

Systemy Obliczeniowe

Laboratorium 2 – Lista A (1 października 2024)

dr inż. Paweł Trajdos

1 Uwagi

1. Podczas realizacji dzisiejszych zadań wolno Państwu zaimportować jedynie moduły `numpy`, `matplotlib`, `multiprocessing` oraz `time`.
2. Do wczytywania obrazów proszę wykorzystywać funkcję `imread()`.
3. Linki do dokumentacji modułów oraz pomocy odnośnie konwolucji znajdują się na końcu tego dokumentu.
4. Po zakończeniu realizacji zadań należy załadować na Eportal skrypy pythona. Format nazwy pliku `Zx.py` (`x` to numer zadania). Źle nazwane pliki nie są oceniane.
5. W przypadku niepewności lub zauważenia jakichkolwiek błędów w instrukcji należy niezwłocznie powiadomić prowadzącego laboratorium w celu wyjaśnienia sprawy. Reklamacje po zakończeniu zajęć nie będą uwzględniane.

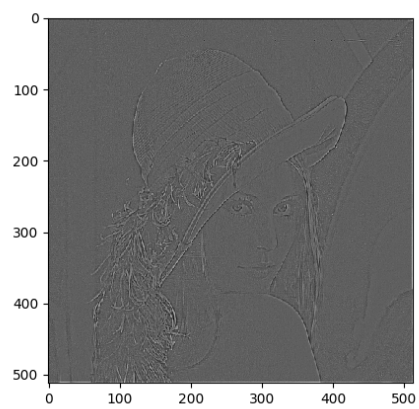
2 Zadania

Zadanie 2.0(Pkt. 5.0):

Jako zadanie rozgrzewkowe należy zaimplementować operację konwolucji wyłącznie z wykorzystaniem biblioteki `numpy` oraz pętli, a następnie przeprowadzić za jej pomocą detekcję krawędzi na załączonym obrazie `lenna.png` z wykorzystaniem podanego niżej filtra o wymiarach 3x3:

```
1 kernel = [[-1, -1, -1],
2           [-1, 8, -1],
3           [-1, -1, -1]]
```

Do obrazu wejściowego należy dodać odpowiedni dobrany `padding`, w taki sposób, aby pozwolić na przejście filtra przez każdy z pikseli obrazu wejściowego. Na rys 1 został przedstawiony rezultat, który powinni Państwo uzyskać:



Rysunek 1: Lenna – wykrywanie krawędzi

Zadanie 2.1(Pkt. 5.0):

W drugim zadaniu należy zrównoleglić wcześniej napisaną operację konwolucji z wykorzystaniem biblioteki `concurrent.futures`, a dokładniej klasy `ProcessPoolExecutor`. W tym celu najlepiej wykorzystać metodę `map`

Dodatkowo, należy rozszerzyć kod z Zadania 2.0. o pomiar czasu przetwarzania zdjęcia, co pozwoli na jego porównanie ze zrównolegloną wersją. Celem powinno być uzyskanie zauważalnej redukcji czasu przetwarzania.

3 Dokumentacja

Dokumentacja:

- `multiprocessing`¹,
- `concurrent.futures`²,
- `time`³,
- Konwolucja 1⁴
- Konwolucja 2⁵

¹<https://docs.python.org/3/library/multiprocessing.html>

²<https://docs.python.org/3/library/concurrent.futures.html>

³<https://docs.python.org/3/library/time.html>

⁴<https://cs231n.github.io/convolutional-networks/>

⁵<https://towardsdatascience.com/a-beginners-guide-to-convolutional-neural-networks-cnns-14649dbddce8>