

ĐẠI HỌC QUỐC GIA TP.HCM
TRƯỜNG ĐẠI HỌC CÔNG NGHỆ
THÔNG TIN



NGÀNH: KHOA HỌC MÁY TÍNH

MÔN HỌC: NHẬP MÔN THỊ GIÁC MÁY TÍNH

**TÌM HIỂU MỘT SỐ
PHƯƠNG PHÁP PHÂN
ĐOẠN ẢNH TRONG THỊ
GIÁC MÁY TÍNH**

GVHD: TS.Mai Tiến Dũng

SV: Phạm Trung Hiếu - 19521512

Võ Khoa Nam - 19521877

Trịnh Minh Hoàng - 19521547

Contents

1	Tổng quan về xử lý ảnh và phân đoạn ảnh	3
1.1	Giới thiệu về xử lý ảnh	3
1.2	Quá trình xử lý ảnh	3
1.3	Giới thiệu phân đoạn ảnh	5
1.3.1	Khái niệm phân đoạn ảnh	5
1.3.2	Phân loại phân đoạn ảnh	6
1.4	Một số khái niệm trong xử lý ảnh và phân đoạn ảnh cần biết .	7
1.4.1	Điểm ảnh - Pixel	7
1.4.2	Mức xám - Grey level	8
1.4.3	Ảnh đen trắng - Ảnh nhị phân - Ảnh màu	9
1.4.4	Biên	9
1.4.5	Làm mịn ảnh	9
2	Phân đoạn dựa trên ngưỡng	10
2.1	Phân đoạn ảnh dựa trên ngưỡng (Thresholding)	11
2.1.1	Chọn ngưỡng T cố định	11
2.1.2	Chọn ngưỡng theo histogram	11
2.1.3	Phân đoạn ảnh theo ngưỡng toàn cục (Global threshold) .	12
2.1.4	Basic global threshold	13
2.1.5	Adaptive threshold (Ngưỡng cục bộ)	15
2.1.6	Phương pháp phân đoạn ảnh Adaptive threshold mean c .	17
3	Contour detection	18
3.1	Đường bao (contour) là gì ?	18
3.2	Ứng dụng của đường bao trong thị giác máy tính	19
3.3	Xác định và cài đặt contour trong OpenCV	19
3.4	Cài đặt và thực nghiệm contour detection sử dụng OpenCV .	20
3.4.1	Dữ liệu thực nghiệm	20
3.4.2	Các hàm và tham số sử dụng cài đặt	21
3.4.3	Các bước cài đặt	22
3.5	Ưu điểm và nhược điểm của phương pháp Contour detection .	22
3.5.1	Ưu điểm	22
3.5.2	Nhược điểm	22

4	Image Segmentation using Color Space	23
4.1	Khái niệm	23
4.2	Ý tưởng	23
4.3	Các bước tiến hành	24
4.4	Ưu và nhược điểm của phương pháp	25
4.4.1	Ưu điểm	25
4.4.2	Nhược điểm	25
5	Image Segmentation using Unet	26
5.1	Giới thiệu mạng Unet	26
5.2	Cách thức hoạt động của mạng Unet	27
5.3	Bộ dữ liệu áp dụng mạng Unet	28
5.4	Kết quả thực nghiệm	29
5.5	Nhận xét	29
6	Tài liệu tham khảo	31

1 Tổng quan về xử lý ảnh và phân đoạn ảnh

1.1 Giới thiệu về xử lý ảnh

Xử lý ảnh ngày nay là một ngành khoa học lớn và có mặt trong nhiều lĩnh vực của cuộc sống. Xử lý ảnh là ngành khoa học nghiên cứu các quá trình xử lý thông tin dạng hình ảnh, mà hình ảnh là một trong những dạng thông tin phong phú nhất đối với chúng ta. Trong quá trình xử lý ảnh bước quan trọng nhất và cũng là khó khăn nhất là bước phân đoạn ảnh. Phân đoạn ảnh nhằm mục đích phân tách các đối tượng cấu thành nên ảnh thô để có thể sử dụng cho các ứng dụng về sau.

Xử lý ảnh là một ngành khoa học mới mẻ so với các ngành khác, song nó lại có tốc độ phát triển rất nhanh, xử lý ảnh được đưa vào giảng dạy ở bậc đại học ở nước ta. Nó là môn học liên quan đến nhiều lĩnh vực và cần nhiều kiến thức cơ sở khác.

Xử lý ảnh số là thao tác trên các ảnh tự nhiên thông qua các phương pháp và kỹ thuật mã hoá. Ảnh sau khi được thu nhận bằng các thiết bị thu nhận sẽ được biến đổi thành ảnh số theo các phương pháp số hoá được nhún trong các thiết bị kỹ thuật khác nhau và được biểu diễn trong máy tính dưới dạng ma trận 2 chiều hoặc 3 chiều.

Mục đích của việc xử lý ảnh để:

- Biến đổi ảnh làm tăng chất lượng ảnh
- Tự động nhận dạng, đoán ảnh, đánh giá nội dung của ảnh

1.2 Quá trình xử lý ảnh

Quá trình xử lý một hình ảnh thì bao gồm 5 bước quan trọng là:

- Thu nhận ảnh đầu vào
- Tiền xử lý
- Phân đoạn ảnh

- Biểu diễn và mô tả hình ảnh
- Nhận dạng và giải thích

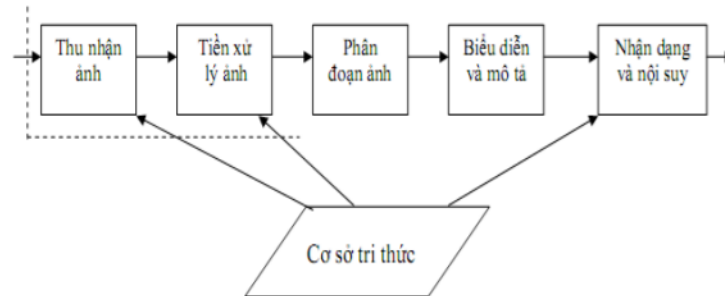


Figure 1: Quá trình xử lý ảnh

Thu nhận ảnh: đây là công đoạn đầu tiên mang tính quyết định đối với cả quá trình. Ảnh đầu vào được thu nhận qua các thiết bị như camera, sensor, máy scanner,... và sau đó các tính hiệu này được số hoá. Các thông số quan trọng của bước này cần quan tâm là độ phân giải, chất lượng màu, dung lượng ảnh, tốc độ thu nhận ảnh.

Tiền xử lý: ở bước này ảnh sẽ được cải thiện về độ tương phản, khử nhiễu, khử bóng, khử độ lệch,... với mục đích làm cho chất lượng ảnh trở nên tốt hơn.

Phân đoạn ảnh: là bước then chốt trong qua trình xử lý ảnh, quá trình này nhằm phân tích ảnh thành những thành phần có cùng tính chất nào đó, dựa theo biên hay các vùng liên thông. Đây là phần phức tạp và khó khăn nhất trong xử lý ảnh và cũng dễ gây lỗi, làm mất độ chính xác của ảnh.

Biểu diễn và mô tả ảnh: kết quả của bước phân đoạn ảnh thường được cho dưới dạng dữ liệu điểm ảnh thô như hàm chứa các biên của một vùng ảnh hay là tập hợp các điểm ảnh thuộc về vùng nào đó. Việc biến đổi các số liệu này thành dạng thích hợp là rất cần thiết. Quá trình chọn các tính chất để biểu diễn hình ảnh còn được gọi là rút trích đặc trưng ảnh.

Nhận dạng và giải thích: là bước cuối cùng trong quá trình xử lý ảnh. Nhận dạng có thể nhìn nhận một cách đơn giản là việc gán nhãn cho các đối tượng trong ảnh hay là quá trình xác định ảnh. Giải thích là phán đoán theo ý nghĩa trên cơ sở nhận dạng.

1.3 Giới thiệu phân đoạn ảnh

1.3.1 Khái niệm phân đoạn ảnh

Phân đoạn ảnh (hay còn gọi là phân vùng ảnh) là bước then chốt trong xử lý ảnh. Giai đoạn này nhằm phân tích ảnh thành những thành phần có cùng tính chất nào đó dựa theo biên hay các vùng liên thông. Tiêu chuẩn để xác định các vùng liên thông có thể là cùng mức xám, cùng màu hay cùng độ nhám... Trước hết cần làm rõ khái niệm "vùng ảnh" (Segment) và đặc điểm vật lý của vùng.

Vùng ảnh là một chi tiết, một thực thể trong toàn cảnh. Nó là một tập hợp các điểm có cùng hoặc gần cùng một tính chất nào đó : mức xám, mức màu, độ nhám... Vùng ảnh là một trong hai thuộc tính của ảnh. Nói đến vùng ảnh là nói đến tính chất bề mặt. Đường bao quanh một vùng ảnh (Boundary) là biên ảnh. Các điểm trong một vùng ảnh có độ biến thiên giá trị mức xám tương đối đồng đều hay tính kết cấu tương đồng.

Dựa vào đặc tính vật lý của ảnh, người ta có nhiều kỹ thuật phân vùng : nếu phân vùng dựa theo miền liên thông, ta gọi là phân vùng dựa theo miền đồng nhất hay miền kề; phân vùng dựa vào biên gọi là phân vùng biên. Ngoài ra còn có các kỹ thuật phân vùng khác dựa vào biên độ, phân vùng dựa theo kết cấu (texture segmentation).

Mục đích của phân tích ảnh là để có một miêu tả tổng hợp về nhiều phần tử khác nhau cấu tạo nên ảnh thô (brut Image). Vì lượng thông tin chứa trong ảnh là rất lớn, trong khi đó đa số ứng dụng chỉ cần một số thông tin đặc trưng nào đó, do vậy cần có một quá trình để giảm lượng thông tin khổng lồ ấy. Quá trình này bao gồm phân đoạn ảnh và trích chọn đặc tính chủ yếu.

Để phân tích các đối tượng trong ảnh, chúng ta cần phải phân biệt được các đối tượng cần quan tâm với phần còn lại của ảnh. Những đối tượng này có thể tìm ra được nhờ các kỹ thuật phân đoạn ảnh, theo nghĩa tách phần tiền

cảnh ra khỏi hậu cảnh trong ảnh.

Có 2 điều cần ghi nhớ khi nghiên cứu áp dụng các phương pháp phân đoạn ảnh:

- Không có kỹ thuật phân đoạn ảnh nào là vạn năng , nghĩa là áp dụng được cho mọi loại ảnh.
- Không có kỹ thuật phân đoạn nào là hoàn hảo

1.3.2 Phân loại phân đoạn ảnh

Phân đoạn ảnh là bước then chốt trong xử lý ảnh. Giai đoạn này nhằm phân tích ảnh thành các thành phần có cùng tính chất dựa vào biên hay những vùng liên thông. Tiêu chuẩn để xác định các vùng liên thông có thể là cùng mức xám, cùng màu,... Vùng ảnh là một thuộc tính quan trọng của ảnh. Đường bao quanh một vùng ảnh gọi là biên ảnh.

Có thể hiểu phân đoạn là tiến trình chia ảnh thành nhiều vùng, mỗi vùng chứa một đối tượng hay nhóm đối tượng cùng kiểu. Chẳng hạn, một đối tượng có thể là một ký tự trên một trang văn bản hoặc một đoạn thẳng trong một bản vẽ kỹ thuật hoặc một nhóm các đối tượng có thể biểu diễn một từ hay hay đoạn thẳng tiếp xúc nhau. Ta có một số phương pháp phân đoạn ảnh như sau:

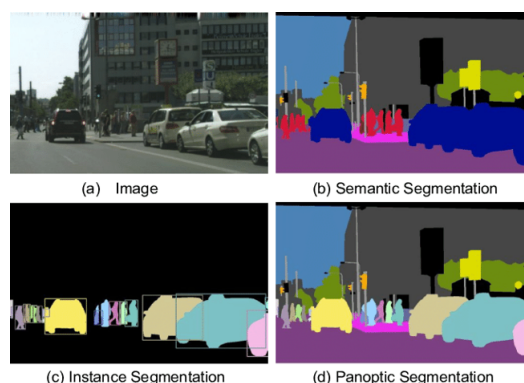


Figure 2: Các loại phân đoạn ảnh

- Phân đoạn theo cá thể (Instance Segmentation)
- Phân đoạn theo ngữ nghĩa (Semantic Segmentation)
- Phân đoạn khái quát hay phân đoạn toàn cảnh

Phân đoạn cá thể (Instance Segmentation): Nó phân đoạn các trường hợp khác nhau của mỗi loại ngữ nghĩa và do đó nó xuất hiện như một phần mở rộng của phân đoạn ngữ nghĩa. Mỗi cá thể của một đối tượng (vật, ví dụ: người) được gán nhãn riêng biệt và các phân đoạn không có cá thể (thứ) nào bị bỏ qua.

Phân đoạn ngữ nghĩa (Semantic Segmentation): Nó đề cập đến nhiệm vụ xác định các lớp đối tượng khác nhau trong một hình ảnh. Nó phân loại rộng rãi các đối tượng thành các danh mục ngữ nghĩa như người, sách, hoa, xe hơi, ... Ở đây tất cả các vùng của hình ảnh (vật + thứ) được xem xét và các đối tượng cùng loại được coi là một nhãn duy nhất. Ví dụ: Nếu nhìn vào hình (b), tất cả các trường hợp của đối tượng là ô tô hoặc người có cùng màu, cũng có thể hiểu là chúng được gán một nhãn chung.

Phân đoạn toàn cảnh: Nó là sự kết hợp giữa Phân đoạn ngữ nghĩa và Phân đoạn cá thể sao cho tất cả các pixel được gán nhãn lớp và tất cả các thể hiện đối tượng được phân đoạn duy nhất. Ở đây, mỗi pixel được gán một nhãn cho từng lớp (ví dụ: người) và nếu một phân đoạn thuộc danh mục thứ, thì mỗi thể hiện của nó được gán nhãn riêng lẻ. Chú thích mẫu, cú pháp cho từng chú thích: [`<class label>`, `<instance id>`], for thing: ["person", "1"], for things: ["building", "None"].

1.4 Một số khái niệm trong xử lý ảnh và phân đoạn ảnh cần biết

1.4.1 Điểm ảnh - Pixel

Ảnh trong thực tế là một ảnh liên tục về không gian và về giá trị độ sáng. Để có thể xử lý ảnh bằng máy tính ta cần phải tiến hành số hoá ảnh. Quá trình số hoá là biến đổi tín hiệu liên tục sang tín hiệu rời rạc thông qua quá trình lấy mẫu và lượng hoá thành phần giá trị mà về nguyên tắc bằng mắt thường không phân biệt được hai điểm kề nhau. Mỗi một điểm như vậy gọi là pixel hay còn được gọi là Picture element (Pel). Như vậy một ảnh là một

tập hợp các pixel.

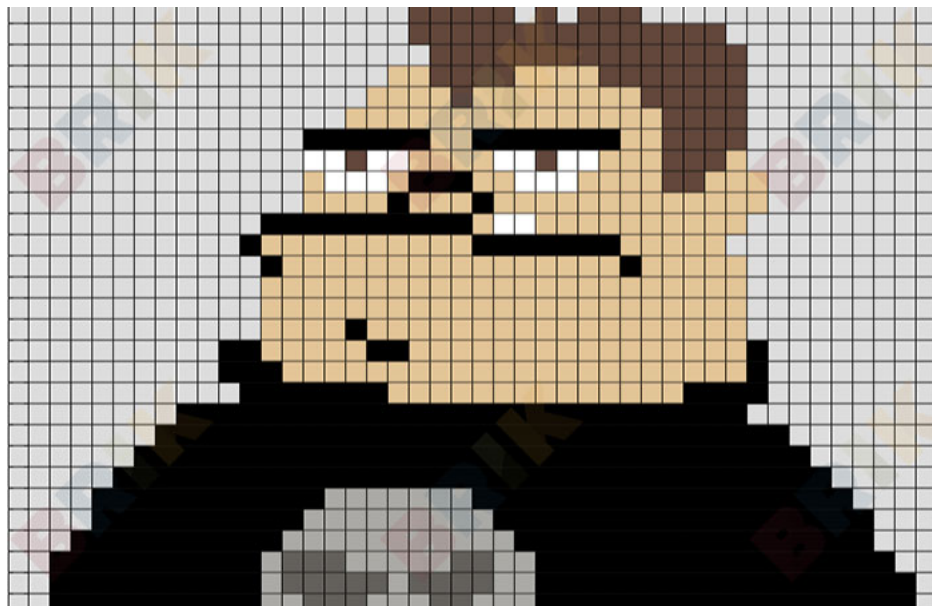


Figure 3: Điểm ảnh

Trong figure 3, mỗi ô vuông đại diện cho một điểm ảnh(pixel).

1.4.2 Mức xám - Grey level



Figure 4: Ảnh mức xám

Mức xám là kết quả sự mã hoá tương ứng một cường độ sáng của mỗi điểm ảnh với một giá trị số, kết quả của quá trình lượng hoá. Dễ hiểu hơn là mức xám được định nghĩa là cường độ sáng của điểm ảnh được gán bằng

giá trị số tại điểm đó.

Các thang giá trị mức xám thường dùng là 16, 32 hay 64. Phổ biến nhất ở mức 256 ở mức này mỗi Pixel sẽ được mã hoá bởi 8 bit.

1.4.3 Ảnh đen trắng - Ảnh nhị phân - Ảnh màu

Ảnh đen trắng là ảnh có hai màu đen trắng (không chứa màu khác) với mức xám ở các điểm ảnh có thể khác nhau.

Ảnh nhị phân có hai mức đen trắng phân biệt tức là dùng 1 bit mô tả 21 mức khác nhau. Nói cách khác là mỗi điểm ảnh của ảnh nhị phân chỉ có thể là 0 và 1.

Ảnh màu là một ma trận các pixel mà mỗi pixel biểu diễn một điểm màu. Mỗi điểm màu được biểu diễn bằng bộ 3 số (r,g,b).

1.4.4 Biên

Biên là một đặc tính rất quan trọng của đối tượng trong ảnh, nhờ vào biên mà chúng ta phân biệt được đối tượng này với đối tượng khác. Một điểm ảnh có thể gọi là điểm biên nếu ở đó có sự thay đổi đột ngột về mức xám. Tập hợp các điểm biên tạo thành biên hay còn gọi là đường bao ảnh (contour).

1.4.5 Làm mịn ảnh

Làm mịn ảnh (smoothing) là công việc quen thuộc trong xử lý ảnh. Nó giúp loại bỏ các dữ liệu không cần thiết như hạt nhiễu, làm mượt biên ảnh,... Có nhiều phương thức khác nhau được OpenCV cung cấp sẵn (Average, Gaussian,...).

Để hiểu rõ về phương pháp làm trơn ảnh, cần phải nắm một số khái niệm sau:

- Convolution: Là các phép tính toán giữa ảnh và toán tử (kernel).
- Kernel: Là mảng hai chiều (ma trận) có kích thước cố định do người dùng chỉ định.

- Anchor point: Là điểm neo nằm bên trong kernel, thường là vị trí trung tâm kernel.

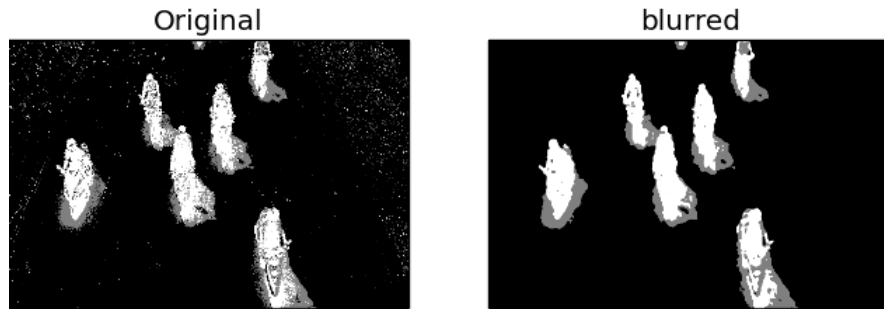


Figure 5: Ảnh trước và sau khi làm mịn

CHƯƠNG 2: Một Số Kỹ Thuật Phân Đoạn Ảnh

2 Phân đoạn dựa trên ngưỡng

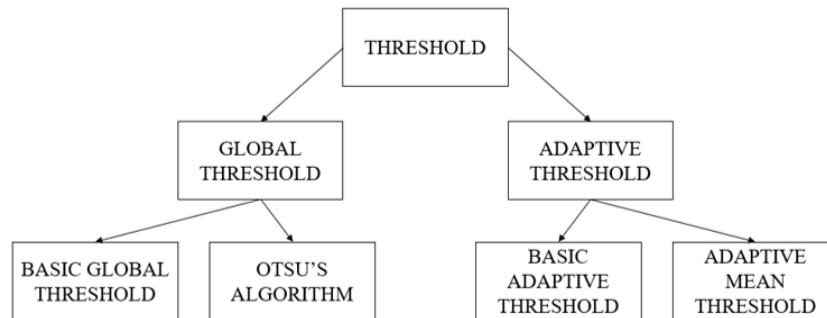


Figure 6: Một số kỹ thuật toán phân đoạn ảnh theo ngưỡng

Trong OpenCV, ngưỡng là một số nằm trong đoạn từ 0 đến 255. Giá trị ngưỡng sẽ chia tách giá trị độ xám của ảnh thành hai miền riêng biệt, mỗi miền có thể là vật hoặc nền.

Ngoài ra, sau khi chia tách giá trị độ xám thành hai miền riêng biệt, giá trị

độ sáng của các pixel trong hai miền này có thể có giá trị là 1 hoặc 0 (Ví dụ: Phân tách vật có mức xám là 1 và Phân tách nền có mức xám là 1). Vì vậy, kỹ thuật phân đoạn theo ngưỡng có thể hiểu đơn thuần là kỹ thuật nhị phân hóa ảnh.

2.1 Phân đoạn ảnh dựa trên ngưỡng (Thresholding)

Việc phân ngưỡng cho một ảnh thường chỉ hoạt động ở ảnh xám, nếu như ảnh ở dạng màu RGB chưa được chuyển thành ảnh xám thì phải chuyển về ảnh xám.

Việc chọn ngưỡng(T) trong kỹ thuật này là một bước vô cùng quan trọng, thông thường người ta tiến hành theo các bước chung như sau:

- Xem xét histogram của ảnh để xác định đỉnh.
- Chọn ngưỡng T sao cho một phần xác định trước của toàn bộ số mẫu là thấp hơn.
- Điều chỉnh ngưỡng dựa trên xét histogram của các điểm lân cận.
- Chọn ngưỡng bằng cách xem xét histogram của những điểm thỏa tiêu chuẩn đã chọn.

2.1.1 Chọn ngưỡng T cố định

Trong trường hợp ảnh có độ tương phản rất cao, trong đó các đối tượng quan tâm rất tối còn nền gần như là đồng nhất và rất sáng thì việc chọn ngưỡng cố định $T=128$ (xét trên thang độ sáng từ 0 đến 255) là một giá trị chọn khá chính xác.

Tuy nhiên, nó không thành công nếu có độ tương phản giữa đối tượng và nền thấp

2.1.2 Chọn ngưỡng theo histogram

Trong hầu hết các trường hợp, ngưỡng được chọn từ lược đồ độ sáng của vùng hay ảnh cần phân đoạn. Có rất nhiều kỹ thuật chọn ngưỡng tự động xuất phát từ histogram đã được nghiên cứu và đưa ra.

Một số kỹ thuật phổ biến sẽ được biểu diễn trong phần tiếp theo cùng với các thuật toán của phân đoạn ảnh theo ngưỡng mà nhóm đã biểu diễn trong sơ đồ phân loại thuật toán Threshold.

Những kỹ thuật này có thể tận dụng những lợi thế do sự làm trơn dữ liệu lược đồ ban đầu mang lại nhằm loại bỏ những dao động nhỏ về độ sáng. Tuy nhiên các thuật toán làm trơn cần phải cẩn thận, không được làm dịch chuyển các vị trí đỉnh của lược đồ. Vì vậy để tăng chất lượng sau khi phân đoạn theo ngưỡng, ta cần phải sử dụng bộ lọc để làm trơn ảnh.

2.1.3 Phân đoạn ảnh theo ngưỡng toàn cục (Global threshold)

Ngưỡng toàn cục là một phương pháp phân đoạn ảnh ở mức đơn giản, dễ hiểu và dễ cài đặt cũng như chọn ngưỡng. Nó chỉ có một số mức độ hữu dụng nhất định khi ta cần phân đoạn các ảnh xám đơn giản, các đối tượng Vật và Nền tách biệt và ảnh không bị ảnh hưởng quá nhiều ở từng vùng ảnh khác nhau bởi môi trường.

Việc phân đoạn ảnh theo ngưỡng toàn cục có thể hiểu là khi mà ta chọn một ngưỡng T nào đó với yêu cầu đặt ra là để phân chia Nền và đối tượng, đối tượng có thể được trích xuất khỏi nền bằng một thao tác đơn giản so sánh giá trị hình ảnh với giá trị ngưỡng T .

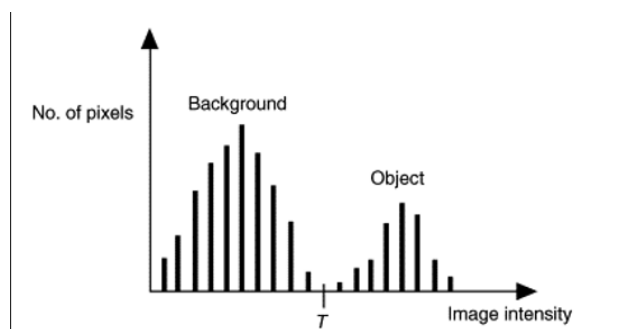


Figure 7: Ngưỡng T phân tách nền và đối tượng

Kết quả của việc xác định ngưỡng là một hình ảnh nhị phân, trong đó các pixel có giá trị cường độ 1 tương ứng với các đối tượng, trong khi các pixel

có giá trị 0 tương ứng với nền.

Ưu điểm: Tính toán ngưỡng toàn cục đơn giản và nhanh chóng. Nó có mức độ hữu dụng nhất định khi ta cần phân đoạn các ảnh xám đơn giản, hình ảnh có chứa các đối tượng có giá trị cường độ đồng nhất trên nền tương phản.

Nhược điểm: không thành công nếu có độ tương phản thấp giữa đối tượng và nền, nếu hình ảnh bị nhiễu hoặc nếu cường độ nền thay đổi đáng kể trên toàn ảnh.

2.1.4 Basic global threshold

Basic Global Threshold thuộc nhóm Basic của bài toán phân đoạn ảnh theo ngưỡng. Vậy, việc chọn ngưỡng cho thuật toán này dựa vào việc xem xét Histogram và phải tự chọn ngưỡng dựa trên thử nhiều ngưỡng hoặc xem xét các khe giữa các đỉnh trong Histogram.

Đối với Basic Global Threshold thì Histogram chúng ta cần xem xét để chọn ngưỡng có hai dạng, tương ứng với hai cách chọn ngưỡng:

- Chọn ngưỡng cho Histogram có hai đỉnh (Bimodal Histogram).
- Chọn ngưỡng cho Histogram có nhiều hơn hai đỉnh (Multimodal Histogram).

Chọn ngưỡng cho Histogram có hai đỉnh:

- B1: Chọn ngưỡng T .
- B2: Tính toán kết quả phân đoạn:

$$g(x, y) = \begin{cases} 255 & \text{if } f(x, y) > T \\ 0 & \text{if } f(x, y) \leq T \end{cases}$$

$g(x, y)$: giá trị điểm ảnh tại tọa độ (x, y) sau khi phân đoạn ảnh.

$f(x, y)$: giá trị điểm ảnh tại tọa độ (x, y) trước khi phân đoạn ảnh.

Hoặc ta có thể hiểu là nếu pixel có giá trị lớn hơn giá trị ngưỡng thì nó được

gán giá trị là 255, ngược lại nhỏ hơn giá trị ngưỡng thì nó được gán giá trị khác là 0.

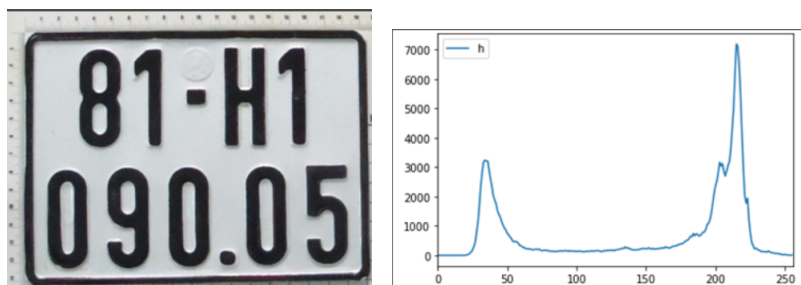


Figure 8: Ảnh grayscale với histogram tương ứng

Đối figure 8 ở trên, ta có thể thấy histogram của ảnh này có hai đỉnh nằm cách xa nhau, hai đỉnh này tương ứng với Nền và Vật. Vì Vật và Nền khá tách biệt nhau nên khe giữa hai đỉnh và vật sẽ là một khoảng các giá trị (80-160) dựa theo histogram, ta chỉ cần chọn một giá trị ngẫu nhiên trong khoảng này. Ở đây, ta chọn ngưỡng là 120. Khi đó, các giá trị pixel mà lớn hơn 120 thì sẽ được gán cho giá trị mới là 255, ngược lại gán bằng 0.



Figure 9: Ảnh kết quả khi sử dụng Global Threshold với $T = 120$

Chọn ngưỡng cho Histogram có nhiều hơn hai đỉnh:

- B1: Chọn ra hai ngưỡng $T1$ và $T2$
- B2: Tính toán kết quả phân đoạn:

$$g(x, y) = \begin{cases} 255 & \text{if } f(x, y) > T1 \\ 0 & \text{if } f(x, y) < T2 \\ f(x, y) & \text{otherwise} \end{cases}$$

$g(x,y)$: giá trị điểm ảnh tại tọa độ (x,y) sau khi phân đoạn.

$f(x,y)$: giá trị điểm ảnh tại tọa độ (x, y) trước khi phân đoạn ảnh.

Hoặc ta có thể hiểu là nếu pixel có giá trị lớn hơn giá trị ngưỡng $T1$ và bé hơn giá trị ngưỡng $T2$ thì nó được gán bằng 255, ngược lại nhỏ hơn giá trị ngưỡng thì nó được gán bằng 0.

2.1.5 Adaptive threshold (Ngưỡng cục bộ)

Một trong những nhược điểm của việc sử dụng các phương pháp tạo ngưỡng đơn giản là chúng ta cần cung cấp giá trị ngưỡng của mình, T theo cách thủ công. Hơn nữa, việc tìm kiếm một giá trị tốt của T có thể yêu cầu nhiều thí nghiệm thủ công và điều chỉnh tham số, điều này đơn giản là không thực tế trong hầu hết các tình huống.

Global Threshold đã giải quyết được nhu cầu cơ bản của người dùng là phân đoạn ảnh ra thành ảnh nhị phân với thông tin trong ảnh là 0 và 255 (tương ứng với Nền hoặc Vật). Nhưng vì ngưỡng được gán toàn cục cho mọi pixel trong ảnh nên đối với việc phân đoạn một số ảnh bị nhiễu, bị phơi sáng, đèn flash, làm cho mức xám trong vùng vùng của ảnh bị khác nhau, khi áp dụng Global Threshold sẽ bị giảm đi chất lượng phân đoạn, hoặc thậm chí không phân đoạn được rõ Vật và Nền.

Ý tưởng: Adaptive Threshold chia nhỏ hình ảnh thành các khối hoặc vùng khác nhau có kích thước giống, sau đó giá trị ngưỡng được tính toán cho các vùng chia nhỏ và do đó, sẽ có các giá trị ngưỡng khác nhau cho các vùng khác nhau.

Cách chọn ngưỡng: Tương tự như phương pháp chọn ngưỡng của bimodal histogram, có thể sử dụng kiến thức về tính chất của histogram để nhìn trực tiếp vào Histogram tìm ra các khe giữa các đỉnh. Lặp lại lần lượt đối với toàn bộ vùng mà ta đã chia nhỏ ảnh gốc ra.

Ví dụ: yêu cầu của bài toán là phân đoạn chữ ra khỏi nền trắng của trang

giấy, cách thực hiện.

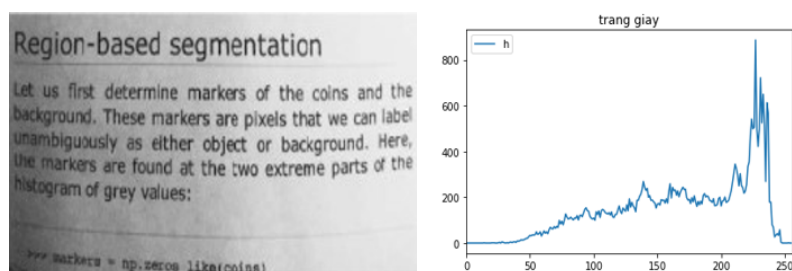


Figure 10: Ảnh grayscale với histogram tương ứng

Với giá trị $T = 158$ ta thu được ảnh kết quả sau:

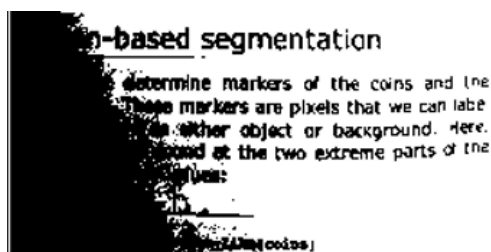


Figure 11: Ảnh kết quả khi sử dụng Global Threshold với $T = 158$

Ta có thể thấy là việc sử dụng Global Threshold trong trường hợp này thì không thể giải quyết được yêu cầu bài toán, vì ảnh hưởng bởi độ sáng thay đổi theo từng vùng.

Sau khi chia ảnh thành 5×3 vùng với kích thước giống nhau và chọn ngưỡng tương ứng, thì kết quả sau khi phân đoạn cũng ổn hơn so với việc áp dụng Global Threshold. Nhưng các vùng đen vẫn còn tồn tại cũng vì do trong mỗi vùng sau khi chia ảnh vẫn còn tồn tại nhiều sáng trong chính vùng đó. Cách giải quyết đơn giản đó là chia ảnh thành nhiều vùng hơn.

Tuy nhiên, sẽ xuất hiện vấn đề, nếu chia ảnh ra thành nhiều vùng hơn thì tức là số lượng vùng ảnh ta cần phải xem xét và chọn ngưỡng trên từng histogram sẽ tăng lên. Việc chọn ngưỡng cho từng vùng sẽ tốn rất nhiều thời

gian và công sức. Adaptive Mean Threshold có khả năng khắc phục được vấn đề này.

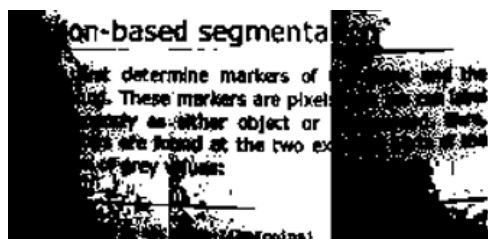


Figure 12: Kết quả ảnh sau khi áp dụng Basic Adaptive Threshold

2.1.6 Phương pháp phân đoạn ảnh Adaptive threshold mean c

Một trong những nhược điểm của việc sử dụng các phương pháp tạo ngưỡng đơn giản là chúng ta cần cung cấp giá trị ngưỡng của mình, T theo cách thủ công. Hơn nữa, việc tìm kiếm một giá trị tốt của T có thể yêu cầu nhiều thí nghiệm thủ công và điều chỉnh tham số, điều này đơn giản là không thực tế trong hầu hết các tình huống.

Để khắc phục vấn đề này, chúng ta có thể sử dụng ngưỡng thích ứng, xem xét các điểm ảnh lân cận nhỏ và sau đó tìm một giá trị ngưỡng tối ưu T cho mỗi lân cận. Phương pháp này cho phép chúng tôi xử lý các trường hợp có thể có phạm vi cường độ pixel đáng kể và giá trị tối ưu của T có thể thay đổi đối với các phần khác nhau của hình ảnh.

Trong ngưỡng thích ứng, đôi khi được gọi là ngưỡng cục bộ, mục tiêu của chúng tôi là kiểm tra thống kê các giá trị cường độ pixel trong vùng lân cận của một pixel nhất định.

Giả định chung làm cơ sở cho tất cả các phương pháp tạo ngưỡng cục bộ và thích ứng là các vùng nhỏ hơn của hình ảnh có nhiều khả năng có độ chiếu sáng xấp xỉ đồng đều hơn. Điều này ngụ ý rằng các vùng cục bộ của một hình ảnh sẽ có ánh sáng tương tự, trái ngược với toàn bộ hình ảnh, có thể có ánh sáng khác nhau đáng kể cho từng vùng.

Tuy nhiên, việc chọn kích thước của vùng lân cận pixel cho ngưỡng cục bộ

là hoàn toàn quan trọng.

Vùng lân cận phải đủ lớn để bao phủ đủ pixel nền và nền trước, nếu không giá trị của T sẽ ít nhiều không liên quan.

Nhưng nếu chúng ta đặt giá trị vùng lân cận của mình quá lớn, thì chúng ta hoàn toàn vi phạm giả định rằng các vùng cục bộ của hình ảnh sẽ có độ chiếu sáng xấp xỉ đồng đều. Một lần nữa, nếu chúng ta cung cấp một vùng lân cận rất lớn, thì kết quả của chúng ta sẽ rất giống với việc tạo ngưỡng toàn cục bằng cách sử dụng phương pháp tạo ngưỡng đơn giản.

Trong thực tế, việc điều chỉnh kích thước vùng lân cận (thường) không phải là vấn đề khó. Bạn sẽ thường thấy rằng có một loạt các kích thước vùng lân cận cung cấp cho bạn kết quả thích hợp - nó không giống như việc tìm kiếm một giá trị T tối ưu có thể tạo ra hoặc phá vỡ đầu ra ngưỡng của bạn.

Thực tế phổ biến là sử dụng trung bình cộng hoặc trung bình Gauss của cường độ pixel trong mỗi vùng (các phương pháp khác tồn tại, nhưng trung bình số học và trung bình Gauss cho đến nay là phổ biến nhất).

Trong trung bình cộng, mỗi điểm ảnh trong góp phần khu phố như nhau đối với máy tính T . Và trong các giá trị trung bình, điểm ảnh Gaussian xa từ (x, y) Trung tâm Phối của khu vực đóng góp ít hơn để tính toán tổng thể của T .

Công thức chung để tính T là:

$$T = \text{mean}(I_L) - C$$

Nơi giá trị trung bình là một trong hai giá trị trung bình số học hoặc Gaussian, L là địa phương tiểu vùng của hình ảnh, I , và C là một số không đổi mà chúng ta có thể sử dụng để tinh chỉnh các giá trị ngưỡng T .

3 Contour detection

3.1 Đường bao (contour) là gì ?

Khi chúng ta nối tất cả các điểm trên ranh giới của một đối tượng, chúng ta sẽ có một đường khép kín bao quanh đối tượng (đường bao). Một đường bao

cụ thể đề cập đến các pixel ranh giới có cùng màu sắc và cường độ. Vùng có cùng cường độ và màu sắc này tạo thành đường đối tượng mà chúng ta cần phát hiện và đường cong khép kín bao quanh đường này là đường bao thể hiện hình dạng của đối tượng. Các đường bao là một công cụ hữu ích để phân tích hình dạng, phát hiện và nhận dạng đối tượng. Vì vậy, về cơ bản phát hiện đường bao (đường viền) hoạt động tương tự như phát hiện cạnh nhưng với điều kiện là các cạnh phải tạo thành một đường khép kín.

3.2 Ứng dụng của đường bao trong thị giác máy tính

Một số ứng dụng đã được xây dựng, sử dụng các đường bao để phát hiện hoặc phân đoạn chuyển động :

- Phát hiện chuyển động: trong các video giám sát, công nghệ phát hiện chuyển động có nhiều ứng dụng, từ môi trường an ninh trong nhà và ngoài trời, kiểm soát giao thông, phát hiện hành vi trong các hoạt động thể thao, phát hiện các đối tượng không được giám sát và thậm chí là nén video. Trong hình bên dưới, việc phát hiện chuyển động của mọi người trong luồng video rất hữu ích trong việc giám sát. Hình ảnh bên dưới chỉ phát hiện những người chuyển động trong video, còn nhóm người đang đứng yên bên trái sẽ không bị phát hiện.
- Phát hiện đối tượng không có người giám sát: bất kỳ đối tượng nào không được giám sát ở những nơi công cộng thường được coi là đối tượng khả nghi. Một giải pháp cho vấn đề đó là phát hiện đối tượng không cần giám sát bằng việc hình thành đường viền.

Thay thế nền của hình ảnh với nhau, sử dụng đường viền để phân đoạn hình ảnh.

3.3 Xác định và cài đặt contour trong OpenCV

Contour rất hữu ích trong phân tích hình dạng, phát hiện đối tượng và nhận diện đối tượng. Vì vậy cần lưu ý khi sử dụng contours:

Để độ chính xác cao hơn thì nên sử dụng hình ảnh nhị phân (chỉ gồm hai màu là trắng và đen). Do đó trước khi phát hiện contours thì nên áp dụng threshold hoặc thuật toán canny để chuyển sang hình ảnh nhị phân

Hàm `findContours()` và hàm `drawContours()` sẽ thay đổi hình ảnh gốc. Do đó muốn thu được hình ảnh gốc với contours vừa tìm được thì phải lưu hình ảnh gốc vào một biến khác

Trong OpenCV, tìm contours như là tìm các vật thể màu trắng từ nền màu đen. Do đó object cần tìm nên là màu trắng và background màu đen.

3.4 Cài đặt và thực nghiệm contour detection sử dụng OpenCV

3.4.1 Dữ liệu thực nghiệm

Nhóm tiến hành thực nghiệm trên một số ảnh và tập dữ liệu ảnh do nhóm thu thập riêng cho phương pháp Contour Detection Về tập dữ liệu nhóm cho



Figure 13: Ảnh nhóm thực nghiệm

contour detection:

- Số lượng: gồm 10 ảnh (10 ảnh dùng để predict và 10 ảnh ground-truth)
- Toàn bộ đều là ảnh mức xám (grey level)
- Kích thước mỗi ảnh 512x512

Một số hình ảnh về bộ dữ liệu:

Ảnh dùng để predict

Ảnh ground-truth



Figure 14: Ảnh dùng để predict bộ dữ liệu



Figure 15: Ảnh ground-truth bộ dữ liệu

3.4.2 Các hàm và tham số sử dụng cài đặt

Hàm `findContours()` sử dụng để tìm đường viền, trả về danh sách các đường viền (contours):

- Input image: hình ảnh đầu vào
- Contour retrieval mode
- Contour approximation mode

Hàm `drawContours()` sử dụng để vẽ các đường viền vừa tìm được trên ảnh nguồn:

- Source image: ảnh nguồn (nên sử dụng ảnh copy)
- Contours: danh sách các cạnh vừa tìm được
- Color, thickness: màu sắc và kích cỡ của đường viền

3.4.3 Các bước cài đặt

Đọc hình ảnh và chuyển đổi hình ảnh sang thang độ xám: đọc hình ảnh và chuyển đổi hình ảnh sang định dạng thang độ xám. Chuyển đổi hình ảnh sang thang độ xám là rất quan trọng vì nó chuẩn bị hình ảnh cho bước tiếp theo. Việc chuyển đổi hình ảnh sang hình ảnh thang độ xám kênh đơn là rất quan trọng đối với việc tạo ngưỡng, do đó điều này cần thiết để thuật toán phát hiện đường viền hoạt động bình thường.

Áp dụng ngưỡng nhị phân hoặc thuật toán tìm cạnh canny: trong khi tìm đường bao trước tiên luôn áp dụng ngưỡng nhị phân hoặc phát hiện cạnh Canny cho hình ảnh thang độ xám. Điều này sẽ chuyển đổi hình ảnh sang màu đen và trắng, làm nổi bật các đối tượng quan tâm để giúp thuật toán phát hiện đường viền trở nên dễ dàng. Thresholding biến đường viền của đối tượng trong ảnh thành màu trắng hoàn toàn, với tất cả các pixel có cùng cường độ. Thuật toán hiện có thể phát hiện đường viền của các đối tượng từ các pixel màu trắng này. Các pixel có giá trị 0, được coi là pixel nền và bị bỏ qua.

Tìm các đường viền: sử dụng hàm `findContours()` trong OpenCV để phát hiện các đường viền trong ảnh

Vẽ các đường viền trên ảnh RGB gốc: sau khi các đường viền đã được xác định, sử dụng hàm `drawContours()` chức năng để phủ các đường viền lên hình ảnh RGB ban đầu

3.5 Ưu điểm và nhược điểm của phương pháp Contour detection

3.5.1 Ưu điểm

Thuật toán phát hiện đường viền hoạt động tốt khi ảnh có nền tối và các đối tượng quan tâm được xác định rõ ràng.

3.5.2 Nhược điểm

Thuật toán hoạt động không tốt đối với các ảnh đầu vào với nhiều đối tượng được quan tâm cùng với các cường độ pixel với nhau.

4 Image Segmentation using Color Space

4.1 Khái niệm

Không gian màu chính là phương pháp định tính màu sắc được thiết lập công thức một cách khoa học. Hệ thống không gian màu cho phép mỗi màu được xác định theo số học. Bằng cách đó ta có thể chọn và lặp lại những màu đó thật chính xác.

Một số không gian màu thường gặp:

- RGB: red(đỏ), green(xanh lục), blue(xanh lam). Không gian màu khá gần màu mắt người. Sử dụng phổ biến máy tính, máy ảnh, điện thoại.
- CMYK: cyan(Lục lam), magenta(Đỏ tươi), yellow(Vàng), Key(Đen). Không gian màu này dùng cho in ấn.
- HSV and HSL: hue(màu sắc), saturation(Độ bão hòa), brightness/luminance(Độ sáng/Độ chói).

4.2 Ý tưởng

Các vật trong ảnh được biểu diễn bằng những màu sắc nếu ta có thể lọc ra những màu sắc đó thì ta có thể lấy được vật thể.



Figure 16: Ảnh đối tượng cần phân đoạn

Như hình ảnh trên ta thấy con cá hề được biểu diễn bởi hai màu chủ đạo là:

Cam và Trắng, các đối tượng khác con cá đều không có hai màu này. Nếu ta có thể lấy ra hai màu đó thì ta có thể lấy được con cá.

4.3 Các bước tiến hành

Đầu tiên ta phải chuyển đổi bức hình thành không gian màu phù hợp để có thể lấy ra màu của đối tượng.

Sau đó ta chỉ cần chọn ra khoảng màu của đối tượng.

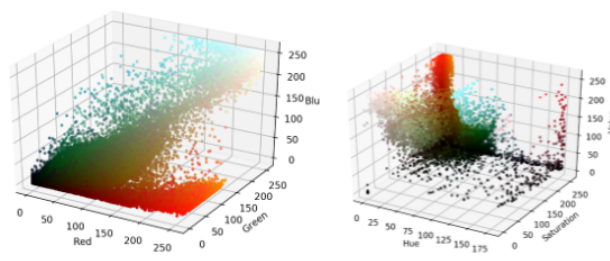


Figure 17: Không gian màu của ảnh

Ở ảnh trên ảnh bên trái biểu diễn cho không gian màu RGB, ảnh bên phải biểu diễn cho không gian màu HSV.

Ta có thể thấy ở hình ảnh bên trái phân phối màu kẻ nhau khó có thể lấy từng màu ra. Nhưng ở hình ảnh bên phải phân phối màu đã có sự phân biệt và dễ dàng lấy ra.

Để áp dụng phương pháp này ta có thể sử dụng 2 hàm cơ bản trong opencv:

- `cv2.cvtColor()`: để chuyển đổi không gian màu.
- `cv2.inRange()`: để tạo mask bằng vùng màu đã chọn

Như hình ảnh (figure 18) ta có thể lấy ra được phần màu cam của con cá, tương tự ta cũng có thể lấy được phần màu trắng của con cá và kết hợp chúng lại được kết quả là hình ảnh (figure 18).

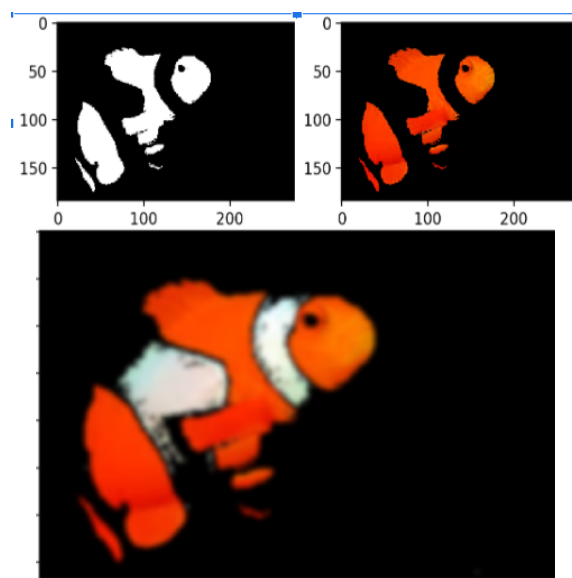


Figure 18: Ảnh áp dụng opencv để tách các không gian màu

4.4 Ưu và nhược điểm của phương pháp

4.4.1 Ưu điểm

Phương pháp dễ cài đặt và có thể dễ dàng chọn ra khoảng màu cần chọn.

4.4.2 Nhược điểm

Nếu như trong màu của đối tượng cần chọn có màu của các đối tượng khác trong hình thì sẽ xảy ra hai trường hợp. Một là lấy cả màu của đối tượng cần chọn và những đối tượng khác. Hai là bỏ cả màu của đối tượng cần chọn và những đối tượng khác. Cho dù gặp phải trường hợp nào cũng cho ra kết quả ta không mong muốn.

Nếu như đối tượng được biểu diễn bằng quá nhiều màu sắc. Thì ta cũng tốn rất nhiều thời gian để có thể lấy ra được đối tượng.

Do phương pháp này phụ thuộc vào màu sắc của đối tượng nên cũng phụ thuộc vào các yếu tố tạo thành nên bức ảnh như: Điều kiện chụp ảnh, thiếu bị chụp ảnh.

5 Image Segmentation using Unet

5.1 Giới thiệu mạng Unet

Unet là một kiến trúc được phát triển bởi Olaf Ronneberger và các cộng sự phát triển nhằm phân vùng các cấu trúc nơ ron thần kinh trong não người. Kiến trúc này lần đầu áp dụng đã dành được chiến thắng trong cuộc thi EM segmentation challenge at ISBI 2012.

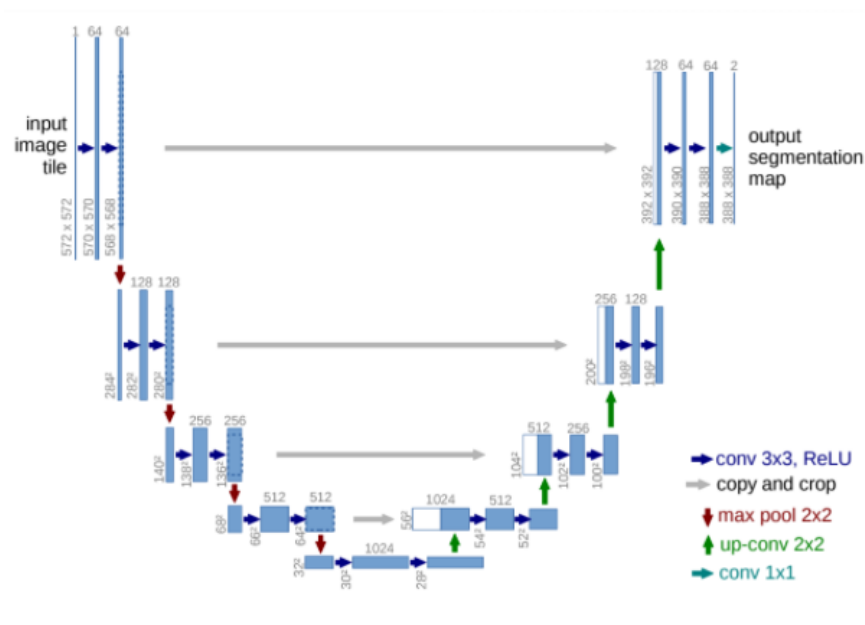


Figure 19: Mạng Unet

Có hai lý do chúng em chọn mạng U-Net để trình bày trong phần báo cáo

của mình là:

- Mạng U-Net được ra đời để phân đoạn hình ảnh y sinh. Điều đó giúp chúng em dễ dàng hình dung được ứng dụng của phân đoạn ảnh trong cuộc sống.
- Mạng U-Net có cấu trúc đơn giản dễ dàng hiểu cách hoạt động với những người chưa biết hoặc mới biết về mạng nơ-ron trong thị giác máy tính.

5.2 Cách thức hoạt động của mạng Unet

Ở hình ảnh trên (figure 19) ta có thể dễ dàng thấy phần bên trái và phần bên phải đối xứng nhau và tạo thành hình chữ U.

Ở phần bên trái người ta gọi là phần thu hẹp (contraction) có nhiệm vụ trích xuất đặc trưng của ảnh ta thu được các khối đặc trưng.

Ở phần bên phải người ta gọi là phần mở rộng (expansion) có các layer đối xứng với phần thu hẹp. Ở phần này ta sử dụng một kỹ thuật gọi là mạng giải chập có shape của các layer tăng dần lên. Mạng giải chập giúp giải mã các khối đặc trưng. Cuối cùng ta thu được một ảnh mask đánh dấu nhãn dự báo của từng pixel. Trong bài báo cáo này chúng em sử dụng Conv2DTranspose của Keras.

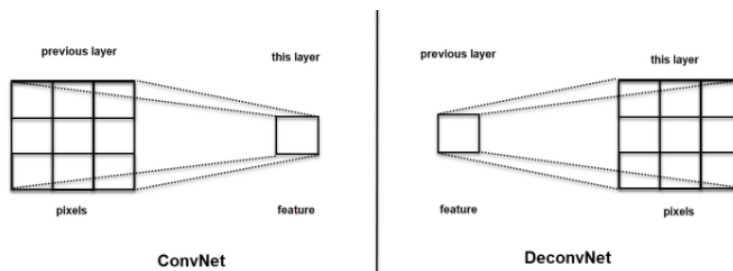


Figure 20: Mạng Unet

Hình bên trái figure 20 mô tả từ ảnh là trích xuất ra một khối đặc trưng. Hình bên phải figure 20 mô tả từ một khối đặc trưng ta thu được ảnh.

5.3 Bộ dữ liệu áp dụng mạng Unet

Trong phần mạng U-Net này chúng em sử dụng 2 bộ dữ liệu để đánh giá mạng U-Net.

Bộ dữ liệu 1: Là bộ dữ liệu trong lĩnh vực y sinh. Chúng em lấy từ Kaggle có tên là “2018 Data Science Bowl”. Bộ dữ liệu này chứa các ảnh phân đoạn hạt nhân tế bào. Hình ảnh được thu ở nhiều điều kiện khác nhau và khác nhau về loại tế bào, độ phóng đại và phương thức hình ảnh (trường sáng so với huỳnh quang (brightfield vs. fluorescence)).

Bộ dữ liệu 2: Có tên là “COCO-2017”. Chúng em lấy bộ dữ liệu này từ Open Images Dataset của Google. Bộ dữ liệu này có một số lượng lớn bức ảnh phục vụ cho bài toán object detection, và các bức ảnh đều được phân đoạn. Trong phần trình bày của chúng em, chúng em chọn đối phân đoạn là “chó” nên chúng em chỉ lấy những bức ảnh có đối tượng “chó” trong bộ dữ liệu “COCO-2017”.

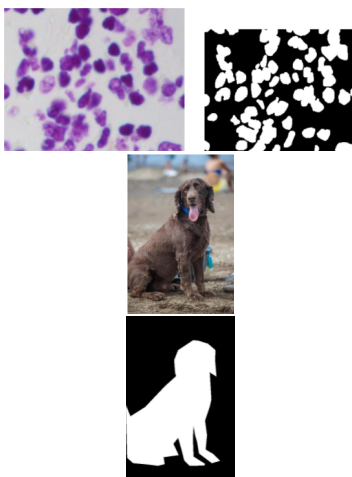


Figure 21: Một số hình ảnh về bộ dữ liệu

Bộ dữ liệu 1 gồm 670 ảnh và 670 ảnh mask chứa phân đoạn nhân tế bào kèm theo.

Bộ dữ liệu 2 gồm 4000 hình ảnh kèm theo 4000 ảnh mask phân đoạn đối tượng chó.

5.4 Kết quả thực nghiệm

Đối với bộ dữ liệu 1 áp dụng Unet

Input	Batch size	epochs	IoU
(None,128,128,3)	16	25	0.839
(None,128,128,3)	16	50	0.835
(None,128,128,3)	16	75	0.860

Table 1: Bảng kết quả thực nghiệm với bộ dữ liệu 1

Đánh giá trên tập test đạt được chỉ số IoU cao nhất: 0.860. Chúng em cho là khá tốt. Dưới đây là kết quả của một bức ảnh được phân đoạn.

Đối với bộ dữ liệu 2 áp dụng Unet

Input	Batch size	epochs	IoU
(None,128,128,3)	16	25	0.227
(None,128,128,3)	16	50	0.247
(None,128,128,3)	16	75	0.241

Table 2: Bảng kết quả thực nghiệm với bộ dữ liệu 2

Đánh giá trên tập test đạt được chỉ số IoU cao nhất: 0.247. Rất thấp so với bộ dữ liệu 1.

5.5 Nhận xét

Qua quan sát hai bộ dữ liệu em đưa ra nhận xét

Bộ dữ liệu 1:

- Các đối tượng được phân đoạn thì nổi bật trong ảnh.
- Các cạnh trong bức ảnh chủ yếu thuộc về các đối tượng.
- Hướng chụp các bức ảnh giống nhau.

- Màu sắc của các bức ảnh chủ yếu thuộc thang màu sáng và tính huỳnh quang (từ đậm đến nhạt) màu sắc đơn giản.
- Bố cục của các bức ảnh gần giống nhau. Nên khi rút trích đặc trưng và tính toán, model sẽ phù hợp với các ảnh trong bộ dữ liệu 1 và đạt kết quả chính xác cao.

Bộ dữ liệu 2:

- Màu sắc đa dạng
- Có nhiều vật thể trong ảnh dẫn đến nhiều cạnh không thuộc về đối tượng cần trích xuất đặc trưng gây khó khăn trong việc tính toán. Nhưng vật thể không thuộc về đối tượng cần phân đoạn là những vật thể gây nhiễu
- Góc chụp khác nhau.

Để phân đoạn tốt bộ dữ liệu thứ 2 chúng em đưa ra giải pháp là giảm thiểu yếu tố gây nhiễu bằng cách.

Đầu tiên là phát hiện vị trí đối tượng và đóng bounding box đối tượng.

Sau đó là phân đoạn đối tượng trong bounding box.

Điều này dẫn đến các yếu tố gây nhiễu trong bounding box sẽ ít hơn toàn bộ bức ảnh và kết quả sẽ được cải thiện. Phương pháp này tương tự như ý tưởng mạng Mask-rcnn nên nó có khả năng thực hiện được.

6 Tài liệu tham khảo

1. Thuật toán CNN, từ <https://topdev.vn/blog/thuat-toan-cnn-convolutional-neural-network/>
2. Image Segmentation, từ <https://phamdinhkhanh.github.io/2020/06/10/ImageSegmentation>
3. Semantic Segmentation using Unet, từ https://www.youtube.com/watch?v=68HR_eyzk00&list=RDCMUC34rW-HtPJulxr5wp2Xa04w&start_radio=1&t=1149s
4. Contour Detection using OpenCV, từ <https://learnopencv.com/contour-detection-using-opencv/>
5. Global threshold, từ <https://www.sciencedirect.com/topics/engineering/global-thresholding>