

# Raport lab7 – Metody filtrowania obrazów

## Zadanie 1

funkcja filtruj(obraz, kernel, scale):

```
# zadanie 1
def filtruj(obraz, kernel, scale):
    obraz_size = int(math.sqrt(len(kernel))), int(math.sqrt(len(kernel)))
    return obraz.filter(ImageFilter.Kernel(size=obraz_size, kernel= kernel, scale= scale))

# plt.imshow(filtruj(obraz, (-1, -1, -1, -1, 6, -1, -1, -1, -1), 1))
# plt.show()
#-----
```

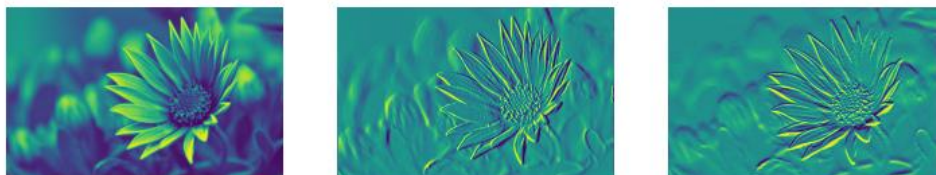
## Zadanie 2

diagram blur, blur1, porównanie:



## Zadanie 3

diagram obraz 'L', sobel1, sobel2:



#### Zadanie 4

diagram z porównaniami dla filtrów 2,4,6,8:

DETAIL



EDGE\_ENHANCE\_MORE



SHARPEN



SMOOTH\_MORE



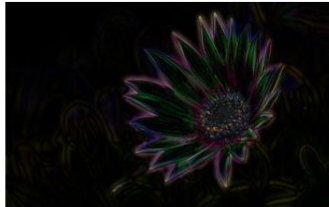
#### **Zadanie 4a**

11. BoxBlur: Rozmywa obraz przez zmianę wartości wszystkich pixeli w kwadracie na ich średnią wartość, radius – rozmiar kwadratu
12. GaussianBlur: Rozmywa obraz korzystając z aproksymacji Gaussian, radius – odchylenie of kernel Gaussian
13. UnsharpMask: radius – obszar rozmycia, percent – moc rozmycia w procentach, threshold – minimalna zmiana jasności, do której zostanie zastosowany filtr
14. Kernel: tworzy Kernel, size – rozmiar, kernel – wagi/wartości w tablicy, scale – skala, jeśli podana to przez nią dzielimy piksele, offset – jeśli podane, dodajemy to do wyniku
15. RankFilter: Sortuje piksele w danym oknie, wynik to rank-ta wartość, size – rozmiar, rank – który pixel wybiera, np. dla min rank = 0
16. MedianFilter: Średnia wartość w oknie
17. MinFilter: Minimalna wartość w oknie
18. MaxFilter: Maksymalna wartość w oknie

## Zadanie 4b

diagram dla filtrów 12,13,16,17,18 i ich porównań:

GaussianBlur, radius=5



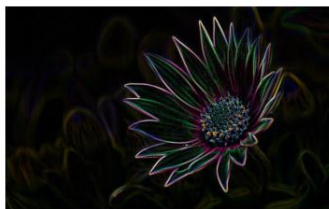
UnsharpMask, radius=10, percent=150, threshold=2



MedianFilter, size=5



MinFilter, size=5



MaxFilter, size=5

