



**HỌC VIỆN
CÔNG NGHỆ BƯU CHÍNH VIỄN THÔNG**

Posts and Telecommunications Institute of Technology

Làm rõ ảnh bị mờ sử dụng Gaussian Blur và Sharpening kernel

HỌC PHẦN: THỰC TẬP CƠ SỞ

Sinh viên thực hiện: Nguyễn Thị Lan

Giáo viên hướng dẫn: Ths. Đinh Xuân Trường

Mã sinh viên: B21DCCN818 - Lớp: D21CQCN09-B

Nội dung

1. Sử dụng Gaussian Blur

- 1.1 Giới thiệu về Gaussian Blur
- 1.2 OpenCV Gaussian Blur

2. Sử dụng sharpening kernel

- 2.1 Nguyên lý
- 2.2 Cách hoạt động
- 2.3 Ví dụ

Sử dụng Gaussian Blur

Giới thiệu về Gaussian Blur

Gaussian Blur được sử dụng để làm mờ hoặc làm trơn ảnh bằng cách áp dụng một bộ lọc Gaussian (một bộ lọc phi tuyến tính) lên mỗi điểm ảnh trong ảnh đầu vào. Bằng cách áp dụng bộ lọc Gaussian, các điểm ảnh gần nhau sẽ có tác động lên nhau và tạo ra hiệu ứng làm mờ ảnh.

Hiểu đơn giản là kernel này có giá trị lớn nhất bên trong và nhỏ dần khi ra ngoài biên. Kernel này phải là số dương và là số lẻ (3, 5, 7, ...)

```
1 blur = cv2.GaussianBlur(img, (5, 5), 0)
```

0	0	0	5	0	0	0
0	5	18	32	18	5	0
0	18	64	100	64	18	0

OpenCV Gaussian Blur

—Để thực hiện Gaussian Blur ta có thể sử dụng hàm GaussianBlur của thư viện OpenCV để áp dụng làm mịn hình ảnh. Hàm này nhận vào ảnh đầu vào và các tham số kết quả là ảnh đã được làm mờ bằng cách áp dụng bộ lọc Gaussian. Cú pháp như sau:

```
blurred_image = cv2.GaussianBlur(image, (35, 35), 25)  
sharpened_image = cv2.addWeighted(image, 1.5, blurred_image, -0.5, 0)
```

Sử dụng sharpening kernel

Nguyên lý

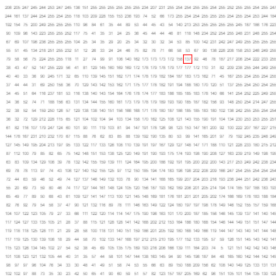
- Sharpening kernel hoạt động bằng cách tăng độ tương phản giữa các pixel gần nhau trong ảnh.
- Bằng cách áp dụng sharpening kernel lên ảnh, các cạnh và đặc trưng trong ảnh sẽ được làm nổi bật hơn, làm tăng sắc nét của ảnh. Điều này làm cho hình ảnh trông sống động hơn.

$$\begin{pmatrix} 0 & -1 & 0 \\ -1 & 5 & -1 \\ 0 & -1 & 0 \end{pmatrix}$$

Cách hoạt động

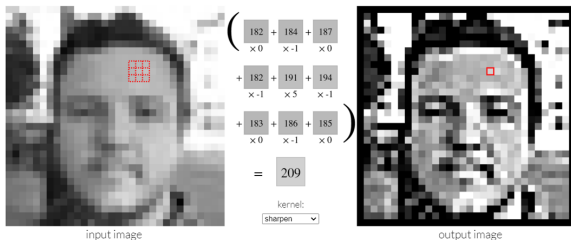
- Sharpening kernel thường được thiết kế để làm nổi bật các biên trong ảnh. Một cách thông thường là giảm độ sáng của pixel ở vị trí trung tâm và tăng độ sáng của các pixel xung quanh.
- Một sharpening kernel cơ bản thường có giá trị âm ở vị trí trung tâm và giá trị dương ở các vị trí xung quanh.
- Các pixel ở vị trí trung tâm của kernel thường nhận một trọng số lớn, trong khi các pixel xung quanh nhận trọng số nhỏ hơn. Điều này giúp tăng độ tương phản và làm nổi bật các biên trong ảnh.
- Khi áp dụng sharpening kernel lên ảnh, các pixel trong ảnh sẽ được tính toán lại dựa trên trọng số của kernel. Kết quả là các biên và đặc trưng trong ảnh được làm nổi bật hơn, làm tăng sắc nét của ảnh.

Để xem chúng hoạt động như thế nào, hãy bắt đầu bằng cách kiểm tra một hình ảnh đen trắng. Ma trận bên trái chứa các số, từ 0 đến 255, mỗi số tương ứng với độ sáng của một pixel trong ảnh khuôn mặt. Hình ảnh lớn, dạng hạt đã được phóng to để dễ nhìn hơn; hình ảnh cuối cùng là kích thước "thật".



Ví dụ

Bên dưới, đối với mỗi khối pixel 3x3 trong hình ảnh bên trái, chúng tôi nhân mỗi pixel với mục nhập tương ứng của hạt nhân rồi lấy tổng. Tổng đó trở thành một pixel mới trong hình ảnh bên phải. Di chuột qua một pixel trên một trong hai hình ảnh để xem giá trị của nó được tính như thế nào.





**HỌC VIỆN
CÔNG NGHỆ BƯU CHÍNH VIỄN THÔNG**

Posts and Telecommunications Institute of Technology

Cảm ơn thầy cô đã chú ý lắng nghe!

Sinh viên thực hiện: Nguyễn Thị Lan

Giáo viên hướng dẫn: Ths. Đinh Xuân Trường

Mã sinh viên: B21DCCN818 - Lớp: D21CQCN09-B



**HỌC VIỆN
CÔNG NGHỆ BƯU CHÍNH VIỄN THÔNG**

Posts and Telecommunications Institute of Technology

PHỤ LỤC: Một số mẫu Latex

Sinh viên thực hiện: Nguyễn Thị Lan

Giáo viên hướng dẫn: Ths. Đinh Xuân Trường

Mã sinh viên: B21DCCN818 - Lớp: D21CQCN09-B