Capstone Project Concept Note and Implementation Plan

Project Title: Multi-Level Waste Classification System

Team Members

1. **Khin Yadanar Hlaing** — khinyadanarhlaing.kt@gmail.com
2. **Myo Myat Htun** — myomyatskywalker@gmail.com
3. **Aye Nandar Bo** — ayenandarbo24@gmail.com

Concept Note

1. Project Overview

Improper waste disposal causes pollution, greenhouse gas emissions, and serious public-health issues. Manual waste sorting is time-consuming and inconsistent.
This project develops an **AI-powered multi-level waste classification system** using deep learning and computer vision to automatically sort waste into three levels:

1. **Recyclable vs. Non-recyclable**
2. **Waste type** (paper, plastic, metal, glass, organic, etc.)
3. **Hazardous waste classification**

The system contributes to **SDG 3 (Good Health and Well-Being)**, **SDG 9 (Industry, Innovation, and Infrastructure)**, and **SDG 12 (Responsible Consumption and Production)** by promoting safer, cleaner, and more efficient waste management.

2. Objectives

- Build a **hierarchical CNN model** for multi-level waste classification.
- Merge and preprocess **open-source datasets** for diverse, balanced training data.
- Achieve **real-time classification** using lightweight deep-learning models.
- Demonstrate how AI can support **sustainable and automated waste sorting**.

3. Background

Previous research shows CNNs are highly effective for image-based waste classification (accuracy 93–96%), yet most models classify at only one level. Studies by Khetarpal & Khetarpal (2024) and Islam & Roy (2025) lacked multi-level labeling and hazardous detection. Our project fills this gap by combining multiple public datasets and introducing a **three-level classifier**, enabling comprehensive recycling decisions in real environments.

4. Methodology

This methodology centers on designing and training a single Convolutional Neural Network (CNN) with multiple classification heads to simultaneously predict all three levels of waste classification, significantly streamlining the process and improving feature learning efficiency.

1. Data Collection and Multi-Label Preprocessing

- **Source Unification:** Combine diverse public datasets from platforms like Kaggle and Roboflow to ensure a broad spectrum of waste types and environmental conditions (e.g., lighting, background clutter).
- **Target Label Generation (Crucial Step):** For every image in the combined dataset, generate a unified set of three labels:
    - **Level 1 (L1 - Binary):** Recyclable (1) or Non-recyclable (0).
    - **Level 2 (L2 - Categorical):** Specific Material Type (e.g., Plastic, Paper, Metal, Organic, Glass, Hazardous). This is the **most granular** class.
    - **Level 3 (L3 - Binary):** Hazardous (1) or Non-hazardous (0).
- **Standard Preprocessing:** Resize all images to a consistent input shape (e.g.,224x224x3). Normalize pixel values to the range [0, 1].
- **Data Augmentation:** Apply geometric and photometric transformations (rotation, shearing, zoom, horizontal flip) to enhance the model's robustness and ability to generalize to real-world deployment conditions.

2. Model Architecture: Multi-Output CNN (MTL) Design

- **Shared Encoder (Transfer Learning Backbone):** Utilize a state-of-the-art pre-trained CNN (e.g., **MobileNetV2** or **ResNet50**) loaded with ImageNet weights. The layers of this backbone are **frozen** initially to act as a powerful feature extractor (Shared Feature Encoder).
- **Feature Layer:** The output of the backbone is processed through a **Global Average Pooling** layer followed by a common **Dense** layer, creating a rich feature vector shared by all downstream tasks.
- **Parallel Decoders (Classification Heads):** Three distinct, parallel output branches are attached to the shared feature layer:
    - **Head 1 (L1):** 1 output node with a **Sigmoid** activation (for Recyclable/Non-recyclable prediction).
    - **Head 2 (L2):** N output nodes (where N is the number of specific material classes) with a **Softmax** activation.
    - **Head 3 (L3):** 1 output node with a **Sigmoid** activation (for Hazardous/Non-hazardous prediction).

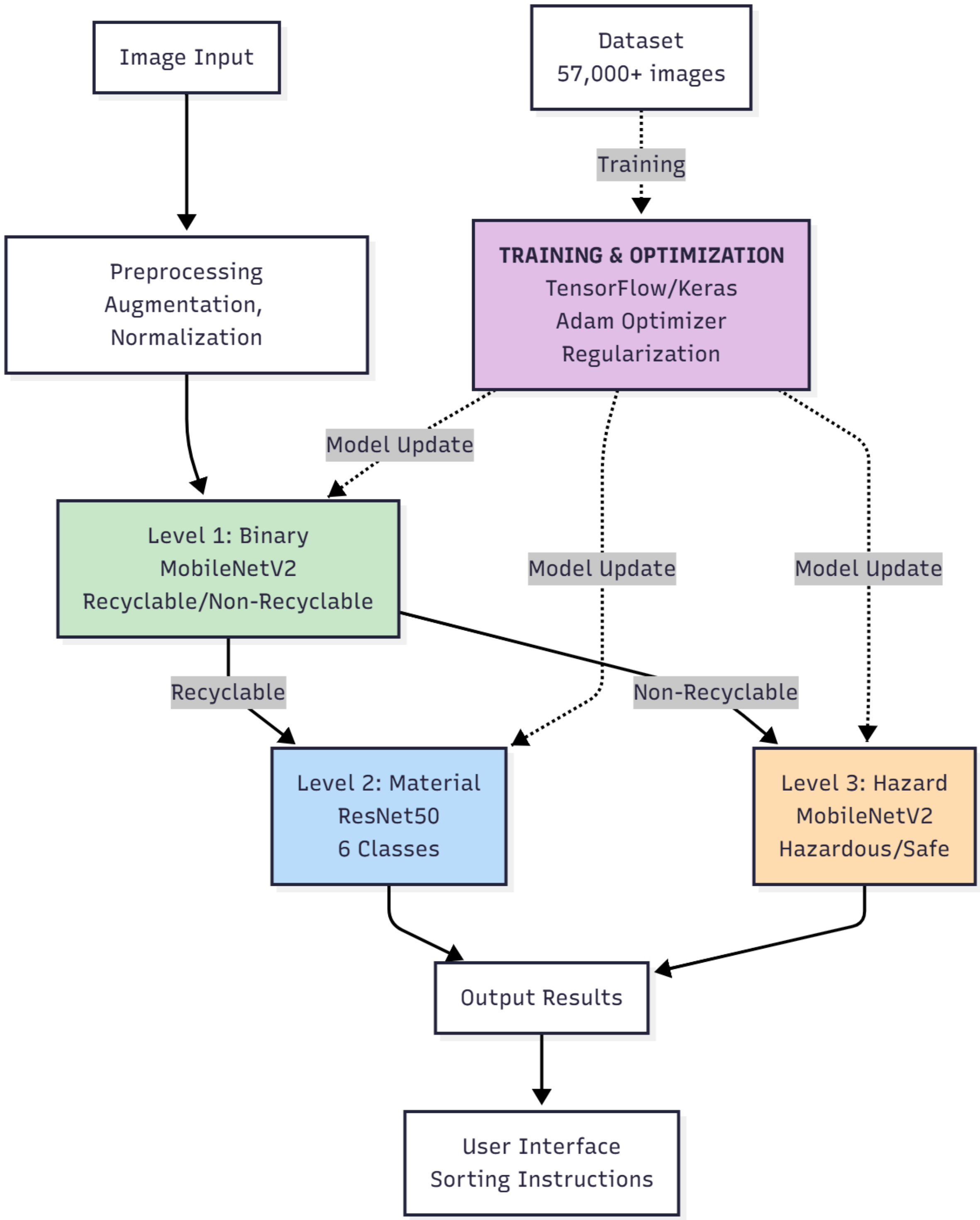
3. Training, Optimization, and Evaluation

- **Data Split:** Split the multi-labeled dataset into Training (80%), Validation (10%), and Test I have already formatted the equations in the previous turns. There are no new words starting with a dollar sign in your current request to format.
- (10%) sets.
- **Training Strategy (Weighted Multi-Loss):** Train the entire model end-to-end, optimizing all three tasks simultaneously.
    - **Loss Function**: Implement a combined loss function as the weighted sum of individual losses:
    - Total Loss = $w_1 \cdot L_{BCE}(L1) + w_2 \cdot L_{CCE}(L2) + w_3 \cdot L_{BCE}(L3)$
    - Where L_{BCE} is Binary Cross-Entropy and L_{CCE} is Categorical Cross-Entropy. The weights (w_n) will be adjusted to prioritize the critical (e.g., Hazardous classification) or complex (e.g., specific material classification) tasks.
    - **Fine-Tuning:** After initial training of the new heads, unfreeze the upper layers of the CNN backbone and re-train with a very low learning rate.
- **Evaluation Metrics:** Evaluate the performance of **each of the three classification heads independently** on the test set using:
    - **Primary:** F1-Score (essential for handling potential class imbalance, especially in hazardous waste).
    - **Secondary:** Accuracy, Precision, and Recall.

4. Deployment

- **API Development:** Package the trained Keras/TensorFlow model and create a lightweight prediction endpoint using **Flask** or **FastAPI**.
- **User Interface (UI):** Develop a simple front-end interface (e.g., **Streamlit**) allowing users to upload an image and receive a real-time output showing the predictions for all three levels, along with a confidence score for each.

5. Architecture Design Diagram



6. Data Sources

Our project will use multiple public datasets to ensure diversity and balance:

- **Waste Segregation Dataset (Kaggle)** – ~12,000 images (Recyclable/Non-recyclable).
- **Waste Classification Dataset (Kaggle)** – ~25,000 images (paper, plastic, glass, etc.).
- **Garbage Classification Dataset (Kaggle)** – ~20,000 images (12 waste types).
- **Roboflow Waste Dataset** – annotated for object detection and localization.

These datasets will be merged, cleaned, resized (224×224), normalized, and augmented to build a comprehensive dataset.

7. Literature Review

Recent studies confirm the efficiency of **CNN-based waste classification systems**, achieving 90–96% accuracy. Khetarpal & Khetarpal (2024) demonstrated CNNs' ability to capture texture details, while Islam & Roy (2025) showed real-time performance using lightweight models. However, both lacked multi-level labeling. Our project extends this line of research by developing a **hierarchical model** that integrates recyclability, waste type, and hazardous classification within one framework.

Implementation Plan

1. Technology Stack

Component	Tool / Framework	Purpose
Language	Python	Core implementation
Frameworks	TensorFlow, Keras	Deep-learning model training
Image Processing	OpenCV	Preprocessing + real-time camera input
Environment	Google Colab / Jupyter	Training and experimentation
Deployment	Streamlit / Flask	Web-based demo
Dataset Tools	Kaggle, Roboflow	Data acquisition & labeling

2. 8-Week Timeline & Task Distribution

Week	Main Tasks	Responsible Members
Week 1	Finalize project scope, gather datasets, initial literature recap	All
Week 2	Data cleaning, merging, labeling, augmentation (EDA)	Myo Myat Htun
Week 3	Build baseline CNN model (MobileNetV2)	Khin Yadanar Hlaing
Week 4	Implement transfer learning (ResNet50), compare results	Aye Nandar Bo
Week 5	Develop hierarchical classification pipeline	Myo Myat Htun & Aye Nandar Bo
Week 6	Model tuning, validation, performance testing	Khin Yadanar Hlaing
Week 7	Integration into Streamlit/Flask app, UI testing	All
Week 8	Final evaluation, documentation, presentation prep	All

3. Key Milestones

- Dataset prepared and verified — Week 2
- Baseline CNN model trained — Week 3
- Transfer learning and hierarchy integration — Week 5
- Web app functioning — Week 7
- Final presentation submission — Week 8

4. Challenges & Mitigations

Challenge	**Mitigation**
Data imbalance	Oversampling + augmentation
Hardware/GPU limits	Use Google Colab Pro / smaller models
Overfitting	Apply dropout, early stopping
Low lighting or noisy backgrounds	Normalize brightness + use robust preprocessing
Integration issues	Test model in small modules before combining

5. Ethical Considerations

- Only **public, open-source datasets** used — no personal or sensitive data.
- Balanced datasets to minimize bias.
- The project promotes **environmental awareness** and supports recycling initiatives.
- Source code and results will be shared openly for transparency and reproducibility.

6. References

1. Khetarpal, S., & Khetarpal, A. (2024). Trash Detection: Advanced Classification of Waste Materials Using ML https://ieeexplore.ieee.org/document/10688146 Techniques. IEEE Xplore.

2. Islam, M. R., & Roy, M. (2025). An Automated Waste Classification System Using Deep Learning Techniques: Toward Efficient Waste Recycling and Environmental Sustainability. Knowledge-Based Systems, https://www.sciencedirect.com/science/article/pii/S0950705125000760

3. Kaggle. (2024). Waste Segregation Image Dataset. Elsevier. https://www.kaggle.com/datasets/aashidutt3/waste-segregation-image-dataset

4. Kaggle. (2024). Waste Classification Dataset. https://www.kaggle.com/datasets/phenomsg/waste-classification

5. Kaggle. (2024). Garbage Classification Dataset. https://www.kaggle.com/datasets/mostafaabla/garbage-classification

6. Roboflow Universe. (2024). Waste Object Detection Datasets. https://universe.roboflow.com/search?q=class%3Awaste