

SPRAWOZDANIE

Zajęcia: Matematyka Konkretna

Prowadzący: prof. dr hab. inż. Vasyl Martsenyuk

Laboratorium Nr 1 Data 24.02.2024 Temat: „Zadanie SVD” Wariant 6	Adam Kubliński Informatyka II stopień, niestacjonarne, zaoczne, II semestr, gr. 1A
---	---

Repozytorium GitHub: https://github.com/Adamadacho/MK_REP

1. Polecenie: wariant 6 zadania

1.1. Zadanie I

Zadanie dotyczy kompresji obrazu metodą SVD zgodnie z wariantem zadania. Jaka powinna być użyta liczba wartości singularnych żeby zachować 90% informacji na obrazie.

2. Opis opracowanego programu

Zastosowany kod:

```
import numpy as np
import matplotlib.pyplot as plt
from matplotlib.image import imread
from numpy.linalg import svd

# Wczytanie obrazu (zakładamy, że jest w skali szarości)
img = imread('6.webp')
if img.ndim == 3:
    img = img.mean(axis=2) # Konwersja do skali szarości, jeśli jest potrzebna

# Wyświetlenie oryginalnego obrazu
plt.imshow(img, cmap='gray')
plt.title('Oryginalny obraz')
plt.show()

# Rozkład SVD
U, S, VT = svd(img, full_matrices=False)
```

```

# Obliczenie całkowitej energii (suma kwadratów wartości osobliwych)
total_energy = np.sum(S**2)

# Obliczenie, ile wartości osobliwych zachować, aby uzyskać 90% energii
energy_accumulated = 0
k = 0
while energy_accumulated / total_energy < 0.9:
    energy_accumulated += S[k]**2
    k += 1

print(f"Liczba wartości osobliwych do zachowania 90% energii: {k}")

# Rekonstrukcja obrazu przy użyciu k wartości osobliwych
S_k = np.diag(S[:k])
U_k = U[:, :k]
VT_k = VT[:, :k]
img_compressed = U_k @ S_k @ VT_k

# Wyświetlenie skompresowanego obrazu
plt.imshow(img_compressed, cmap='gray')
plt.title(f'Obraz po kompresji z {k} wartościami osobliwymi')
plt.show()

```

3. Wyniki



