**Phần 1: Lý thuyết**

1. **Mô hình màu cơ bản**

Mô hình màu cơ bản là các hệ thống được sử dụng để biểu diễn và tổ chức màu sắc theo một cách dễ dàng và có thể ứng dụng trong thực tế, đặc biệt là trong đồ họa và công nghệ số. Trong đó mô hình màu RGB là từ viết tắt trong tiếng Anh có nghĩa là đỏ (red), xanh lục (green) và xanh lam (blue), là ba màu cơ bản trong các mô hình ánh sáng bổ sung.

1. **Với ảnh đa mức xám thì tăng giảm độ sáng của ảnh như thế nào? Chúng ta phải sử dụng toán tử loại nào đề thực hiện phép toán này?**

Với ảnh đa mức xám, để tăng hoặc giảm độ sáng, chúng ta cần thay đổi giá trị của mỗi pixel trong ảnh. Các pixel này có các giá trị cường độ từ 0 (đen) đến 255 (trắng) đối với ảnh 8-bit. Để thực hiện tăng giảm độ sáng, ta sử dụng toán tử **cộng** hoặc **trừ** một hằng số cho từng giá trị pixel trong ảnh:

A white sheet with black text

Description automatically generated

1. **Cân bằng tần suất là gì? Tại sao phải cân bằng tần suất? Kỹ thuật cân bằng?**

**Cân bằng tần suất** (hay còn gọi là cân bằng histogram, *histogram equalization*) là một kỹ thuật xử lý ảnh nhằm cải thiện độ tương phản của hình ảnh bằng cách phân phối lại các giá trị tần suất của pixel để nó trải đều hơn trong khoảng giá trị cường độ. Mục tiêu là làm cho histogram có dạng gần như phẳng, tức là các pixel có giá trị cường độ được phân phối đồng đều hơn, thay vì tập trung ở một số giá trị cụ thể.

**Tại sao phải cân bằng tần suất?**

* **Cải thiện độ tương phản**: Hình ảnh gốc có thể có độ tương phản thấp (như hình ảnh tối hoặc mờ), làm cho chi tiết khó nhìn thấy. Cân bằng tần suất giúp tăng cường độ tương phản và làm cho các chi tiết rõ ràng hơn.
* **Khắc phục vấn đề ánh sáng**: Trong nhiều trường hợp, hình ảnh có thể bị chiếu sáng không đều hoặc bóng tối, cân bằng tần suất giúp điều chỉnh lại ánh sáng trong hình ảnh.
* **Chuẩn bị dữ liệu cho phân tích**: Trong các ứng dụng máy học và phân tích hình ảnh, việc có một histogram cân bằng có thể cải thiện hiệu suất của các thuật toán.

1. **Biến đổi cửa sổ di chuyển hay còn gọi là biến đổi cuộn là biến đối sử dụng toán tử gì? Ý tưởng cơ bản của biến đổi này là gì?**

**Biến đổi cửa sổ di chuyển** (hay còn gọi là biến đổi cuộn, *moving window transformation*) là một kỹ thuật trong xử lý ảnh dùng để thực hiện các phép toán trên các vùng lân cận (nhóm pixel) trong một hình ảnh.

**Toán tử sử dụng**

Biến đổi cửa sổ di chuyển thường sử dụng các **toán tử không gian** (spatial operators) để tính toán giá trị mới cho một pixel dựa trên các pixel lân cận trong một cửa sổ (window) nhất định. Một số loại toán tử phổ biến bao gồm:

* **Toán tử trung bình (Average Filter)**: Tính giá trị trung bình của các pixel trong cửa sổ.
* **Toán tử trung vị (Median Filter)**: Tính giá trị trung vị của các pixel trong cửa sổ.
* **Toán tử Gauss (Gaussian Filter)**: Sử dụng một hàm Gauss để trọng số các pixel trong cửa sổ, nhằm làm mờ ảnh.
* **Toán tử phát hiện cạnh (Edge Detection Operators)**: Như Sobel hoặc Prewitt, để phát hiện các cạnh trong ảnh.

**Ý tưởng cơ bản**

* **Cửa sổ di chuyển** là một khung hình chữ nhật (hoặc hình vuông) với kích thước cố định, di chuyển qua mỗi pixel trong hình ảnh.
* Khi cửa sổ di chuyển đến mỗi pixel, nó sẽ xem xét các giá trị pixel trong cửa sổ để tính toán giá trị mới cho pixel trung tâm của cửa sổ.
* Biến đổi này giúp xử lý các vùng lân cận của mỗi pixel để thực hiện các tác vụ như làm mờ, phát hiện cạnh, nâng cao độ tương phản, và giảm nhiễu trong ảnh.

1. **Mô hình nhiễu là gì? Các loại nhiễu?** P.318

**Mô hình nhiễu** trong xử lý ảnh là một cách để mô tả và phân tích các tác động không mong muốn mà nhiễu có thể gây ra đối với các tín hiệu hoặc hình ảnh. Nhiễu là những biến đổi ngẫu nhiên trong tín hiệu, gây ra bởi các yếu tố như cảm biến, môi trường, hoặc quy trình thu thập dữ liệu.

Các mô hình/loại nhiễu:

* **Salt Noise**: nhiễu muối (Là loại nhiễu mà trong ảnh xuất hiện các chấm trắng (hoặc giá trị sáng hơn) ngẫu nhiên, làm cho bức ảnh có những điểm sáng bất thường. Điều này thường xảy ra khi các pixel bị thay đổi thành giá trị cực đại, tương tự như các hạt muối trong một bức ảnh.)
* **Pepper noise**: nhiễu tiêu (Là loại nhiễu mà trong ảnh xuất hiện các chấm đen (hoặc giá trị tối hơn) ngẫu nhiên, tạo ra các điểm tối bất thường trên bức ảnh. Điều này xảy ra khi các pixel bị thay đổi thành giá trị cực tiểu, giống như các hạt tiêu trong một bức ảnh)
* **Impulse noise**: nhiễu xung hay nhiễu muối hạt tiêu (Salt-and-Pepper Noise)
* **Gauss noise**: Nhiễu Gauss có đặc điểm là làm cho toàn bộ hình ảnh trở nên mờ hoặc không đồng đều.
* **Speckle noise**: nhiễu lốm đốm
* **Periodic noise**: Nhiễu định kỳ

1. **Biên là gì? Kỹ thuật phát hiện biên?**

**Biên** trong xử lý ảnh là khu vực chuyển tiếp giữa các vùng có độ sáng, màu sắc hoặc kết cấu khác nhau trong một hình ảnh. Biên thường chỉ ra sự thay đổi đột ngột trong cường độ ánh sáng hoặc màu sắc và có thể giúp nhận diện các đối tượng, chi tiết và hình dạng trong ảnh.

**Kỹ thuật phát hiện biên**

Phát hiện biên là quá trình xác định và làm nổi bật các đường biên trong một hình ảnh. Có nhiều kỹ thuật khác nhau để phát hiện biên, dưới đây là một số phương pháp phổ biến:

1. **Toán tử Sobel:**
   * Sử dụng hai bộ lọc (kernel) để tính toán độ gradient theo cả hai hướng x và y.
   * Biểu thức của toán tử Sobel là:

;

* + Giá trị gradient được tính bằng: G =
  + Phát hiện biên dựa trên độ lớn của gradient.

1. **Toán tử Prewitt:**
   * Tương tự như Sobel, nhưng bộ lọc có trọng số khác nhau.
   * Bộ lọc Prewitt cũng được áp dụng để tìm gradient trong các hướng x và y.
2. **Toán tử Canny:**
   * Là một trong những phương pháp phát hiện biên hiệu quả nhất.
   * Bao gồm các bước:
     1. Làm mờ ảnh: Sử dụng bộ lọc Gauss để giảm nhiễu.
     2. Tính gradient: Sử dụng toán tử Sobel để tìm độ lớn và hướng của gradient.
     3. Phát hiện biên nén: Sử dụng ngưỡng để xác định các biên mạnh và yếu.
     4. Phát hiện biên bằng đường dẫn: Kết nối các biên yếu với các biên mạnh để tạo ra đường biên liên tục.
3. **Phát hiện biên Laplacian:**
   * Sử dụng toán tử Laplacian để phát hiện biên bằng cách tính toán sự thay đổi thứ hai trong cường độ pixel.
   * Thường được sử dụng cùng với bộ lọc Gauss để làm mờ nhiễu trước khi phát hiện biên.
4. **Biến đổi Hough:**
   * Kỹ thuật này không chỉ phát hiện biên mà còn nhận diện các hình dạng cụ thể như đường thẳng hoặc đường cong.
   * Nó chuyển đổi không gian hình ảnh sang không gian tham số, nơi mà các đường thẳng có thể dễ dàng nhận diện.
5. **Tại sao chúng ta phải nén ảnh? Các kỹ thuật nén ảnh**

Nén ảnh là một trong những kỹ thuật trong xử lý ảnh số có chức năng làm giảm dung lượng cần sử dụng để biểu diễn một bức ảnh. Ngày nay, với sự phát triển của công nghệ Internet, việc sử dụng ảnh số ngày một gia tăng. Vì vậy, vai trò của nén ảnh trở nên rất quan trọng trong việc truyền và lưu trữ ảnh số

1. **Các phương pháp lọc ảnh trên miền không gian? Lọc trên miền tần số**

**Phương pháp lọc ảnh trên miền không gian**

Lọc trên miền không gian là quá trình thực hiện phép toán trực tiếp trên các giá trị pixel trong ảnh. Các kỹ thuật lọc phổ biến trên miền không gian bao gồm:

* **Lọc trung bình (Mean Filtering)**:
  + Sử dụng một bộ lọc hình vuông (hoặc hình chữ nhật) để tính giá trị trung bình của các pixel trong cửa sổ.
  + Giúp làm mờ ảnh và giảm nhiễu.
* **Lọc trung vị (Median Filtering)**:
  + Tính giá trị trung vị của các pixel trong cửa sổ.
  + Hiệu quả trong việc loại bỏ nhiễu muối và tiêu đường mà không làm mất nhiều chi tiết của ảnh.
* **Lọc Gaussian (Gaussian Filtering)**:
  + Sử dụng bộ lọc Gaussian để làm mờ ảnh.
  + Có trọng số cho các pixel gần trung tâm cửa sổ cao hơn, giúp giảm nhiễu một cách hiệu quả.
* **Lọc cạnh (Edge Filtering)**:
  + Sử dụng các toán tử như Sobel, Prewitt, hoặc Roberts để phát hiện biên trong ảnh.
  + Tạo ra các ảnh chỉ chứa thông tin về các biên và cạnh.
* **Lọc theo ngưỡng (Thresholding)**:
  + Biến đổi ảnh thành ảnh nhị phân dựa trên một ngưỡng cường độ.
  + Tách biệt các đối tượng trong ảnh bằng cách loại bỏ các pixel có giá trị cường độ dưới ngưỡng.

**Phương pháp lọc ảnh trên miền tần số**

Lọc trên miền tần số sử dụng phân tích tần số để xử lý ảnh. Quá trình này thường bao gồm việc biến đổi ảnh từ miền không gian sang miền tần số, thực hiện phép lọc, và sau đó biến đổi trở lại. Các kỹ thuật lọc phổ biến trên miền tần số bao gồm:

* **Biến đổi Fourier (Fourier Transform)**:
  + Biến đổi ảnh từ miền không gian sang miền tần số.
  + Tần số thấp đại diện cho các đặc trưng mượt mà trong ảnh, trong khi tần số cao đại diện cho các chi tiết và biên.
* **Lọc tần số cao (High-Pass Filtering)**:
  + Giúp làm nổi bật các chi tiết và biên trong ảnh bằng cách loại bỏ tần số thấp.
  + Thường sử dụng các bộ lọc như bộ lọc Laplacian hoặc các bộ lọc Gaussian có tần số cao.
* **Lọc tần số thấp (Low-Pass Filtering)**:
  + Sử dụng để làm mờ ảnh và giảm nhiễu bằng cách loại bỏ tần số cao.
  + Có thể thực hiện bằng cách áp dụng bộ lọc Gaussian hoặc bộ lọc trung bình trong miền tần số.
* **Bộ lọc Butterworth, Chebyshev, và các bộ lọc khác**:
  + Các loại bộ lọc này có thể được thiết kế để lọc tần số cao hoặc thấp trong miền tần số với các thuộc tính cụ thể như độ dốc và chiều cao.
* **Biến đổi Wavelet**:
  + Cung cấp cách tiếp cận linh hoạt hơn để phân tích ảnh, cho phép lọc và nhận diện các đặc trưng tại nhiều tần số và vị trí khác nhau.
  + Thích hợp cho các bài toán nén ảnh và phát hiện đặc trưng.

1. **Khái niệm, ý nghĩa của phân vùng ảnh? Các kỹ thuật thuật phân vùng ảnh**

**Khái niệm phân vùng ảnh**

**Phân vùng ảnh** (*image segmentation*) là quá trình chia một hình ảnh thành nhiều phần (hay vùng) khác nhau nhằm mục đích đơn giản hóa hoặc thay đổi nội dung của hình ảnh để dễ dàng phân tích hơn. Mỗi vùng trong hình ảnh thường đại diện cho một đối tượng hoặc một phần của đối tượng trong cảnh.

**Ý nghĩa của phân vùng ảnh**

* **Phân tích hình ảnh**: Giúp nhận diện và phân tích các đối tượng hoặc các phần của đối tượng trong ảnh.
* **Nâng cao độ chính xác**: Cải thiện độ chính xác của các ứng dụng như nhận diện đối tượng, theo dõi chuyển động, và phân tích cảnh.
* **Tách biệt thông tin**: Cho phép xử lý và tách biệt thông tin cần thiết cho các bước xử lý tiếp theo như nhận diện, phân loại, và trích xuất đặc trưng.
* **Ứng dụng trong các lĩnh vực khác nhau**: Phân vùng ảnh được sử dụng trong nhiều lĩnh vực như y tế (phân tích hình ảnh y khoa), an ninh (nhận diện khuôn mặt), và xe tự lái (nhận diện đối tượng xung quanh).

**Các kỹ thuật phân vùng ảnh**

Có nhiều kỹ thuật phân vùng ảnh khác nhau, mỗi kỹ thuật có ưu nhược điểm và ứng dụng riêng. Dưới đây là một số kỹ thuật phổ biến:

1. **Phân vùng theo ngưỡng (Thresholding)**:
   * Phân vùng dựa trên việc xác định một hoặc nhiều ngưỡng cường độ pixel.
   * Pixels được phân thành hai hoặc nhiều vùng dựa trên các giá trị cường độ của chúng (ví dụ: ảnh nhị phân).
2. **Phân vùng theo vùng (Region-Based Segmentation)**:
   * Tập trung vào việc tìm các vùng liên thông trong ảnh có đặc trưng tương tự nhau.
   * **Phương pháp vùng lớn nhất (Region Growing)**: Bắt đầu từ một pixel seed và mở rộng vùng bằng cách thêm các pixel lân cận tương tự.
   * **Phân vùng theo tách biệt (Region Splitting and Merging)**: Chia nhỏ hình ảnh thành các vùng nhỏ hơn và sau đó hợp nhất các vùng có đặc trưng tương tự.
3. **Phân vùng biên (Edge-Based Segmentation)**:
   * Sử dụng thông tin về biên (đường chuyển tiếp giữa các vùng khác nhau) để phân vùng ảnh.
   * Các kỹ thuật như toán tử Sobel, Prewitt, hoặc Canny để phát hiện biên và xác định các vùng.
4. **Phân vùng theo thuật toán phân cụm (Clustering-Based Segmentation)**:
   * Sử dụng các thuật toán phân cụm như K-means hoặc Mean Shift để nhóm các pixel tương tự vào cùng một vùng.
   * Các pixel có độ tương đồng về màu sắc hoặc cường độ sẽ được phân vào cùng một cụm.
5. **Phân vùng theo mô hình (Model-Based Segmentation)**:
   * Sử dụng mô hình hình học hoặc thống kê để phân vùng hình ảnh.
   * Ví dụ: Mô hình Gaussian Mixture Model (GMM) cho phép phân vùng dựa trên các phân phối xác suất.
6. **Phân vùng theo Deep Learning**:
   * Sử dụng các mạng nơ-ron sâu (deep neural networks) như U-Net hoặc Mask R-CNN để phân vùng ảnh với độ chính xác cao.
   * Phương pháp này thường cho kết quả tốt trong các bài toán phức tạp như nhận diện đối tượng trong ảnh y tế hoặc ảnh tự nhiên.
7. **Một số phương pháp mã hóa**
8. **Biến đổi Cosin trong nén JPEG nhằm mục đích gì? Và bước nào trong nén JPEG sẽ làm cho quá trình nén ảnh là không bảo toàn?**

**Biến đổi Cosin rời rạc (DCT - Discrete Cosine Transform)** là một kỹ thuật quan trọng trong quy trình nén ảnh JPEG. Mục đích chính của DCT trong nén JPEG là chuyển đổi các giá trị pixel trong miền không gian thành các hệ số tần số trong miền tần số.

**Mục đích của biến đổi Cosin trong nén JPEG:**

1. **Biến đổi từ miền không gian sang miền tần số**: DCT giúp phân tách ảnh thành các tần số khác nhau, cho phép xác định tần số nào là quan trọng hơn và tần số nào có thể bị giảm hoặc loại bỏ mà không làm mất nhiều chất lượng hình ảnh.
2. **Tập trung thông tin**: DCT giúp tập trung thông tin vào một số ít các hệ số tần số thấp, trong khi các hệ số tần số cao (chứa thông tin chi tiết và nhiễu) thường có giá trị nhỏ và có thể bị loại bỏ hoặc giảm đi trong quá trình nén.
3. **Giảm kích thước dữ liệu**: Bằng cách loại bỏ các tần số cao và chỉ giữ lại các tần số thấp có ảnh hưởng lớn đến chất lượng hình ảnh, quá trình nén có thể giảm thiểu kích thước tệp mà vẫn giữ được chất lượng hình ảnh tốt.

**Bước không bảo toàn trong nén JPEG**

Quá trình nén JPEG bao gồm nhiều bước, trong đó một trong những bước quan trọng nhất dẫn đến việc nén ảnh không bảo toàn là **quá trình lượng tử hóa (quantization)**.

**Quá trình lượng tử hóa:**

* **Lượng tử hóa** là quá trình làm tròn các hệ số DCT (các giá trị tần số) đến các giá trị gần nhất có thể, dựa trên một bảng lượng tử hóa cụ thể.
* Bước này làm giảm độ chính xác của các hệ số DCT và loại bỏ một số thông tin tần số cao, dẫn đến việc mất thông tin. Đây là lý do tại sao nén JPEG là một phương pháp nén **không bảo toàn** (lossy compression).
* Khi lượng tử hóa được thực hiện, một số chi tiết trong ảnh gốc không thể được khôi phục khi ảnh được giải nén, do đó dẫn đến việc mất chất lượng hình ảnh.

1. **Xử lý hình thái- khái niệm, ý nghĩa, các phép xử lý hình thái**

Khái niệm:

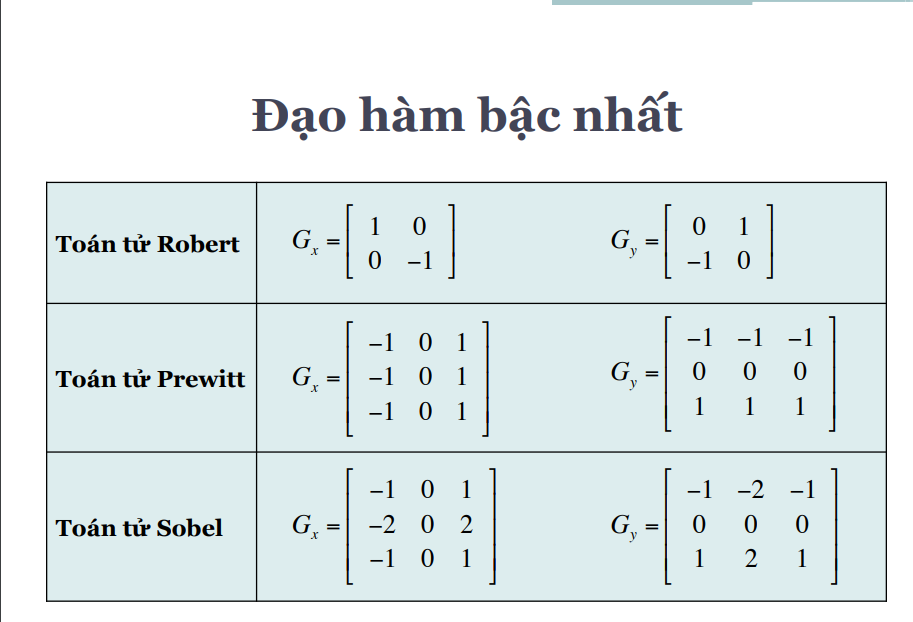
Khi ảnh được xử lý để tăng cường ảnh và khi thực hiện một số thao tác như phân ngưỡng, các ảnh có nhiều khả năng bị méo do nhiễu. Do đó, trong cấu trúc ảnh xuất hiện các khiếm khuyết. Mục tiêu chính của xử lý ảnh hình thái là loại bỏ các khiếm khuyết này, các khiếm khuyết ảnh hưởng chủ yếu đến hình dạng và kết cấu của bức ảnh. Hình thái trong xử lý ảnh nghĩa là mô tả về hình dáng và cấu trúc của đối tượng trong ảnh

Một số phép xử lý hình thái

* Erosion (Xói mòn): Thu hẹp ranh giới của các đối tượng trong một hình ảnh.
* Dilation (Giãn nở): Mở rộng ranh giới của các đối tượng trong một hình ảnh.
* Opening (Mở): Xóa các đối tượng nhỏ và nhiễu khỏi một hình ảnh.
* Closing (Đóng): Lấp đầy các lỗ hổng bên trong các đối tượng và xóa các khoảng trống nhỏ giữa các đối tượng.
* Hit-or-miss: Phát hiện các mẫu hoặc hình dạng cụ thể trong một hình ảnh.
* Skeletonization: Giảm một đối tượng thành bộ xương hoặc trục giữa của nó.

**Phần 2: Bài tập**

**BÀI 1: Tìm biên của ảnh sau sau khi áp dụng các bộ lọc Robert (prewitt, sobel)**



* **Theo phương pháp Sobel**

Hx =

Hy =

I =

Giải: I \* Hx + I \* Hy

I \* Hx =

* Phương pháp Robert

- Phương pháp prewit

Hx =

Hy =

**BÀI 2: Cho biết kết quả của ảnh sau khi tăng cường sử dụng các hàm biến đổi mức xám kỹ thuật lọc không gian, lọc thông thấp Gaussian cửa số 3x3**

I =

**BÀI 3: Cân bằng histogram**

**Lí thuyết:**

: mức xám đầu vào

: mức xám đầu ra

: tần suất cường độ xám

n: tổng số điểm ảnh

L’: số mức xám ảnh đầu ra (thường bằng số mức xám đầu vào)

Cho bức ảnh I. Hãy vẽ lược đồ xám và thực hiện cân bằng lược đồ xám. Tìm ảnh I’ sau khi đã cân bằng lược đồ xám

I =

**GIẢI: 1**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | 0 | 1 | 2 | 4 | 7 |
|  | 5/20 | 7/20 | 5/20 | 3/20 | 1/20 |

**A line graph with numbers

Description automatically generated**

Cân bằng lược đồ xám:

**=** 4 p(0) = 4 \* = 1

= 4 [p(0) + p(1)]

= 4 ( = 2.4 ~ 2

= 4 [p(0) + p(1) +p(2)]

= 4 ( = 3.4 ~ 3

= 4 [p(0) + p(1) + p(2) + p(4)]

= 4 ( = 3.8 ~ 4

= 4 [p(0) + p(1) + p(2) + p(4) + p(7)]

= 4 ( = 4.4 ~ 4

I = 🡪 I\_kq =

**GIẢI: 2 của cô**

Histogram ảnh gốc:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| rk | nj | PDF | CDF | 7\*CDF | sk = round(5CDF) |
| 0 | 5 | 0.25 | 0.25 | 1.75 | 1 |
| 1 | 7 | 0.35 | 0.6 | 4.2 | 4 |
| 2 | 5 | 0.25 | 0.85 | 5.95 | 6 |
| 3 | 0 | 0 | 0.85 | 5.95 | 6 |
| 4 | 2 | 0.1 | 0.95 | 6.65 | 7 |
| 5 | 0 | 0 | 0.95 | 6.65 | 7 |
| 6 | 0 | 0 | 0.95 | 6.65 | 7 |
| 7 | 1 | 0.05 | 1 | 7 | 7 |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Mức xám đầu ra (sk) | 1 | 4 | 6 | 7 |
| tần suất | 5 | 7 | 5 | 3 |

**BÀI 4: Xử lý hình thái**

**A diagram of a math problem

Description automatically generated with medium confidence**

A white and blue squares with black text

Description automatically generated

A grid with blue and white squares

Description automatically generated

Tìm kết quả của phép: giãn, co, đóng, mở của ảnh với phần tử cấu trúc B

X = với B =

GIẢI:

Triển khai ma trận với mặt nạ:

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 1 | 1 | 0 | 0 | 1 | 0 | 0 | 0 |
| 0 | 1 | 1 | 0 | 0 | 0 | 1 | 1 | 0 | 0 |
| 0 | 0 | 1 | 1 | 0 | 1 | 1 | 1 | 1 | 0 |
| 0 | 1 | 1 | 0 | 1 | 1 | 1 | 1 | 1 | 0 |
| 0 | 1 | 0 | 1 | 1 | 1 | 1 | 0 | 0 | 0 |
| 0 | 1 | 0 | 0 | 1 | 0 | 1 | 1 | 1 | 0 |
| 0 | 1 | 0 | 1 | 1 | 1 | 0 | 1 | 0 | 0 |
| 0 | 1 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

Phép co: X1= (fit (B,X) = 1 else 0)

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 1 | 0 | 0 | 0 | 0 | 1 | 0 | 0 |
| 0 | 0 | 0 | 0 | 1 | 0 | 1 | 1 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

Phép giãn: X2 = ( Hit(B,X1) = 1 else 0)

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 1 | 0 | 0 | 0 | 0 | 1 |
| 1 | 1 | 0 | 0 | 0 | 1 | 1 | 1 |
| 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 |
| 0 | 0 | 0 | 1 | 1 | 0 | 1 | 1 |
| 0 | 0 | 0 | 0 | 1 | 1 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

**BÀI 5: Phân vùng ảnh**

A white paper with black text

Description automatically generated

**GIẢI:**

Chọn ngưỡng t0 = 5

t1 = = 5.0574452

Xét < 5.05

t2 = = 5.05

t1 = t2 🡪 ngưỡng tự động = 5

A white background with black text

Description automatically generated

Được biết độ chính xác cần tính là 88%.

**GIẢI:**

Tìm mức xám có h(g) lớn nhất thì có max(p) = 5 (g =112)

Tổng số pixel = 456 (cộng tổng các giá trị h(g) với nhau)

Số pixel từ 0 -> p là: 456 \* 88% = 401,28 pixel

=> p = 6 (do 39+45+53+72+40+112+25 = 386 < 401)

=> Ngưỡng delta = max(p) – (p-max(p)) = 5 – (6-5) = 4

**c. Tìm ngưỡng tự động cục bộ theo thuật toán otsu**

I =

**GIẢI:**

Ma trận N x M = 6 x 5 =30

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| i |  | Pi(k)= | m(k)= | mg= | = |
| 0 | 15/30 | 15/30 = 0.5 | 15/30 \*0=0 | 7/6 | = 1.36 |
| 1 | 5/30 | 15/30+5/30=20/30 | 0+5/30 \*1=5/30 | 7/6 | = 1.68 |
| 2 | 4/30 | 20/30+4/30=24/30 | 5/30+4/30 \*2= 13/30 | 7/6 | 1.56 |
| 3 | 3/30 | 24/30+3/30=27/30 | 13/30+3/30 \*3=11/15 | 7/6 | 1.11 |
| 4 | 2/30 | 27/30+2/30=29/30 | 11/15+2/30 \*4=1 | 7/6 | 0.51 |
| 5 | 1/30 | 29/30+1/30=1 | 1+1/30 \*5 =7/6 | 7/6 | 0.00 |

Như vậy ngưỡng T = 1 với = 1.68

**BÀI 6: Thuật toán nén ảnh:**

Mã và giải mã bằng thuật toán LZW với bộ từ điển gốc gồm các khối ảnh kích thức 1x3 tương ứng với các giá trị từ 0 đến 7, từ điển sẽ được xây dựng tiếp theo từ các giá trị 8 và cọi từ điển là đủ lớn

I =

**GIẢI:**

Các khối ảnh có kích thước 1x3:

🡪A number of binary code

Description automatically generated with medium confidence



Giá trị từ 0-7:

Đặt:

Từ điển gốc:

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 |
| 000 | 001 | 010 | 011 | 100 | 101 | 110 | 111 |

|  |  |  |
| --- | --- | --- |
| output | Dictionary | |
| Code word | String |
| 4 | 8 | 100-111 |
| 7 | 9 | 111-010 |
| 2 | 10 | 010-000 |
| 0 | 11 | 000-101 |
| 5 | 12 | 101-110 |
| 6 | 13 | 110-001 |
| 1 | 14 | 001-011 |
| 3 | 15 | 011-111 |
| 7 | 16 | 111-111 |
| 7 | 17 | 111-000 |
| 0 | 18 | 000-110 |

🡪Mã: 4-7-2-0-5-6-1-3-7-7-0

;

**BÀI 7: Hãy nén và giải nén ảnh bằng thuật toán Huffman**

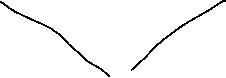
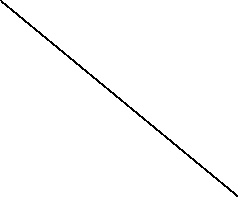
**GIẢI:**

Bảng tần suất:

|  |  |  |  |
| --- | --- | --- | --- |
| Ký tự | 12 | 15 | 18 |
| Tần suất: n | 6 | 6 | 4 |
| Xác suất: p | 3/8 | 3/8 | 1/4 |

12 15 18

3/8 3/8 ¼



Các giá trị ít xuất hiện hơn sẽ được gán mã dài hơn, trong khi các giá trị xuất hiện nhiều hơn sẽ được gán mã ngắn hơn.

12 là 0

15 là 10

18 là 11

🡪 ma trận mã hoá:

🡪 chuỗi nhị phân đã nén:

0 10 0 10 10 0 10 11 11 0 11 0 10 11 10 0