Progress Report - Embedded Systems Workshop Course

Team Name: HBVR

Date: October 1, 2024

Team Members:

Vijay Aravynthan

Harshil Singh

Bhaskar Itekela

Monoseij Roy

Task Overview:

Today's work focused on the development of object detection functionality using a combination of a camera module and YOLOv8-based segmentation for detecting red objects in a live video feed. We also integrated a feature that draws a line from the center of the frame to the center of the detected red object, enhancing visualization in the embedded system.

Accomplishments:

1. YOLOv8 Model Integration:
    - Successfully loaded the pre-trained YOLOv8 model for segmentation and configured it for real-time object detection.
    - Adjusted confidence and IOU thresholds for reliable detection performance during live video processing.
2. Red Object Detection
    - Implemented a function to detect objects in real-time video, identify red objects using HSV color segmentation, and apply a binary mask to isolate red objects in the frame.
    - Defined color thresholds for red in HSV space and tested the detection under different lighting conditions.
3. Mask Overlay Implementation:
    - Created a semi-transparent black mask overlay that covers detected red objects in the live feed.
    - Applied morphological operations to clean up segmentation masks, ensuring accurate detection of object boundaries.
4. Centroid Calculation:
    - Used contour-based methods to calculate the centroid of detected red objects.
    - Calculated the moments of the detected red objects and derived the centroid position for each detected object.
5. Line Drawing Feature:

- Implemented a feature that draws a line from the center of the video frame to the centroid of the detected red object.
- Added dynamic recalculation of object centroids and updated the line position in real-time as objects move in the frame.

6. Webcam Integration
    - Integrated the code with a webcam feed for real-time object detection and masking, allowing continuous testing of red object detection.

Challenges:

1. Lighting Variability: The accuracy of red object detection was initially affected by varying lighting conditions in the environment. We improved robustness by fine-tuning the HSV threshold ranges.

2. Mask Quality: Ensuring that the segmentation masks fit the object boundaries well required several iterations of morphological operations and contour refinement.

3. Real-Time Processing: Processing speed was critical, and resizing frames to a smaller resolution helped balance accuracy and performance.

Next Steps:

1. Optimize Performance:
  - Explore optimizations to improve the speed of segmentation and line-drawing while maintaining accuracy.

2. Test in Real Embedded Environment:
  - Deploy the detection system on the embedded hardware platform and perform real-time testing using the camera module attached to the board.

3. Fine-Tune Color Detection:
  - Refine HSV thresholds and test with different shades of red objects to ensure robustness across varying conditions.

4. Finalize Documentation:
  - Update the team documentation and code comments to reflect changes in the implementation and describe key features for future reference.

Conclusion:

The team successfully implemented a real-time red object detection system using YOLOv8 segmentation and enhanced visualization by drawing lines from the frame center to detected objects. With further optimization and testing, this system will be a robust feature in our embedded systems project.