

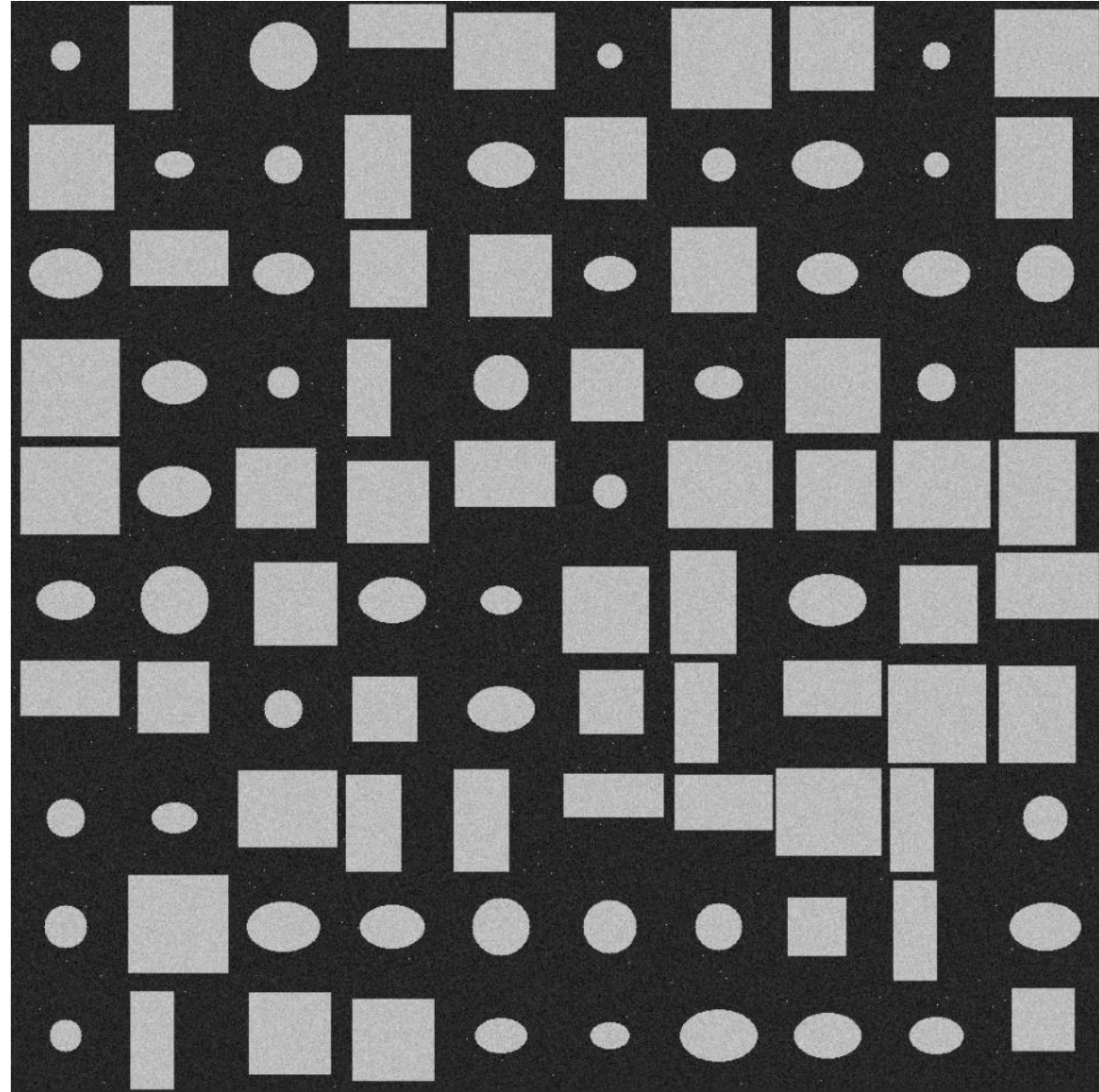
Geometric shapes

How many squares?

How many circles?

How many rectangles?

How many ellipses?

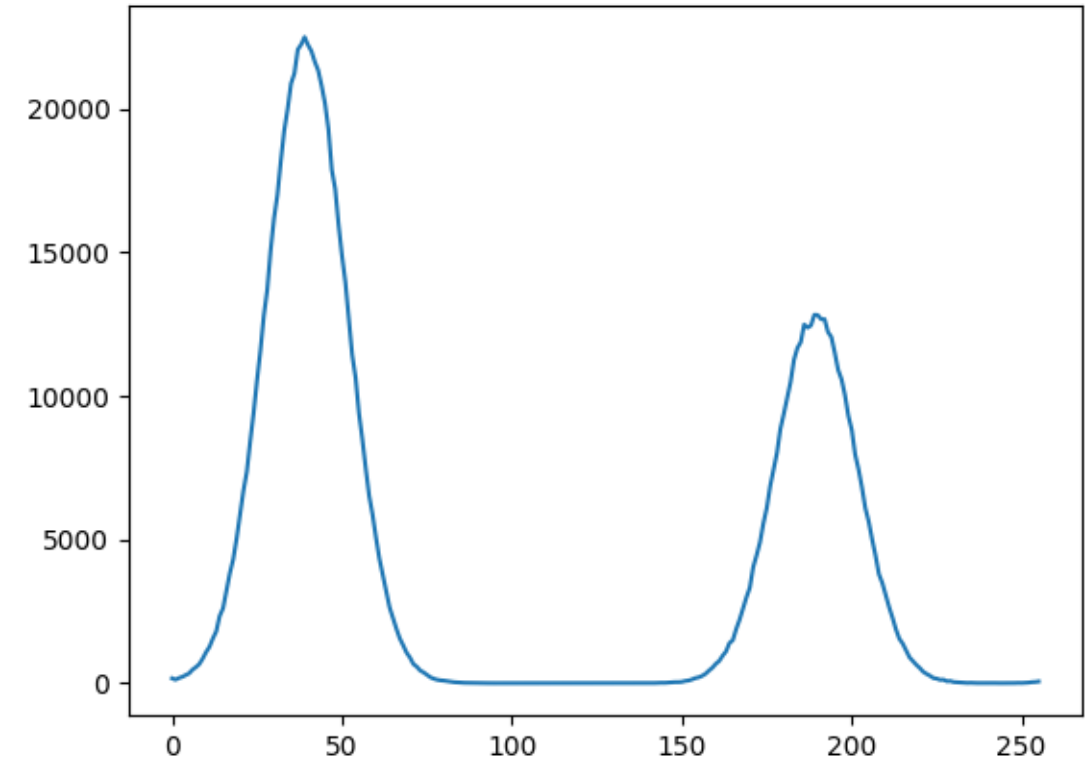
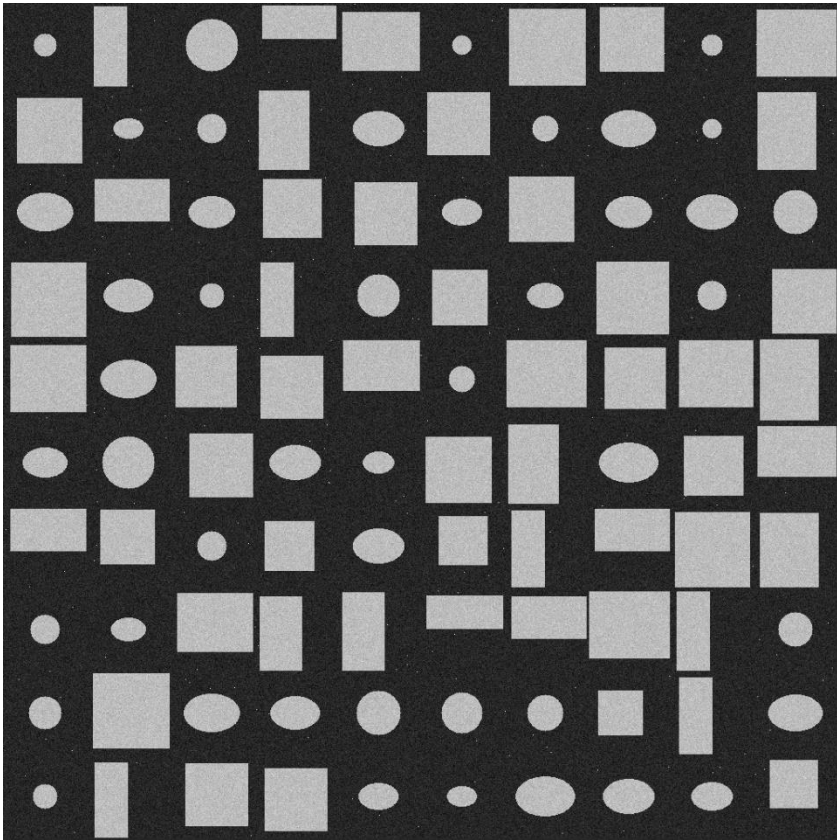


Assignment - 2

1. Binary Image Processing
 - a. Thresholding
 - b. Blob Coloring
 - c. Region Analysis
2. Compression
 - a. Run-length encoding
 - b. Decoding

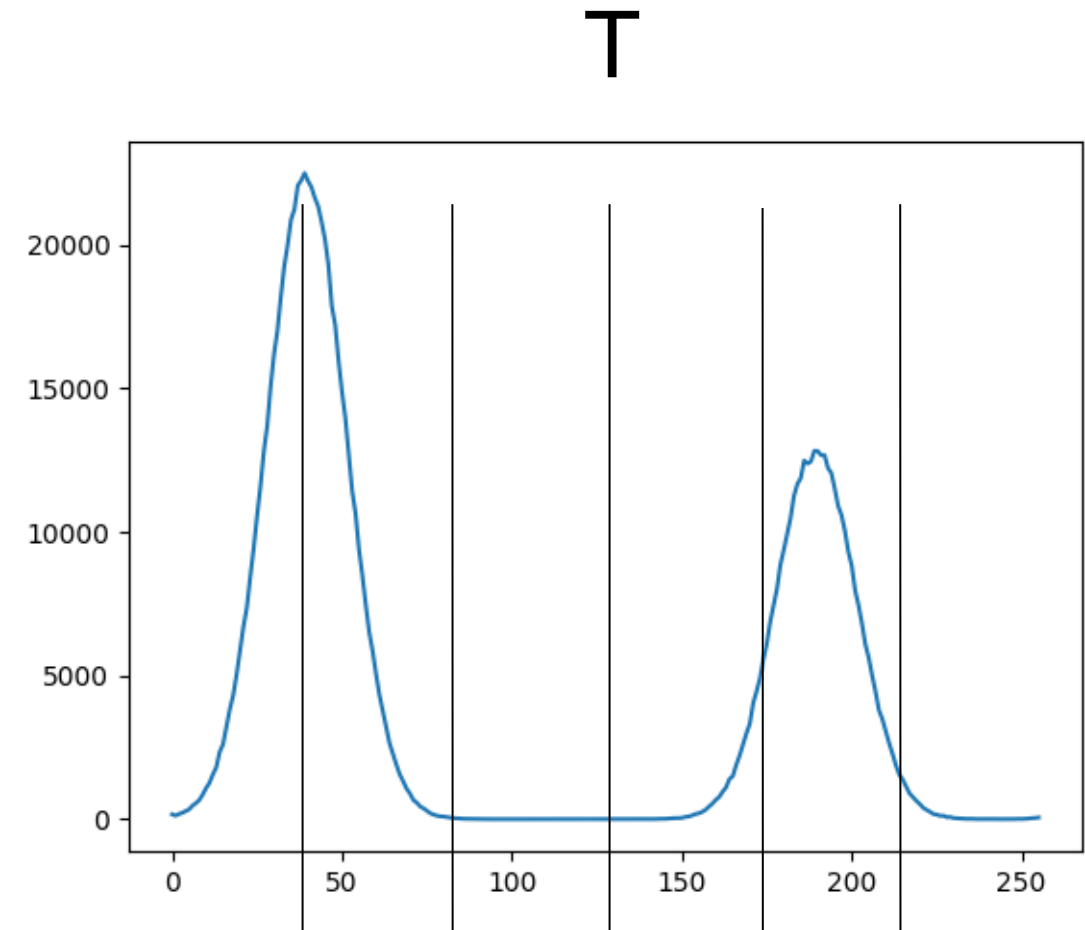
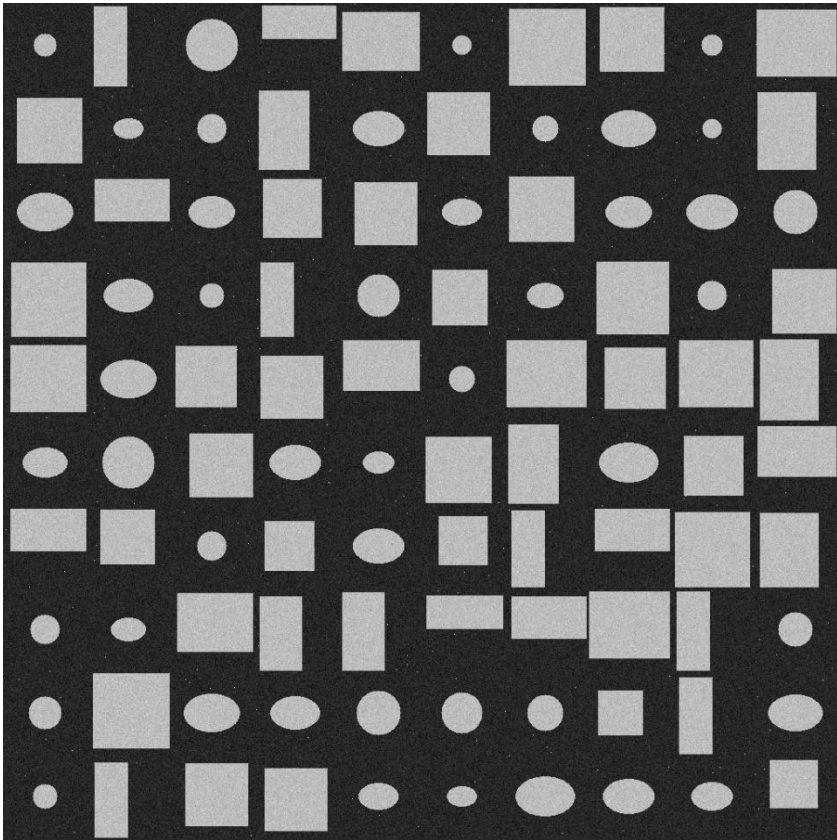
1. Binary Image Processing

- a. Thresholding
 - Compute Histogram



1. Binary Image Processing

- a. Thresholding
 - Compute Histogram
 - Optimal Threshold (Iterative Method: Week 4, Lecture 7, slide 40)
 - Do not implement Otsu's method



Algorithm (Optimal Threshold)

Initialize $T = K/2$

Do

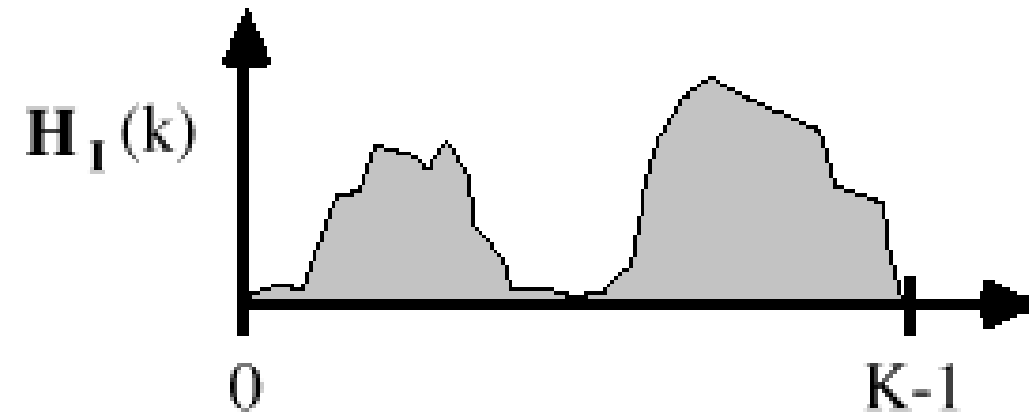
Compute $\mu_1 = E(X) \forall X < T$

Compute $\mu_2 = E(X) \forall X \geq T$

Set $T = \frac{\mu_1 + \mu_2}{2}$

While $\Delta\mu_1 \neq 0 \ \& \ \Delta\mu_2 \neq 0$

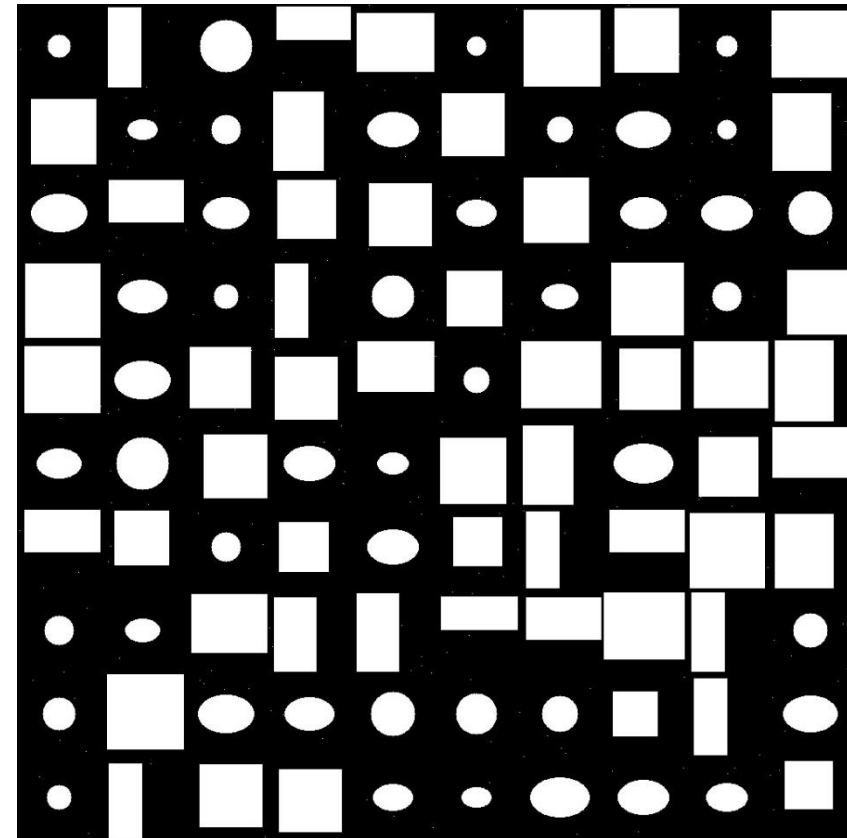
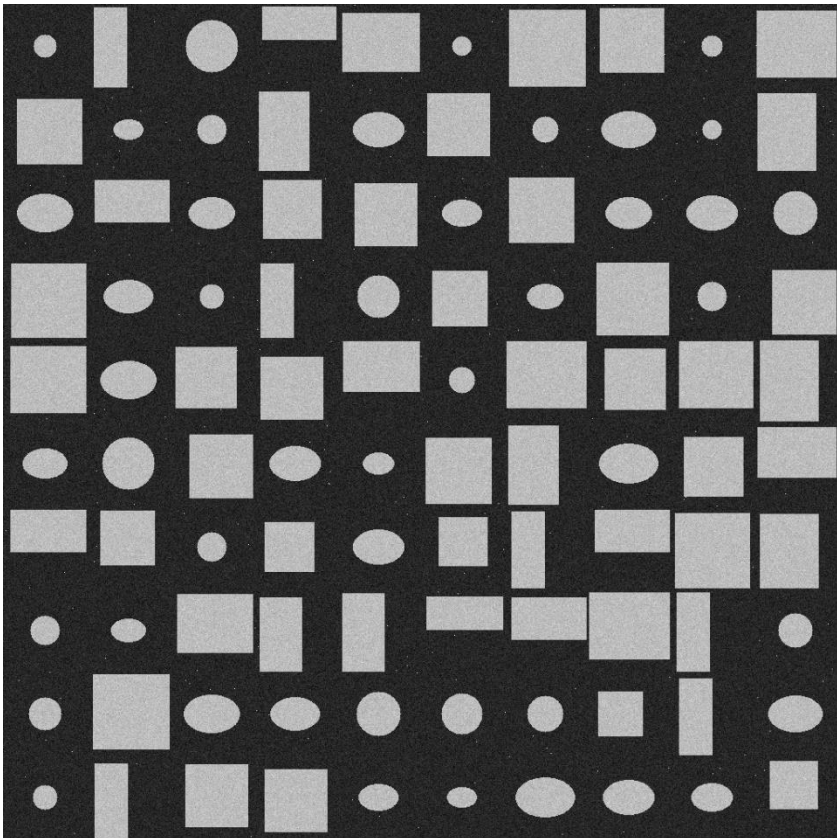
AKA: Expectation Maximization (simple version)



bimodal histogram
well separated peaks

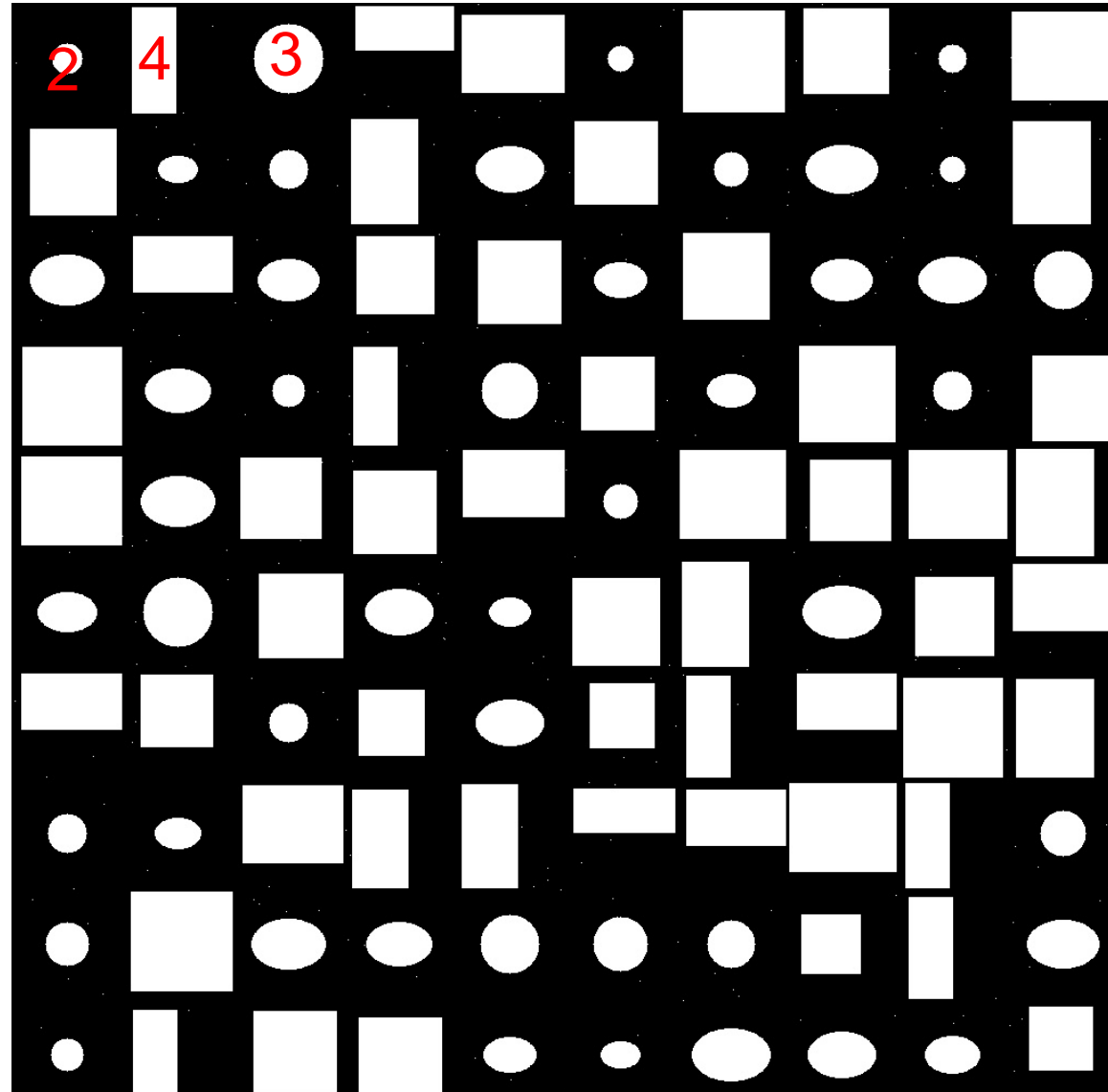
1. Binary Image Processing

- a. Thresholding
 - Compute Histogram
 - Optimal Threshold
 - Create Binary Image (Thresholding)



1. Binary Image Processing

- b. Blob Coloring



1. Binary Image Processing

- c. Region Analysis
 - C.1 compute shape statistics
 - For each region Compute Statistics
 - Compute area
 - Compute centroid
 - Identify shape (circle, square, rectangle, or ellipse)
 - Print statistics
 - Example: Region: 871, centroid: (969.11, 51.11), area: 707, shape: c

1. Binary Image Processing

- c. Region Analysis

- C.2 Count shapes

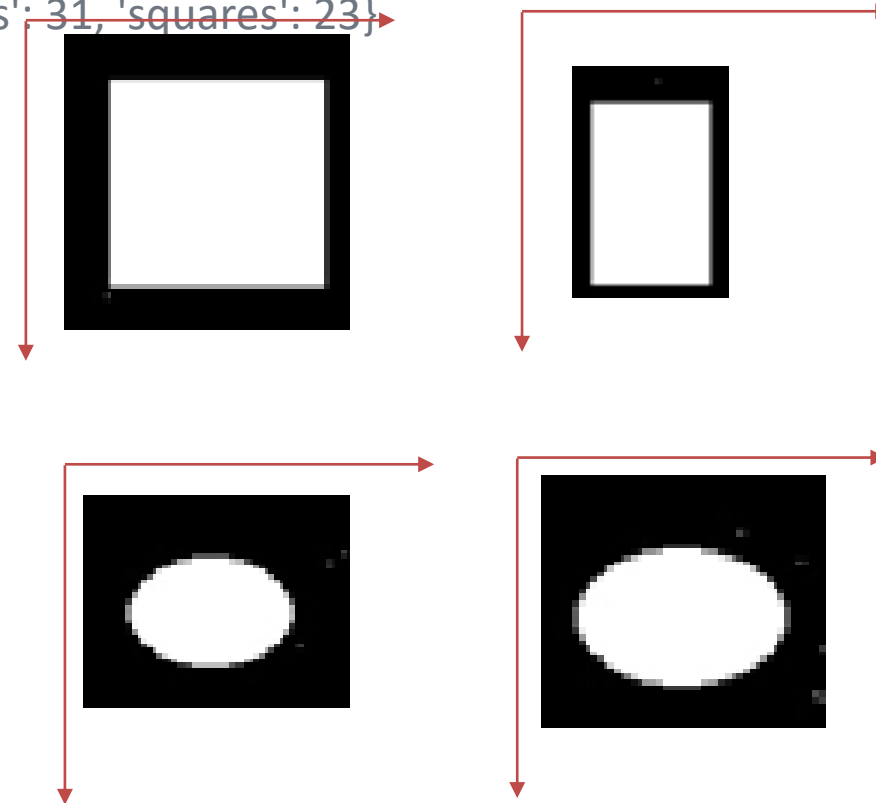
- Count Shapes:

- Count number of circles
 - Count number of squares
 - Count number of rectangles
 - Count number of ellipses

- Example: {'circles': 21, 'ellipses': 25, 'rectangles': 31, 'squares': 23}

- Assumption

- Squares and rectangles have side always parallel to the image co-ordinate system.
 - Ellipses have major and minor axis parallel to the image co-ordinate system.
 - Shapes are not rotated.



1. Binary Image Processing

- c. Region Analysis

- C.2 Count shapes

- Count Shapes:

- Count number of circles
 - Count number of squares
 - Count number of rectangles
 - Count number of ellipses

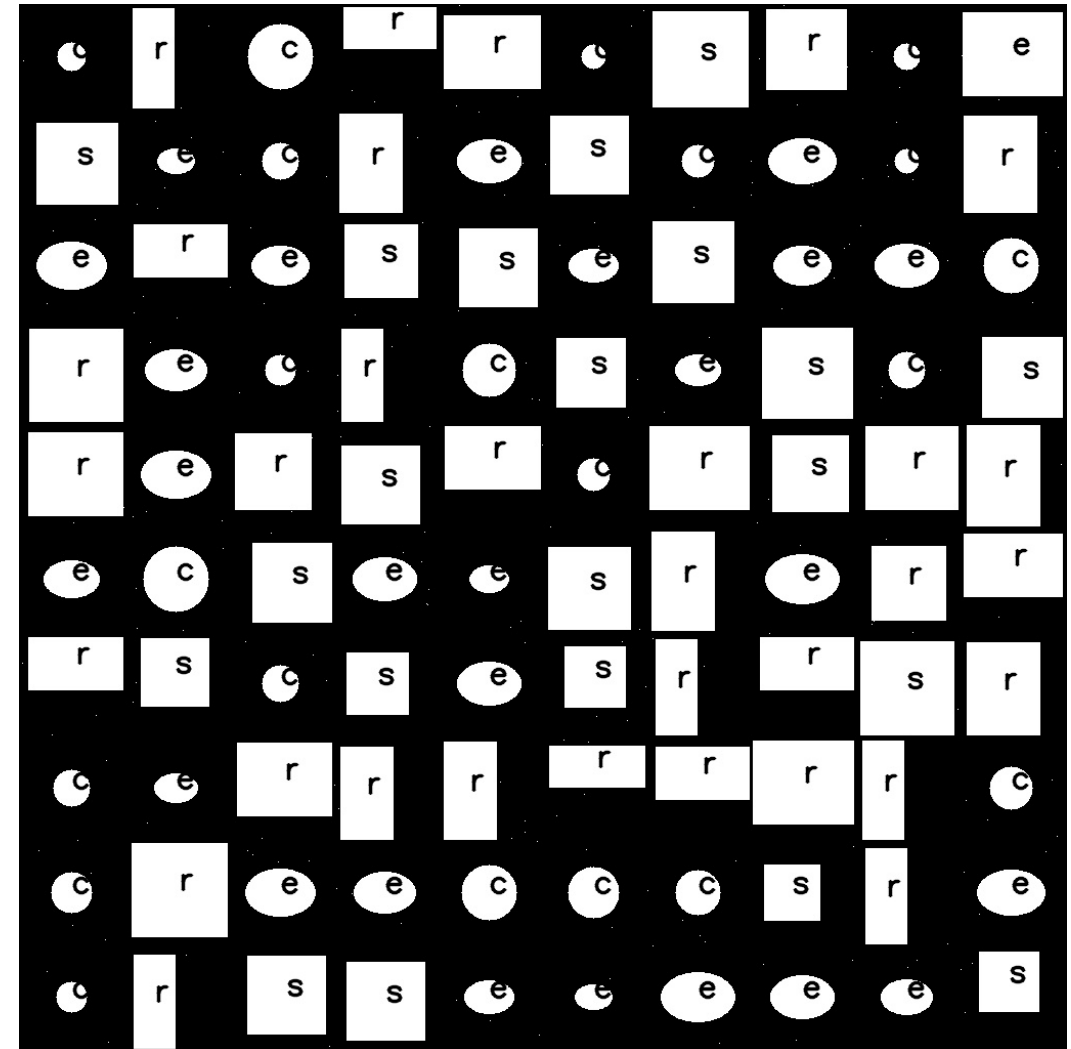
- Example: {'circles': 21, 'ellipses': 25, 'rectangles': 31, 'squares': 23}

Groundtruth ==> {'circles': 26, 'ellipses': 23, 'rectangles': 23, 'squares': 28}

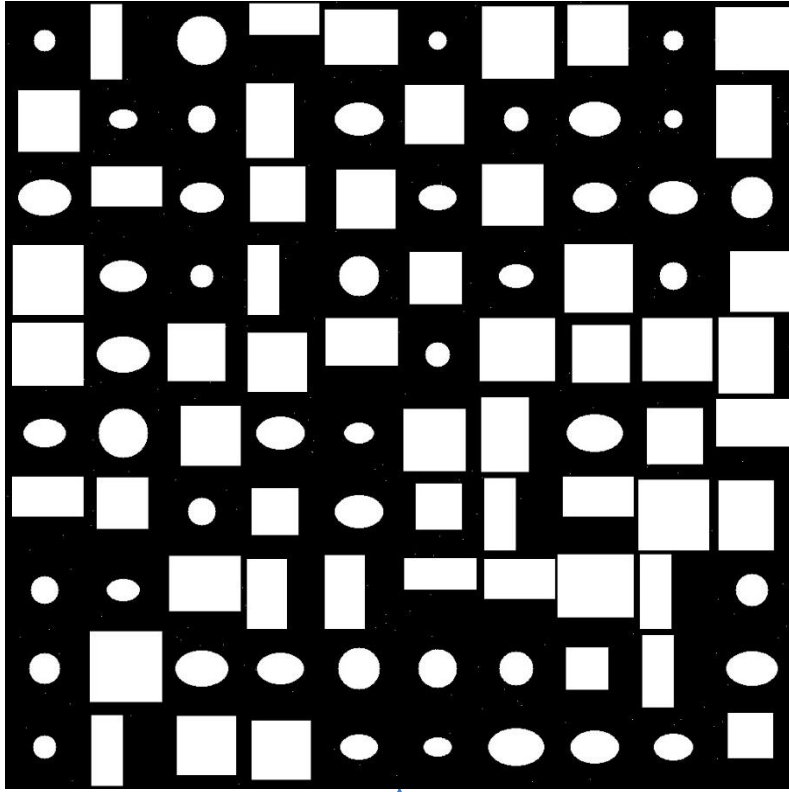
- Acceptable for the counts to be slightly off.
 - Sometimes it may be difficult to distinguish between rectangle and square, for example, if the height and width of the rectangle is only one pixel.
 - Similarly, if the ellipse is almost a circle, it may be difficult to distinguish them.

1. Binary Image Processing

- c. Region Analysis
 - C.3 Mark regions
 - Mark each shape center with a character.
 - c – circle
 - r – rectangle
 - s – square
 - e - ellipse

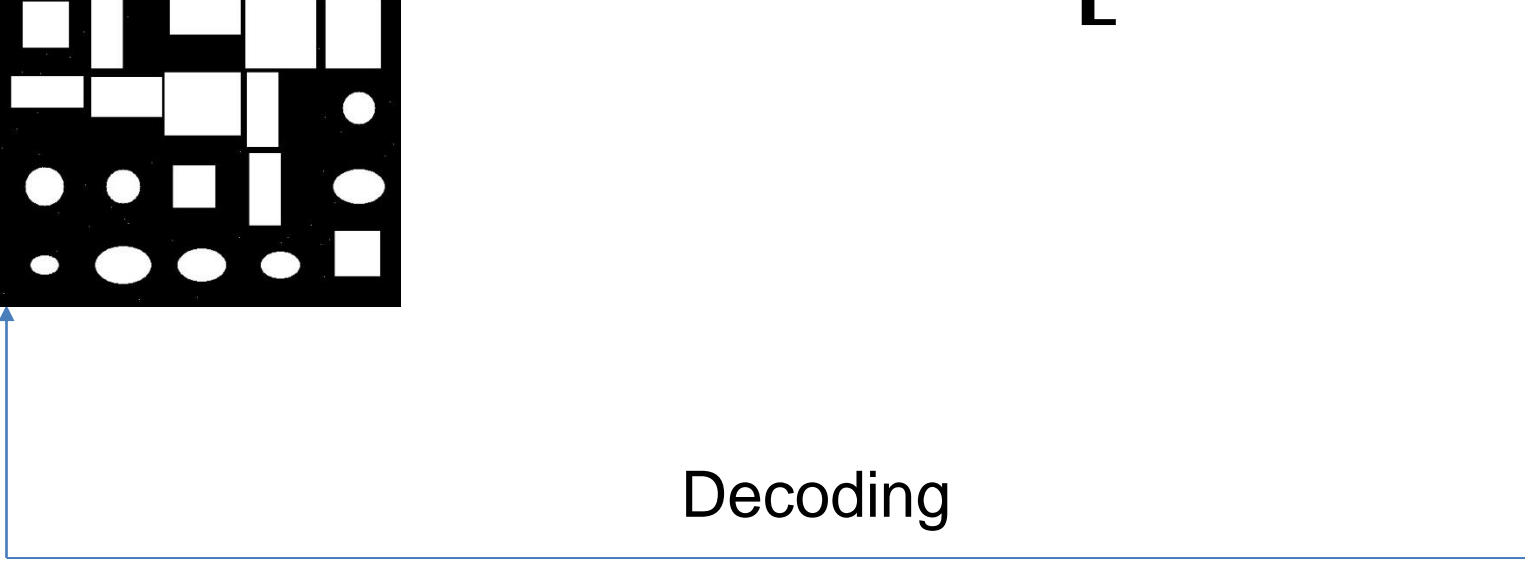


2. Compression



Run Length Encoding [1 10 100 ...]

Decoding



Assignment -2

1. Binary Image Processing
 - a. Thresholding (15 Pts.)
 - b. Blob Coloring (30 Pts.)
 - c. Region Analysis (25 Pts.)
2. Compression (30 Pts)
 - a. Run-length encoding
 - b. Decoding

Total: 100 Pts.

Due Date: March 21st

Submission Instructions

- Must use the **starter code** available in **Github**
- Submission allowed only through **Github**
- You will receive an email with invitation to join **Github** classroom
- Start by reading the **readme.md** file.
Instructions are available here
- Github will **automatically** save the **last commit** as a **submission** before the deadline