

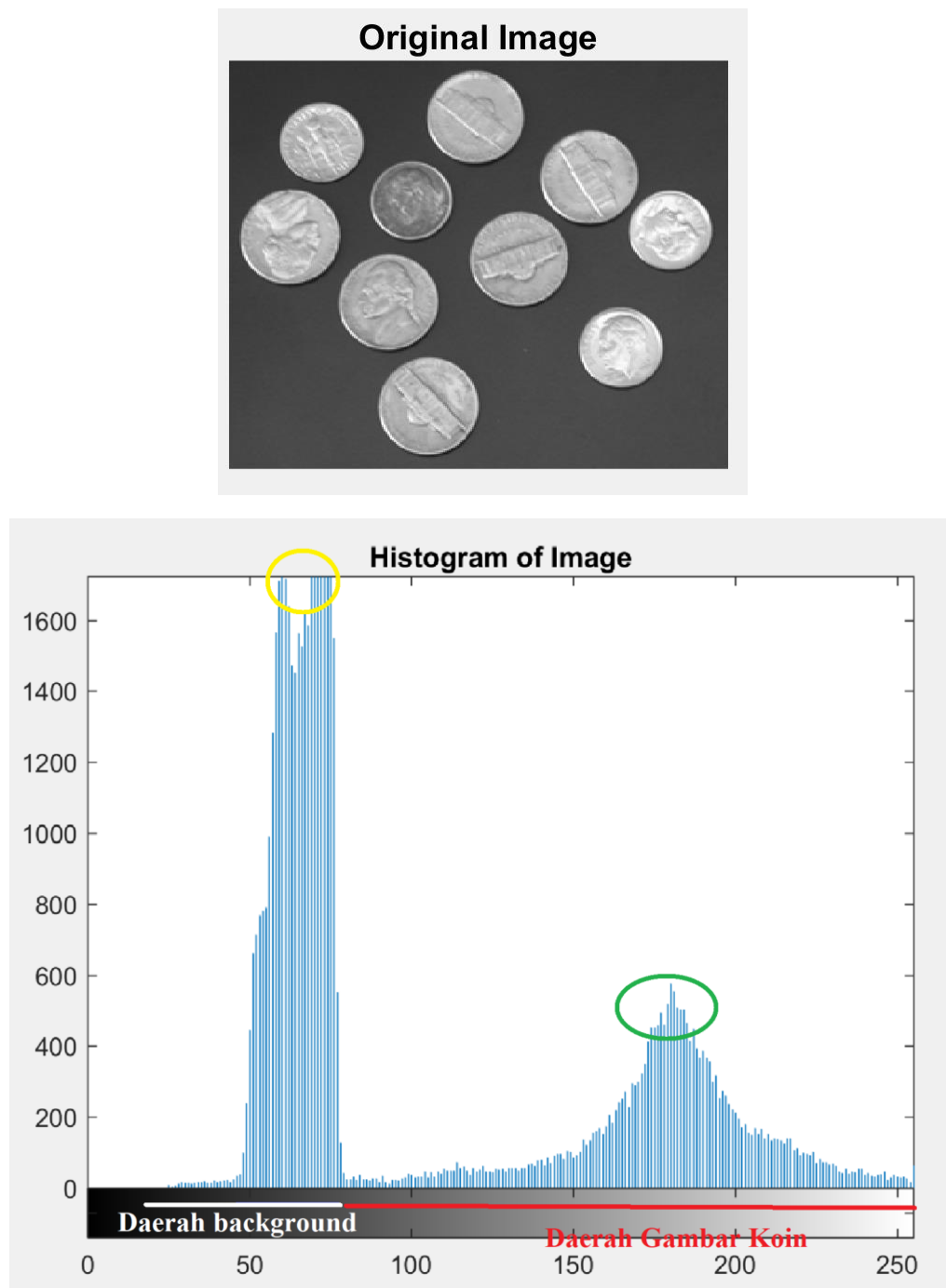
Nama : Johan Ambarita

Nim : 14S15045

15.1: IMAGE THRESHOLDING

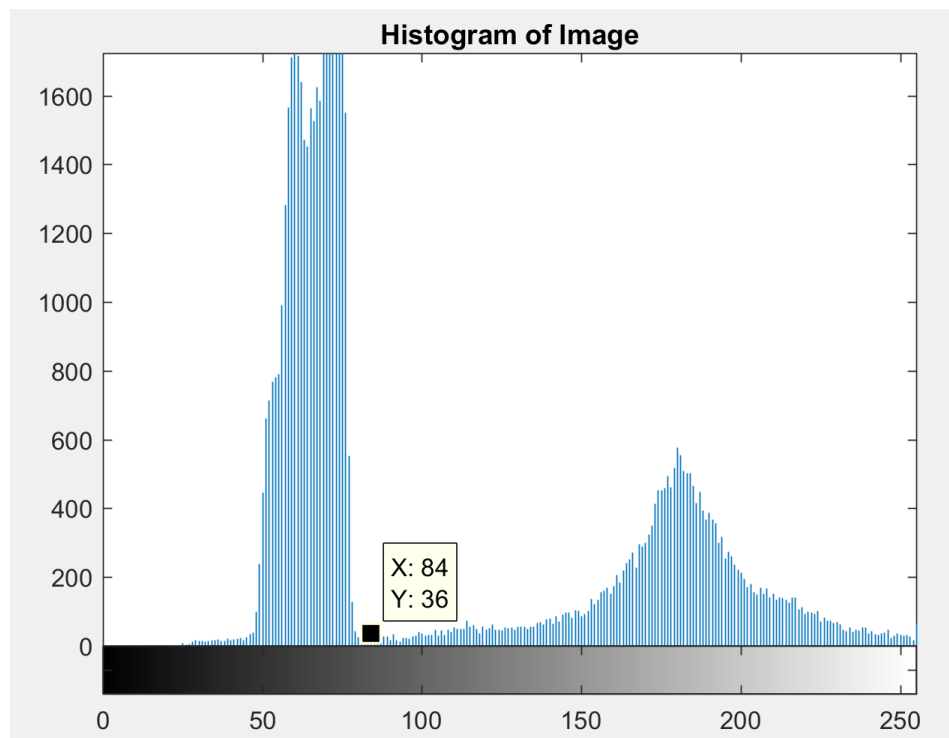
Global Thresholding

Question 1



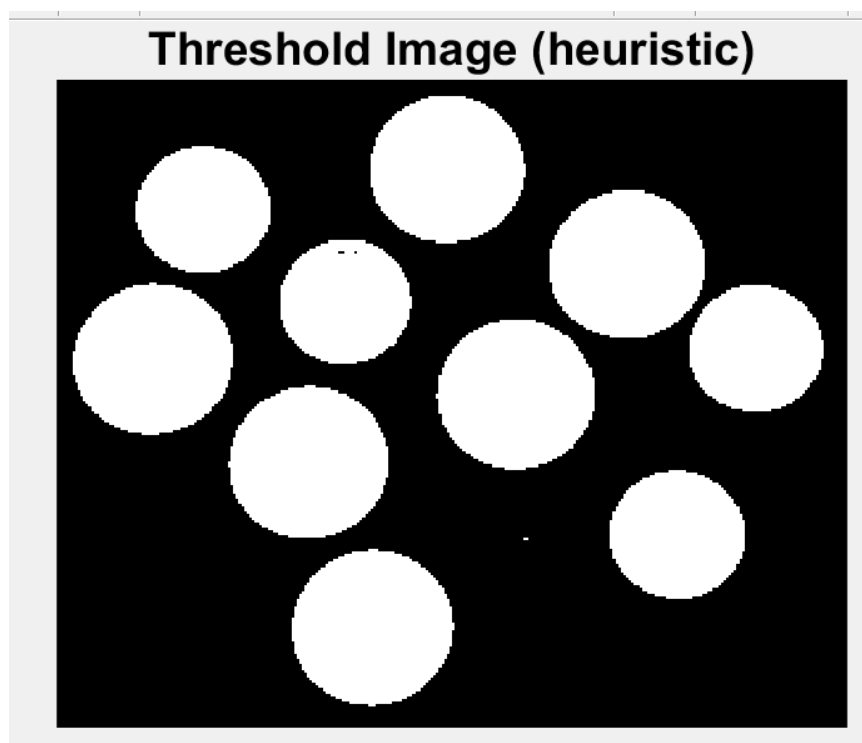
Pada gambar dapat dilihat bahwa gambar dapat dikelompokkan menjadi 2 bagian yaitu daerah yang mewakili background gambar dan daerah objek dari gambar yaitu koin. Untuk dari puncak tiap daerah dapat lihat pada gambar dengan keterangan

- Lingkaran kuning mewakili puncak dari daerah background
- Lingkaran hijau mewakili puncak dari daerah objek koin



Pada gambar diatas dapat dilihat dengan menggunakan fungsi *inspect* , user dapat mengetahui informasi (frekuensi dan intensitas) dari titik yang diplot.

Dengan membuat *threshold* senilai 85, maka gambar yang akan didapat sebagai berikut :



Question 2

Fungsi `im2bw` memiliki kemampuan untuk mengubah gambar menjadi gambar biner dengan menentukan *threshold*-nya. Dengan ketentuan :

- Nilai pixel gambar $> threshold$, maka nilai pixel diubah menjadi 1 (putih)
- Nilai pixel gambar $< threshold$, maka nilai pixel diubah menjadi 0 (hitam)

Question 3

Karena pada output dari fungsi ini, bertipe logical disebabkan hanya 2 nilai saja yang diketahui yaitu hitam dan putih, untuk itu nilai pixel harus berada di rentang 0 sampai 1. Untuk itu nilai *threshold* yang ditentukan yaitu :

$$X = \frac{85}{255} = 0,333$$

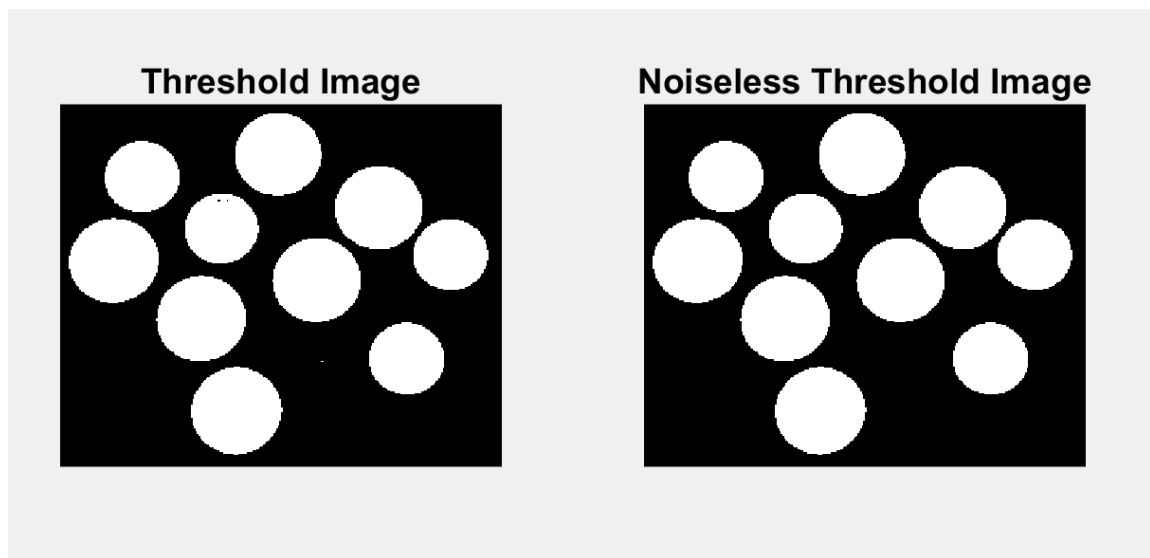
Nilai *threshold* dibagi 255 karena nilai maksimal dari pixel gambar yaitu 255.

Question 4

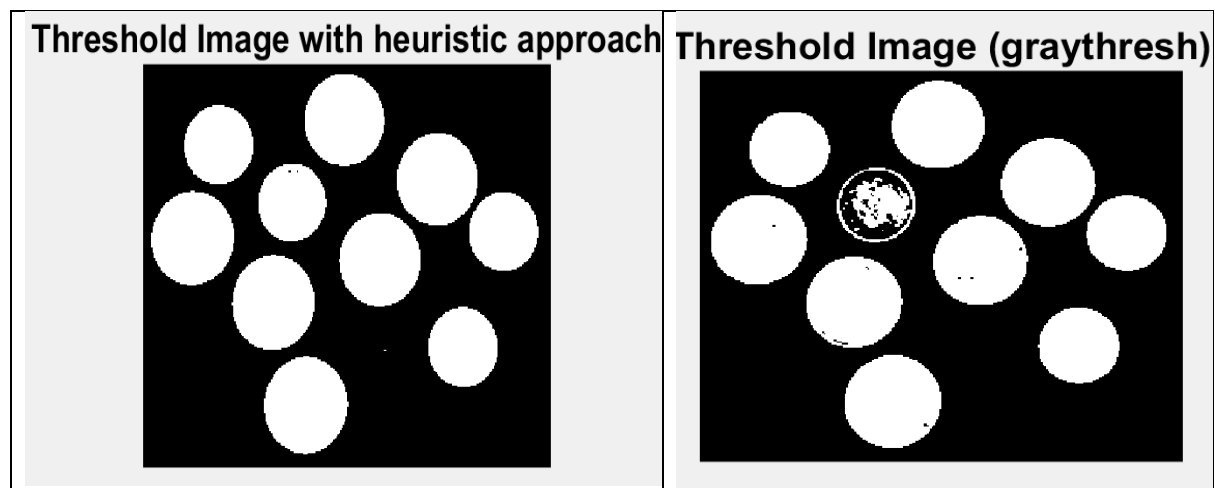
Dengan menggunakan median filter berukuran 3 x 3, noise pada *threshold image* dapat dihilangkan.

```
T = 85;
I = imread('coins.png');
I_thresh = im2bw(I, (T / 255));
I_filt = medfilt2(I_thresh,[3 3]);
subplot(1,2,1), imshow(I_thresh), title('Threshold Image');
subplot(1,2,2), imshow(I_filt), title('Noiseless Threshold Image');
```

Dengan menggunakan kode diatas, gambar yang dihasilkan terbukti dapat menghilangkan noise pada gambar.



Question 5



Jika membandingkan kedua gambar, maka image thresholding yang lebih baik menggunakan metode heuristic karena noise yang dihasilkan lebih sedikit dari pada menggunakan metode otsu, tetapi metode ini memiliki kelemahan yaitu memerlukan nilai threshold yang pas agar noise yang dihasilkan gambar sedikit. Untuk mencari nilai threshold ini dilakukan dengan melakukan uji coba pada gambar yang memakan waktu dan tenaga.

Adaptive Thresholding

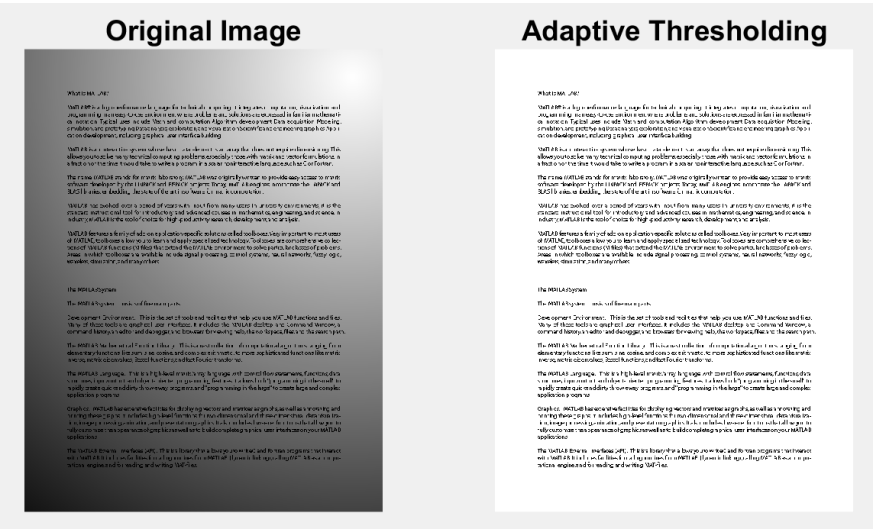
Question 6

Perbedaannya pada variabel std_without_text bernilai standar deviasi dari matriks dengan ukuran 10 x 10 pada kolom & baris ke 1 sampai 10, sedangkan std_with_text bernilai standar deviasi dari matriks dengan ukuran 11 x 11 pada kolom & baris ke 100 sampai 110

Question 7

Dengan memanfaatkan standar deviasi dari gambar yang ada teks dan tidak ada teks, dapat digunakan untuk membedakan posisi/bagian yang mana saja yang ada teksnya dan tidak ada teksnya. Setelah mendapatkan standar deviasinya, maka gambar akan disortir pada thresholdnya sesuai dengan operasi tertentu pada bagian tertentu sesuai dengan ada/tidaknya teks dari gambar.

Question 8



Dari gambar dapat dilihat bahwa dengan metode adaptive thresholding dengan memanfaatkan standar deviasi dari gambar yang tidak ada teks dan ada teksnya, terlihat image thresholding yang dilakukan berhasil dilakukan dengan output yang cukup memuaskan.

Question 9

Kelemahannya jelas pada gambar bahwa gambar yang dihasilkan memiliki kualitas yang buruk dan apabila gambar background dari gambar sama dengan nilai grayscale pada teks, maka teks akan hilang juga.