**Step 1: Feature Extraction from Segmented Objects**

1. **Segment the Image**
   * **Segmentation** is the process of dividing an image into parts or regions, usually by separating the objects of interest from the background.
   * **How to Do It**: Use techniques like thresholding, edge detection, or more advanced methods like watershed or contour-based segmentation.
   * Example:
     + **Thresholding**: Converts the image to binary (black and white) based on a certain intensity value.
     + **Contours**: Finds the outlines of objects in the image.
2. **Identify the Objects of Interest**
   * After segmentation, identify the specific objects in the image. Each object will be a separate region or set of pixels.
   * **How to Do It**: You can use contours or connected component analysis to label and isolate each object.
3. **Extract Features from Each Object**
   * Once the objects are isolated, you can extract features from each segmented object.

**Feature Types:**

* + **Shape Features**: Describe the geometry of the object.
    - **Area**: The number of pixels inside the object.
    - **Perimeter**: The length of the boundary of the object.
    - **Bounding Box**: The smallest rectangle that can enclose the object.
    - **Centroid**: The center of the object.
  + **Edge Features**: Extract edges within the segmented object.
    - **How to Do It**: Apply edge detection (e.g., Canny) only to the pixels within the object's region.
  + **Texture Features**: Analyze the surface quality of the object.
    - **How to Do It**: Compute texture descriptors like Local Binary Pattern (LBP) or Gray-Level Co-occurrence Matrix (GLCM) for the object's pixels.
  + **Color Features**: If working with color images, analyze the color distribution within the object.
    - **How to Do It**: Compute color histograms or color moments for the segmented object.

Analyze and Use the Features

* + Once the features are extracted, you can analyze them for tasks like object recognition, classification, or further image analysis.
  + For example, you might compare the shape or texture features of different objects to classify them into categories.

**Step 2 : Object Detection using the extracted features**

**1. Extract Features from a Reference Object**

* First, you need a reference object, which is the object you want to detect in other images.
* Extract features from this reference object as described previously (shape, edge, texture, color).

**2. Extract Features from the Target Image**

* Segment the target image where you want to detect the object.
* Extract the same set of features from all segmented regions or objects in the target image.

**3. Compare Features for Object Detection**

* Compare the features extracted from the reference object with the features extracted from each segmented object in the target image.
* Use similarity measures to find the closest match. Here’s how you can do it:

**Similarity Measures:**

* **Euclidean Distance**: Compare feature vectors by calculating the Euclidean distance. The smaller the distance, the more similar the objects are.
* **Correlation**: Measure the correlation between feature vectors. Higher correlation means higher similarity.
* **Histogram Comparison**: If using color histograms, compare them using methods like Chi-square or Intersection.

**4. Locate and Identify the Object**

* Once you find the best match (i.e., the segmented object with features most similar to the reference object), you can identify and locate the object in the target image.
* Mark the detected object using bounding boxes or contours.

**5. Visualize the Results**

* Draw the bounding box or highlight the detected object in the target image to show the result of the object detection.

**Finally based on the objects and features matching the target image label them and create a labelled dataset.**