**Computer Vision Engineer Assessment Task**

### **Objective**

The goal of this assessment is to evaluate your knowledge and expertise in computer vision, object detection techniques, and understanding of camera properties. You are required to develop a complete vision pipeline that detects and classifies objects from a live camera feed while analyzing the impact of camera properties on detection performance, including intrinsic camera parameters.

### **Task Description**

You will be designing and implementing a computer vision system that:

1. **Acquires video frames** from a camera or a streaming source.
2. **Extracts camera properties** including intrinsic parameters (focal length, principal point, distortion coefficients) and modifies them programmatically.
3. **Performs object detection** using classical computer vision methods.
4. **Implements object detection** using deep learning-based approaches.
5. **Analyzes performance differences** between various approaches and camera configurations.

### **Task Breakdown**

#### **1. Image Acquisition & Camera Properties**

* Capture real-time video frames from a connected camera.
* Extract and display key camera properties such as resolution, frame rate, brightness, exposure, and intrinsic parameters.
* Modify these properties programmatically and observe the effect on the captured frames.

#### **2. Object Detection Using Classical Computer Vision**

* Implement an object detection pipeline based on classical image processing techniques.
* Use feature extraction and image segmentation to detect objects.
* Apply filtering techniques to improve detection accuracy.

#### **3. Object Detection Using Deep Learning**

* Implement a deep learning-based object detection system using a pre-trained model.
* Process and annotate detected objects with bounding boxes and classification labels.
* Compare results obtained from classical and deep learning-based methods.

#### **4. Performance Analysis & Camera Impact**

* Analyze how different camera properties, including intrinsic parameters, impact object detection performance.
* Investigate the effect of lighting conditions on the accuracy and robustness of the detection.
* Document the key findings with visual examples.

#### **5. Code Structure & Best Practices**

* Organize the code into modular and reusable components.
* Ensure proper documentation and adherence to best coding practices.
* Handle possible failure cases such as missing frames or incorrect camera configurations.

### **Deliverables**

* **Well-structured source code** for the entire implementation.
* **A short report** summarizing the methodologies, findings, and observations.
* **Visual results** showcasing the performance of different techniques.

### **Evaluation Criteria**

Your work will be evaluated based on:

* **Technical Depth:** Understanding of classical and deep learning-based object detection.
* **Code Quality:** Clean, well-documented, and modular code structure.
* **Understanding of Camera Properties:** Ability to manipulate and analyze their effect on detection, including intrinsic camera parameters.
* **Problem-Solving Approach:** How well challenges are tackled and solutions are justified.
* **Performance & Optimization:** Efficiency and effectiveness of the implemented pipeline.

### **Submission Guidelines**

* Submit the complete project as a zip file or a Git repository link.
* Provide a detailed explanation in the report regarding the choices made in the implementation.
* Ensure the project can be executed without modifications on a standard system.

**Deadline:** 10th March

### **Additional Instructions**

* You are free to choose libraries, frameworks, and models that you find most suitable.
* Be prepared to explain your approach and reasoning during the technical discussion.
* If any assumptions are made, document them clearly in the report.

Good luck! We look forward to reviewing your work.