Machine Learning and Pattern Recognition
Lab 1: Spring Semester 2026

Note: This is an individual assignment. You may not consult your peers or AI tools to do these tasks except where explicitly asked for. Any misconduct will result in a 0 score in this entire assignment and will be noted and reported to the Academic Integrity Committee.

1. In this assignment you are allowed to use the required Python packages.
2. Rules for submission of assignment
   - *You can use any IDEs you want*. You need to submit your code along with the outputs and the report in one file in **pdf format**.
   - Write your comments along with the code for each step as required
   - The file should specify the steps taken, followed by observations and answering the question as specified in the assignment.
   - Your file should be as follows. **Yourname_lab1.pdf**

1 Introduction

In this assignment, we will learn how to extract some features from images so that we can use them later for a machine learning (ML) model. We will use a chromosome image for this purpose. For each chromosome present in the image, we will extract some shape and size characteristics that can be used later as features in an ML model. Such a model can be trained on the features to detect, for example, if a chromosome is healthy or deformed. The output for this assignment will be a 2D data frame with rows being chromosomes and columns being their features. You can follow the workflow below to perform this task.

1. **Step 1**: Open your choice of IDE and create a file with the name as specified above.

2. **Step 2**: Install the following libraries:

   - Install OpenCV by using pip install opencv-python
   - Install Numpy by using pip install numpy
   - Install Matplotlib by using pip install matplotlib

3. **Step 3**: Import the above libraries:

   - Import Open CV by using import cv2
   - Import Numpy by using import numpy as np
   - Import Matplotlib using import matplotlib.pyplot as plt

4. **Step 4**: Read the image Chromosomes.jpg file using cv2.imread()
   Plot the image using plt.imshow()

5. **Step 5**: Convert the image into GRAYSCALE using cv2.cvtColor().
   Check for documentation on how to use cv2.cvtColor method

6. **Step 6**: Apply morphological opening for background removal using *cv2.getStructuringElement() and cv2.morphologyEx().*
   Then threshold the image for binarization using *cv2.threshold()* and find the contours using *cv2.findContours()*

7. **Step 7**: Check if the number of contours is greater than or equal to threshold (5 is recommended) so that it can draw a bounding box over set of contours. Because less than number of contours than threshold value might not consist of pair of chromosomes in it.

Find the features for each chromosome (Height, Width, Shape, Area, Perimeter and Circularity and store it in a data-frame.

Area= cv2.contourArea()

Perimeter = cv2.arcLength(contour)

Circularity = (4 ∗ np.pi ∗ area) / (perimeter 2)))

8. **Step 8:** Draw a bounding box for each chromosome using cv2.Rectangle() and display it over the chromosomes.

9. **Useful resources**:

   (a) Resource1 for cv2.getStructuringElement and cv2.morphologyEx methods.

   (b) Resource2 for documentation on thresholding.

   (c) Resource3 for documentation on Contours and Bounding Boxes.

10. **Output Reference:** The bounding box will comprise of information such as height, width, area, perimeter, and circularity.

    - Perform standardization of all the features and outline your observation.
    - Perform normalization of all the features and outline your observation.

2 Report

Your report should include all intermediate figures depicting the various steps and any other pertinent visuals, accompanied by your observations. Subsequently, answer the following questions within your report (you may use markdown cells for writing the report section).

- **Q1**: How can contour detection be used to identify objects in an image?
- **Q2**: What is the importance of standardization of data? What difference did you observe before and after standardization?
- **Q3**: Let's consider one of the values in the width column is missing. How to handle this missing value?
- **Q4**: What is the importance of data normalization? What difference did you observe before and after normalization?
- **Q5**: How might you adapt the bounding box construction process to handle overlapping or touching chromosomes?

REFERENCE OUTPUTS:

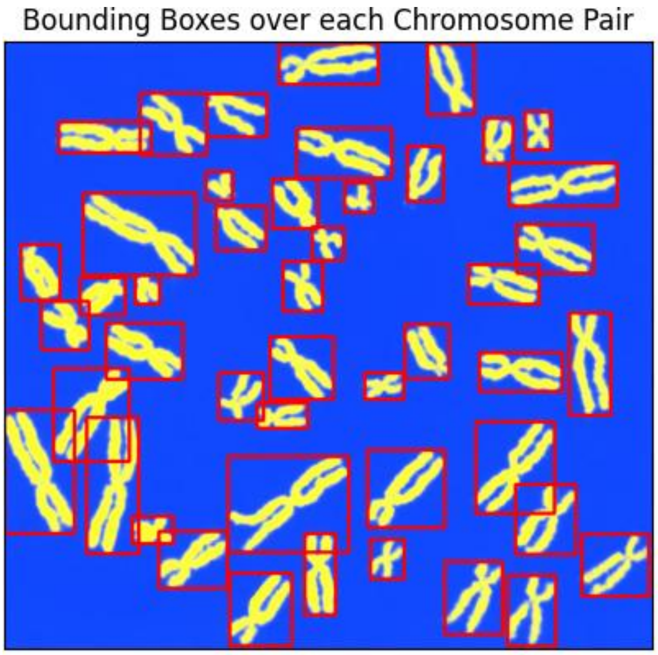
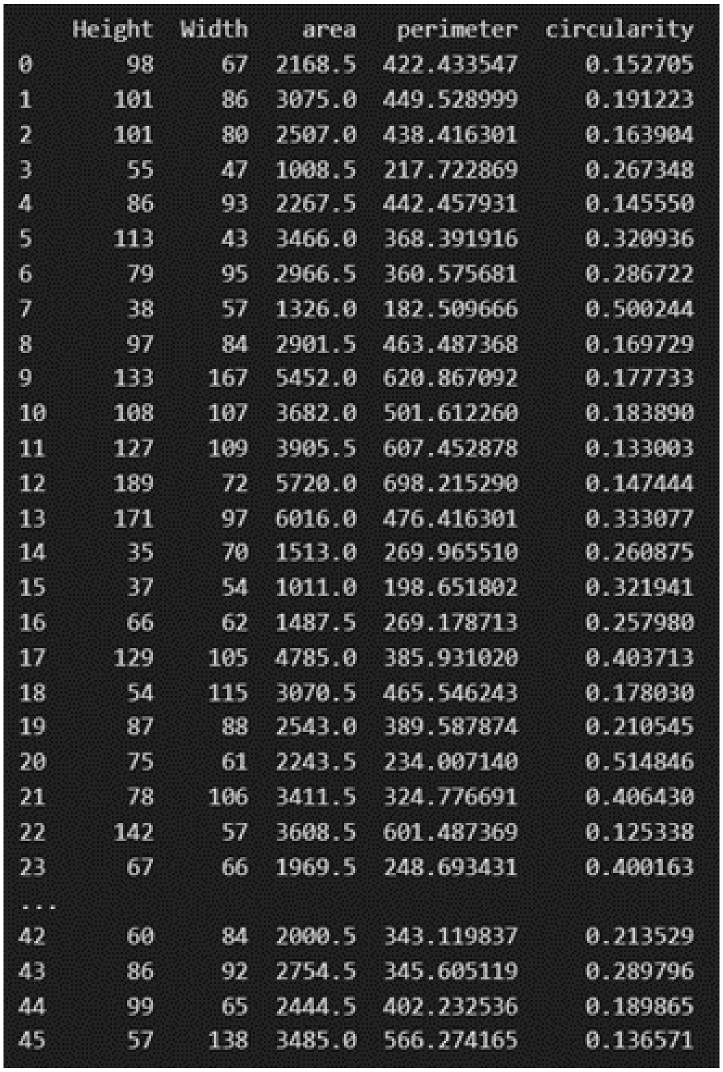


Figure 1: Output figure reference



```
    Height  Width    area  perimeter  circularity
0       98     67  2168.5  422.433547     0.152705
1      101     86  3075.0  449.528999     0.191223
2      101     80  2507.0  438.416301     0.163904
3       55     47  1008.5  217.722869     0.267348
4       86     93  2267.5  442.457931     0.145550
5      113     43  3466.0  368.391916     0.320936
6       79     95  2966.5  360.575681     0.286722
7       38     57  1326.0  182.509666     0.500244
8       97     84  2901.5  463.487368     0.169729
9      133    167  5452.0  620.867092     0.177733
10     108    107  3682.0  501.612260     0.183890
11     127    109  3905.5  607.452878     0.133003
12     189     72  5720.0  698.215290     0.147444
13     171     97  6016.0  476.416301     0.333077
14      35     70  1513.0  269.965510     0.260875
15      37     54  1011.0  198.651802     0.321941
16      66     62  1487.5  269.178713     0.257980
17     129    105  4785.0  385.931020     0.403713
18      54    115  3070.5  465.546243     0.178030
19      87     88  2543.0  389.587874     0.210545
20      75     61  2243.5  234.007140     0.514846
21      78    106  3411.5  324.776691     0.406430
22     142     57  3608.5  601.487369     0.125338
23      67     66  1969.5  248.693431     0.400163
...
42      60     84  2000.5  343.119837     0.213529
43      86     92  2754.5  345.605119     0.289796
44      99     65  2444.5  402.232536     0.189865
45      57    138  3485.0  566.274165     0.136571
```

Figure 2: Bounding box information