sqrt{\sum_{x', y'} T(x', y')^2 \cdot \sum_{x', y'} I(x + x', y + y')^2}}$
cv2.TM_CCOEFF	$R(x, y) = \sum_{x', y'} (T'(x', y') \cdot I'(x + x', y + y'))$ <p><i>where,</i></p> $T'(x', y') = T(x', y') - 1/(w \cdot h) \cdot \sum_{x'', y''} T(x'', y'')$ $I'(x + x', y + y') = I(x + x', y + y') - 1/(w \cdot h) \cdot \sum_{x'', y''} I(x + x'', y + y'')$
cv2.TM_CCOEFF_NORMED	$R(x, y) = \frac{\sum_{x', y'} (T'(x', y') \cdot I'(x + x', y + y'))}{\sqrt{\sum_{x', y'} T'(x', y')^2 \cdot \sum_{x', y'} I'(x + x', y + y')^2}}$

Khi ta trượt ảnh template trên ảnh đầu vào, một metric sẽ được tính toán tại mỗi pixel của ma trận đầu ra RR. Ma trận RR sẽ có kích thước nhỏ hơn ảnh đầu vào, giống với kết quả khi thực hiện tích chập. Giá trị mỗi pixel của RR biểu diễn mức độ trùng khớp của ảnh template với vị trí tương ứng trên ảnh đầu vào. Từ đó ta có thể chọn ra pixel có giá trị cao nhất (Nếu muốn detect 1 vật thể) hoặc lấy threshold RR (nếu detect nhiều vật thể)

Ưu điểm:

- Nhanh đơn giản
- Không tốn công làm data

Nhược điểm:

- Template phải rất giống với vật thể trong ảnh cả về kích thước độ nghiêng. Nếu khác biệt quá lớn sẽ không phát hiện được.

2. GIẢI THÍCH CODE TRONG OPENCV:

Opencv cung cấp hàm `cv2.matchTemplate()` cho thuật toán template matching. Hàm này nhận vào 3 tham số chính

- Ảnh đầu vào chứa vật thể cần detect
- Ảnh template
- Method là cách tính ma trận output RR

Em sẽ sử dụng phần mềm Colaboratory của Google. Phần mềm này không hỗ trợ lưu ảnh sau khi import nên em sẽ gửi kèm ảnh ạ.

Đầu tiên import các thư viện và đọc ảnh:

```
import cv2
import numpy as np
import matplotlib.pyplot as plt

# Read the main image
img_rgb = cv2.imread(' /rabbit2.png')

# Convert it to grayscale
img_gray = cv2.cvtColor(img_rgb, cv2.COLOR_BGR2GRAY)

# Read the template
template = cv2.imread(' /template.PNG',0)
```

```
# Store width and height of template in w and h
```

```
w, h = template.shape[::-1]
```

Sau đó ta gọi hàm `cv2.matchTemplate()` với `img_gray` và `template`

```
# Perform match operations.
```

```
res = cv2.matchTemplate(img_gray,template,cv2.TM_CCOEFF_NORMED)
```

Ta lấy threshold để lọc ra những vị trí, threshold có thể tùy ý, dùng hàm len để đếm object giống nhau

```
# Specify a threshold
```

```
threshold = 0.8
```

```
# Store the coordinates of matched area in a numpy array
```

```
loc = np.where( res >= threshold)
```

```
# Draw a rectangle around the matched region.
```

```
for pt in zip(*loc[::-1]):
```

```
    cv2.rectangle(img_rgb, pt, (pt[0] + w, pt[1] + h), (0,255,255), 2)
```

```
sensitivity = 100
```

```
f.add((round(pt[0]/sensitivity), round(pt[1]/sensitivity)))
```

```
# Show the final image with the matched area.
```

```
# cv2.imshow('Detected',img_rgb)
```

```
imgplot = plt.imshow(img_rgb)
```

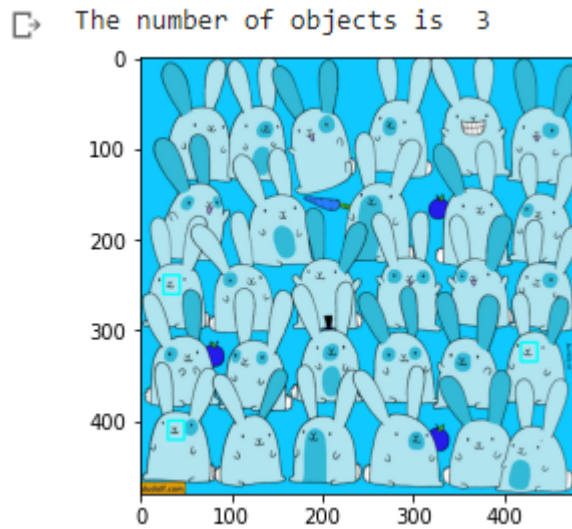
Chúng ta đếm object

```
#Counting Objects
```

```
loc = np.where( res >= threshold)
```

```
print('The number of objects is', len(loc[0]))
```

Kết quả:



Số lượng object trùng khớp (miệng thỏ) là 3

Khung chữ nhật màu xanh dương hiển thị vị trí object

Nếu chỉnh threshold nhỏ hơn sẽ cho nhiều kết quả hơn, tuy nhiên nếu quá nhỏ sẽ bắt đầu có sự sai lệch giữa kết quả trên máy và nhìn bằng mắt thường.