**Assignment 2**

**Instructions:**

1. **Submission Requirements:**
   - Use the provided template and upload the Jupyter notebook in HTML or PDF format with outputs and its corresponding PDF.
   - You may also choose to use Google Colab for working on the project.
   - Data files do not need to be uploaded with the submission.
   - Ensure the updated Jupyter notebook is submitted with properly formatted and aligned outputs. Incomplete outputs, misalignments, or poorly written comments will result in a deduction of marks.
   - Partial code and partial output will be evaluated, and marks will be awarded based on the PDF file.
   - Only the latest submission would be considered for marking.
2. **File Naming Convention:**
   - Name the file as: CV_assignment2_group_problem statement number
3. **Plagiarism Policy:**
   - Any form of plagiarism will be taken very seriously, resulting in zero (0) marks.
   - All submissions must be the result of your original effort.
   - Copying from any sources, whether online or from peers, is strictly prohibited.
   - Unauthorized collaboration to gain an unfair advantage is prohibited.
   - Both the person sharing resources and the one receiving them will face consequences for plagiarism.
   - Identical or significantly similar submissions will be investigated, and severe punishments will be imposed on those found guilty.
4. **Late Submission Policy:**
   - Late submissions will incur a penalty of -2 marks.
5. **Queries:**
   - For any questions regarding the assignment, use the discussion forum.
6. **Programming Instructions:**
   - Use appropriate Python programming techniques to read and process the dataset.
   - Convert the notebook into HTML or PDF format along with outputs and upload the file.
   - Avoid submitting excessively long notebooks by eliminating irrelevant lengthy prints.

**7. Problem Statement Selection:**

- Choose **ANY ONE** problem statement from the three provided below.
- Enter the problem statement number in the following sheet (Do not edit others' responses):

https://docs.google.com/spreadsheets/d/1xetRC_POpf8bbztxZ_b_mNiipzHUKpG20YFoPUspfGY/edit?usp=drive_link

================================================================================

**Assignment 2 Rubric**

| Criteria | Description | Pts |
|---|---|---|
| **Data Preprocessing** | Implement necessary preprocessing steps such as normalization, resizing, and semantic segmentation to prepare data for model input. | 2.5 pts |
| **Model Development** | Implement the model (e.g., Faster R-CNN) and integrate relevant techniques (contextual awareness, multi-task learning, etc.) for improved performance. | 5 pts |
| **Evaluation Metrics** | Evaluate model performance using appropriate metrics (precision, recall, F1-score, speed, etc.) and justify their relevance to the task. | 2.5 pts |
| **Justification** | Analyze and explain the results, including reasons for the model's success or poor performance (e.g., overfitting, underfitting, model choices). | 2.5 pts |
| **Documentation, Study Presentation, and Code Quality** | Ensure clear, readable code and well-organized documentation. Present the study logically, summarizing the problem and key findings. | 2.5 pts |
| | | **Total:15pts** |

**Problem Statement 1: Cross-Modal Attention Framework for Robust $NO_2$ Prediction.**

**1.Dataset name**: Sentinel-5P TROPOMI Tropospheric $NO_2$ (Level-2)

- Source: Copernicus Sentinel-5 Precursor (ESA); processed by ESA & KNMI, archived by NASA GES DISC.

- Alternate (high-res) version: Sentinel-5P TROPOMI Tropospheric $NO_2$ 1-Orbit L2 V2 HiR (~5.5 km spatial resolution) with relevant DOI and metadata.

Paper link:https://acp.copernicus.org/articles/23/10267/2023/

**2.Problem statement**: Develop a cross-modal deep learning framework to predict tropospheric nitrogen dioxide ($NO_2$) concentrations by fusing Sentinel-5P TROPOMI satellite observations with auxiliary spatio-temporal and environmental data using cross-modal attention mechanisms. The objective is to improve prediction accuracy, spatial consistency, and robustness compared to single-source models.

**Objectives:**

**1. Data Preprocessing & Augmentation**

- Ingest TROPOMI Level-2 $NO_2$ products:
    - Tropospheric $NO_2$ column density
    - Quality assurance (QA) filtering

- Perform preprocessing:
    - Cloud masking and QA-based filtering
    - Regridding to a uniform spatial resolution
    - Temporal aggregation (daily / weekly averages)
    - Normalization and outlier removal

- Integrate auxiliary data (if available):
    - Meteorological variables (temperature, wind, humidity)
    - Land-use / population density
    - Temporal indicators (season, day of week)

- Apply data augmentation:
    - Temporal jittering
    - Noise injection
    - Spatial smoothing and masking to simulate missing data

**2. Cross-Modal Feature Extraction & Fusion**

- Design a multi-branch architecture:
    - TROPOMI branch: CNN / Vision Transformer for spatial $NO_2$ patterns
    - Auxiliary data branch: MLP or Transformer for non-image features

- Extract modality-specific features:
    - Spatial features from satellite imagery
    - Contextual features from meteorological and temporal data

- Implement cross-modal attention:

    - Allow satellite features to attend to meteorological and temporal features
    - Capture dynamic interactions affecting $NO_2$ distribution

- Fuse features using:
    - Attention-weighted fusion
    - Cross-modal transformer layers

**3. $NO_2$ Prediction Task**

- Predict:
  Tropospheric $NO_2$ column concentration (continuous regression)

- Output:
  - High-resolution $NO_2$ maps
  - Temporally consistent prediction

**4. Evaluation & Benchmarking**

- Compare the proposed fusion model against:
  - TROPOMI-only model
  - Auxiliary-data-only model
  - Simple feature concatenation model

- Evaluation metrics:
  - RMSE
  - MAE
  - $R^2$ score
  - Spatial correlation coefficient

- Perform:
  - Seasonal and regional performance analysis
  - Ablation studies on attention mechanisms

## Problem Statement 2: Unsupervised Cross-Modal Anomaly Detection in Brain CT–MRI Imaging

**1. Dataset:** https://www.kaggle.com/datasets/darren2020/ct-to-mri-cgan

**2.Problem Statement:** Develop an unsupervised AI-based anomaly detection framework that learns normal anatomical patterns from paired CT and MRI brain images, and identifies anomalous deviations by analyzing reconstruction errors or latent feature inconsistencies across modalities.

**Objectives**
**1. Preprocessing & Augmentation**

- Normalize CT and MRI images independently
- Resize and align paired images
- Apply data augmentation:

- Random noise injection
- Spatial distortions
- Intensity variations
- Maintain anatomical correspondence between modalities

**2. Feature Extraction**

- Use deep learning architectures to learn latent representations:
- Convolutional Autoencoders
- Variational Autoencoders (VAE)
- Transformer-based encoders
- Learn:
- Modality-specific features
- Shared latent representations between CT and MRI

**3. Unsupervised Anomaly Detection Methods**

Implement and compare multiple approaches:

**a) Autoencoder-Based Detection**
Train on normal CT–MRI pairs
Detect anomalies using:
Reconstruction error
Cross-modal reconstruction mismatch

**b) One-Class Learning**
Extract deep features
Apply:
One-Class SVM
Isolation Forest
Identify outliers in feature space

**c) Cross-Modal Consistency-Based Detection**
Measure divergence between CT and MRI latent representations
Large inconsistencies indicate potential anomalies

**4. Comparison & Evaluation**
Since ground-truth anomaly labels are unavailable:
Use:
- Reconstruction error distributions
- Anomaly score histograms
- Synthetic anomaly injection for validation

Metrics:
- AUC-ROC (using simulated anomalies)
- Mean reconstruction error
- Latent space distance measures
- Qualitative visual inspection of anomaly maps

## Problem statement 3 :Advanced Object Tracking and Detection in Video Streams (Situated Learning )

**1.Dataset**: https://motchallenge.net/data/

**Sample Papers:**

- "Real-Time Object Detection and Tracking Using Faster R-CNN" - https://arxiv.org/abs/2006.04567
- "A Survey on Object Detection and Tracking" - https://www.sciencedirect.com/science/article/pii/S0031320321001234

**2.Problem Statement**: Develop an advanced object tracking and detection system that utilizes the Faster R-CNN model to accurately identify and track multiple objects in video streams. The system should incorporate novel techniques such as temporal consistency checks and adaptive tracking to enhance performance in dynamic environments.

- **Objectives**:
  - **Data Preprocessing**:
    - Extract frames from video sequences and perform normalization to standardize input data.
    - Implement data augmentation techniques such as random cropping, flipping, and color jittering to improve model robustness.
  - **Model Development**:
    - Design a Faster R-CNN model for object detection, fine-tuning it on the selected dataset.
    - Integrate a temporal consistency check mechanism to ensure that detected objects maintain consistent identities across frames.
    - Implement adaptive tracking algorithms (e.g., Kalman filter or SORT) that adjust tracking parameters based on object speed and direction.
  - **Evaluation**:
    - Evaluate the model's performance using metrics such as mean Average Precision (mAP), tracking accuracy, and identity switch rate.
    - Compare the performance of the proposed system against baseline models and other state-of-the-art tracking algorithms. (Optional)

## Problem statement 4: Fine-Tuning a Text-to-Image Model Using Crowd-Sourced Text–Image Pairs

**Problem Statement:** You are given a small set of **crowd-sourced images containing text**, paired with corresponding text prompts. Your task is to **fine-tune a pre trained text-to-image model** so that it can generate images that **visually resemble real operational text images** when given a text prompt.

**Objective:**

Fine-tune a small text-to-image generation model to generate simple, readable text images using crowd-sourced text–image pairs collected from real-world operational contexts.

**The goal is to understand:**

- Domain shift from synthetic to real data
- Model adaptation using limited, noisy samples
- Practical constraints of fine-tuning generative models on Colab

**Model Selection: Stable Diffusion (v1.5 or v1.4) with LoRA fine-tuning**

- Fine-tune attention layers only
- Freeze base model
- Use diffusers + peft

**Evaluation Criteria**

**1.Qualitative Evaluation**

- Visual clarity of generated text
- Font and spacing realism
- Background consistency

**2.Quantitative Evaluation (Mandatory)**

- OCR-based readability test:
  text → generated image → OCR → recovered text
- Metrics:
  Exact string match
  Character accuracy