**Name: Aleena Omar**

**BSIT 21043 7th Semester (Morning)**

**ASSIGNMENT 6**

**Collect some bottles at your home, partially and completely filled with some liquid. Take pictures of these bottles. Write a program which can label only incompletely filled bottles and test on these pictures.**

**Approach:**

1. **Preprocessing:**
   * Convert the image to grayscale.
   * Apply edge detection to find bottle contours.
   * Use thresholding or adaptive thresholding to differentiate liquid levels.
2. **Bottle Detection:**
   * Identify bottle shapes using contour detection.
   * Filter out irrelevant objects based on size and aspect ratio.
3. **Liquid Level Detection:**
   * Analyze the detected bottles to identify liquid levels.
   * Compare the liquid height to the bottle height.
   * If the liquid level is below a certain percentage of the bottle height (e.g., 90%), label it as "incompletely filled."
4. **Labeling and Output:**
   * Draw bounding boxes around incompletely filled bottles.
   * Save or display the annotated image.

**function detect\_incomplete\_bottles(image\_path, fill\_threshold)**

**if nargin < 2**

**fill\_threshold = 0.9; % Default fill threshold**

**end**

**% Read the image**

**image = imread(image\_path);**

**gray = rgb2gray(image);**

**% Edge detection (Canny)**

**edges = edge(gray, 'Canny');**

**% Find contours (bottles)**

**bottle\_stats = regionprops(edges, 'BoundingBox');**

**% Convert to HSV for liquid detection**

**hsv\_image = rgb2hsv(image);**

**% Define threshold for detecting liquid (adjust based on liquid color)**

**lower\_liquid = 0.2; % Adjust for actual color**

**upper\_liquid = 0.8;**

**% Extract saturation and value channels**

**liquid\_mask = (hsv\_image(:,:,2) > lower\_liquid) & (hsv\_image(:,:,3) < upper\_liquid);**

**% Find contours for liquid levels**

**liquid\_stats = regionprops(liquid\_mask, 'BoundingBox');**

**% Display the original image**

**figure;**

**imshow(image);**

**hold on;**

**% Process detected bottles**

**for i = 1:length(bottle\_stats)**

**bottle\_box = bottle\_stats(i).BoundingBox;**

**x\_b = bottle\_box(1);**

**y\_b = bottle\_box(2);**

**w\_b = bottle\_box(3);**

**h\_b = bottle\_box(4);**

**for j = 1:length(liquid\_stats)**

**liquid\_box = liquid\_stats(j).BoundingBox;**

**x\_l = liquid\_box(1);**

**y\_l = liquid\_box(2);**

**w\_l = liquid\_box(3);**

**h\_l = liquid\_box(4);**

**% Ensure liquid is inside the bottle**

**if x\_l >= x\_b && (x\_l + w\_l) <= (x\_b + w\_b) && ...**

**y\_l >= y\_b && (y\_l + h\_l) <= (y\_b + h\_b)**

**fill\_ratio = h\_l / h\_b; % Liquid height compared to bottle height**

**% Mark incompletely filled bottles**

**if fill\_ratio < fill\_threshold**

**rectangle('Position', bottle\_box, 'EdgeColor', 'r', 'LineWidth', 2);**

**text(x\_b, y\_b - 10, 'Incomplete', 'Color', 'r', 'FontSize', 12, 'FontWeight', 'bold');**

**end**

**end**

**end**

**end**

**title('Labeled Incompletely Filled Bottles');**

**hold off;**

**end**

**% Example usage**

**detect\_incomplete\_bottles('image.png');**

