

Sygnały i obrazy cyfrowe

Sprawozdanie z przetwarzania obrazów

Michał Wróblewski

272488

Data wykonania sprawozdania: 10.01.2024 r.

1 Wstęp

Kod ogólny

```
1 import numpy as np
2 import cv2
3
4 # Definicja funkcji filtra obrazu
5 # ...
6
7 # Wczytanie obrazu
8 # ...
9
10 # Odszumianie obrazu za pomocą funkcji filtru splotowego
11 # ...
12
13 # Wyświetlenie obrazów
14 # ...
```

Kod - filtr splotowy i medianowy

```
def apply_filter(image, filter_type='average'):
    # Przechodzenie maską 3x3 po pikselach obrazu i zastosowanie
    # filtru splotowego
    filtered_image = np.zeros_like(image)

    for channel in range(3):  # Przechodzenie przez kanały (RGB)
        for i in range(1, image.shape[0] - 1):
            for j in range(1, image.shape[1] - 1):
                if filter_type == 'average':
                    # Przykładowa maska splotowa (uśredniająca)
                    mask = np.array([[1, 1, 1],
```

```

[1, 1, 1],
[1, 1, 1]])
# Zastosowanie filtra splotowego na danym
# fragmencie obrazu
filtered_image[i, j, channel] = np.sum(image[i -
    1:i + 2, j - 1:j + 2, channel] * mask) /
    np.sum(mask)
elif filter_type == 'median':
    # Uzyskanie otoczenia 3x3 dla danego piksela
    surroundings = image[i - 1:i + 2, j - 1:j + 2,
    channel]
    # Zastosowanie filtra medianowego na otoczeniu
    filtered_image[i, j, channel] =
        np.median(surroundings)

return filtered_image

```

Kod - użycie i wyświetlenie

```

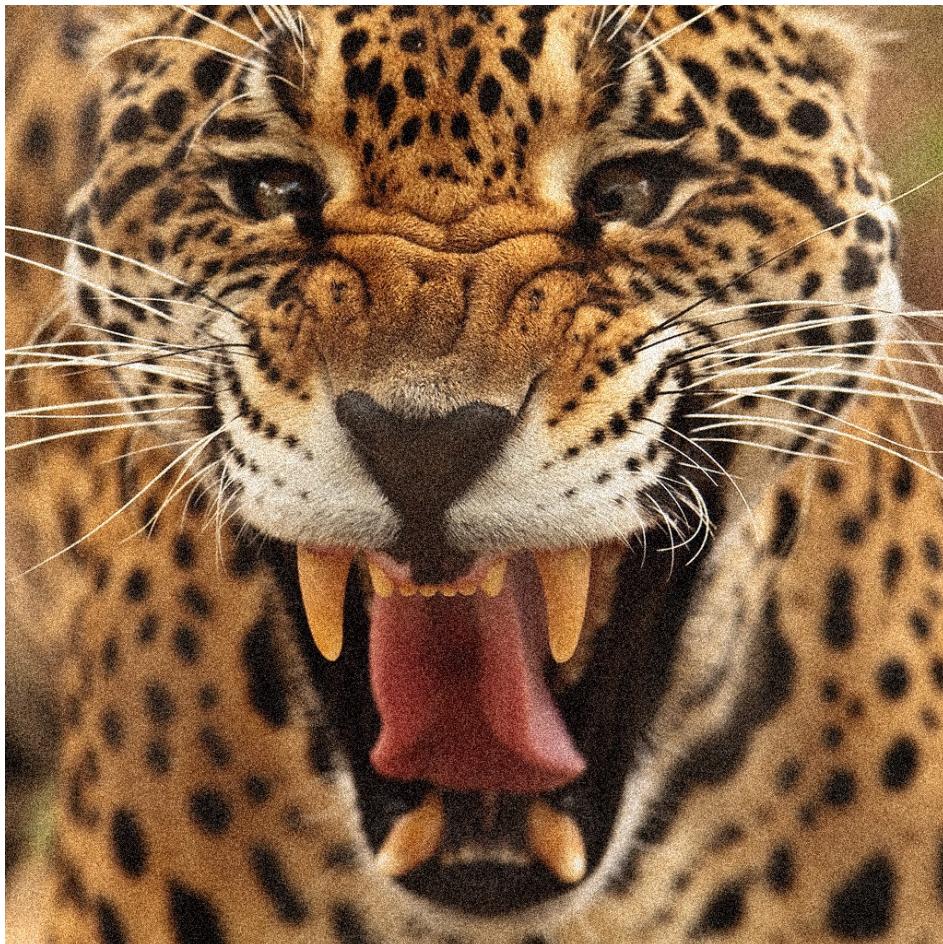
# Wczytanie obrazu
image_path = 'Leopard-with-noise.jpg'
image = cv2.imread(image_path) # Wczytanie obrazu kolorowego

# Odszumianie obrazu za pomocą funkcji filtru splotowego
filtered_image_average = apply_filter(image, filter_type='average')
filtered_image_median = apply_filter(image, filter_type='median')

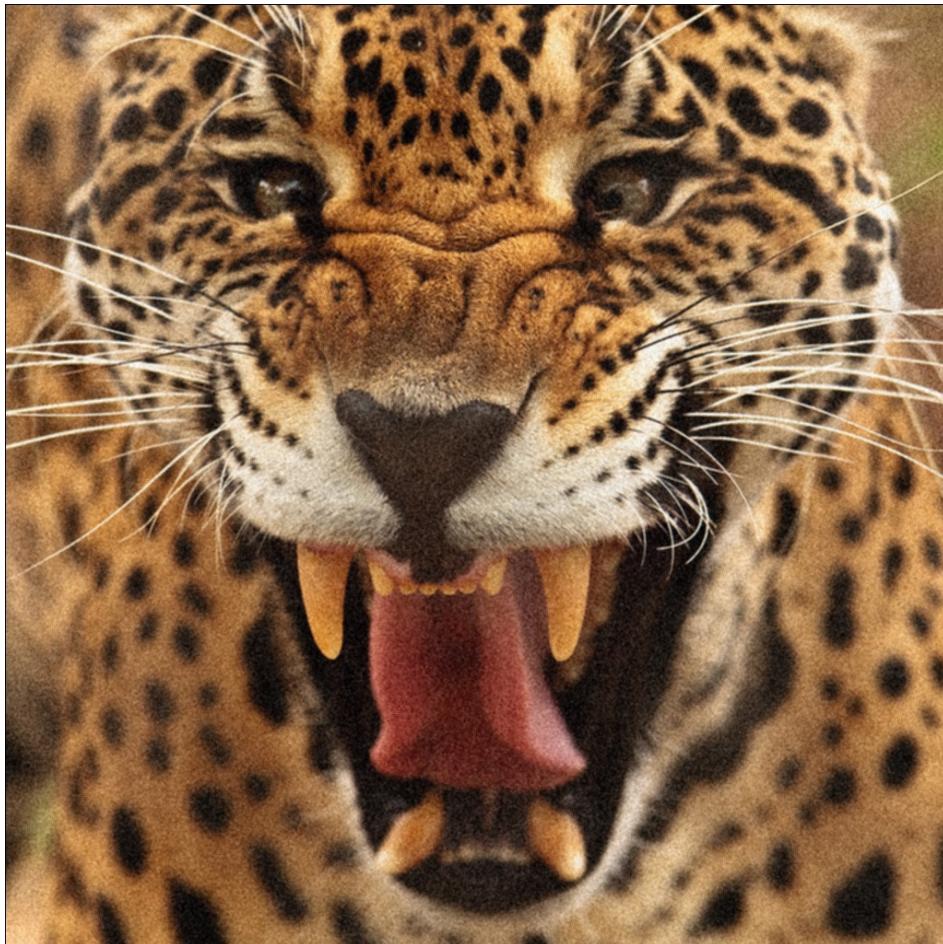
# Wyświetlenie obrazów
cv2.imshow('Original Image', image)
cv2.imshow('Filtered Image (Average)', filtered_image_average)
cv2.imshow('Filtered Image (Median)', filtered_image_median)
cv2.imshow('Diff (Average)', image - filtered_image_average)
cv2.imshow('Diff (Median)', image - filtered_image_median)
cv2.waitKey(0)
cv2.destroyAllWindows()

```

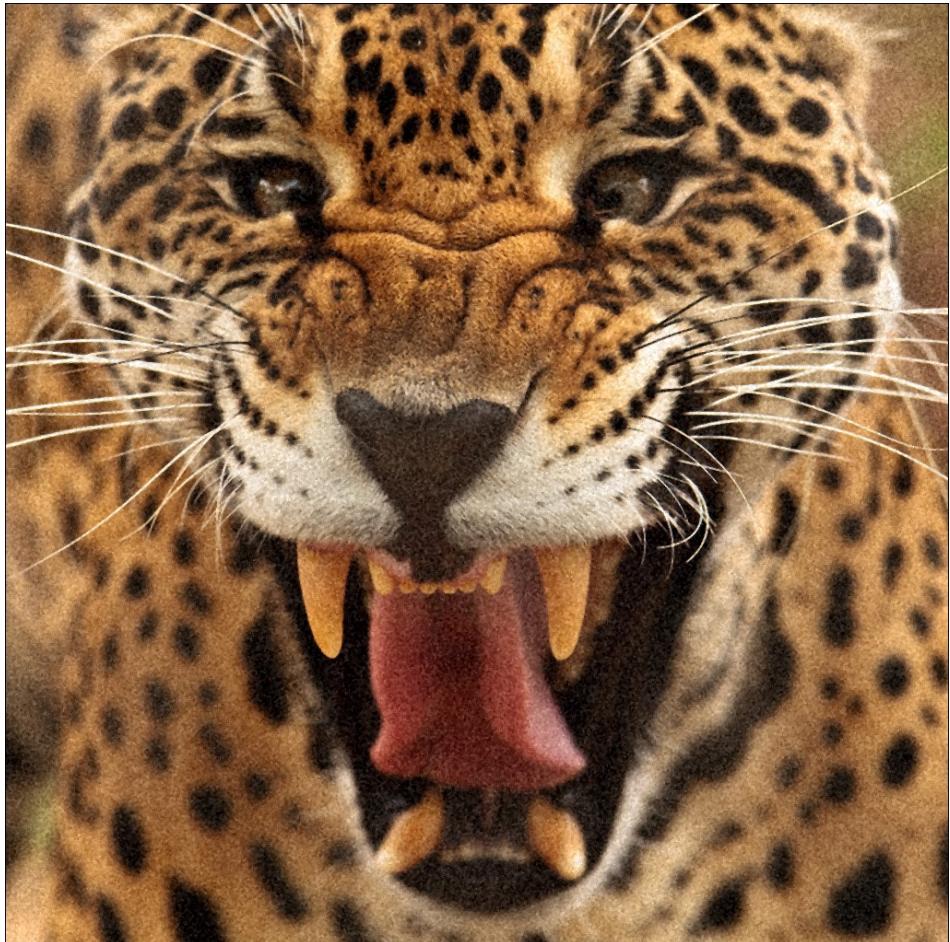
2 Obrazy



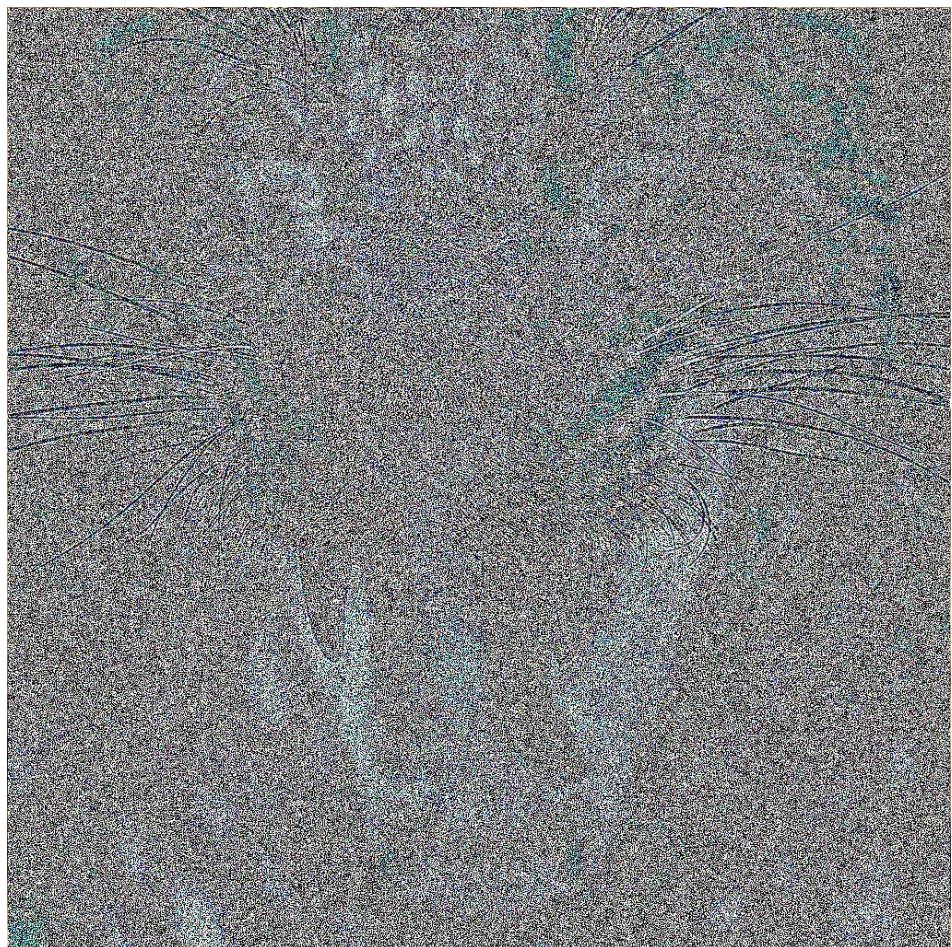
Rysunek 1: Oryginalny Obraz



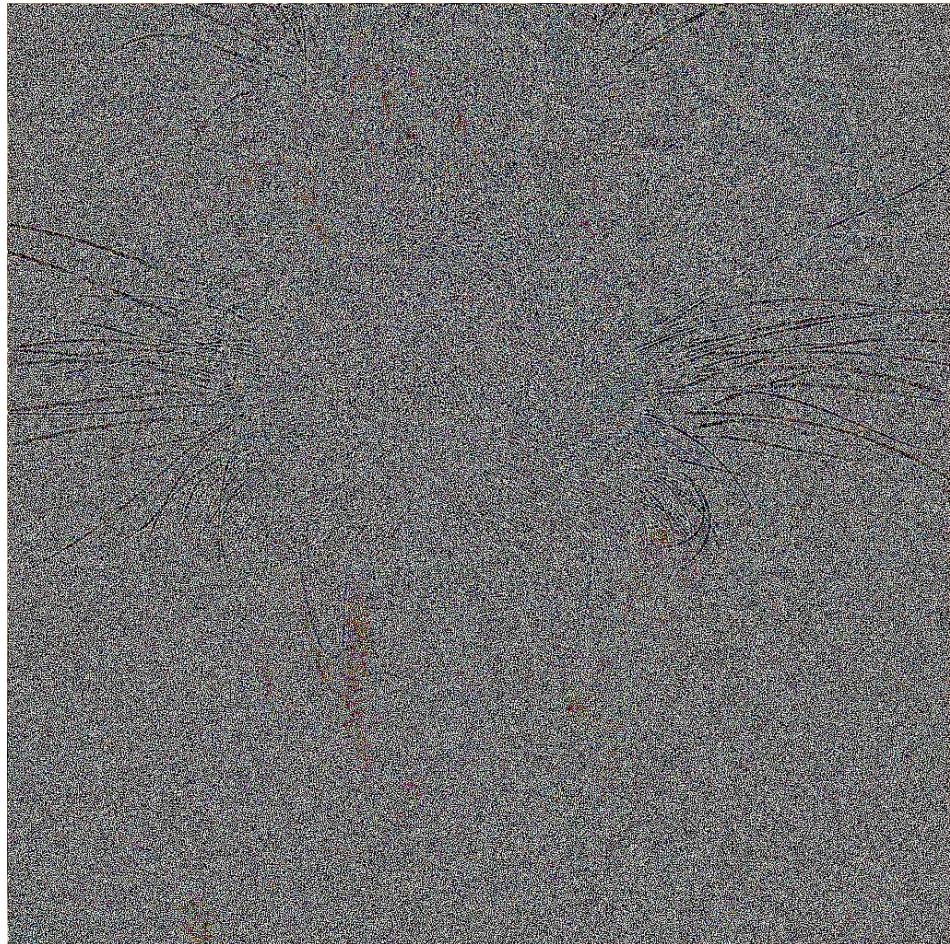
Rysunek 2: Przefiltrowany Obraz (Średnia)



Rysunek 3: Przefiltrowany Obraz (Medianiana)



Rysunek 4: Różnica (Średnia)



Rysunek 5: Różnica (Medianą)