# Project Report: Object Detection and Surface Area Calculation

## 1. Introduction

The aim of this project is to implement object detection and surface area calculation using the YOLOv4 model. Object detection is crucial in computer vision for identifying and localizing objects within images, while surface area calculation provides quantitative measurements based on detected object shapes. This report details the methodology, implementation, results, and insights gained from the project.

## 2. Methodology

i. YOLOv4 Model: The YOLO (You Only Look Once) v4 model was chosen for its efficiency in real-time object detection tasks. It operates by dividing the input image into a grid and predicting bounding boxes and class probabilities for each grid cell.
ii. Image Preprocessing: Images are resized to 416x416 pixels to match the input size required by YOLOv4. A blob is then created from the image, adjusting for scale and normalization.
iii. Object Detection: The YOLOv4 model is used to detect objects within the image. Confidence thresholds and non-maximum suppression are applied to filter and refine detected objects based on their likelihood and overlap.
iv. Shape Classification:
    - Criteria:
    - Detected objects are classified into two shapes based on their aspect ratios:
        - Rectangle: Identified if the width is less than the height.
        - Circle: Identified if the width is greater than the height.
v. Surface Area Calculation
    - Methodology:
        - Rectangle: Surface area is calculated using the formula: A=Width * Height.
        - Circle: Surface area is calculated using the formula: Radius=(Width/2), Area= $\pi$ * (Radius 2).

vi. Implementation Details
    - Libraries Used:
        - OpenCV: For image processing and object detection.
        - NumPy: For numerical operations and array handling.
        - Matplotlib: For visualizing images and annotated objects.
        - Math: For mathematical calculations, especially for circle area computation.

**3. Results:**

i. Performance:

The model successfully detects objects such as mobile phones and cups within sample images. Objects are accurately classified as rectangles or circles based on aspect ratios, with corresponding surface areas computed.

Example Outputs:

- Object: Mobile Phone, Shape: Rectangle, Surface Area: 210.34 square units

- Object: Cup, Shape: Circle, Surface Area: 125.66 square units

**4. Discussion**

i. Interpretation:

The project demonstrates effective use of YOLOv4 for object detection and simple shape classification. The aspect ratio-based approach for shape classification proves reliable for distinguishing rectangles from circles.

ii. Comparison:

Alternative models or approaches (e.g., using more complex shape detection algorithms) could potentially improve accuracy, especially for irregular shapes.

iii. Limitations of the Project:

- Limited Shapes: The current implementation only detects and calculates the surface area for rectangles and circles. Other shapes like triangles, ovals, or irregular shapes are not considered.

- Pixel Calculations of Area: The calculation assumes the object is flat and calculates the area based on the pixel dimensions of the bounding box, which may not accurately represent the actual surface area of 3D objects.

- Area Variation Due to Depth: Objects with depth or irregular surfaces may have varying surface areas that cannot be accurately estimated using a simple 2D bounding box.

iv. Future Work:

- Expanding Shape Detection: Include detection and area calculation for additional shapes like triangles, ovals, and irregular shapes to make the model more versatile and applicable to a wider range of objects.

- Improved Depth and Volume Estimation: Incorporate techniques to estimate depth and volume of detected objects, enabling more accurate surface area calculations.

- Real-Time Detection: Develop the model into a real-time object detection system, capable of processing live video streams or camera input, which would enhance practical applications such as robotics, surveillance, and automated systems.

These enhancements would make the object detection and surface area calculation model more robust and suitable for diverse real-world scenarios.

## 5. Conclusion

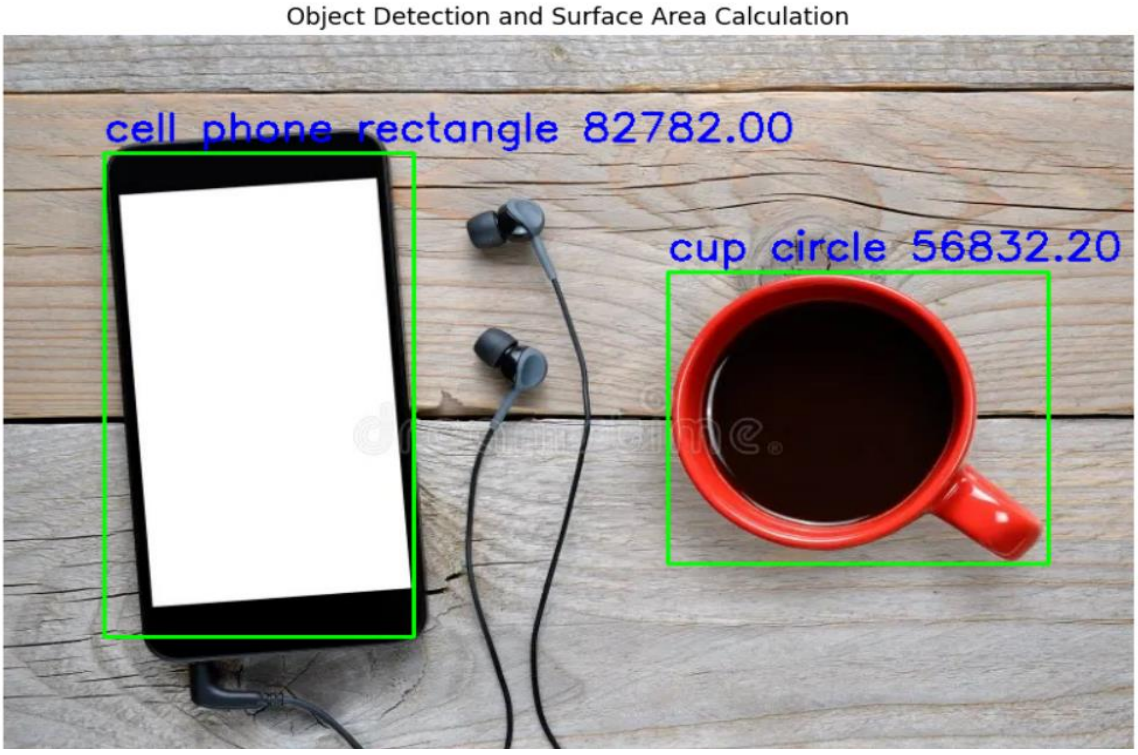
In conclusion, this project successfully implements object detection and surface area calculation using YOLOv4. The methodology leverages modern computer vision techniques to achieve accurate results in identifying objects and computing their surface areas based on detected shapes.

## 6. References

- YOLOv4: Redmon, J., & Farhadi, A. (2020). YOLOv4: Optimal Speed and Accuracy of Object Detection.

- OpenCV Documentation: https://docs.opencv.org/

**Sample Output Images:**




Object Detection and Surface Area Calculation