Garbage Classification Using Deep Learning Neural Networks

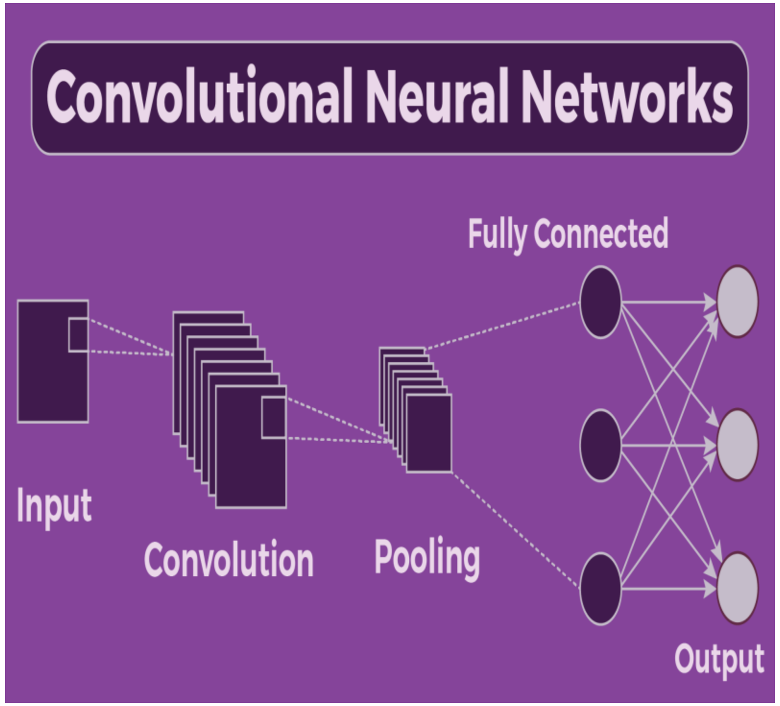
PRML Intermediate Report

Anek Anjireddy

ee22btech11007

Motivation & Context

- The problem of improper waste segregation remains a major challenge in sustainable urban management.
- An efficient solution to this problem are deep learning neural networks such as Convolutional Neural Networks (CNNs).
- CNNs are powerful models capable of learning spatial and visual patterns directly from image data.
- The model takes input as waste images and predicts the category each image belongs to.
- The goal is to build a deployable and efficient classification framework that generalizes across different datasets and real-world waste images.



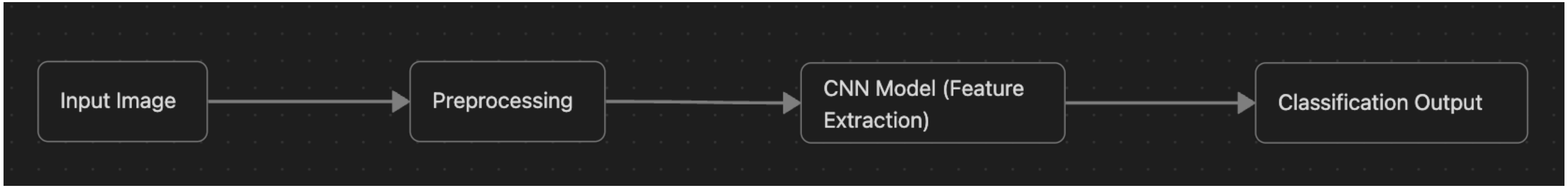
Motivation & Context

- Efficient waste management has become a global challenge, as manual segregation is slow, unsafe, and prone to errors.
- Deep learning and computer vision enable automated waste classification, improving accuracy and efficiency in recycling and disposal processes.
- The project addresses the challenge of robust waste classification using deep learning, aiming to improve generalization across diverse datasets and real-world waste images.
- A key motivation is to develop an efficient yet deployable architecture that maintains high classification quality while reducing computational cost.

Problem Statement and Goals

- **Problem Statement:** Automate the process of garbage classification using deep learning for accurate and scalable waste segregation.
- **Objective:** Develop a model that can classify waste images into various categories. (Binary for now, to be expanded to 5-6 classes).
- **Goals**

1. Build a baseline CNN model capable of achieving consistent performance on curated datasets.
2. Extend the framework to multi-class classification (5–6 categories) using diverse datasets.
3. Test the trained model on real-life waste images from the IITH campus to validate robustness.



Approach

- **Data Collection and Preprocessing:**
  The dataset is sourced from publicly available datasets. Images are resized, normalized, and augmented (rotation, flip, zoom, contrast) to improve generalization and simulate real-world variability.
- **Model Architecture (Version 1):**
  The baseline model is EfficientNetB0 trained for binary or limited-class classification. It serves as a foundation for performance benchmarking and metric tuning.
- **Model Optimization (Version 2):**
  The second version introduces deeper architectures, dropout regularization, and learning rate scheduling. Evaluation uses Precision, Recall, and F1-best threshold to optimize classification balance beyond raw accuracy.
- **Multi-Class Expansion:**
  The framework will be extended to handle 5–6 distinct classes (e.g., paper, plastic, metal, glass, organic, others).
- **Cross-Dataset and Real-World Testing:**
  The final model will be validated on unseen datasets and real images from the IITH campus, assessing robustness under varied lighting and background conditions.

Intermediate Result

After running the model a couple of times with different learning rates and small modifications in neural network's layers, these are the best results i've gotten.
acc: 0.9587 , val_auc: 0.9866, loss: 0.2061

Here is the Output:

```
Epoch 1/3
706/706 ━━━━━━━━━━━━━━━━━━━━ 0s 3s/step - acc: 0.9509 - auc: 0.7592 - loss: 0.2297
Epoch 1: val_auc improved from -inf to 0.98487, saving model to best_freeze.keras
706/706 ━━━━━━━━━━━━━━━━━━━━ 2207s 3s/step - acc: 0.9509 - auc: 0.7595 - loss: 0.2297 - val_acc: 0.9483 - val_auc: 0.9849 - val_loss: 0.2496 - learning
Epoch 2/3
706/706 ━━━━━━━━━━━━━━━━━━━━ 0s 3s/step - acc: 0.9543 - auc: 0.7913 - loss: 0.2115
Epoch 2: val_auc improved from 0.98487 to 0.98509, saving model to best_freeze.keras
706/706 ━━━━━━━━━━━━━━━━━━━━ 2180s 3s/step - acc: 0.9543 - auc: 0.7916 - loss: 0.2115 - val_acc: 0.9459 - val_auc: 0.9851 - val_loss: 0.2514 - learning
Epoch 3/3
706/706 ━━━━━━━━━━━━━━━━━━━━ 0s 3s/step - acc: 0.9587 - auc: 0.7865 - loss: 0.2061
Epoch 3: val_auc improved from 0.98509 to 0.98657, saving model to best_freeze.keras
706/706 ━━━━━━━━━━━━━━━━━━━━ 2162s 3s/step - acc: 0.9587 - auc: 0.7868 - loss: 0.2061 - val_acc: 0.9503 - val_auc: 0.9866 - val_loss: 0.2385 - learning
```

Improvements: Will check for other metrics like precision,recall,etc.

Learning

- Learned (and learning) end-to-end workflow of data preprocessing, model tuning, and evaluation.
- Faced a challenge where the dataset had no validation split — resolved by creating a custom, class-balanced split with fixed seed for reproducibility.
- Dealt with class imbalance and low-quality images through augmentation and weighted loss functions.
- Understood the value of optimizing models using F1-best threshold, not just accuracy, for balanced performance.
- Gained insight into practical deployment trade-offs between model complexity, accuracy, and inference time.

Further Steps

- The immediate nest step is to work on Version 2 and fine tuning the model. I'm looking to optimise the model on metrics (like Precision, Recall,etc.) other than just raw accuracy. To do so I plan on using the F1-best threshold metric.
- Following this, I'll expand the project to multi-class classification.I'm currently aiming for 5-6 classes. I will train this model on various datasets to ensure generalization.
- Test the model on real-life waste images from the IITH campus to assess real-world performance.
- In the longer term, I aim to extend the framework toward object detection, using architectures such as YOLO or Faster R-CNN, enabling the system to not only classify but also locate and label individual waste items within an image.

Thank You