

Лабораторна робота №1

Бінаризація зображень

Мета: вивчення засобів по бінаризації зображення, набуття практичних навичок їх використання. Вивчити основи роботи з цифровими зображеннями в системі MatLab.

Варіант 10

Варіант	Формат вхідних зображень				Поріг
	1	2	3	4	
	<i>Rock.bmp</i>	<i>kamen1.jpg</i>	<i>Rock.bmp</i>	<i>kamen.bmp</i>	<i>graythresh</i>

Хід роботи:

1. Вивчити основні можливості системи MatLab по перетворенню кольорових зображень в бінарне.
2. Виконати бінаризацію зображень для вхідних форматів, що представлені в таблиці 1.1, згідно варіанту:

1) Виконати бінаризацію палітрового зображення з використанням автоматичного порогу.

На рисунку 1.1 представлений код для перетворення зображення rock.bmp(обране за 10 варіантом) в бінарне за допомогою автоматичного вибору порогу.

```
lab1_1.m  x  +
1  [X,map]=imread('D:\Study\5 курс\1 семестр\Цифрова обробка сигналів і систем\lab_1\rock.bmp');
2  I=im2double( ind2gray(X,map));
3  figure,imshow(I)
4  T = 0.5*(min(I(:)) + max(I(:)));
5  done = false;
6  while ~done g=I>= T;
7      Tnext = 0.5*(double(min(I(g))) + double(max(I(~g))));
8      done = abs(T - Tnext) < 0.5;
9      T = Tnext;
10 end
11 bw = I> T;
12 figure, imshow(bw)
```

Рис. 1.1. Код для перетворення зображення rock.bmp в бінарне за допомогою автоматичного вибору порогу

Виконання коду:

На початку ми мали вхідне зображення рис.1.2.



Рис. 1.2. Вхідне зображення

Після запуску коду ми отримали результат перетворення у бінарне зображення показане на рисунку 1.3.

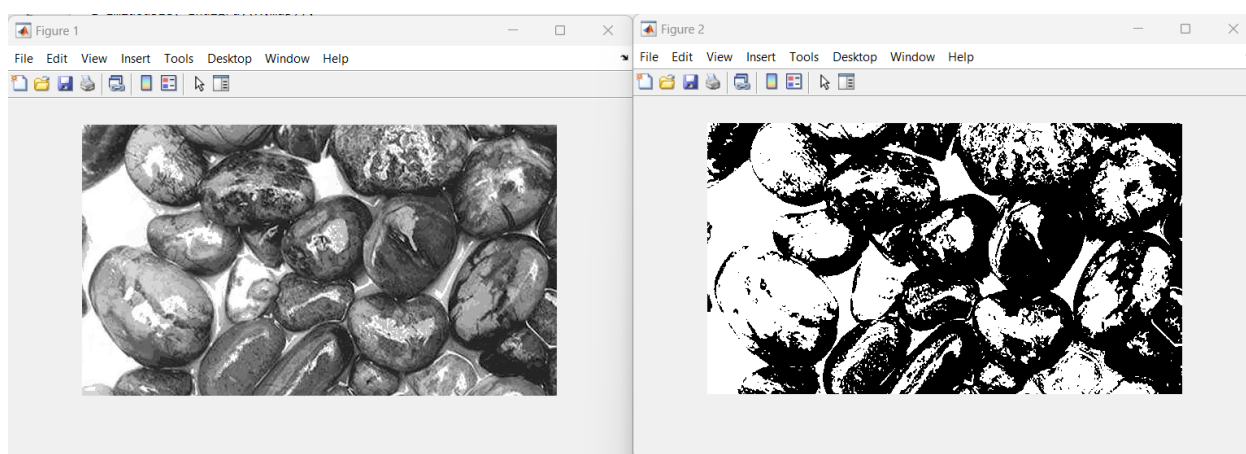


Рис. 1.3. Результат виконання коду.

2) Виконати бінаризацію кольорового зображення з використанням функції `graythresh`.

Функція `graythresh` пакета IPT, обчислює поріг по методу Отса, та має наступний синтаксис:

$T = \text{graythresh}(S),$

де S - вихідне напівтонове зображення; T - глобальний поріг в інтервалі $[0 \ 1]$.

За 10-м варіантом необхідно виконати бінаризацію зображення `камен1.jpg`. Але в наданих нам файлах ми маємо зображення з розширенням `.bmp`. Тому конвертуємо зображення в необхідних нам формат(рис. 2.1), та створюємо код, зображений на рис. 2.2.



Рис. 2.1. Вхідне зображення для перетворення

```
lab1_1.m  lab1_2.m  +
1  rgb = imread('D:\Study\5 курс\1 семестр\Цифрова обробка сигналів і систем\lab_1\kamen1.jpg');
2  I= im2double(rgb2gray(rgb));
3  figure,imshow(I)
4  T = graythresh(I);
5  Bw=I > T;
6  figure, imshow(Bw)
```

Рис. 2.2. Код для перетворення зображення kamen1.jpg в бінарне за допомогою функції *graythresh*

Після запуску програми ми отримали наступні бінарні зображення(рис. 2.3).

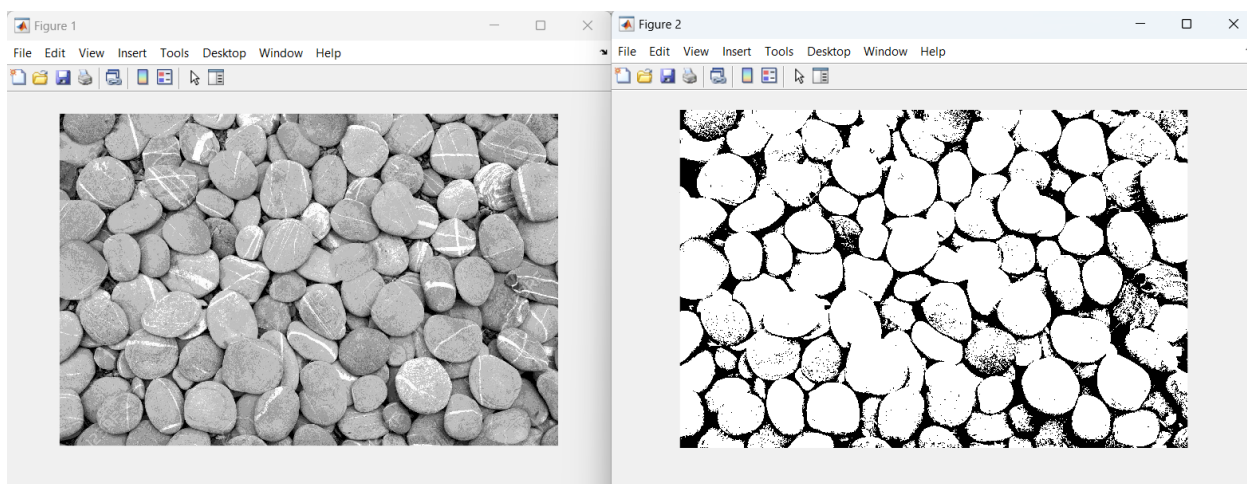


Рис. 2.3. Результати перетворення в бінарне зображення

3)Отримати негатив за допомогою бінаризації по порогу палітрового зображення.

За нашим варіантом необхідно отримати негатив зображення rock.bmp. На початку ми мали вхідне зображення рис.3.1.



Рис. 3.1. Вхідне зображення

Створимо код для даної операції(рис. 3.2).

```
lab1_1.m  lab1_2.m  lab1_3.m  +
1  [X,map] = imread('D:\Study\5 курс\1 семестр\Цифрова обробка сигналів і систем\lab_1\rock.bmp' );
2  I = ind2gray(X,map);
3  figure,imshow(I)
4  T=graythresh(I);
5  BW = I < T;
6  figure,imshow(BW)
```

Рис. 3.2. Код для отримання негативу

Та після запуску ми отримали результат зображень на рис. 3.3.

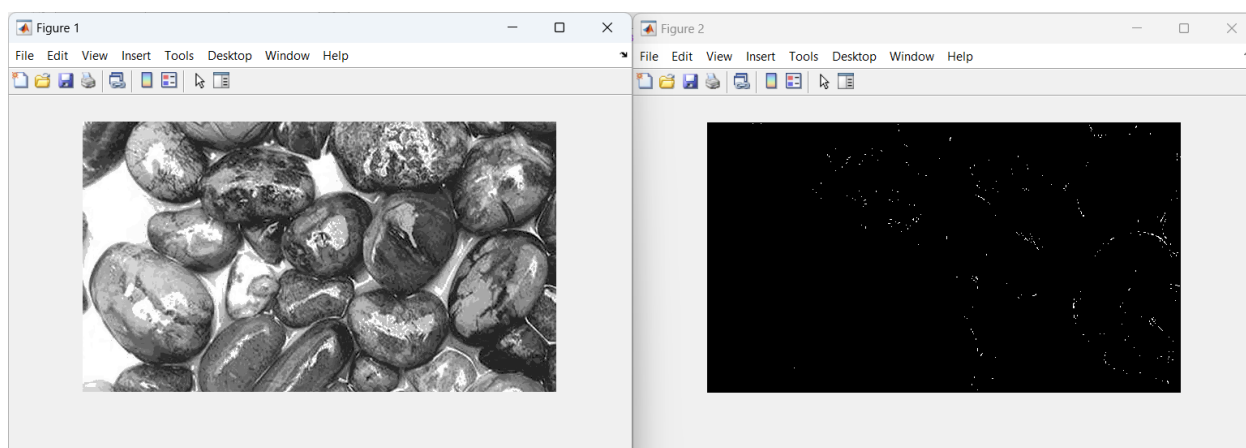


Рис. 3.3. Результат виконання коду наведеного вище

4) Виконати бінаризацію палітрового зображення з використанням заданого значення порогу.

В системі MatLab бінаризація зображення відсіканням по порогу яскравості виконується функцією `im2bw`.

Синтаксис:

$BW = im2bw(I, threshold)$

$BW = im2bw(X, map, threshold)$

$BW = im2bw(RGB, threshold)$

Функція `im2bw` створює бінарне зображення, використовуючи відсікання по порогу яскравості `threshold`. Для цієї мети функція конвертує повнокольорові і палітрові зображення в напівтонові. Попіг `threshold` повинен задаватися в діапазоні $[0,1]$. За замовчуванням `threshold = 0.5`.

За варіантом 10 виконаємо бінаризацію палітрового зображення `kamen.bmp`(рис. 4.1).



Рис. 4.1. Вхідне зображення

Створимо код, для бінаризації палітрового зображення з використанням заданого значення порогу(рис. 4.2).

```
lab1_1.m x lab1_2.m x lab1_3.m x lab1_4.m x +
1 [X,map] = imread('D:\Study\5 курс\1 семестр\Цифрова обробка сигналів і систем\lab_1\kamen.bmp');
2 figure,imshow(X,map)
3 BW = im2bw(X,map);
4 figure,imshow(BW)
```

Рис. 4.2. Код для бінаризації

Нижче представлений результат коду наведеного вище(рис. 4.3).

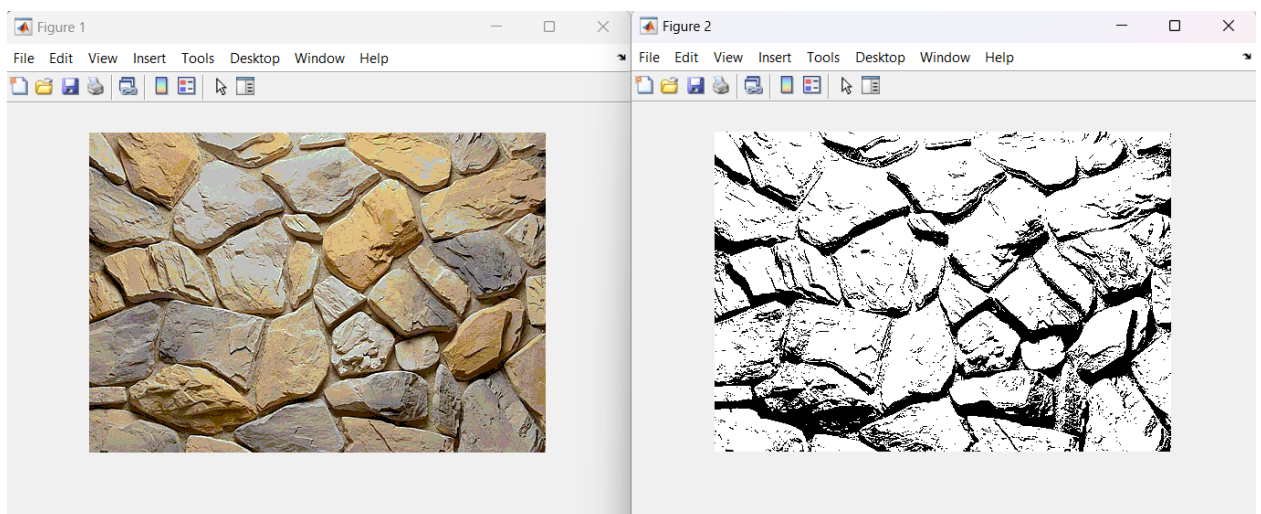


Рис. 4.3. Бінаризація палітрового зображення

5) Обчислити поріг за допомогою функції `graythresh` для 1,2,3 вхідних зображень.

Обчислимо поріг за допомогою функції `graythresh` для зображень:

а) `rock.bmp`;

б) `kamen1.bmp`;

в) `rock.bmp`.

а) Створимо код для обчислення порогу заданого зображення(рис.5.1).

```
lab1_1.m x lab1_2.m x lab1_3.m x lab1_4.m x a.m x +
1 [X,map] = imread('D:\Study\5 курс\1 семестр\Цифрова обробка сигналів і систем\lab_1\rock.bmp');
2 I = im2double(ind2gray(X,map));
3 figure,imshow(I)
4 T = graythresh(I);
5 BW = im2bw(I,T);
6 figure,imshow(BW)
```

Рис. 5.1. Код програми для обчислення порогу першого зображення

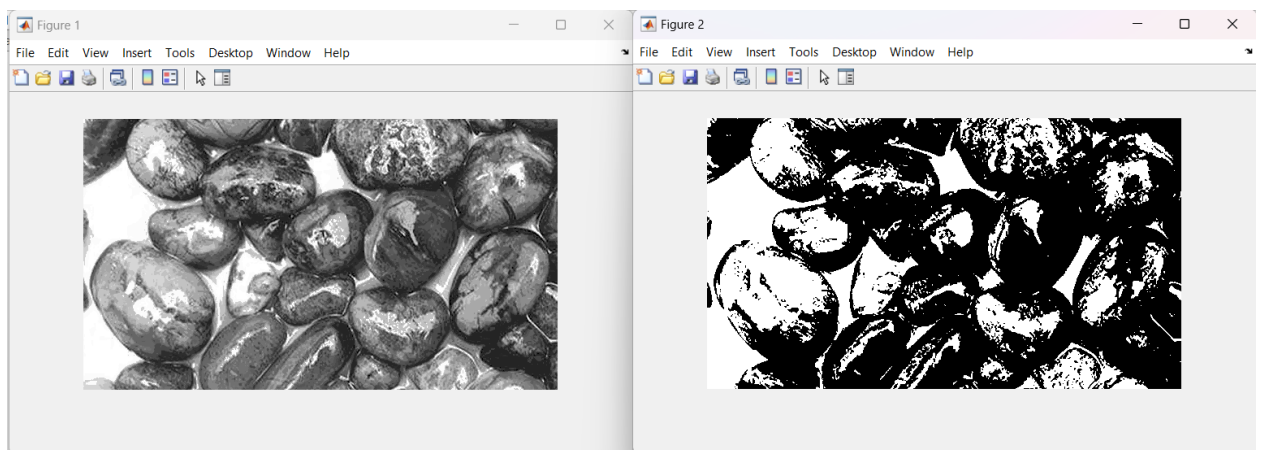


Рис. 5.2. Результат роботи коду.

а) Створимо код для обчислення порогу заданого зображення(рис.6.1).

```
lab1_1.m x lab1_2.m x lab1_3.m x lab1_4.m x a.m x b.m x +
1 [X,map] = imread('D:\Study\5 курс\1 семестр\Цифрова обробка сигналів і систем\lab_1\kamen1.bmp');
2 I = im2double(ind2gray(X,map));
3 figure,imshow(I)
4 T = graythresh(I);
5 BW = im2bw(I,T);
6 figure,imshow(BW)
```

Рис. 6.1. Код програми для обчислення порогу першого зображення

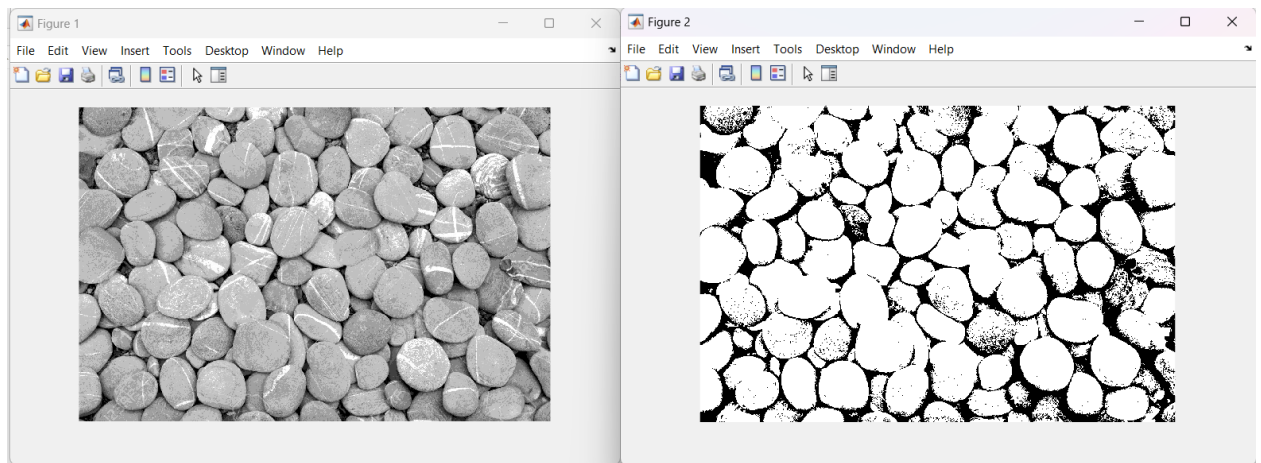


Рис. 6.2. Результат роботи коду.

Висновок: У ході даної лабораторної роботи було досліджено методи бінаризації напівтонових зображень, зокрема пороговий поділ та використання функції `graythresh`. Основним завданням бінаризації є перетворення напівтонового зображення у бінарне, що полегшує його подальшу обробку та аналіз. Було розглянуто глобальне та локальне порогове розділення. При глобальному підході використовується один поріг для всього зображення, що підходить для зображень з однорідними значеннями фону та об'єкта. Локальне порогове розділення, у свою чергу, є ефективним для зображень з неоднорідними областями, оскільки передбачає поділ зображення на менші частини та застосування індивідуального порогу для кожної з них.

Практична частина роботи продемонструвала застосування різних підходів до бінаризації за допомогою Matlab. Особливо корисними виявились автоматичні методи вибору порогових значень, такі як метод Отсу, що дозволяє уникнути ручного налаштування порогу і підвищує точність обробки.

Відповідь на контрольні запитання:

1. Як можна отримати бінарне зображення?