

TRƯỜNG ĐẠI HỌC CÔNG THƯỜNG TP. HCM
KHOA CÔNG NGHỆ THÔNG TIN



BÁO CÁO MÔN: XỬ LÝ ẢNH
ĐỀ TÀI: PHÂN VÙNG ẢNH

THÀNH PHỐ HỒ CHÍ MINH, NĂM 2024

TRƯỜNG ĐẠI HỌC CÔNG THƯỜNG TP. HCM
KHOA CÔNG NGHỆ THÔNG TIN



BÁO CÁO MÔN: XỬ LÝ ẢNH
ĐỀ TÀI: PHÂN VÙNG ẢNH

Lớp danh nghĩa: 12DHTH14

TKB chính thức: Thứ 2 – Tiết 4_6

GVHD: Trần Đình Toàn

Sinh viên thực hiện:

Phạm Hồ Đăng Huy

THÀNH PHỐ HỒ CHÍ MINH, NĂM 2024

MSSV	Họ Tên	Nhiệm Vụ Hàng Tuần	Kết Quả	GV đánh giá
2001215828	Phạm Hồ Đăng Huy	Tìm hiểu và cài đặt	Hoàn Thành	

LỜI CẢM ƠN

Lời nói đầu tiên cho phép em – Phạm Hồ Đăng Huy cùng các cộng sự gửi lời cảm ơn đến thầy Trần Đình Toàn, hiện đang là giảng viên - người hướng dẫn chúng em để hoàn thiện đề tài báo cáo này.

Đề án này được hoàn thành dưới sự hướng dẫn một cách chi tiết và khoa học của thầy Trần Đình Toàn. Nhóm chúng em xin được bày tỏ lòng biết ơn sâu sắc đến thầy Trần Đình Toàn – người đã tận tình giúp nhóm chúng tôi nâng cao kiến thức và tác phong làm việc bằng tất cả sự mẫu mực của một người giảng viên và tinh thần trách nhiệm của người làm khoa học. Đồng thời, thầy cũng là người đã hỗ trợ nhóm chúng tôi và cho chúng tôi nhiều ý kiến đóng góp quý giá để giúp nhóm có thể hoàn thành luận án một cách hoàn thiện nhất.

Kế đến, nhóm chúng em cũng xin cảm ơn quý thầy cô giáo trường Đại Học Công Thương TP. Hồ Chí Minh đã giảng dạy và truyền đạt những kiến thức, kinh nghiệm và kỹ năng nền tảng cốt lõi vô cùng bổ ích để cho nhóm em làm đề án hiệu quả.

Hơn thế nữa, nhóm chúng em xin gửi lời cảm ơn chân thành đến Ban giám hiệu trường Đại Học Công Thương TP. Hồ Chí Minh đã giúp đỡ và tạo mọi điều kiện thuận lợi cho nhóm có việc học tập và nghiên cứu của nhóm.

Nhóm em xin chúc thầy luôn luôn mạnh khỏe và hạnh phúc!

Nhóm Trưởng

(Ký tên)

Phạm Hồ Đăng Huy

MỤC LỤC HÌNH ẢNH

Hình 1. Phân biệt giữa Object Detection và Instance Segmentation.....	2
Hình 2. Các dạng Image Segmentation	3
Hình 3. Ứng dụng trong y tế.....	4
Hình 4. Ứng dụng với xe tự hành	4
Hình 5. Ứng dụng với định vị vệ tinh.....	5
Hình 6. Ví dụ sử dụng Hough để tìm đường thẳng trong ảnh.....	7
Hình 7. Các loại đường biên phổ biến.....	7
Hình 8. Các dạng đường biên trong xử lí ảnh trong y học	8
Hình 9. Xử lí ảnh toàn cục bằng Hough	11
Hình 10. Minh họa ngưỡng toàn cục	12
Hình 11. Minh họa thuật toán Otsu	13
Hình 12. Minh họa ngưỡng thích nghi	16
Hình 13. Minh họa kết hợp giữa ngưỡng toàn cục và thích nghi tối ưu	17
Hình 14. Minh họa khi sử dụng phân vùng ảnh dựa trên nhiều biến	17
Hình 15. Phân vùng ảnh dựa trên miền bằng cách rút trích đặc trưng	18
Hình 16. Minh họa thuật toán K-means Clustering.....	18
Hình 17. Minh họa các hàm trong OpenCV	20

MỤC LỤC

LỜI CẢM ƠN.....	ii
MỤC LỤC HÌNH ẢNH.....	iii
MỤC LỤC.....	iv
CHƯƠNG I. GIỚI THIỆU	1
1. Lý do chọn đề tài	1
2. Mục tiêu của báo cáo	1
CHƯƠNG II. PHÂN VÙNG ẢNH	2
1. Kiến thức cơ bản	2
1.1. Khái niệm Image Segmentation	2
1.2. Các dạng Image Segmentation khác nhau.....	3
1.3. Ứng dụng của Image Segmentation	3
2. Phân vùng ảnh dựa trên biên (Edge Based Segmentation).....	6
2.1. Phát hiện điểm (Point detection)	6
2.2. Phát hiện đường thẳng (Line detection)	6
2.3. Phát hiện cạnh (Edge detection)	7
2.4. Xử lý toàn cục bằng biến đổi Hough (Global processing using Hough transform)	8
3. Phân vùng dựa trên ngưỡng (Threshold Based Segmentation)	11
3.1. Khái niệm	11
3.2. Ưu điểm và nhược điểm	11
3.3. Ngưỡng toàn cục (Global thresholding)	12
3.4. Ngưỡng thích nghi (Adaptive thresholding)	15

3.5. Ngưỡng tự động.....	16
3.6. Ngưỡng toàn cục và thích nghi tối ưu (Global and adaptive optimal thresholding).....	17
3.7. Căn cứ dựa trên nhiều biến (Multivariable-based thresholding):.....	17
4. Phân vùng ảnh dựa trên miền (Region Based Segmentation)	18
4.1. Phân vùng dựa trên đặc trưng và thuộc tính:.....	18
4.2. Phương pháp k-means clustering	18
4.3. Phân vùng dựa trên đồ thị (Graph-based segmentation):	19
4.4. Phương pháp dựa trên mô hình (Model-based segmentation):	19
4.5. Phương pháp dựa trên học máy (Machine learning-based segmentation): ...	19
5. Các hàm tách biên của OpenCV	19
5.1. Canny Edge Detection:.....	20
5.2. Sobel Edge Detection:	20
5.3. Laplacian Edge Detection:	21
5.4. Scharr Edge Detection:.....	21
5.5. Công cụ phát hiện biên (Edge Detection Tool):.....	21
CHƯƠNG III. KẾT LUẬN.....	22
TÀI LIỆU THAM KHẢO	23

CHƯƠNG I. GIỚI THIỆU

Trong thế giới ngày nay, phân vùng ảnh đã trở thành một lĩnh vực nghiên cứu quan trọng trong xử lý ảnh. Phân vùng ảnh là quá trình chia hình ảnh thành các phần nhỏ gọi là segments, với mục tiêu là làm giảm độ phức tạp của hình ảnh và tạo ra các vùng có ý nghĩa trong hình ảnh. Phân vùng ảnh đóng vai trò quan trọng trong nhiều ứng dụng thực tiễn như nhận diện đối tượng, xử lý hình ảnh y học, và tự động hóa công nghiệp.

1. Lý do chọn đề tài

Trong lĩnh vực xử lý ảnh, phân vùng ảnh là một chủ đề nghiên cứu quan trọng nhằm tạo ra các vùng đồng nhất trong hình ảnh, giúp cho quá trình xử lý và phân tích hình ảnh trở nên hiệu quả hơn. Hiểu biết về các phương pháp phân vùng ảnh không chỉ giúp cải thiện hiệu suất của các ứng dụng xử lý ảnh mà còn mở ra nhiều cơ hội mới trong nghiên cứu và phát triển công nghệ.

2. Mục tiêu của báo cáo

Mục tiêu của báo cáo này là cung cấp một cái nhìn tổng quan về phân vùng ảnh, bao gồm các phương pháp, kỹ thuật và ứng dụng. Chúng tôi sẽ giải thích cách các phương pháp phân vùng ảnh được áp dụng và biết cách chúng đóng vai trò quan trọng trong thực tiễn. Bằng cách này, chúng tôi hy vọng sẽ giúp độc giả hiểu rõ hơn về tầm quan trọng của phân vùng ảnh và ứng dụng của nó trong các lĩnh vực khác nhau.

CHƯƠNG II. PHÂN VÙNG ẢNH

1. Kiến thức cơ bản

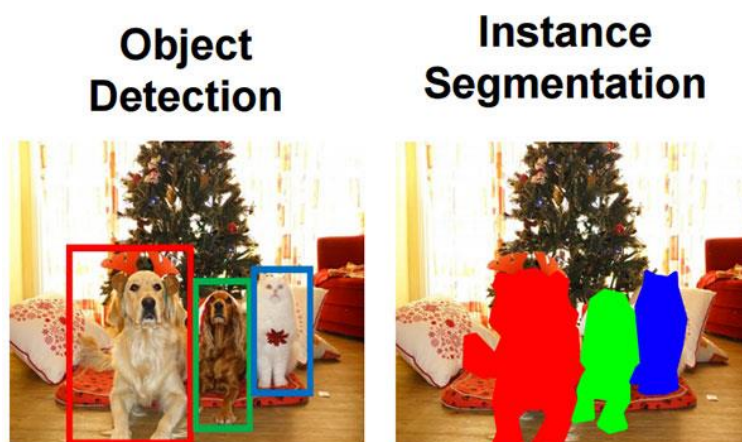
1.1. Khái niệm Image Segmentation

Phân vùng ảnh (Image Segmentation) là một phương pháp mà trong đó, hình ảnh kỹ thuật số được chia thành nhiều nhóm con khác nhau được gọi là segments. Mục tiêu của phân vùng ảnh là làm giảm độ phức tạp của hình ảnh, phân tách ảnh thành từng vùng, mỗi vùng sẽ tương ứng với một đối tượng, giúp cho quá trình xử lý hoặc phân tích hình ảnh sau đó trở nên đơn giản hơn. Nói một cách dễ hiểu, phân vùng là dán nhãn cho từng pixel. Tất cả các yếu tố hình ảnh hoặc pixel thuộc cùng một danh mục sẽ có chung một nhãn. Ví dụ: Đối với bài toán phát hiện đối tượng, thay vì xử lý toàn bộ hình ảnh, máy có thể chỉ thực hiện trên một đoạn được chọn bởi thuật toán phân vùng. Điều này sẽ ngăn máy xử lý toàn bộ hình ảnh, do đó làm giảm thời gian suy luận.



⇒ Thay vì xử lý toàn bộ hình ảnh, máy chỉ thực hiện trên một đoạn được chọn bởi thuật toán phân vùng.

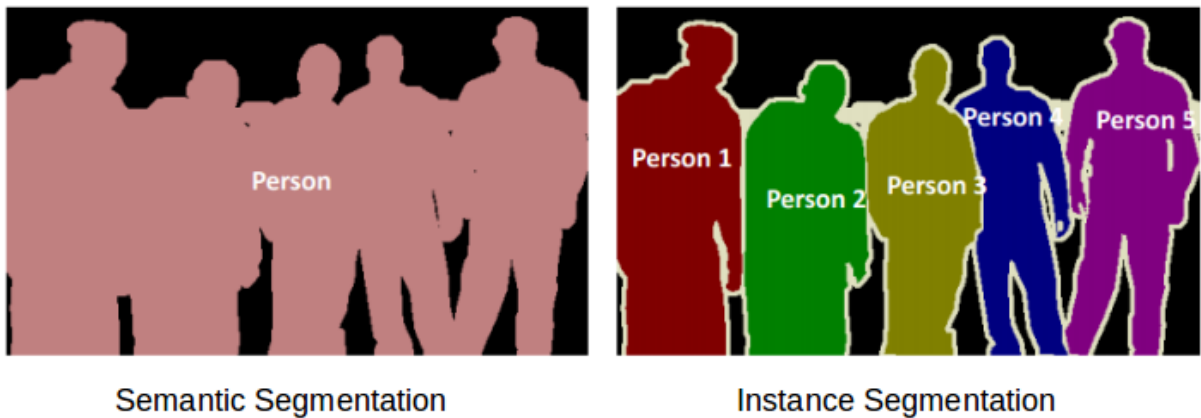
Image Segmentation cũng có chung mục tiêu như object detection là phát hiện ra vùng ảnh chứa vật thể và gán nhãn phù hợp cho chúng. Tuy nhiên tiêu chuẩn về độ chính xác của Image Segmentation ở mức cao hơn so với Object Detection khi nó yêu cầu nhãn dự báo đúng tới từng pixel.



Hình 1. Phân biệt giữa Object Detection và Instance Segmentation

1.2. Các dạng Image Segmentation khác nhau

Trong lĩnh vực phân vùng ảnh, chúng ta thường gặp hai bài toán chính là Semantic Segmentation và Instance Segmentation:



Hình 2. Các dạng Image Segmentation

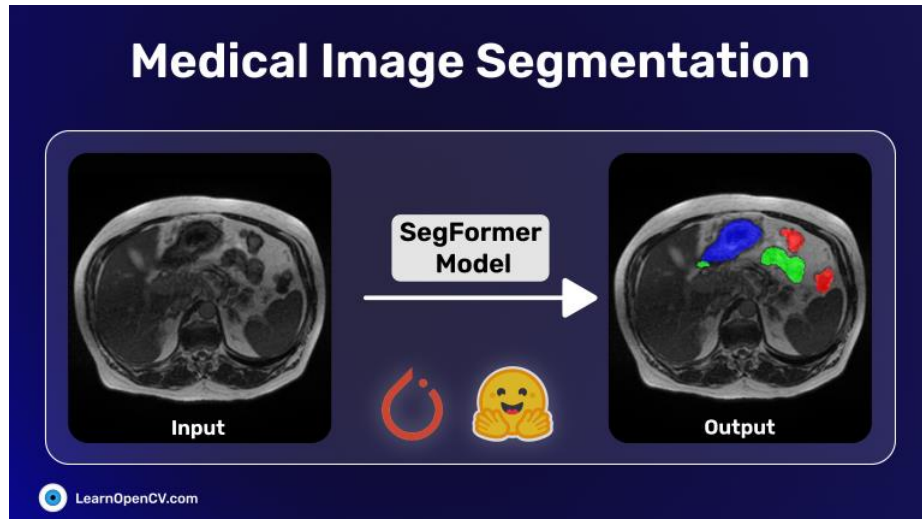
- **Semantic Segmentation:** là quá trình phân đoạn các vùng ảnh thành các nhóm dựa trên các nhãn khác nhau mà không phân biệt sự khác biệt giữa các đối tượng trong từng nhãn. Về cơ bản, chúng ta nhận biết và phân loại các vùng ảnh theo các đặc điểm chung của chúng. Ví dụ, trong một bức ảnh có nhiều người và nền, Semantic Segmentation sẽ chỉ cho chúng ta biết pixel nào thuộc về người và pixel nào thuộc về background, mà không phân biệt được từng người cụ thể.

- **Instance Segmentation:** là quá trình phân đoạn các vùng ảnh chi tiết đến từng đối tượng trong mỗi nhãn. Trong bài toán này, chúng ta không chỉ nhận biết các vùng ảnh theo nhãn mà còn phân biệt được mỗi đối tượng riêng lẻ trong từng nhãn. Ví dụ, đối với một bức ảnh chứa nhiều người, Instance Segmentation sẽ phân chia chi tiết từng người thành các đối tượng riêng biệt, giúp ta xác định được số lượng và vị trí của mỗi người trong hình ảnh.

1.3. Ứng dụng của Image Segmentation

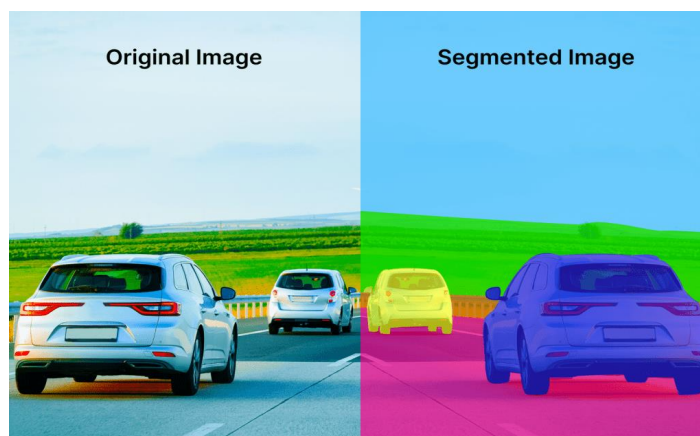
Image Segmentation đóng vai trò không thể phủ nhận trong nhiều lĩnh vực ứng dụng như y học, xe tự hành và xử lý ảnh vệ tinh.

- Y học: Trong lĩnh vực y học, các thuật toán Image Segmentation được áp dụng để hỗ trợ việc chẩn đoán các khối u từ ảnh x-quang. Sự ưu việt của Image Segmentation không chỉ là khả năng xác định vị trí của các khối u trong ảnh mà còn là khả năng trích xuất hình dạng chi tiết của chúng, đóng vai trò quan trọng trong quá trình chẩn đoán và điều trị.



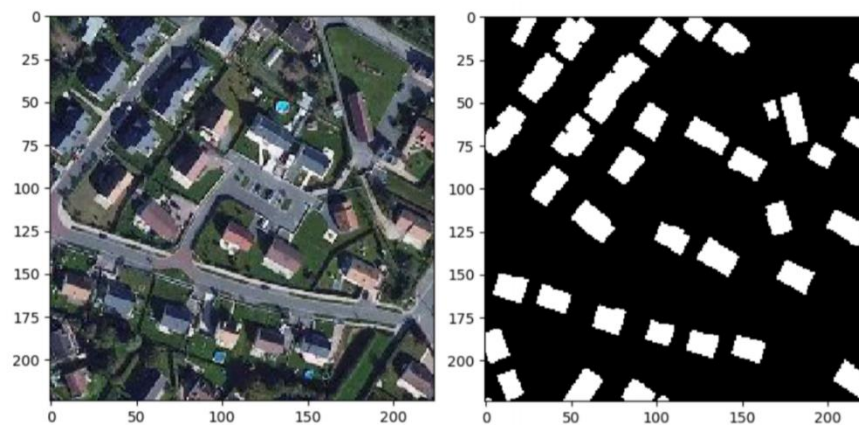
Hình 3. Ứng dụng trong y tế

- Xe tự hành: Trong môi trường động và không ngừng thay đổi của giao thông, hệ thống xe tự hành yêu cầu khả năng nhận diện, xử lý và lập kế hoạch một cách chính xác. Để đảm bảo an toàn và độ tin cậy, các thuật toán Image Segmentation được sử dụng để chính xác xác định các vật thể như người đi bộ, đèn giao thông, biển báo, và dấu vạch đường.



Hình 4. Ứng dụng với xe tự hành

- Xử lý ảnh vệ tinh: Với việc thu thập liên tục hình ảnh từ các vệ tinh quan sát trái đất, Image Segmentation đóng vai trò quan trọng trong việc phân đoạn hình ảnh thành các phần khác nhau như đường phố, khu dân cư, mặt nước, và cây cỏ. Thông qua quá trình này, các nhà nghiên cứu và quản lý môi trường có thể phân tích và theo dõi các biến đổi môi trường và đánh giá tác động của các hoạt động nhân tạo và tự nhiên.



Hình 5. Ứng dụng với định vị vệ tinh

- Ứng dụng trong nông nghiệp: Image Segmentation cũng có ứng dụng trong việc tăng cường hiệu suất và giảm thiểu sử dụng hóa chất trong nông nghiệp. Hệ thống tự động phun thuốc trừ sâu có khả năng phân biệt diện tích của cỏ và cây trồng dựa trên Image Segmentation, giúp định lượng và chỉ đạo việc sử dụng thuốc trừ sâu một cách hiệu quả và tiết kiệm. Đồng thời, nó cũng giúp giảm thiểu rủi ro cho môi trường và con người.

- Cảnh báo cháy rừng: Trong việc kiểm soát cháy rừng, các hệ thống Image Segmentation có thể xác định chính xác vị trí và phạm vi của các đám cháy từ hình ảnh vệ tinh. Thông qua việc phân đoạn đám cháy, các cơ quan chức năng có thể cung cấp cảnh báo và đưa ra các biện pháp kiểm soát và ứng phó phù hợp.

2. Phân vùng ảnh dựa trên biên (Edge Based Segmentation)

2.1. Phát hiện điểm (Point detection)

Phát hiện điểm nhằm xác định các điểm đặc biệt trong hình ảnh như điểm góc, điểm đỉnh của các hình dạng, hoặc các điểm quan trọng khác bằng cách sử dụng các bộ lọc đạo hàm (gradient) như Laplacian hoặc Sobel để nổi bật các điểm ấy.

Ý tưởng các bước phát hiện điểm dựa trên bộ lọc Laplacian:

- Bước 1: Áp dụng bộ lọc Laplacian vào hình ảnh để thu được $R(x, y)$
- Bước 2: Tạo ảnh nhị phân với ngưỡng

$$g(x, y) = \begin{cases} 1, & \text{if } |R(x, y)| \geq T \\ 0, & \text{otherwise} \end{cases}$$

Trong đó T: là ngưỡng không âm

-1	-1	-1
-1	8	-1
-1	-1	-1

Bộ lọc Laplacian

2.2. Phát hiện đường thẳng (Line detection)

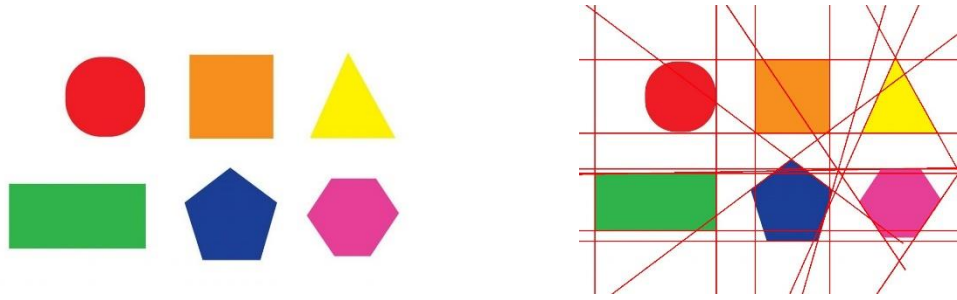
Phát hiện đường thẳng nhằm xác định các đường thẳng trong hình ảnh.

Chúng ta sẽ tạo một mảng (mặt nạ) dựa theo nhu cầu để phát hiện các sự thay đổi trên đường thẳng.

Ví dụ: Với mảng đường thẳng nằm ngang thì với các giá trị điểm ảnh sẽ nổi lên khi nằm trên giá trị này.

Ngang			+ 45 độ			Dọc			- 45 độ		
-1	-1	-1	-1	-1	2	-1	2	-1	2	-1	-1
2	2	2	-1	2	-1	-1	2	-1	-1	2	-1
-1	-1	-1	2	-1	-1	-1	2	-1	-1	-1	2

Biến đổi Hough là một phương pháp mạnh mẽ và phổ biến được sử dụng để phát hiện đường thẳng trong ảnh.



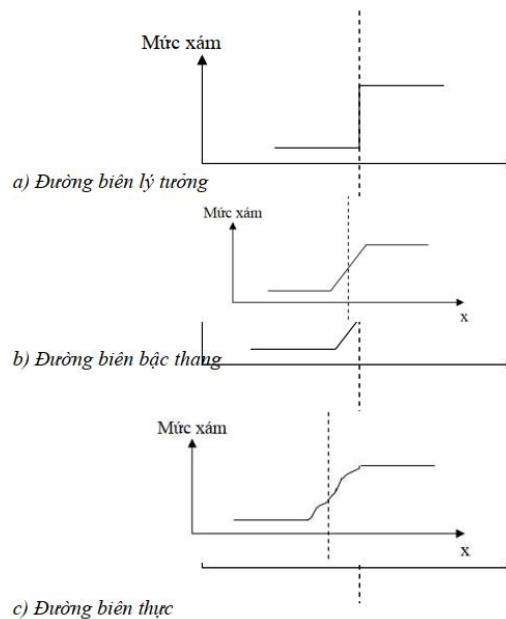
Hình 6. Ví dụ sử dụng Hough để tìm đường thẳng trong ảnh

2.3. Phát hiện cạnh (Edge detection)

Phát hiện cạnh là phương pháp phân đoạn hình ảnh dựa trên sự thay đổi đột ngột trong cường độ. Nó được sử dụng để phát hiện ranh giới hoặc để tìm kích thước hoặc vị trí của một đối tượng trong hình ảnh.

Cạnh sẽ như thế nào?

- Một cạnh là một tập hợp các pixel kết nối nằm trên biên giữa hai khu vực.
- Một cạnh là một khái niệm "địa phương" trong một vùng không gian, do cách nó được xác định, là một ý tưởng toàn cục. Các loại cạnh (biên) thường gặp:



Hình 7. Các loại đường biên phổ biến

- Đường biên lý tưởng (Hình 7 a) được xác định là sự thay đổi trong giá trị cấp xám tại một vị trí cụ thể. Vị trí của đường biên chính là vị trí của sự thay đổi cấp xám.
- Đường biên bậc thang (Hình 7 b) xuất hiện khi sự thay đổi cấp xám lan rộng qua nhiều điểm ảnh. Vị trí của đường biên được coi là vị trí trung tâm giữa cấp xám thấp và cấp xám cao.
- Đường biên thực (Hình 7c): Đây là sự thay đổi cấp xám tại nhiều điểm, nhưng không liên tục.



Hình 8. Các dạng đường biên trong xử lý ảnh trong y học

Các phương pháp như Sobel, Prewitt và Canny thường được sử dụng để phát hiện cạnh.

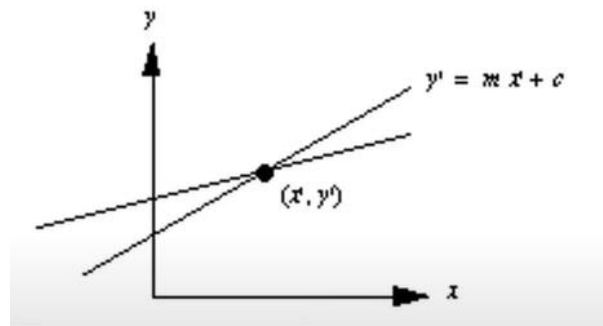
2.4. Xử lý toàn cục bằng biến đổi Hough (Global processing using Hough transform)

Biến đổi Hough là một phương pháp xử lý toàn cục, thường được sử dụng để phát hiện các hình dạng đặc biệt như đường thẳng, đường tròn trong ảnh.

Hãy giả sử rằng chúng ta đang tìm kiếm các đường thẳng trong ảnh. Nếu chúng ta lấy một điểm (x', y') trên hình ảnh, tất cả các đường thẳng đi qua điểm đó có dạng:

$$y' = mx' + c$$

Với các giá trị khác nhau của m và c .

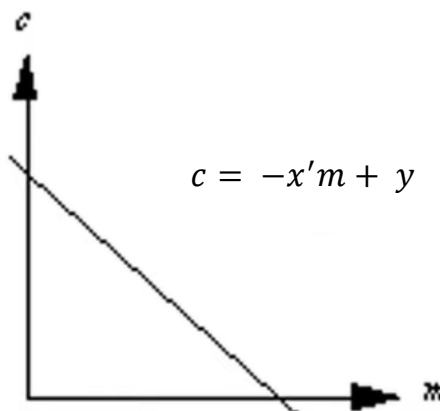


Tuy nhiên, phương trình này cũng có thể được viết lại dưới dạng

$$c = -x'm + y$$

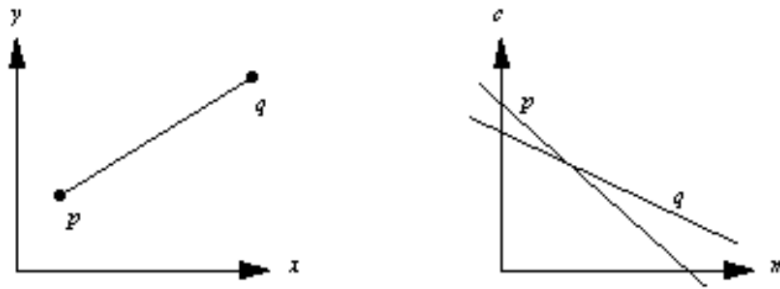
nơi chúng ta giờ xem x' và y' là các hằng số, và m và c là các biến thiên.

Đây là một đường thẳng trên đồ thị của c so với m như được thể hiện trong Hình dưới đây.



Mỗi đường thẳng khác nhau qua điểm (x', y') tương ứng với một trong các điểm trên đường thẳng trong không gian (m, c) .

Xét hai điểm p và q trong không gian (x, y) nằm trên cùng một đường thẳng. Đối với mỗi điểm, tất cả các đường thẳng có thể đi qua nó được biểu diễn bằng một đường thẳng duy nhất trong không gian (m, c) .



Do đó, đường thẳng duy nhất trong không gian (x, y) đi qua cả hai điểm p và q nằm trên giao điểm của hai đường thẳng đại diện cho p và q trong không gian (m, c) , như được thể hiện trong hình. Nếu p và q là các điểm được kết nối bởi một đường trong miền không gian, thì chúng sẽ là các đường chéo trong không gian Hough $(m - c)$.

Đưa điều này một bước xa hơn, tất cả các điểm ảnh nằm trên cùng một đường trong không gian (x, y) được biểu diễn bằng các đường thẳng đi qua một điểm duy nhất trong không gian (m, c) .

Điểm duy nhất mà tất cả chúng đi qua cho các giá trị của m và c trong phương trình của đường thẳng $y = mx + c$.

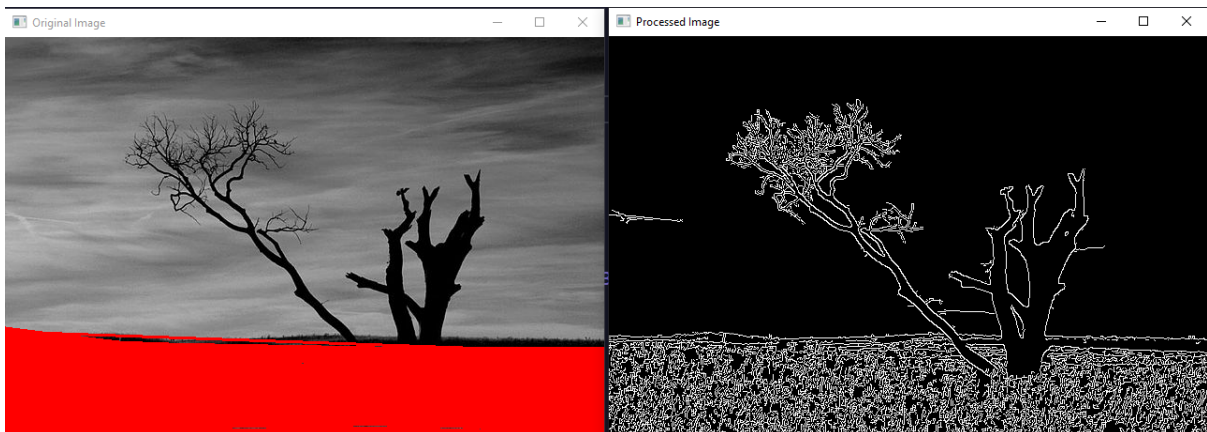
Để phát hiện các đường thẳng trong một hình ảnh, chúng ta thực hiện như sau:

- Chuyển không gian (m, c) thành một mảng hai chiều A với các bước phù hợp của m và c .
- Khởi tạo tất cả các phần tử của $A(m, c)$ bằng không.
- Đối với mỗi điểm ảnh (x', y') nằm trên một biên nào đó trong hình ảnh, chúng ta thêm 1 vào tất cả các phần tử của $A(m, c)$ mà chỉ số m và c thỏa mãn $y' = mx' + c$.
- Tìm kiếm các phần tử của $A(m, c)$ có giá trị lớn. Mỗi phần tử được tìm thấy tương ứng với một đường trong hình ảnh gốc.

Một thuộc tính hữu ích của biến đổi Hough là các điểm ảnh nằm trên đường không cần phải liên kề nhau.

❖ Các bước thực hiện biến đổi Hough:

1. Tải hình ảnh.
2. Tìm các biên của hình ảnh bằng bất kỳ công cụ phát hiện biên nào.
3. Quy hóa không gian tham số P.
4. Lặp lại các bước sau cho tất cả các điểm ảnh của hình ảnh: nếu pixel là một điểm biên, thì (a) $c = (-x)m + y$ hoặc tính ρ (b) $P(c, m) = P(c, m) + 1$ hoặc tăng vị trí trong P.
5. Hiển thị không gian Hough.
6. Tìm các cực đại cục bộ trong không gian tham số.
7. Vẽ đường bằng cách sử dụng các cực đại cục bộ
 - Vấn đề chính của thuật toán này là nó không hoạt động cho các đường thẳng dọc, vì chúng có độ dốc vô hướng.
 - Giải pháp là chuyển đổi đường thành tọa độ cực $\rho = x \cos\theta + y \sin\theta$, trong đó θ là góc giữa đường và trục x, và ρ là đường kính.



Hình 9. Xử lý ảnh toàn cục bằng Hough

3. Phân vùng dựa trên ngưỡng (Threshold Based Segmentation)

3.1. Khái niệm

Phân vùng dựa trên ngưỡng là phương pháp phân vùng đơn giản và phổ biến. Đây không phải là cách tiếp cận theo vùng hoặc theo biên mà là tiếp cận mức điểm ảnh. Với phương pháp này thường phải có bước hậu xử lý say đó.

3.2. Ưu điểm và nhược điểm

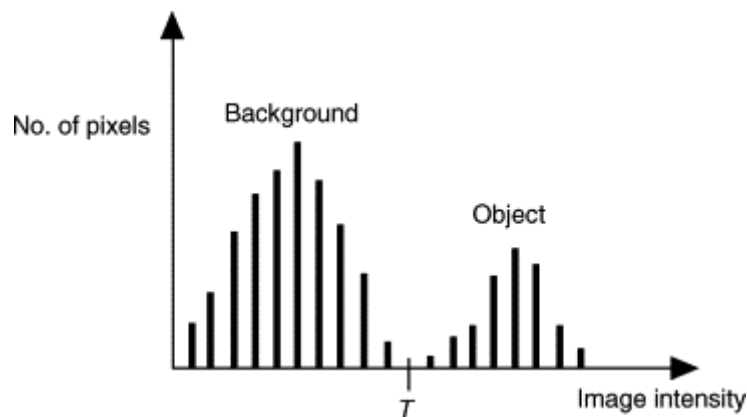
- Ưu điểm: đơn giản, nhanh.
- Nhược điểm: phải biết trước số lớp, các vùng không phù hợp sẽ xuất hiện (không có ràng buộc không gian).

3.3. Ngưỡng toàn cục (Global thresholding)

Ngưỡng toàn cục là phương pháp đơn giản nhất trong đó một ngưỡng cố định được áp dụng cho toàn bộ hình ảnh để phân tách các vùng.

Ý tưởng chính có thể viết dưới dạng như sau: $g(x, y) = \begin{cases} 1, & f(x, y) > T \\ 0, & f(x, y) \leq T \end{cases}$

⇒ Kết quả cuối cùng là ảnh nhị phân → có thể chọn n ngưỡng để ảnh thành n + 1 lớp. Lưu ý, chúng ta phải chọn ngưỡng phù hợp để cho ra kết quả sau khi phân vùng tốt nhất.

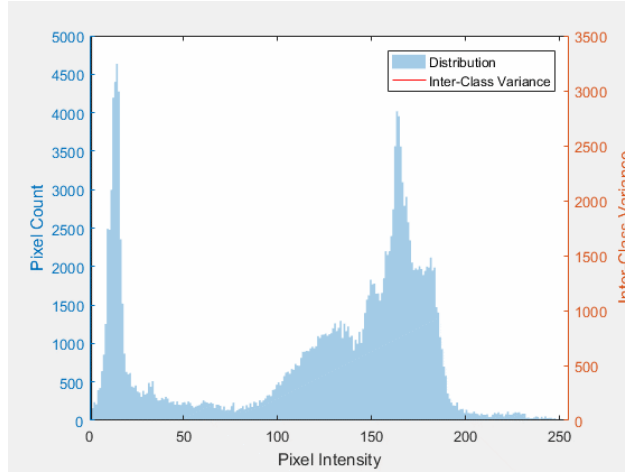


Hình 10. Minh họa ngưỡng toàn cục

Trong hình trên có thể được khi ta chọn ngưỡng T cho một lần xử lý ảnh, ta thấy có sự phân chia rõ ràng giữa nền và các đối tượng.

Các phương pháp như phương pháp Otsu hoặc phương pháp lấy ngưỡng dựa trên histogram thường được sử dụng để chọn ngưỡng tối ưu.

Để rõ hơn thì chúng ta sẽ cần tìm hiểu hơn về thuật toán Otsu, Otsu là tên một nhà nghiên cứu người Nhật đã nghĩ ra ý tưởng cho việc tính ngưỡng một cách tự động dựa vào giá trị điểm ảnh của ảnh đầu vào nhằm thay thế cho việc sử dụng ngưỡng cố định.



Hình 11. Minh họa thuật toán Otsu

Đầu tiên sử dụng lược đồ Histogram biểu diễn tần suất xuất hiện mức xám:

$$p_i = \sum_{i=0}^{L-1} \frac{n_i}{(M \times N)}$$

Trong đó:

- n_i là số lượng điểm ảnh của giá trị i .
- L : 1,2,3, ... 256.
- $p_0 + p_1 + p_2 + \dots + p_{L-1} = 1$

Chọn một ngưỡng $T_k = k$, ($0 < k < L - 1$) để phân ảnh đầu vào thành 2 lớp C1 (tập hợp các điểm ảnh có giá trị $\leq k$) và C2 (tập hợp các điểm ảnh có giá trị lớn hơn k). Tỷ lệ lớp C1 với số lượng điểm ảnh k với tổng số lượng điểm ảnh được ký hiệu $P1(k)$, tương tự C2 ký hiệu là $P2(k)$.

$$P_1 = \sum_{i=0}^k p_i$$

$$P_2(k) = \sum_{i=k+1}^{L-1} p_i = 1 - P_1(k)$$

Sau đó ta tính giá trị trung bình m_1 của lớp C1:

$$m_1(k) = \sum_{i=0}^k iP\left(\frac{i}{C1}\right) = \frac{1}{P_1(k)} \sum_{i=0}^k iP_i$$

Tương tự tính m_2 :

$$m_2(k) = \sum_{i=k+1}^k iP\left(\frac{i}{C2}\right) = \frac{1}{P_2(k)} \sum_{i=k+1}^{L-1} iP_i$$

Theo Otsu, ta sẽ tính ngưỡng k^* mà giá trị tại đó sự chênh lệch giữa hai đoạn (màu nền và màu ký tự) đạt giá trị cực đại, ký hiệu $\sigma_B^2(k^*)$, được tính:

$$\sigma_B^2(k^*) = \text{Max}_{0 \leq k \leq L-1} \sigma_B^2(k)$$

Trong đó σ^B là phương sai hai lớp C1 và C2. Ta có:

$$\begin{aligned} \sigma^B &= P_1(m_1 - m_G)^2 + P_2(m_2 - m_G)^2 \\ &= P_1P_2(m_1 - m_G)^2 \\ &= \frac{(m_GP - m)^2}{P_1(1 - P_1)} \end{aligned}$$

Từ công thức trên ta suy ra:

$$\sigma^B = \frac{(m_GP_1(k) - m(k))^2}{P_1(k)(1 - P_1(k))}$$

Trong đó:

- m_G là giá trị trung bình của ảnh.

$$m_G = \sum_{i=0}^{L-1} iP_i \text{ hoặc } m_G = P_1m_1 + P_2m_2$$

- m_k là giá trị trung bình đến ngưỡng k.

$$m_k = \sum_{i=0}^k iP_i$$

Nếu có nhiều giá trị σ^2 lớn nhất bằng nhau, ta sẽ chọn k có giá trị lớn nhất làm ngưỡng k^* , sau đó ta thực hiện nhị phân biến số theo ngưỡng.

$$g(x, y) = 1 \text{ if } f(x, y) < \text{ và } g(x, y) = 0 \text{ if } f(x, y) \geq$$

Trong đó:

- $g(x, y)$ là ảnh đầu ra.
- $f(x, y)$ là ảnh đầu vào.
- $x = 0, 1, 2, \dots, N - 1$.
- $y = 0, 1, 2, \dots, M - 1$.

⇒ Với các công thức trên cũng là sơ lược về thuật toán Otsu này đối với phân ngưỡng cho ảnh

3.4. Ngưỡng thích nghi (Adaptive thresholding)

Ngưỡng thích nghi là một phương pháp linh hoạt hơn, trong đó ngưỡng được xác định dựa trên các vùng nhỏ trong hình ảnh thay vì trên toàn bộ hình ảnh.

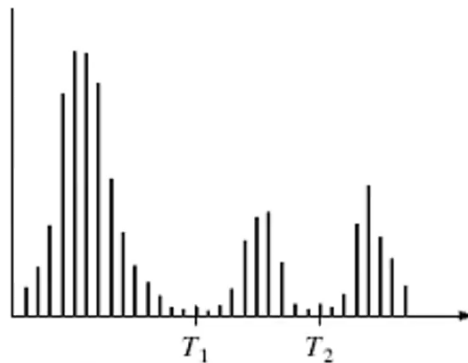
Ý tưởng minh họa, dựa vào n ngưỡng cho phép chia ảnh thành n + 1 lớp ta sẽ có như sau:

- If value(pixel) < threshold_1
then value(pixel) = 0
- If value(pixel) >= threshold_1 && value(pixel) < threshold_2
then value (pixel) = 1
- ...
- If value(pixel) >= threshold_n

then value (pixel) = n

Ý tưởng của ngưỡng thích nghi này sẽ dựa vào n ngưỡng cho phép để chia ảnh thành n + 1 lớp, như hình minh họa ở dưới ta chia thành hai vùng ảnh có thể

viết nó dưới dạng như sau:
$$g(x, y) = \begin{cases} 2, & f(x, y) \geq T_2 \\ 1, & f(x, y) \leq T_2 \wedge f(x, y) > T_1 \\ 0, & f(x, y) < T_1 \end{cases}$$



Hình 12. Minh họa ngưỡng thích nghi

Trong hình trên có thể được khi ta chọn hai ngưỡng T cho xử lí ảnh trên histogram, ta sẽ chia nó thành các vùng đa dạng hơn tùy theo số lượng ngưỡng mà đề bài hoặc nhu cầu mình muốn.

Các phương pháp như ngưỡng thích nghi Gaussian hoặc ngưỡng thích nghi cục bộ được sử dụng để điều chỉnh ngưỡng cho từng phần nhỏ của hình ảnh.

3.5. Ngưỡng tự động

Sau các bước tìm ngưỡng như trên, ta có một vấn đề đặt ra là “Làm thế nào để xác định được giá trị của T (ngưỡng), ta có cách như sau:

- Bảng thử nghiệm
- Giá trị trung bình của mức xám
- Giá trị trung bình giữa max và min
- Giá trị cho phép cân bằng các vùng của phân bố

⇒ Với những cách như trên có thể gây mất thời gian, thì để đơn giản hơn ta có thể thực hiện bằng giải thuật xác định ngưỡng tự động như sau:

- Bước 1: Khởi tạo giá trị T (trung bình, điểm giữa, ...)
- Bước 2: Xác định 2 nhóm điểm ảnh $C1$ nếu $f(x, y) > T$ và $C2$ nếu $f(x, y) \leq T$
- Bước 3: Tính trung bình về mức xám của $C1$ và $C2 \rightarrow \mu_1$ và μ_2
- Bước 4: Tính giá trị mới của $T = \frac{1}{2} (\mu_1 + \mu_2)$
- Bước 5: Lặp lại bước 2 cho đến khi T ổn định

3.6. Ngưỡng toàn cục và thích nghi tối ưu (Global and adaptive optimal thresholding)

Kết hợp cả hai phương pháp, tức là sử dụng ngưỡng toàn cục và ngưỡng thích nghi, có thể mang lại kết quả phân vùng tốt hơn.

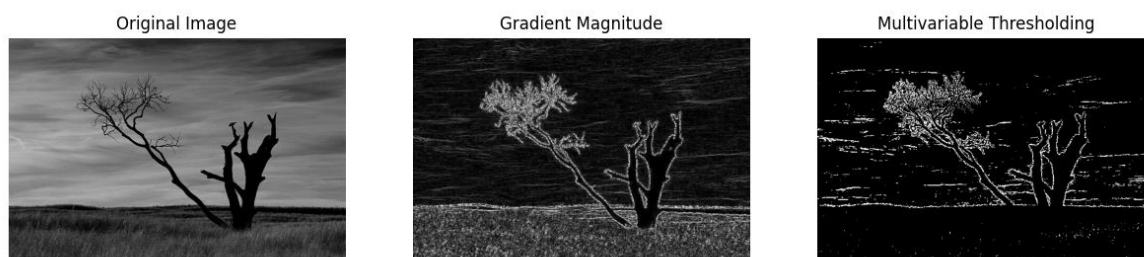
Các phương pháp như kết hợp ngưỡng Otsu và ngưỡng thích nghi cục bộ có thể được áp dụng để đạt được kết quả tối ưu.



Hình 13. Minh họa kết hợp giữa ngưỡng toàn cục và thích nghi tối ưu

3.7. Căn cứ dựa trên nhiều biến (Multivariable-based thresholding):

Các phương pháp này sử dụng nhiều đặc trưng hoặc thuộc tính của hình ảnh để xác định ngưỡng phù hợp.



Hình 14. Minh họa khi sử dụng phân vùng ảnh dựa trên nhiều biến

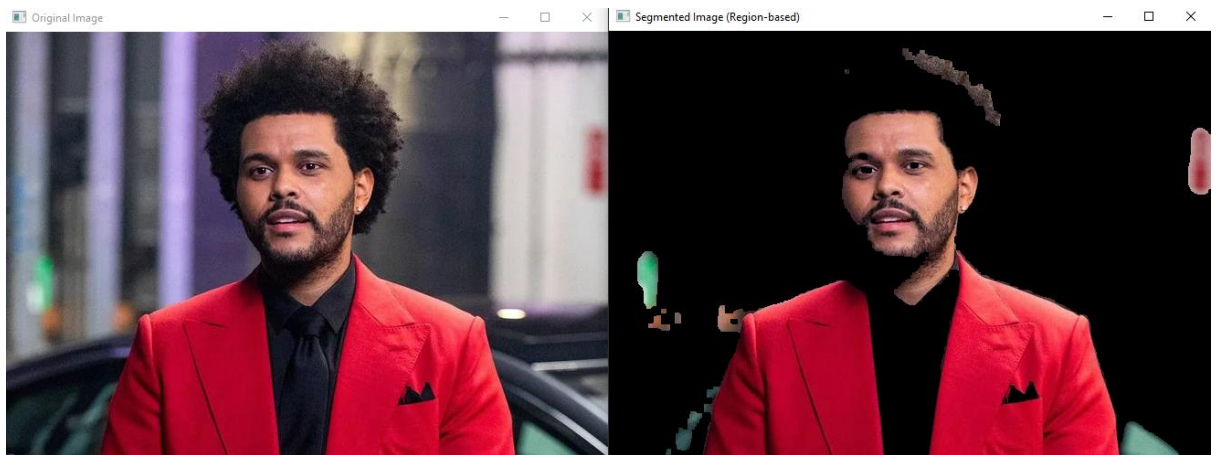
Các phương pháp như phân tích đối tượng (object-based analysis) hoặc phân tích kết cấu (structure-based analysis) có thể được sử dụng để áp dụng ngưỡng dựa trên nhiều biến.

4. Phân vùng ảnh dựa trên miền (Region Based Segmentation)

4.1. Phân vùng dựa trên đặc trưng và thuộc tính:

Phân vùng dựa trên miền thường sử dụng các đặc trưng và thuộc tính của các vùng trong hình ảnh để phân loại chúng.

Các đặc trưng có thể bao gồm màu sắc, độ sáng, cường độ gradient, textural features, hoặc các thuộc tính khác như hình dạng, kích thước và vị trí.

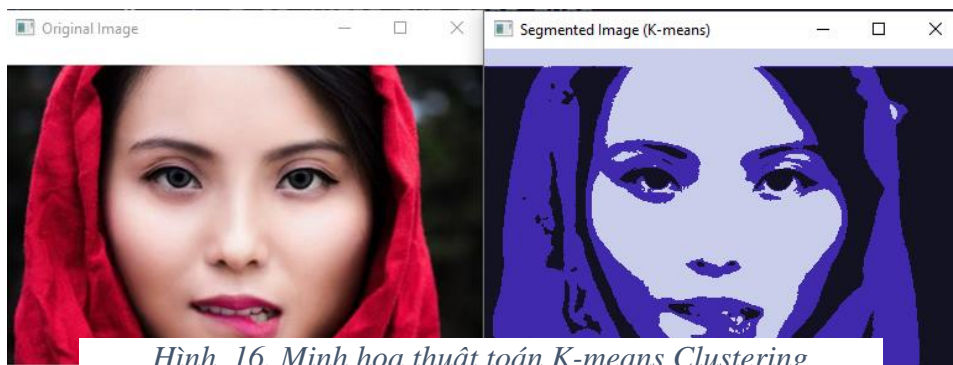


Hình 15. Phân vùng ảnh dựa trên miền bằng cách rút trích đặc trưng

Ví dụ như ảnh trên, khi ta muốn rút trích đối tượng trong khung hình thì tùy theo thuật toán mà muốn nó rút trích với màu sắc hay các đặc trưng khác nhau sẽ cho ra các kết quả khác nhau.

4.2. Phương pháp k-means clustering

Phương pháp này chia hình ảnh thành các vùng (cluster) dựa trên sự tương đồng giữa các điểm ảnh.



Hình 16. Minh họa thuật toán K-means Clustering

K-means clustering là một phương pháp phân cụm phổ biến được sử dụng trong phân vùng dựa trên miền.

4.3. Phân vùng dựa trên đồ thị (Graph-based segmentation):

Các phương pháp này xây dựng một đồ thị từ hình ảnh, trong đó các đỉnh đại diện cho các điểm ảnh và các cạnh đại diện cho mối liên kết giữa chúng.

Sau đó, các thuật toán như min-cut/max-flow hoặc normalized cuts được sử dụng để phân tách hình ảnh thành các miền dựa trên cấu trúc của đồ thị.

4.4. Phương pháp dựa trên mô hình (Model-based segmentation):

Các phương pháp này sử dụng các mô hình hoặc mẫu để mô tả các vùng trong hình ảnh và phân tách chúng dựa trên sự phù hợp với mô hình.

Ví dụ, phương pháp dựa trên Active Shape Models (ASM) hoặc Active Appearance Models (AAM) có thể được sử dụng để phân vùng các vùng khuôn mặt trong hình ảnh.

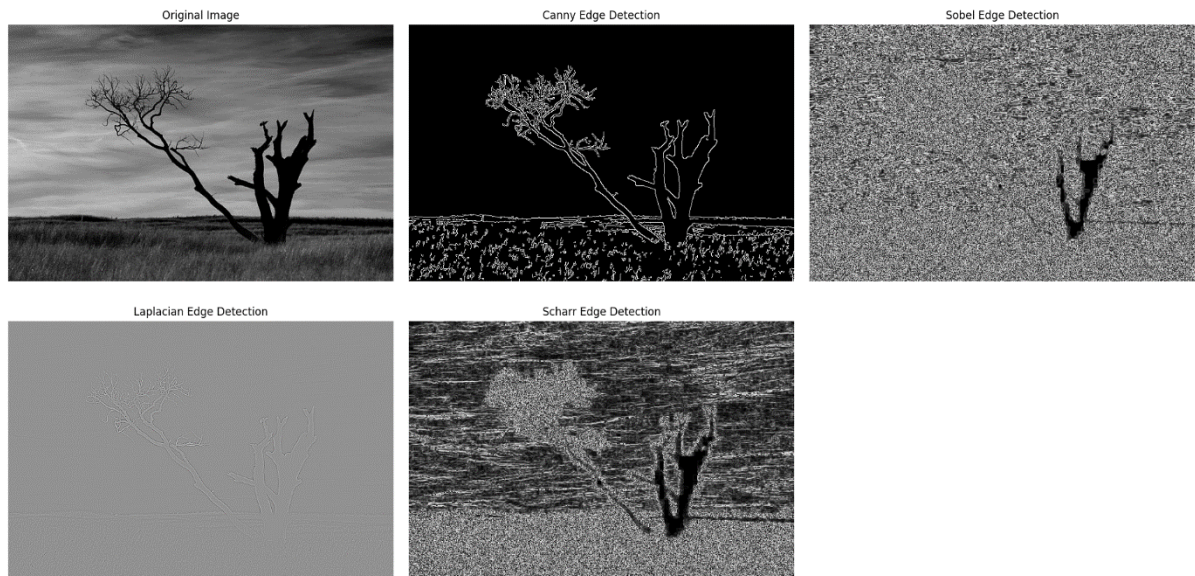
4.5. Phương pháp dựa trên học máy (Machine learning-based segmentation):

Các phương pháp này sử dụng các thuật toán học máy để học và dự đoán các vùng trong hình ảnh.

Ví dụ, các mô hình Deep Learning như Convolutional Neural Networks (CNNs) có thể được sử dụng để thực hiện phân vùng hình ảnh dựa trên miền.

5. Các hàm tách biên của OpenCV

Trong xử lý ảnh, việc tách biên là một bước quan trọng để trích xuất các đặc trưng và thông tin quan trọng từ hình ảnh. OpenCV cung cấp một loạt các phương pháp tách biên mạnh mẽ, bao gồm Canny Edge Detection, Sobel Edge Detection, Laplacian Edge Detection và Scharr Edge Detection. Dưới đây là một ví dụ minh họa cho mỗi loại hàm tách biên này:



Hình 17. Minh họa các hàm trong OpenCV

5.1. Canny Edge Detection:

Hàm: `cv2.Canny(image, threshold1, threshold2[, edges[, apertureSize[, L2gradient]]])`

Trong đó:

- image: Ảnh đầu vào, đây là ảnh mà bạn muốn thực hiện tách biên.
- threshold1, threshold2: Ngưỡng thấp và ngưỡng cao
- edges: (Tùy chọn) Ma trận đầu ra để lưu trữ kết quả của tách biên.
- L2gradient: (Tùy chọn) Một cờ boolean chỉ định liệu sử dụng chuẩn L2 (True) hay chuẩn L1 (False) cho phát hiện gradient. Mặc định là False.

⇒ Canny Edge Detection là một trong những phương pháp phát hiện biên phổ biến nhất trong xử lý ảnh. Nó sử dụng bộ lọc Gaussian để làm mờ ảnh trước khi sử dụng đạo hàm để tìm các biên cạnh sắc nét trong ảnh.

5.2. Sobel Edge Detection:

Hàm: `cv2.Sobel(src, ddepth, dx, dy[, dst[, ksize[, scale[, delta[, borderType]]]])`

Trong đó:

- src: Ảnh đầu vào, đây là ảnh mà bạn muốn thực hiện tách biên.
- ddepth: Kiểu dữ liệu của ảnh đầu ra. Thông thường là -1 để giữ nguyên kiểu dữ liệu của ảnh đầu vào.
- dx, dy: Độ lớn của đạo hàm ứng với các biến x và y. Một trong số chúng có thể là 0, nhưng ít nhất một phải khác không để tính toán được gradient.
- dst: (Tùy chọn) Ma trận đầu ra để lưu trữ kết quả của tách biên. Nếu không được cung cấp, kết quả sẽ được trả về.
- ksize: (Tùy chọn) Kích thước của bộ lọc Sobel. Mặc định là 3.
- scale: (Tùy chọn) Hệ số tỉ lệ được áp dụng cho kết quả. Mặc định là 1.
- delta: (Tùy chọn) Giá trị được thêm vào kết quả. Mặc định là 0.
- borderType: (Tùy chọn) Phương pháp xử lý các biên của ảnh. Mặc định là cv2.BORDER_DEFAULT.

⇒ Sobel Edge Detection là một phương pháp phổ biến để tìm biên trong ảnh bằng cách áp dụng các bộ lọc đạo hàm Sobel theo hai hướng x và y.

5.3. Laplacian Edge Detection:

Hàm: cv2.Laplacian(src, ddepth[, dst[, ksize[, scale[, delta[, borderType]]]])

Laplacian Edge Detection sử dụng bộ lọc Laplacian để tìm các biên trong ảnh. Nó có thể giúp phát hiện các biên cạnh với độ cong lớn.

5.4. Scharr Edge Detection:

Hàm: cv2.Scharr(src, ddepth, dx, dy[, dst[, scale[, delta[, borderType]]]])

Scharr Edge Detection cũng sử dụng bộ lọc đạo hàm tương tự như Sobel, nhưng có hiệu suất tốt hơn để phát hiện biên cạnh.

5.5. Công cụ phát hiện biên (Edge Detection Tool):

OpenCV cung cấp một công cụ giao diện người dùng để tùy chỉnh và thử nghiệm các thuật toán phát hiện biên trong ảnh. Bạn có thể sử dụng công cụ này để thử nghiệm và tinh chỉnh các tham số của các thuật toán phát hiện biên khác nhau.

CHƯƠNG III. KẾT LUẬN

Trong bài báo cáo này, chúng ta đã thảo luận về các phương pháp phân vùng ảnh dựa trên biên, ngưỡng và miền trong lĩnh vực xử lý ảnh. Qua quá trình nghiên cứu và phân tích, chúng ta nhận thấy rằng mỗi phương pháp mang lại những ưu điểm và hạn chế riêng, và sự lựa chọn của phương pháp phù hợp phụ thuộc vào bối cảnh và yêu cầu cụ thể của ứng dụng.

Phương pháp phân vùng dựa trên biên cung cấp các kết quả chính xác và sắc nét, nhưng có thể bị ảnh hưởng bởi nhiễu và chi phí tính toán cao. Trong khi đó, phương pháp phân vùng dựa trên ngưỡng là phương pháp đơn giản và hiệu quả, nhưng thường không thể đáp ứng được sự đa dạng của các vùng trong ảnh. Phân vùng dựa trên miền mang lại kết quả tổng quát và linh hoạt hơn, nhưng yêu cầu nhiều tài nguyên tính toán và đặc biệt là kiến thức về đặc trưng và thuộc tính của các vùng trong ảnh.

Tuy nhiên, cần lưu ý rằng không có một phương pháp phân vùng ảnh nào là hoàn hảo cho tất cả các tình huống. Sự kết hợp và tinh chỉnh giữa các phương pháp có thể được áp dụng để đạt được kết quả tốt nhất cho mỗi bài toán cụ thể.

Trong tương lai, việc nghiên cứu và phát triển các phương pháp phân vùng mới và hiệu quả hơn là cần thiết để nâng cao khả năng xử lý ảnh trong các ứng dụng thực tế. Các phương pháp học máy và deep learning đang trở thành xu hướng quan trọng trong lĩnh vực này, và chúng cung cấp cơ hội lớn cho việc tiến bộ trong phân vùng ảnh.

Tóm lại, qua quá trình nghiên cứu và phân tích, bài luận văn đã đưa ra cái nhìn tổng quan về các phương pháp phân vùng ảnh trong xử lý ảnh, từ đó giúp ta hiểu rõ hơn về ứng dụng và tiềm năng của chúng trong thực tế.

TÀI LIỆU THAM KHẢO

Dr. Dafda. (n.d.). *Image Segmentation*. India.

Geeksforgeeks. (n.d.). *OpenCV - Python Tutorials*. Retrieved from <https://www.geeksforgeeks.org/opencv-python-tutorial/>

[Image Segmentation using K Means Clustering - GeeksforGeeks](#)

Khanh's Blog. (n.d.). *Image Segmentation*. Retrieved from <https://phamdinhkhanh.github.io/2020/06/10/ImageSegmentation.html#12-b%C3%A0i-to%C3%A1n-image-segmentation>

MathWork. (n.d.). *Structuring Elements*. Retrieved from <https://www.mathworks.com/help/images/structuring-elements.html>

Nero Phung. (n.d.). *Otsu Threshold*. Retrieved from <https://nerophung.github.io/2019/09/26/otsu-threshold>

PGS. TS. Đỗ Toàn Năng. (n.d.). *Bài Giảng Xử Lí Ảnh*.

TAPIT. (n.d.). *Morphological Image Processing*. Retrieved from <https://tapit.vn/xu-ly-anh-hinh-thai-hoc-morphological-image-processing/>