КИЇВСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ

ІМЕНІ ТАРАСА ШЕВЧЕНКА

ФАКУЛЬТЕТ КОМП’ЮТЕРНИХ НАУК ТА КІБЕРНЕТИКИ

Звіт

про виконання

лабораторної роботи №2

Виконала

студентка групи ІПС-33

Петрик Юлія Олександрівна

Київ 2023

Зміст

Побудова лiнiйної моделi з допомогою псевдообернених операторiв.

* + - 1. Постановка задачі.
      2. Теоретичні відомості.
      3. Алгоритм виконання.
      4. Розв’язання задачі.
      5. Висновок.
      6. Код програми.

Варіант №8

**Лабораторна робота №2**

Постановка задачі

Потрiбно побудувати лiнiйний оператор перетворення вхiдного сигналу X у вихiдний сигнал Y на основi формули

Y (1)

1. Вивчити означення псевдооберненої матрицi i її основнi властивостi.

2. Створити програму, яка за заданими двома зображеннями знаходить лiнiйний оператор переходу мiж цими зображеннями. Основою для програми є формула (1), де V – довiльна матриця (наприклад, нульова).

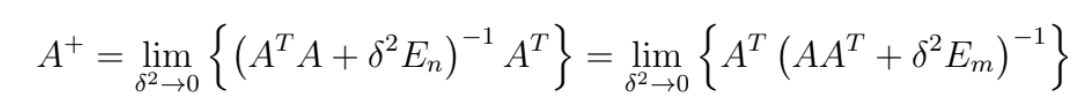
Псевдообернену матрицю в (1) шукати двома методами: на основi

формули Мура-Пенроуза i на основi формули Гревiля. Правильнiсть знаходження псавдооберненої матрицi перевiрити за допомогою теореми про характеристичну властивiсть псевдооберненої матрицi.

3. Вивести вихiдне зображення i образ вхiдного зображення при одержаному перетвореннi. Зробити порiвняння. Проаналiзувати одержаний результат.

Теоретичні відомості

Нехай задана матриця A розмiрностi m × n. За означенням Мура - Пенроуза, **псевдооберненою матрицею** A+ називається матриця розмiрностi n × m вигляду



Тут – одинична матриця розмiрностi n × n. **Властивостi псевдообер-**

**неної матрицi** такi:

1. Якщо матриця A – невироджена, то A+ = A−1.

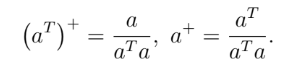
2. A+ =(ATA)+AT, A+ = AT(AAT)+.

3. Якщо матриця ATA – невироджена, то A+ =(ATA)−1AT.

Якщо матриця AAT – невироджена, то A+ = AT(AAT)−1.

4. Якщо a ∈ Rn – вектор розмiрностi n, a a≠ 0, то з означення Мура-

Пенроуза випливає, що



Якщо a = 0, то a+ = 0.

5. (A+)+ = A.

6. (AT)+ = (A+)T.

**Теорема 3.1** (характеристична властивiсть псевдооберненої матрицi).

Матриця A+ розмiрностi n × m є псевдооберненою матрицею до матрицi A розмiрностi m×n тодi i тiльки тодi, якщо виконуються такi умови:

• AA+A = A;

• A+AA+ = A+;

• AA+ – симетрична матриця розмiрностi m × m;

• A+A – симетрична матриця розмiрностi n × n.

Алгоритм виконання

*Алгоритм знаходження псевдооберненої матриці на основi формули Мура-Пенроуза:*

1. Задається початкове значення δ = δ0;

2. Розраховується початкове наближення A0+ = AT(AAT + δ02Em)−1;

3. На кроцi k нове значення ;

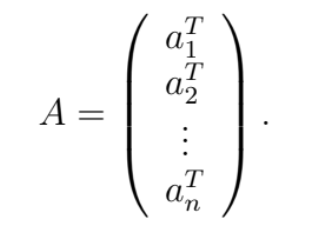
4. Наближення Ak+ = AT(AAT + δk2Em)−1;

5. Якщо || Ak+ − Ak-1+||< ε, то зупинитись з A+ = Ak+, iнакше k := k+1

i продовжити з пункту 3.

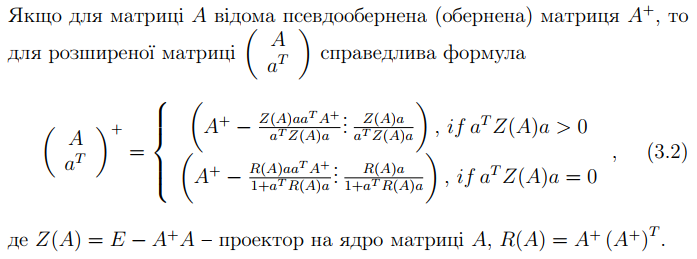
*Алгоритм знаходження псевдооберненої матриці на основi формули Гревіля:*

Цей алгоритм є рекурентним. Представляємо матрицю A у виглядi



Для першого кроку алгоритму , при , якщо . На наступному кроцi додаємо до матрицi другий рядок i шукаємо псевдообернену матрицю згiдно формули Гревiля. Потiм знову додаємо рядок i т.д. поки не вичерпаються всi рядки матрицi A.

Формула Гревіля:

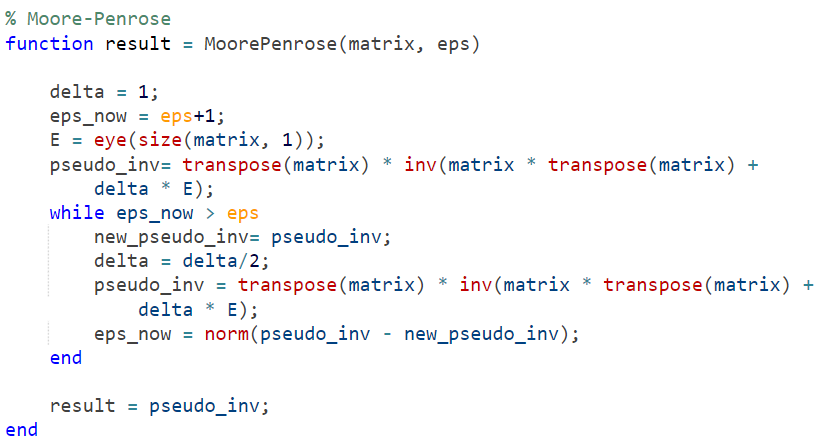


Розв’язання задачі

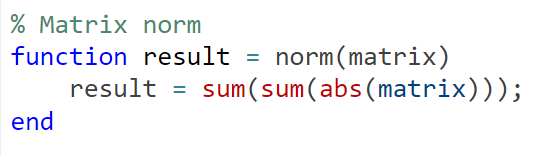
Зчитуємо вхідне і вихідне зображення.



Функція, яка знаходить псевдообернену матрицю на основi формули Мура-Пенроуза. Алгоритм подано вище.



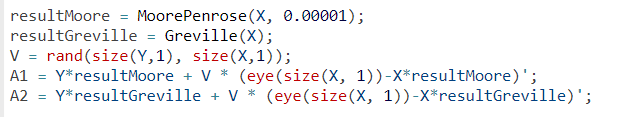
Норму матриці обраховуємо за допомогою функції:



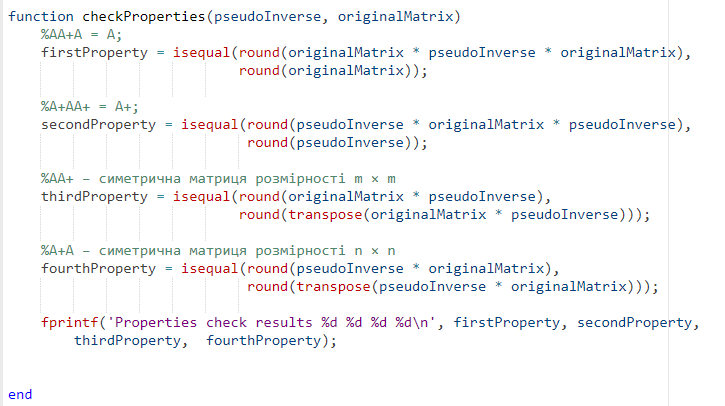
Функція, яка знаходить псевдообернену матрицю на основi формули Гревіля. Алгоритм подано вище.



Знаходимо псевдообернені матриці двома методами:

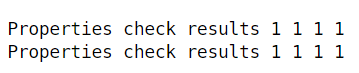


Проводимо перевірку на правильнiсть знаходження псавдооберненої матрицi за допомогою теореми 3.1 про характеристичну властивiсть псевдооберненої матрицi (наведена вище).

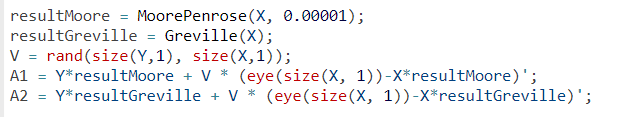




Результат:

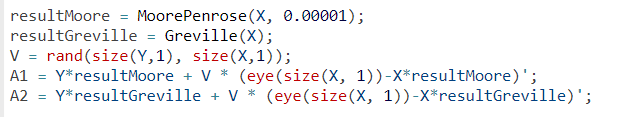


Результати свідчать про те, що і матриця, знайдена за допомогою алгоритма Мура-Пенроуза, і матриця, знайдена за допомогою алгоритма Гревіля, є псевдооберненими матрицями, адже всі чотири властивості виконуються.

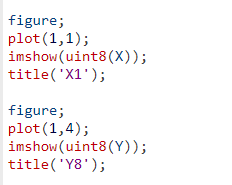
Задаємо рандомну матрицю V:

Знаходимо лiнiйний оператор переходу за допомогою знайдених раніше псевдообернених матриць за допомогою формули:

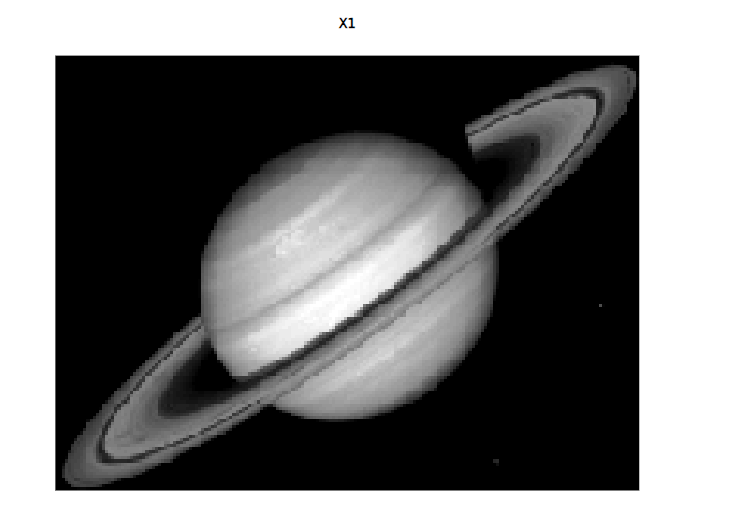




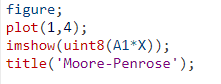
Виводимо результати. Спершу початкові вхідний та вихідний сигнали.

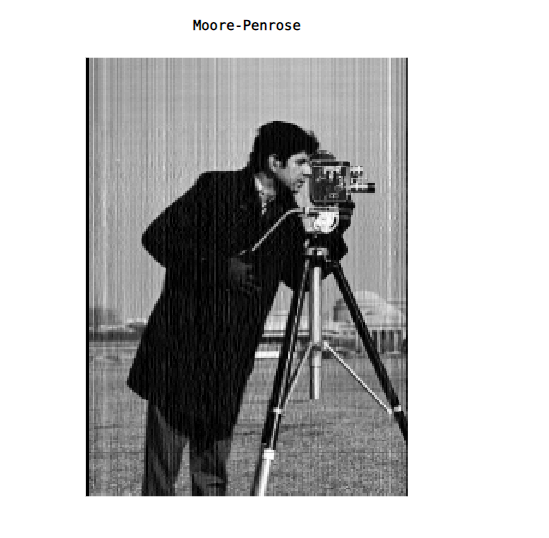


Отримали:

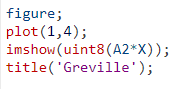
 

Далі виводимо результат для випадку, де псевдообернену матрицю шукали за допомогою формули Мура-Пенроуза:





Далі виводимо результат для випадку, де псевдообернену матрицю шукали за допомогою формули Гревіля:





Висновок

У ході роботи було вивчено означення псевдооберненої матриці та її основні властивості. Розроблено програму, яка приймає на вхід два зображення та знаходить лінійний оператор переходу між ними.

Були використані два методи знаходження псевдооберненої матриці: метод Мура-Пенроуза та метод Гревіля. Псевдообернені матриці, отримані обома методами, відповідають характеристичній властивості псевдооберненої матриці за теоремою 3.1.

Отримані зображення є досить близькими до початкового вихідного зображення, тому можемо зробити висновок, що програма успішно реалізує лінійний оператор перетворення між вхідним та вихідним зображеннями на основі обох формул.

Код програми

% Matrix norm

function result = norm(matrix)

result = sum(sum(abs(matrix)));

end

% Moore-Penrose

function result = MoorePenrose(matrix, eps)

delta = 1;

eps\_now = eps+1;

E = eye(size(matrix, 1));

pseudo\_inv= transpose(matrix) \* inv(matrix \* transpose(matrix) + delta \* E);

while eps\_now > eps

new\_pseudo\_inv= pseudo\_inv;

delta = delta/2;

pseudo\_inv = transpose(matrix) \* inv(matrix \* transpose(matrix) + delta\*delta \* E);

eps\_now = norm(pseudo\_inv - new\_pseudo\_inv);

end

result = pseudo\_inv;

end

% Greville

function result = Greville(matrix)

pseudo\_inv = 0;

a = matrix(1,:)';

matrix\_now = a';

if(a'\*a == 0)

pseudo\_inv = a;

else

pseudo\_inv = a / a' \* a;

end

i = 2;

while i <= size(matrix, 1)

Z = eye(size(pseudo\_inv , 1))-pseudo\_inv \* matrix\_now;

R = pseudo\_inv \*pseudo\_inv ';

row = matrix(i,:)';

matrix\_now = [ matrix\_now ; row'];

aZa = row'\*Z\*row;

aRa = 1+row'\*R\*row;

if(aZa == 0)

pseudo\_inv = [(pseudo\_inv -(R\*row\*row'\*pseudo\_inv )/aRa),(R\*row)/aRa];

else

pseudo\_inv = [(pseudo\_inv -(Z\*row\*row'\*pseudo\_inv )/aZa),(Z\*row)/aZa];

end

i = i + 1;

end

result =pseudo\_inv ;

end

%Pseudoinversion check

function checkProperties(pseudoInverse, originalMatrix)

%AA+A = A

firstProperty = isequal(round(originalMatrix \* pseudoInverse \* originalMatrix),

round(originalMatrix));

%A+AA+ = A+

secondProperty = isequal(round(pseudoInverse \* originalMatrix \* pseudoInverse),

round(pseudoInverse));

%AA+ – симетрична матриця розмiрностi m × m

thirdProperty = isequal(round(originalMatrix \* pseudoInverse),

round(transpose(originalMatrix \* pseudoInverse)));

%A+A – симетрична матриця розмiрностi n × n

fourthProperty = isequal(round(pseudoInverse \* originalMatrix),

round(transpose(pseudoInverse \* originalMatrix)));

fprintf('Properties check results %d %d %d %d\n', firstProperty, secondProperty, thirdProperty, fourthProperty);

end

%Initial signals

X = double(imread('x1.bmp'));

Y = double(imread('y8.bmp'));

%Pseudo-inverse matrices

resultMoore = MoorePenrose(X, 0.00001);

resultGreville = Greville(X);

%Check properties

checkProperties(resultMoore, X)

checkProperties(resultGreville, X)

%Random matrix

V = rand(size(Y,1), size(X,1));

%Сalculation of the linear operator

A1 = Y\*resultMoore + V \* (eye(size(X, 1))-X\*resultMoore)';

A2 = Y\*resultGreville + V \* (eye(size(X, 1))-X\*resultGreville)';

%Display the results

figure;

plot(1,1);

imshow(uint8(X));

title('X1');

figure;

plot(1,4);

imshow(uint8(Y));

title('Y8');

figure;

plot(1,4);

imshow(uint8(A1\*X));

title('Moore-Penrose');

figure;

plot(1,4);

imshow(uint8(A2\*X));

title('Greville');