Republic of the Philippines
Laguna State Polytechnic University
Province of Laguna

Machine Problem No. 5			
Topic:	**Module 2.0: Feature Extraction and Object Detection**	**Week No.**	8-9
Course Code:	**CSST106**	**Term:**	1st Semester
Course Title:	**Perception and Computer Vision**	**Academic Year:**	2024-2025
Student Name		**Section**	
Due date		**Points**	

Machine Problem: Object Detection and Recognition using YOLO.

Objective:
To implement real-time object detection using the YOLO (You Only Look Once) model and gain hands-on experience in loading pre-trained models, processing images, and visualizing results.

Task:

1. **Model Loading:** Use TensorFlow to load a pre-trained YOLO model.
2. **Image Input:** Select an image that contains multiple objects.
3. **Object Detection:** Feed the selected image to the YOLO model to detect various objects within it.
4. **Visualization:** Display the detected objects using bounding boxes and class labels.
5. **Testing:** Test the model on at least three different images to compare its performance and observe its accuracy.
6. **Performance Analysis:** Document your observations on the model's speed and accuracy, and discuss how YOLO's single-pass detection impacts its real-time capabilities.

Key Points:

- YOLO performs detection in a single pass, making it highly efficient.
- Suitable for applications requiring real-time object detection.

Submission Instructions:

1. **Code:** Write your implementation in a Python script or Jupyter Notebook.
2. **Processed Images:** Save the images with bounding boxes and labels in a folder named output_images.
3. **Documentation:** Create a brief document (README.md or PDF) explaining your approach, code, and observations.
4. **Folder Organization:** Create a folder named YOLO_Object_Detection and include the following:
    - code/: Your Python script or Jupyter Notebook.
    - output_images/: Processed images.
    - documentation/: A README file explaining the process.

5. **Filename Format:** Use [SECTION-YOURNAME-MP] for all files (e.g., SECTION-YOURNAME-MP.py).

Penalties:

- **Incorrect Filename:** 5-point deduction.
- **Late Submission:** 5-point deduction per day.
- **Cheating/Plagiarism:** Strict penalties as per academic integrity policies.

Rubric for Machine Problem: Object Detection using YOLO

Criteria	Excellent (90-100%)	Good (75-89%)	Satisfactory (60-74%)	Needs Improvement (0-59%)
Correct Implementation (30%)	Successfully implements YOLO for object detection with no errors; code is efficient and runs smoothly.	Minor issues in implementation but overall functional; code is mostly efficient.	Basic implementation with noticeable errors; the code runs but may have inefficiencies.	Incorrect implementation: code does not run or produces incorrect results.
Visualization and Accuracy (25%)	Bounding boxes and labels are clear, well-placed, and accurate across all test images.	Bounding boxes and labels are mostly correct with minor inaccuracies.	Basic visualization: some bounding boxes are misplaced or missing.	Poor or missing visualization; bounding boxes are largely incorrect or absent.
Code Quality and Comments (15%)	Code is well-structured, follows best practices, and is thoroughly commented for clarity.	Code is mostly organized; comments are present but minimal.	Code runs but is disorganized; lacks detailed comments.	Code is poorly structured, lacks comments, or is hard to follow.
Documentation (20%)	Comprehensive documentation explaining the approach, code, and observations in detail.	Documentation is clear but lacks some details or observations.	Basic documentation present; lacks clarity and depth.	Missing or inadequate documentation that fails to explain the process.
Folder Organization (10%)	All files are correctly named and organized according to submission instructions, with proper use of folders.	Mostly follows the naming and organization requirements with minor errors.	Basic organization present but does not fully adhere to the specified format.	Poor or missing organization; incorrect file names and folder structure.
Testing and Analysis (10%)	Tests the model on multiple images and provides a thorough analysis of its performance, discussing accuracy and speed.	Tests the model on multiple images but provides a limited analysis.	Limited testing; minimal analysis of model performance.	Fails to test the model or provide any analysis.