



Uniwersytet
Bielsko-Bialski

Metoda SVD w celu obliczenia korelacji

Zadanie SVD Korelacja

Sprawozdanie z ćwiczeń
Matematyka Konkretna

Data wykonania:
28.06.2025

Autor:
Bartosz Bieniek 058085

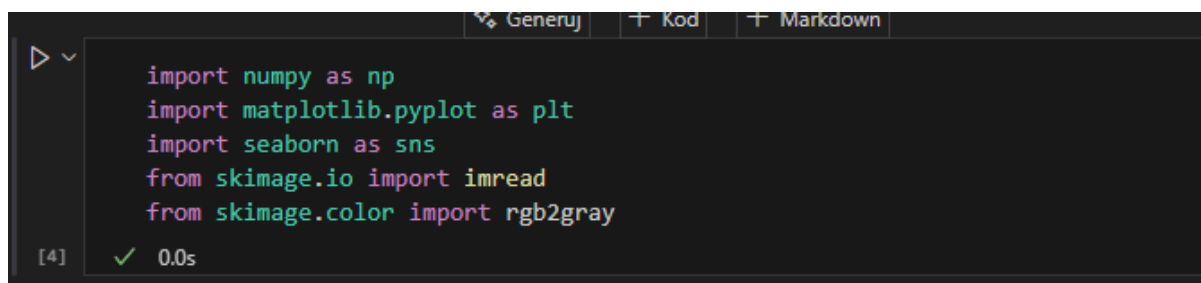
1. Cel ćwiczenia

Zadanie dotyczy obliczenia korelacji obrazu zgodnie z wariantem pod względem zarówno wierszy jak i kolumn z użyciem SVD. Przedstawić macierzy korelacji graficznie. We wniosku potraktować widoki macierzy korelacji. Wstawienie projektu na serwis [Github](#).

2. Przebieg ćwiczenia

1. Import niezbędnych bibliotek

Na początku zaimportowano biblioteki potrzebne do obróbki obrazu, obliczeń oraz wizualizacji:



```
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
from skimage.io import imread
from skimage.color import rgb2gray
```

[4] ✓ 0.0s

Rys. 1. Import bibliotek.

Zaimportowano podstawowe biblioteki umożliwiające wczytanie obrazu, jego przekształcenie do postaci numerycznej oraz wizualizację wyników. Zastosowano numpy do obliczeń macierzowych, matplotlib i seaborn do wizualizacji oraz skimage do przetwarzania obrazów.

2. Wczytanie obrazu i konwersja do skali szarości


Wczytano obraz z pliku w formacie .webp i przekonwertowano go do skali szarości, co umożliwiło uzyskanie dwuwymiarowej macierzy wartości intensywności pikseli. Konwersję wykonano w celu uproszczenia analizy korelacji.

```
# Wczytanie obrazu
img = imread("1.webp")

# Konwersja do skali szarości
gray = rgb2gray(img)

# Wyświetlenie obrazu
plt.imshow(gray, cmap='gray')
plt.title("Obraz w skali szarości")
plt.axis('off')
plt.show()
```

[5] ✓ 0.3s Python



Rys. 2. Wczytanie obrazu i konwersja do skali szarości.

3. Obliczenie SVD obrazu

Na macierzy obrazu przeprowadzono dekompozycję singularną (SVD), uzyskując macierze U , S , V^T , które mogą służyć do dalszych analiz. Choć nie były one bezpośrednio użyte w obliczeniach korelacyjnych, umożliwiły teoretyczne powiązanie ćwiczenia z metodami redukcji wymiaru i kompresji.

```
U, S, VT = np.linalg.svd(gray, full_matrices=False)
```

[6] ✓ 0.5s

Rys. 3. Obliczanie SVD obrazu.

4. Obliczenie macierzy korelacji

W kolejnym kroku obliczono współczynniki korelacji Pearsona osobno dla wierszy i kolumn obrazu. Operacja pozwoliła ocenić stopień liniowej zależności między sąsiednimi elementami macierzy w obu kierunkach.

```

# Korelacja między wierszami
row_corr = np.corrcoef(gray)

# Korelacja między kolumnami
col_corr = np.corrcoef(gray.T)

```

[7] ✓ 0.0s

Rys. 4. Korelacja międzywierszowa i międzykolumnowa.

5. Wizualizacja macierzy korelacji

Dla obu macierzy korelacji wygenerowano mapy cieplne (heatmapy), które umożliwiły graficzną interpretację zależności pomiędzy wierszami i kolumnami. Zastosowano skalę kolorystyczną coolwarm dla lepszej widoczności zmian.

```

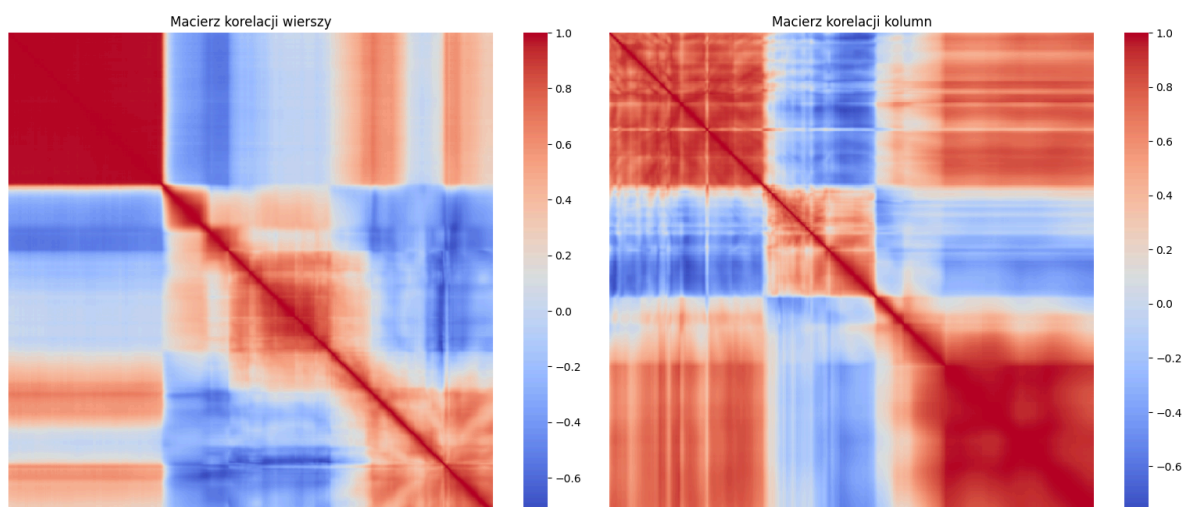
# Wizualizacja korelacji wierszy
plt.figure(figsize=(10, 8))
sns.heatmap(row_corr, cmap='coolwarm', xticklabels=False, yticklabels=False)
plt.title("Macierz korelacji wierszy")
plt.show()

# Wizualizacja korelacji kolumn
plt.figure(figsize=(10, 8))
sns.heatmap(col_corr, cmap='coolwarm', xticklabels=False, yticklabels=False)
plt.title("Macierz korelacji kolumn")
plt.show()

```

[8] ✓ 0.8s Python

Rys. 5. Wizualizacja macierzy korelacji - kod



Rys. 6. Wizualizacja macierzy korelacji - macierz

3. Wnioski

Na podstawie macierzy korelacji stwierdzono, że obraz zawiera obszary o silnej wewnętrznej spójności strukturalnej. Wysokie wartości korelacji ujawniły powtarzające się wzorce zarówno w pionie (wiersze), jak i poziomie (kolumny). Graficzna forma korelacji umożliwiła łatwą detekcję regionów o podobnej strukturze bez konieczności analizy wizualnej samego obrazu.