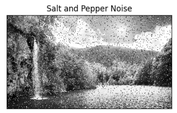
Filtros não Lineares

Conceito

Filtros lineares danificam a qualidade da imagem ao suavizar ou remover ruído, afetando todas as estruturas. Filtros não lineares são explorados como alternativa para essa limitação.



Na filtragem de imagem não linear, a máscara é usada para obter os valores dos pixels vizinhos e os mecanismos de ordenação produzem o pixel de saída, em vez de usar uma máscara espacial em um processo de convolução.

Filtros não lineares serão explorados principalmente para aprimorar imagens ruidosas. Embora possam ser usados para outras tarefas, a remoção de ruído é o uso mais comum desses filtros.

Filtro da Mediana

O filtro de mediana é um filtro não linear que suaviza a imagem substituindo cada pixel pela mediana de seus pixels vizinhos, o que o torna eficaz para remover ruído de impulso e sal e pimenta. Ele não introduz novos valores de pixel e preserva as informações de borda, tornando-o uma boa opção para aprimorar imagens ruidosas.

O filtro da mediana é usado para substituir cada pixel da imagem pela mediana dos pixels na região de filtro correspondente.

```python
import cv2
import numpy as np
import matplotlib.pyplot as plt


img = cv2.imread('./images/image.jpg', cv2.IMREAD_GRAYSCALE)

# salt and pepper noise
noise = np.zeros(img.shape, np.uint8)
cv2.randu(noise, 0, 255)
black = noise < 10
white = noise > 245
img[black] = 0
img[white] = 255

# median filter
filtered = cv2.medianBlur(img, 5)

# Plot images
fig, (ax1, ax2) = plt.subplots(1, 2, figsize=(10, 5))
ax1.imshow(img, cmap='gray', interpolation='nearest')
ax1.set_title('Salt and Pepper Noise')
ax1.set_xticks([])
ax1.set_yticks([])

ax2.imshow(filtered, cmap='gray', interpolation='nearest')
ax2.set_title('Median Filter')
ax2.set_xticks([])
ax2.set_yticks([])

plt.show()
```

Filtros de Máximo e Mínimo

Um filtro de mínimo é um tipo de filtro não linear usado em processamento de imagem para melhorar a qualidade de uma imagem. O filtro de mínimo é uma operação de filtro que é realizada em cada pixel da imagem para obter o valor mínimo de pixel na vizinhança do pixel atual.

Um filtro de máximo é um tipo de filtro não linear que é usado em processamento de imagem para melhorar a qualidade de uma imagem. O filtro de máximo é uma operação de filtro que é realizada em cada pixel da imagem para obter o valor máximo de pixel na vizinhança do pixel atual.

Eficiência

O cálculo dos resultados dos filtros é computacionalmente caro na maioria dos casos, especialmente com imagens grandes e/ou núcleos de filtro grandes. Uma implementação direta requer muitas operações, como multiplicação e adição, e a complexidade de tempo é proporcional ao tamanho da imagem e do kernel, ou seja, O(mnk^2), onde m e n são as dimensões da imagem e k é a dimensão do kernel. Para economizar tempo de computação, é possível decompor grandes filtros bidimensionais em filtros menores, possivelmente unidimensionais, o que pode levar a grandes economias de tempo de computação.

Filtro da Mediana Ponderada

O filtro de mediana ponderado atribui pesos diferentes aos pixels em uma região do filtro, enquanto no filtro de mediana padrão todos os pixels têm a mesma influência. Os pesos são especificados por uma matriz de pesos e o valor central resultante após classificar o vetor de pixels estendido é considerado a mediana.

```python
import cv2
import numpy as np
import matplotlib.pyplot as plt

# Load the image
img = cv2.imread('./images/image.jpg')

# Convert the image to grayscale
gray = cv2.cvtColor(img, cv2.COLOR_BGR2GRAY)

# Apply salt and pepper noise
noise = np.zeros_like(gray)
cv2.randu(noise, 0, 255)
salt = noise > 245
pepper = noise < 10
gray[salt] = 255
gray[pepper] = 0

# Apply median filter with custom kernel
kernel = np.array([[1, 2, 1], [2, 3, 2], [1, 2, 1]], dtype=np.uint8)
filtered = cv2.medianBlur(gray, 5)

# Plot the noisy and filtered images using matplotlib
plt.figure(figsize=(10,10))
plt.subplot(1,2,1)
plt.imshow(gray, cmap='gray')
plt.title('Noisy Image')
plt.axis('off')
plt.subplot(1,2,2)
plt.imshow(filtered, cmap='gray')
plt.title('Filtered Image')
plt.axis('off')
plt.show()
```

Bordas

As bordas da imagem são importantes em filtros, e nenhum resultado de filtro pode ser calculado em posições onde a matriz de filtro não está totalmente contida na matriz de imagem. Isso pode levar à redução do tamanho da imagem, o que não é aceitável em muitos aplicativos. Existem vários métodos para lidar com as regiões de fronteira, incluindo definir os pixels não processados nas bordas para um valor constante, definir os pixels para os valores da imagem original ou estender a imagem preenchendo pixels adicionais em torno dela. Cada método tem suas limitações e a escolha depende do tipo de imagem e filtro aplicado.