National College of Ireland

Configuration Manual

MSc Research Project
MSc in Data Analytics

Gowtham Sudharsan
Student ID: 23294418

School of Computing
National College of Ireland

Supervisor: John Kelly

National College of Ireland

MSc Project Submission Sheet

School of Computing

Student Name:	Gowtham Sudharsan
Student ID:	23294418
Programme:	MSc in Data Analytics **Year:** 2024-25
Module:	MSc Research Project
Lecturer:	John Kelly
Submission Due Date:	11-08-2025
Project Title:	Investigating CNN-based YOLO architectures to improve vehicle detection accuracy for intelligent transportation systems

Word Count: 336 **Page Count: 4**

I hereby certify that the information contained in this (my submission) is information pertaining to research I conducted for this project. All information other than my own contribution will be fully referenced and listed in the relevant bibliography section at the rear of the project.
ALL internet material must be referenced in the bibliography section. Students are required to use the Referencing Standard specified in the report template. To use other author's written or electronic work is illegal (plagiarism) and may result in disciplinary action.

Signature: Gowtham Sudharsan

Date: 11-08-2025

PLEASE READ THE FOLLOWING INSTRUCTIONS AND CHECKLIST

Attach a completed copy of this sheet to each project (including multiple copies)	□
Attach a Moodle submission receipt of the online project submission, to each project (including multiple copies).	□
You must ensure that you retain a HARD COPY of the project, both for your own reference and in case a project is lost or mislaid. It is not sufficient to keep a copy on computer.	□

Assignments that are submitted to the Programme Coordinator Office must be placed into the assignment box located outside the office.

Office Use Only	
Signature:	
Date:	
Penalty Applied (if applicable):	

Configuration Manual

Gowtham Sudharsan
Student ID: 23294418

1 PROJECT OVERVIEW

This is a comprehensive AI-powered system for vehicle detection and weather scene classification designed to work in adverse weather conditions. The project combines two state-of-the-art deep learning models:

1. YOLOv7 - For real-time object detection (vehicles, pedestrians, traffic signs)
2. ResNet50 - For weather scene classification (clear, foggy, rainy, snowy, etc.)

The system is designed to help improve road safety by automatically detecting potential hazards and classifying weather conditions that may affect driving visibility and safety.

2 TECHNICAL SPECIFICATIONS

- Primary Framework: PyTorch with CUDA support
- Object Detection: YOLOv7 (80 COCO classes)
- Scene Classification: ResNet50 (7 weather conditions)
- Input Resolution: 640x640 (YOLOv7), 224x224 (ResNet50)
- Supported Formats: Images, Videos, Real-time Camera Feed
- Platform: Cross-platform (Windows/Linux/MacOS)

3 LIBRARIES

- torch, torchvision, torchaudio: PyTorch with CUDA 12.1 support
- gradio: Web interface framework
- numpy: Numerical computing
- opencv-python: Computer vision operations
- Pillow: Image processing
- pyyaml: YAML file parsing
- scipy: Scientific computing
- tqdm: Progress bars
- pandas: Data manipulation
- seaborn, matplotlib: Data visualization

4 PROJECT STRUCTURE

This project is organized into two main directories, each serving a distinct purpose:

Training_Environment/ - Model training and development

Adverse_Weather_Gradio_App/ - Production deployment application

Training_Environment

- vehicle-detection-yolov7_train_kaggle.ipynb (Training and cloning of yolov7 from github)
- Kaggle Notebooks with Internet On and GPU accessed
- Train using GPU: Estimation 5-6hours
- Output: yolov7_object_detector.pt (75 MB) - Object detection model, resnet50_scene_classifier.pth (95 MB) - Scene classification model

Adverse_Weather_Gradio_App

- app.py – Python File compatible with Gradio web interface

5 SYSTEM REQUIREMENTS

Minimum Requirements:

- Python 3.11.13
- 8GB RAM
- 5GB free storage
- Windows/Linux/MacOS

Recommended for Optimal Performance:

- NVIDIA GPU with CUDA support
- 16GB+ RAM
- SSD storage
- Stable internet connection

Compatible with: Python 3.11.13, PyTorch 2.x, CUDA 12.1

6 STEPS TO RUN

Steps to train the model

- On Kaggle Notebooks, Upload the prepared dataset in zip format (size constraint)
- Upload dataset to Kaggle Notebooks
- Using GPU, Run the ipynb file
- Complete trained models are saved

Steps to run Gradio Web Interface

- pip install torch torchvision torchaudio --index-url https://download.pytorch.org/whl/cu121 gradio numpy opencv-python Pillow pyyaml scipy tqdm pandas seaborn matplotlib
- Navigate to Gradio file location
- Run In PowerShell: Install all necessary libraries, Activate virtual Environment
    - Set-ExecutionPolicy -Scope Process -ExecutionPolicy Bypass – Run without any restrictions on PS
    - .\venv311\Scripts\Activate.ps1 – activate virtual Environment
- Run Python app.py
- Copy the URL and paste in browser