

TRƯỜNG ĐẠI HỌC SƯ PHẠM TP.HCM
KHOA CÔNG NGHỆ THÔNG TIN



BÀI THUYẾT MINH

NHẬN DẠNG BARCODE

Sinh viên thực hiện : Lê Tấn Lộc, Huỳnh Thanh Phong

MSSV : 4501104135, 4501104172

Nhóm : DarkSouls

Giáo viên hướng dẫn : TS. Ngô Quốc Việt

1 Mục tiêu.

- Xác định được vị trí mã vạch qua hình ảnh và camera:
 - + Hình ảnh: xác định được vị trí mã vạch khi ảnh xoay chưa đúng vị trí hay trong vùng có nhiều chữ.
 - + Camera: xác định được vị trí mã vạch trên không gian thực (ghi hình trực tiếp).
- Xây dựng một ứng dụng nhận dạng mã vạch.

2 Phương pháp tiếp cận.

2.1 Đối với hình ảnh

Thực hiện thông qua ba bước xử lý:

2.1.1 Xác định vùng nhận diện

Sử dụng bộ lọc Sobel cả hai chiều ngang và dọc để nhận dạng các cạnh, từ đó xác định vùng cần nhận diện mã vạch. Sobel có khả năng chống nhiễu tốt.

Chúng em sử dụng Scharr kernel: $G_x = \begin{bmatrix} +3 & 0 & -3 \\ +10 & 0 & -10 \\ +3 & 0 & -3 \end{bmatrix}$ và $G_y = \begin{bmatrix} +3 & +10 & +3 \\ 0 & 0 & 0 \\ -3 & -10 & -3 \end{bmatrix}$

Cường độ gradient (gradient magnitude): $G = \sqrt{G_x^2 + G_y^2}$

2.1.2 Xoay hình ảnh về đúng vị trí

Nhận dạng các cạnh với phương pháp tích chập filter2D kết hợp với phương pháp nhận dạng đường thẳng Probabilistic Hough Line Transform.

2.1.3 Nhận diện mã vạch

Về cơ bản cũng thực hiện phương pháp xử lý giống với bước *Xoay ảnh về đúng vị trí* nhưng kết hợp thêm với Morphological Shape.

2.2 Đối với camera

Chúng em sử dụng bộ lọc Sobel cả hai chiều ngang và dọc với Scharr kernel và kết hợp với Morphological Shape.

3 Tài liệu tham khảo

[1] Image Gradients with OpenCV (Sobel and Scharr) – PyImageSearch link: <https://www.pyimagesearch.com/2021/05/12/image-gradients-with-opencv-sobel-and-scharr/>

[2] Convolutions with OpenCV and Python – PyImageSearch link: <https://www.pyimagesearch.com/2016/07/25/convolutions-with-opencv-and-python/>

[3] OpenCV Morphological Operations – PyImageSearch link: <https://www.pyimagesearch.com/2021/04/28/opencv-morphological-operations/>

