**ĐẠI HỌC QUỐC GIA TP. HỒ CHÍ MINH**

**TRƯỜNG ĐẠI HỌC CÔNG NGHỆ THÔNG TIN**

**KHOA KHOA HỌC MÁY TÍNH**

Logo, company name

Description automatically generated

**BÁO CÁO ĐỒ ÁN MÔN HỌC**

**NHẬP MÔN THỊ GIÁC MÁY TÍNH**

**CS231.M22.KHCL**

|  |  |
| --- | --- |
| **SINH VIÊN THỰC HIỆN:** | **PHẠM TRẦN ANH TIÊN - 20522012**  **LƯƠNG LÝ CÔNG THỊNH - 20521960** |

**TP. HỒ CHÍ MINH, 6/2022**

Contents

[**1.Giới thiệu 3**](#_Toc106896422)

[**2.Các bước thực hiện 3**](#_Toc106896423)

[**2.1. Preprocessing image 4**](#_Toc106896424)

[**2.1.1. Chuyển ảnh về mức xám và loại bỏ nhiễu. 5**](#_Toc106896425)

[**2.1.2. Sử dụng các hàm và filter để phát hiện biên cạnh của ảnh. 6**](#_Toc106896426)

[**2.1.3. Morphology (phép biến đổi hình thái) 6**](#_Toc106896427)

[**2.2. Object Detection 7**](#_Toc106896428)

[**2.3. Object Measurement 8**](#_Toc106896429)

[**2.4. Distant Measurement 9**](#_Toc106896430)

[**3.Thực nghiệm 10**](#_Toc106896431)

[**4.Kết luận 14**](#_Toc106896432)

[**5.Tham khảo 15**](#_Toc106896433)

**OBJECT DETECTION AND MEASUREMENT OF ITS SIZE**

# Giới thiệu

Ngày nay, phát hiện đối tượng và xác định kích thước của các đối tượng là một vấn đề quan trọng trong nhiều lĩnh vực công nghiệp. Đặc biệt là được thực hiện trong thời gian thực.

# Các bước thực hiện

Hệ thống bao gồm hai phần là phát hiện đối tượng và đo đối tượng. Trong phần đầu tiên, camera điện thoại được sử dụng để chụp được các khung hình. Trong phần thứ hai, các mô-đun thị giác máy tính sẽ được áp dụng cho các khung hình đã chụp để xác định các đối tượng.Sau đó, đối tượng được phát hiện trong khung hình hiện tại ngay lập tức sẽ được xử lý để trích xuất kích thước của các đối tượng.

Shape

Description automatically generated with low confidenceTuy nhiên trước tiên, chúng ta cần xử lý trước hình ảnh của mình. Máy ảnh sẽ chụp một khung hình và khung hình sẽ chuyển sang mức xám để tăng tốc độ xử lí và độ chính xác. Các đối tượng được phát hiện thông qua thuật toán dò tìm cạnh như canny, sobel, laplacian,… Nó được sử dụng để phát hiện một đối tượng hoặc nhiều đối tượng. Với sự trợ giúp của thuật toán dò tìm cạnh, hình ảnh xám sẽ được xử lý. Sau đó, thực hiện thuật toán dilation và erosion để giữa các cạnh trong khung cạnh

Hình 2.1 Sơ đồ thuật toán

A picture containing text, businesscard

Description automatically generated

Hình 2.2 Ảnh gốc

## Preprocessing image

### Calendar Description automatically generatedChuyển ảnh về mức xám và loại bỏ nhiễu.

Hình 2.3 Ảnh đã được chuyển mức xám

A picture containing text

Description automatically generated

Hình 2.4 Ảnh đã được lọc nhiễu

### Text Description automatically generated with medium confidenceSử dụng các hàm và filter để phát hiện biên cạnh của ảnh.

Hình 2.5 Ảnh biên cạnh các filter khác nhau

### Morphology (phép biến đổi hình thái)

Để có được kết quả tốt hơn và phát hiện đối tượng chính xác hơn, quy trình phát hiện cạnh đã được cải tiến với một số phép biến đổi hình thái. Các phép này thường là sự kết hợp của các các thủ tục được thực hiện tương đối trên việc chuẩn bị các pixel mà không thay đổi các giá trị số của chúng. Erosion và Dilation là các thủ tục cho các phép toán hình thái học.

Phép biến đổi hình thái là sự kết hợp của erosion và dilation. Openinglà quá trình thực hiện “erosion” trước và “dilation” sau đó để loại bỏ nhiễu bên trong của hình ảnh thu được. Closing là quá trình thực hiện ngược lại với việc thực hiện “dilation” trước và sau đó là “erosion” để làm mịn contours và hợp nhất các chỗ biên cạnh của đối tượng bị đứt không liền mạch.

Calendar

Description automatically generated with low confidence

Hình 2.6 Ảnh biên cạnh sử dụng Canny đã qua phép biến đổi hình thái

## Object Detection

A picture containing text, person

Description automatically generatedSau khi phát hiện cạnh và phép đóng bất kỳ khoảng trống nào giữa các cạnh, chúng tôi phát hiện các contours bằng cách sử dụng hàm OpenCV là cv2.findContours để tìm hình dạng của các vật thể. Chúng ta sắp xếp các contours từ trái sang phải. Vật thể tham chiếu trong khung hình là vật thể bên trái nhất. Bằng cách tùy thuộc vào chọn vật thể tham chiếu mà chúng ta điều chỉnh máy ảnh và cài đặt giá trị của tham số.

Hình 2.7 Ảnh contour

## Object Measurement

Để có thể đo được kích thước vật thể trong ảnh, ta sẽ cần có một đại lượng gọi là pixels per metric:

* Biết trước kích thước vật thể làm gốc
* Ví dụ pixels per metric = 400 / 23.5 = 17.02

Tính pixels per metric dựa trên vật thể làm gốc

Tính kích thước các vật thể dựa trên pixels per metric và qua tọa độ contour

Graphical user interface, application

Description automatically generated

Hình 2.8 Ảnh kết quả kích thước các vật

## Distant Measurement

Từ tọa độ tâm bounding box của các vật thể

A picture containing diagram

Description automatically generatedTính khoảng cách giữa các tâm thông qua euclid

Hình 2.9 Hình kết quả khoảng cách giữa các vật

# Thực nghiệm

Nhóm thực nghiệm đo các đối tượng trong hình ảnh và thời gian thực. Nhóm đã chuẩn bị một vài chuẩn bị trước thử nghiệm để kiểm tra tính đúng đắn của phương pháp được đề xuất. Chương trình được viết bằng ngôn ngữ Python.

Kết quả được đánh giá giá dựa trên công thức tính độ chính xác như sau:

𝐴𝑐𝑐𝑢𝑟𝑎𝑐𝑦 = 100% − 𝐸𝑟𝑟𝑜𝑟 𝑅𝑎𝑡𝑒

* Canny:

Table

Description automatically generatedA picture containing diagram

Description automatically generatedBảng 3.1 Kết quả đánh giá với thuật toán Canny

Hình 3.1 Ảnh kết quả với thuật toán Canny

* Sobel

Table

Description automatically generatedBảng 3.2 đánh giá kết quả sử dụng bộ lọc Sobel

A picture containing diagram

Description automatically generated

Hình 3.2 Ảnh kết quả sử dụng bộ lọc Sobel

* Scharr

A picture containing diagram

Description automatically generatedTable

Description automatically generatedBảng 3.3 Kết quả đánh giá sử dụng bộ lọc Scharr

Hình 3.3 Ảnh kết quả sử dụng bộ lọc Scharr

* Laplacian

Table

Description automatically generatedBảng 3.4 Kết quả đánh giá sử dụng bộ lọc Laplacian

A picture containing application

Description automatically generated

Hình 3.4 Ảnh kết quả sử dụng bộ lọc Laplacian

# Kết luận

Hoạt động tốt trong các môi trường có

* Điều kiện ánh sáng tốt
* Góc chụp tốt
* Ảnh rõ nét
* Vật và nền có độ tưởng phản tốt

Hoạt động kém trong các điều kiện sau:

* A screenshot of a computer

  Description automatically generated with medium confidenceẢnh chụp đồ vật không có độ tương phản tốt với nền

Hình 4.1 Ảnh kết quả vật có độ tương phản kém với nền

* Góc chụp bị lệch:

A picture containing text, several

Description automatically generated

Hình 4.2 Ảnh kết quả góc chụp bị lệch

* Ảnh chụp bị bóp méo (distortion): Đối với các ảnh bị bóp méo, dẫn đến độ vật cũng bị bóp méo nên việc tính toán kích thước sẽ sai theo

A picture containing water, outdoor, building, city

Description automatically generated

Hình 4.3 Ví dụ về ảnh distortion

# Tham khảo

[1] Adrian An Rosebrock, March 28, 2016, Measuring size of objects in an image with OpenCV, pyimagesearch. https://pyimagesearch.com/2016/03/28/measuring-size-of-objects-in-an-image-with-opencv/

[2] T. Dhikhi, Allagada Naga Suhas, Gosula Ramakanth Reddy, Kanadam Chandu Vardhan, In Measuring Size of an Object using Computer Vision, ISSN: 2278-3075

[3] Shweta Pardeshi, Pranali Pawar, Nikhil, In Real Time Object Measurement, ISSN 2321-3361

[4] Adrian An Rosebrock, April 4, 2016, Measuring distance between objects in an image with OpenCV, pyimagesearch. https://pyimagesearch.com/2016/04/04/measuring-distance-between-objects-in-an-image-with-opencv/

[5] Nashwan Adnan Othman, Mehmet Umut Salur, Mehmet Karakose, Ilhan Aydin, In An Embedded Real-Time Object Detection and Measurement of its Size. 2018 International Conference on Artificial Intelligence and Data Processing (IDAP). doi:10.1109/idap.2018.8620812

[6] OpenCV, Canny Edge Detection, https://docs.opencv.org/4.x/da/d22/tutorial\_py\_canny.html

[7] OpenCV, Image Gradients https://docs.opencv.org/4.x/d5/d0f/tutorial\_py\_gradients.html