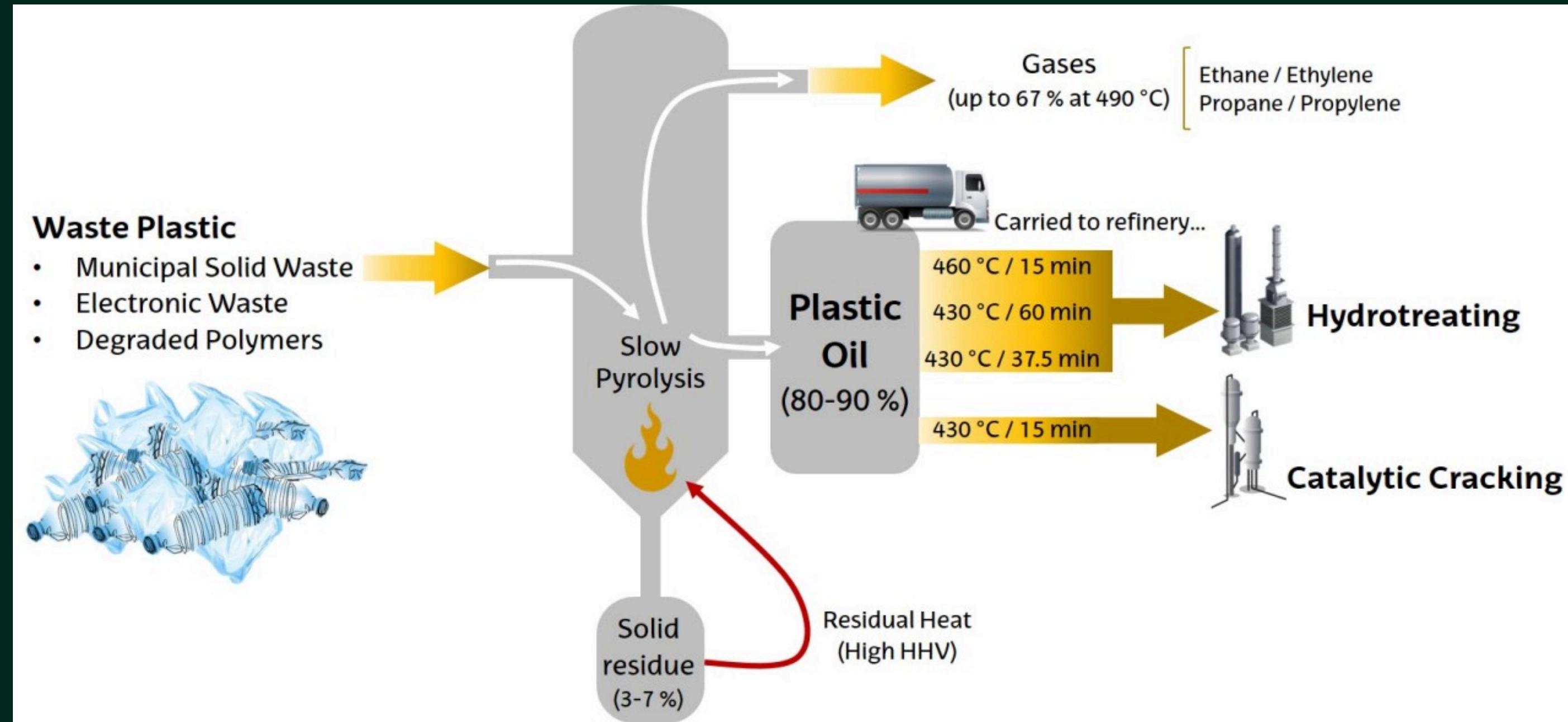


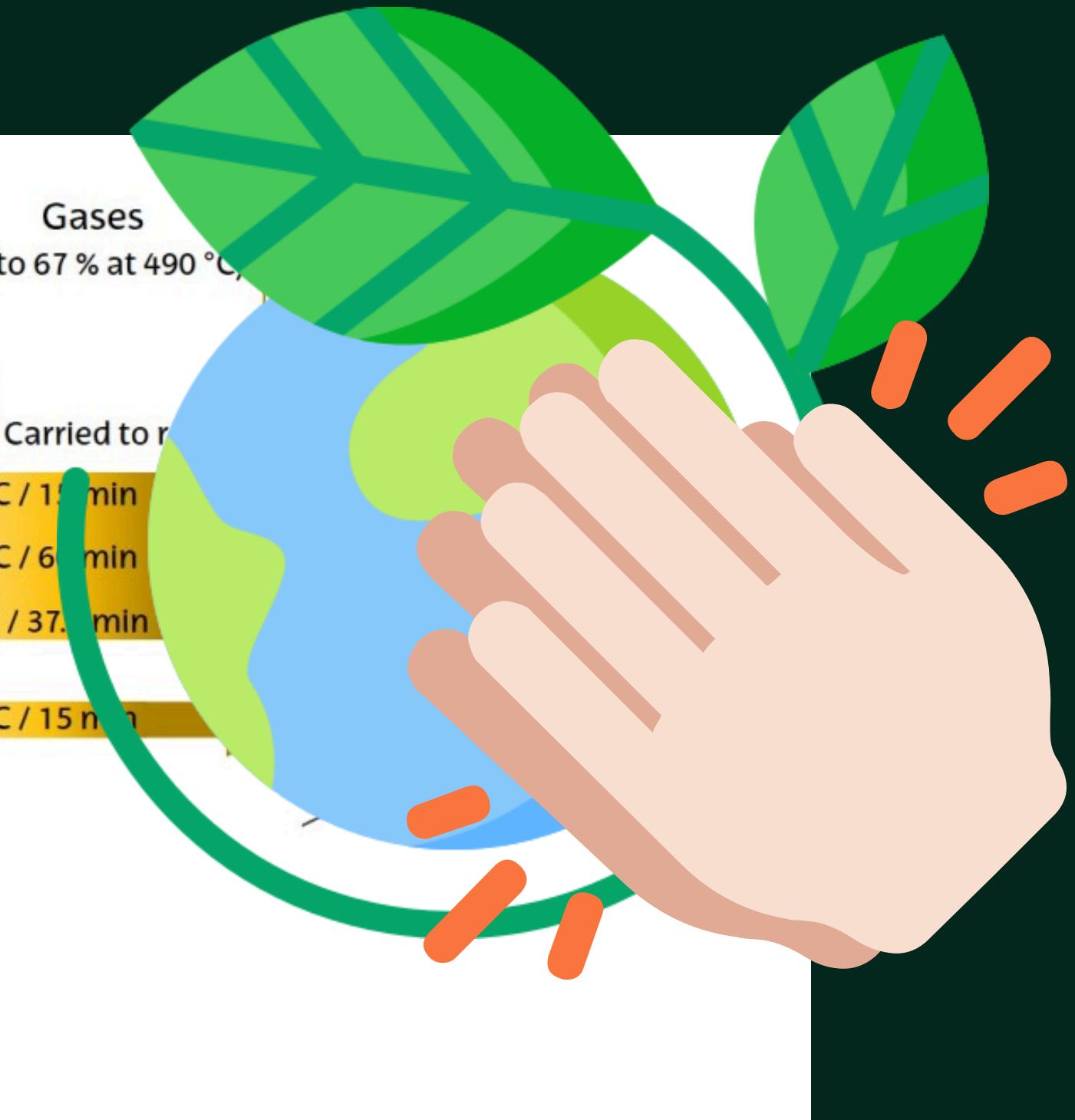
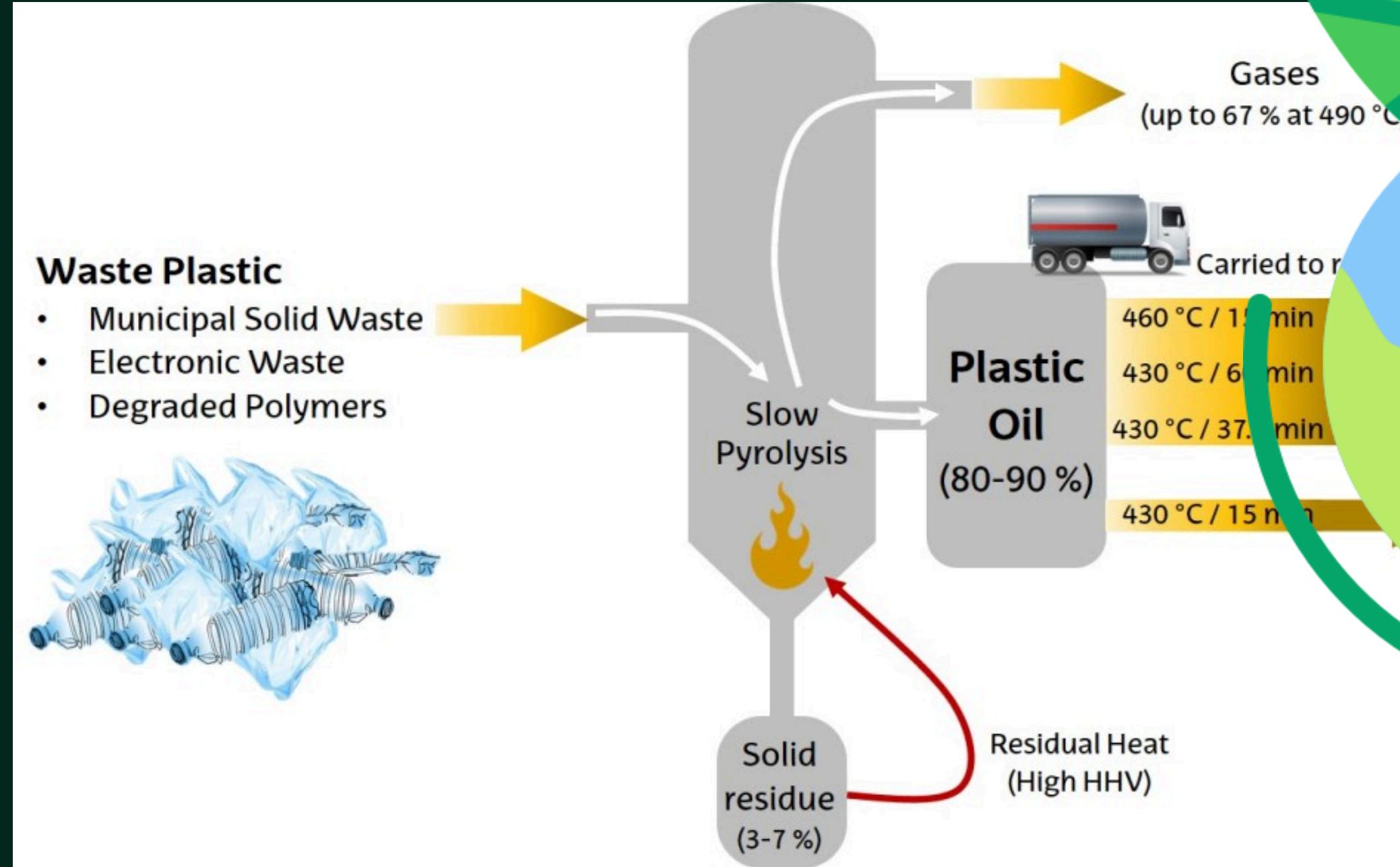
# SANSEVERIA



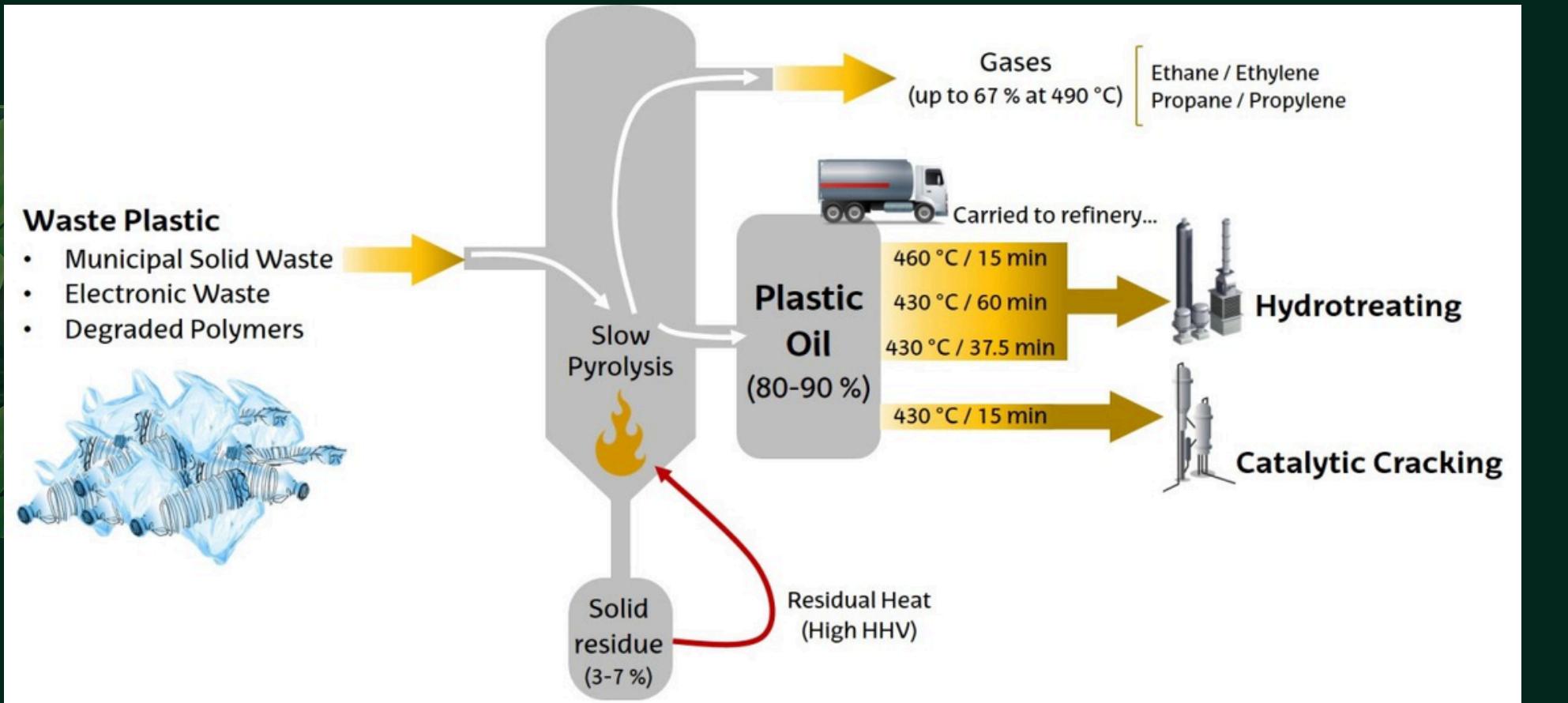
Jasmine, Yurim, Michelangelo, Christopher, Eric,  
Adrian, Keval



Convert plastics into feedstock for the petrochemical industry and green fuels ⇒ reduce 80% in CO<sub>2</sub>



Convert plastics into feedstock for the petrochemical industry and green fuels ⇒ reduce 80% in CO<sub>2</sub>



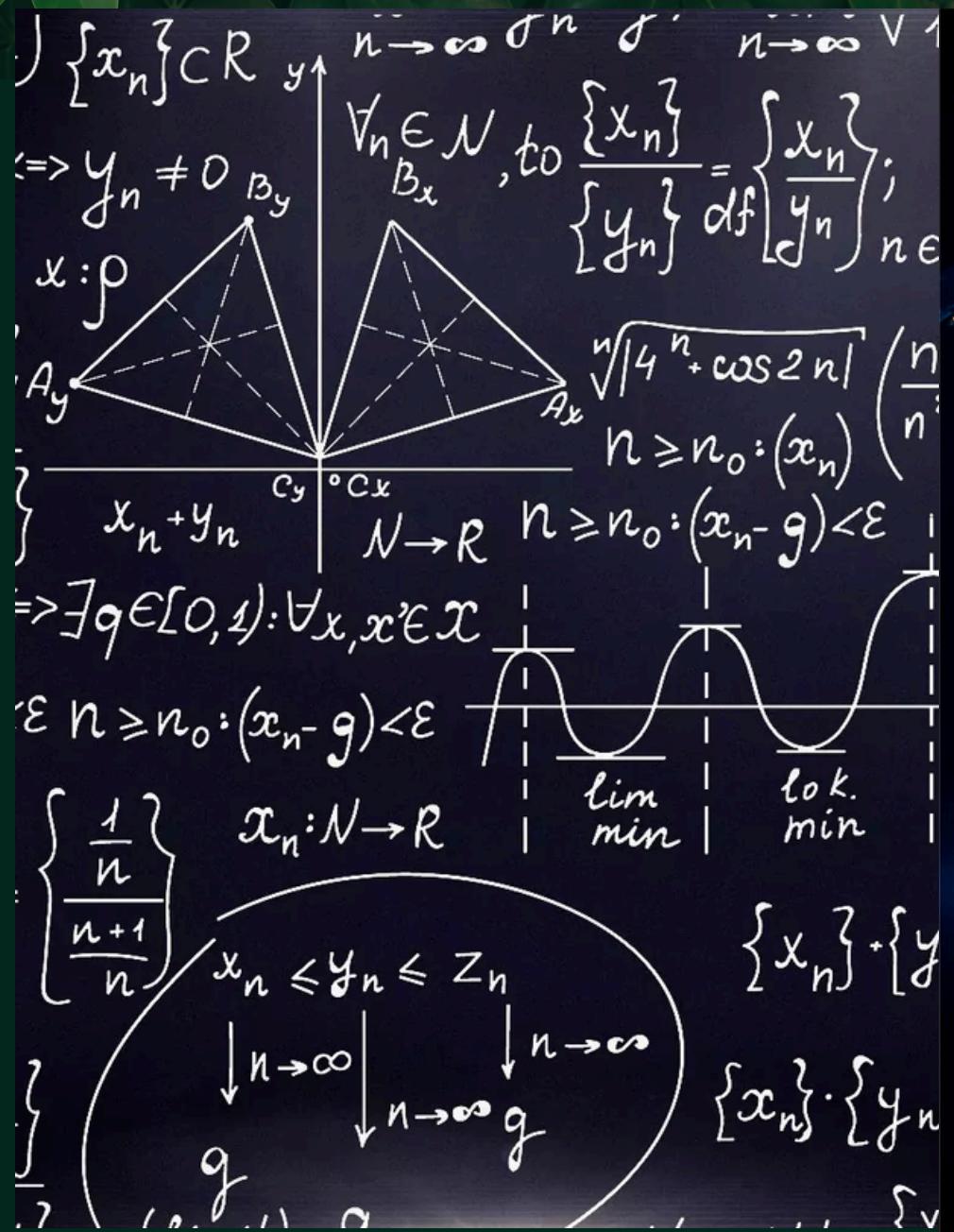
# Introduction |

Sansevieria: converts plastic & tyres into feedstock for the petrochemical industry and green fuels  
 ⇒ an 80% reduction in CO<sub>2</sub> emissions vs. oil refineries



Goal: optimize the process with ML + image processing

⇒ identifying plastic types in the queue reduces emissions + speeds up the process



# Introduction |





# — Overview



Learning Process



Image Classifiers



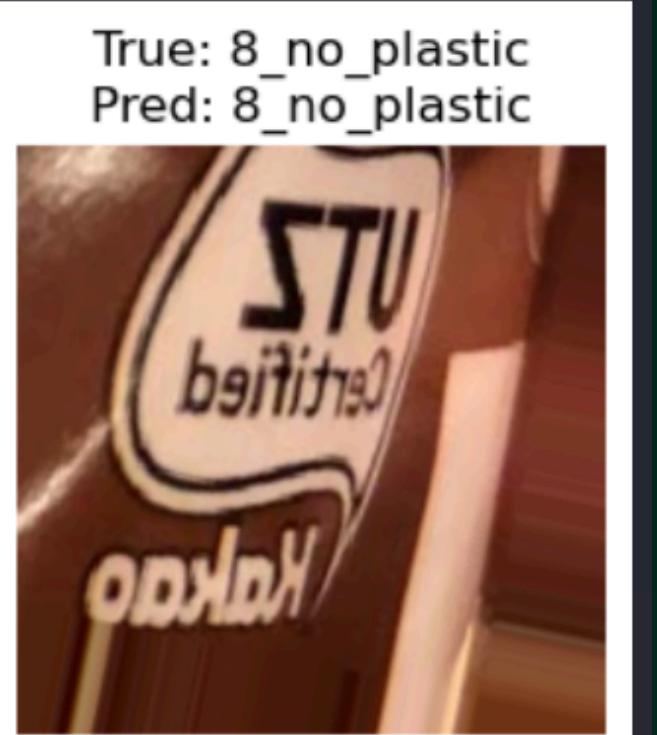
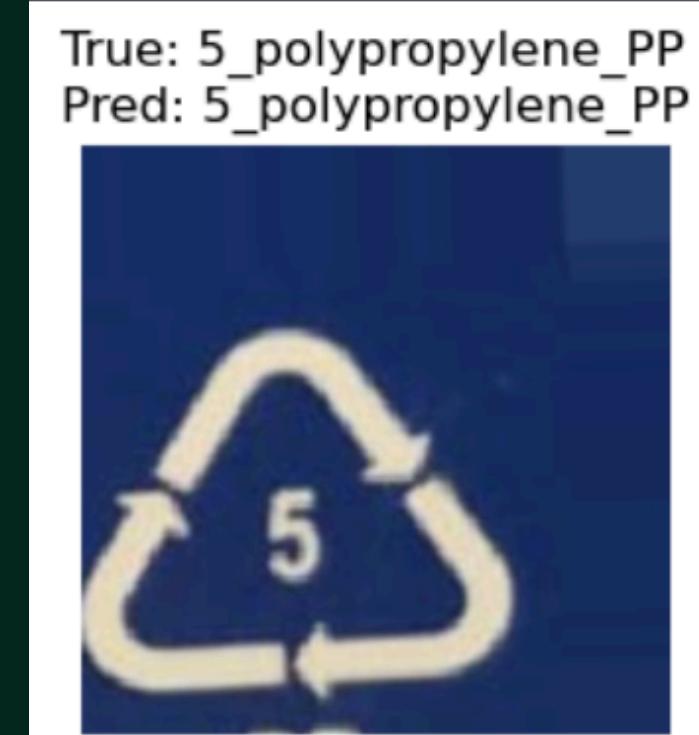
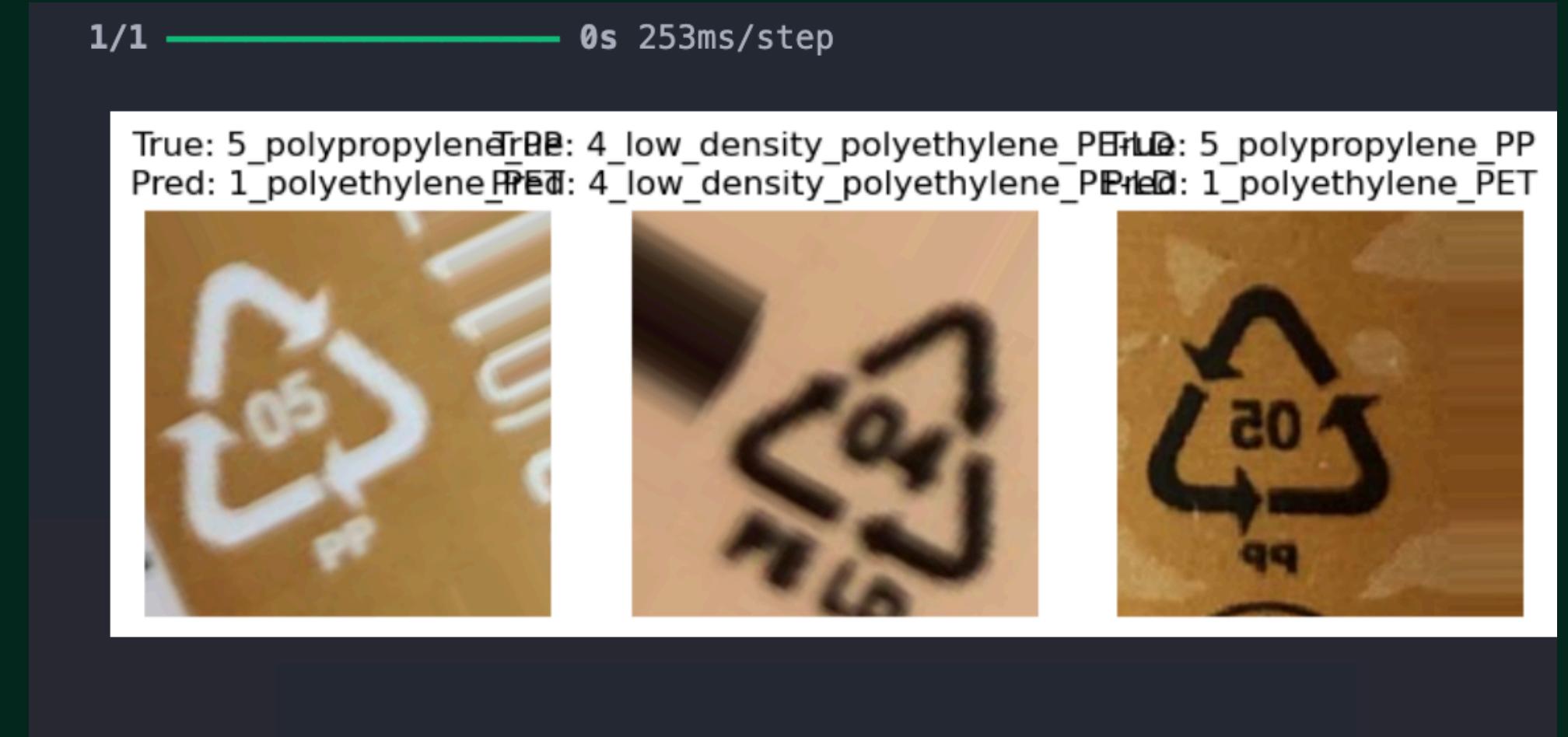
Divide & Conquer

# Group 1:

## Eric, Adrian, and Keval

Build a robust image classifier  
for complex data with CNN

PlasticPredictorModel uses  
ViTforImageClassification  
framework to read PNG files

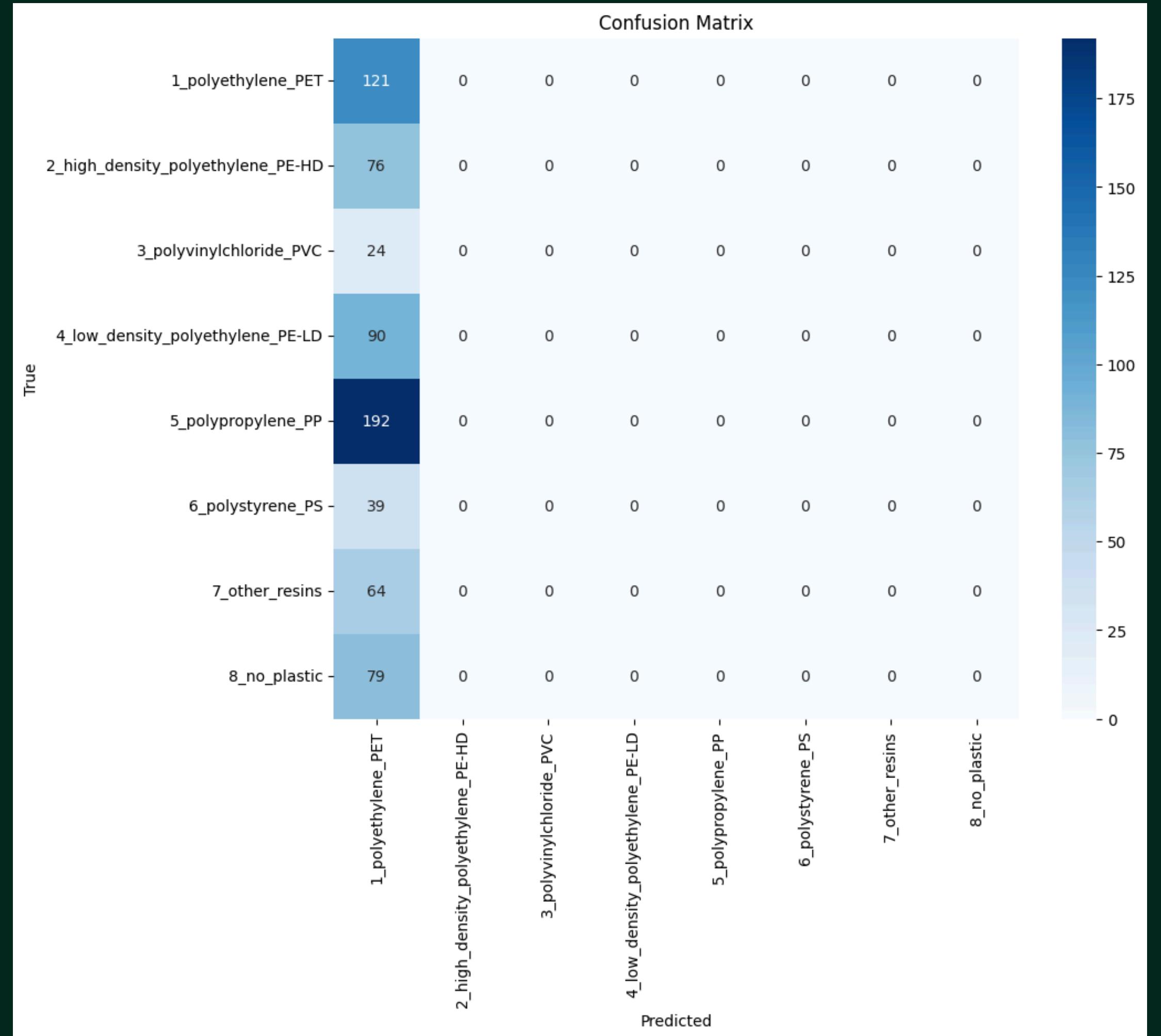


...

Layer (type)	Output Shape	Param #
conv2d (Conv2D)	(None, 224, 224, 32)	896
batch_normalization (BatchNormalization)	(None, 224, 224, 32)	128
conv2d_1 (Conv2D)	(None, 224, 224, 32)	9,248
max_pooling2d (MaxPooling2D)	(None, 112, 112, 32)	0
conv2d_2 (Conv2D)	(None, 112, 112, 64)	18,496
conv2d_3 (Conv2D)	(None, 112, 112, 64)	36,928
max_pooling2d_1 (MaxPooling2D)	(None, 56, 56, 64)	0
conv2d_4 (Conv2D)	(None, 56, 56, 128)	73,856
conv2d_5 (Conv2D)	(None, 56, 56, 128)	147,584
conv2d_6 (Conv2D)	(None, 56, 56, 256)	295,168
conv2d_7 (Conv2D)	(None, 56, 56, 256)	590,080
max_pooling2d_2 (MaxPooling2D)	(None, 28, 28, 256)	0
flatten (Flatten)	(None, 200704)	0
dense (Dense)	(None, 1024)	205,521,920
dropout (Dropout)	(None, 1024)	0
dense_1 (Dense)	(None, 512)	524,800
dropout_1 (Dropout)	(None, 512)	0
dense_2 (Dense)	(None, 8)	4,104

...

Total params: 207,223,208 (790.49 MB)





Google ViT Base

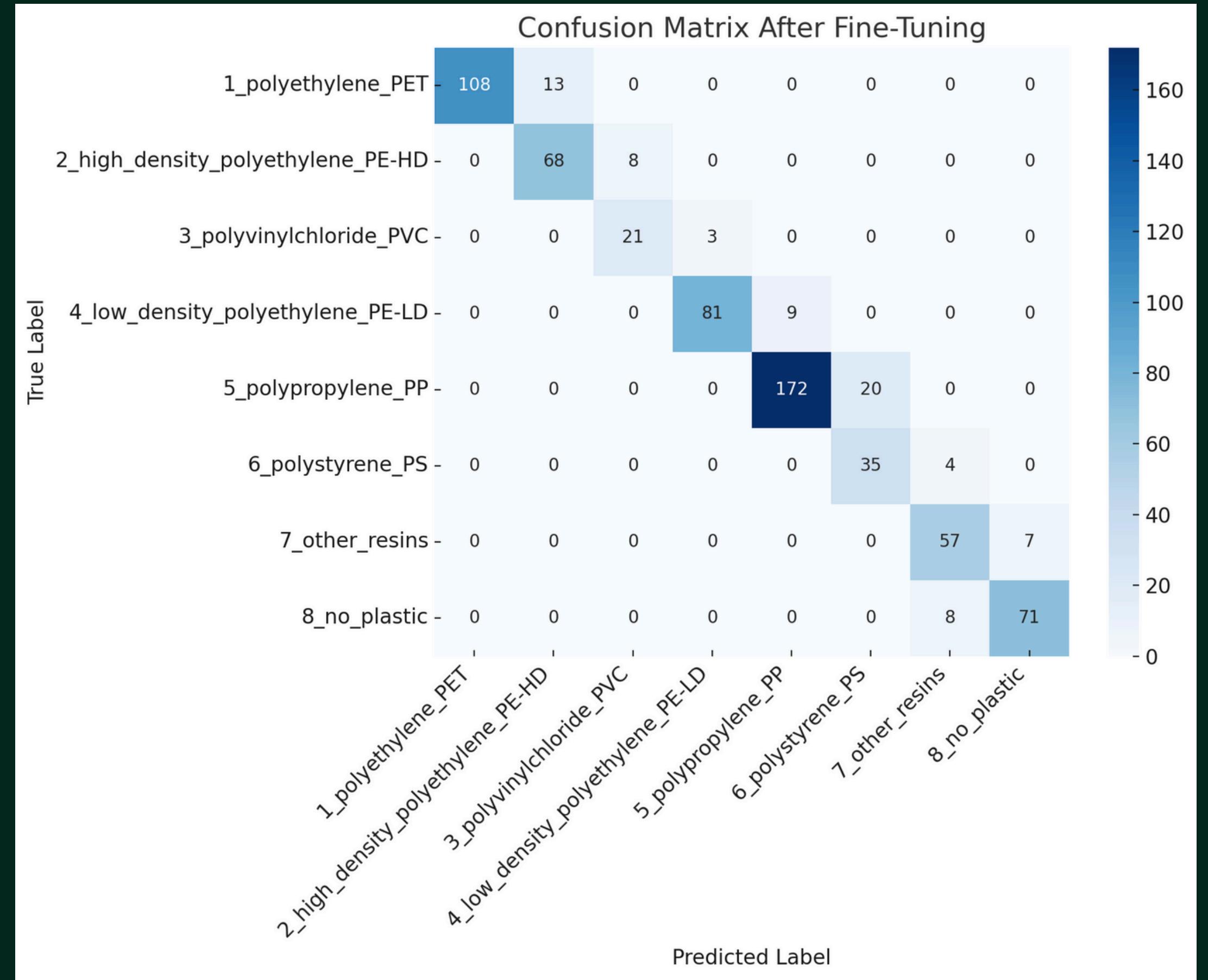
**Images:** Process data with OpenCV for  
224x224 images

**Preprocessing:** Send to a 85/15 test-train  
split for tuning

**Optimization:** Optimized on all-parameters  
across 10 runs with 64-size batches

ROUGE-1 F1 ~ 81%





# Group 2:

## Michael & Christopher

Picture of plastics pile  
→ Identified plastic type

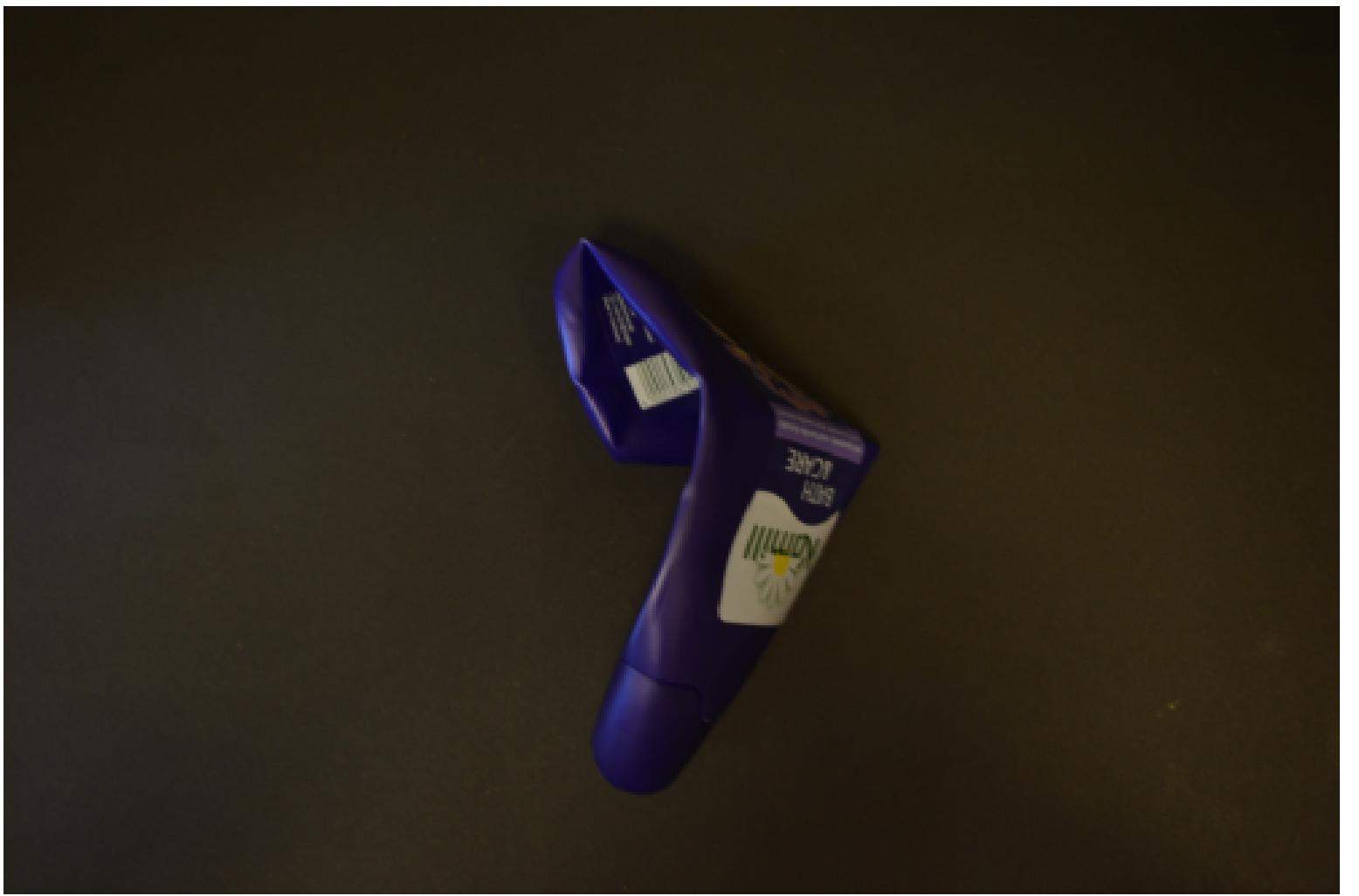


1. Model to classify images

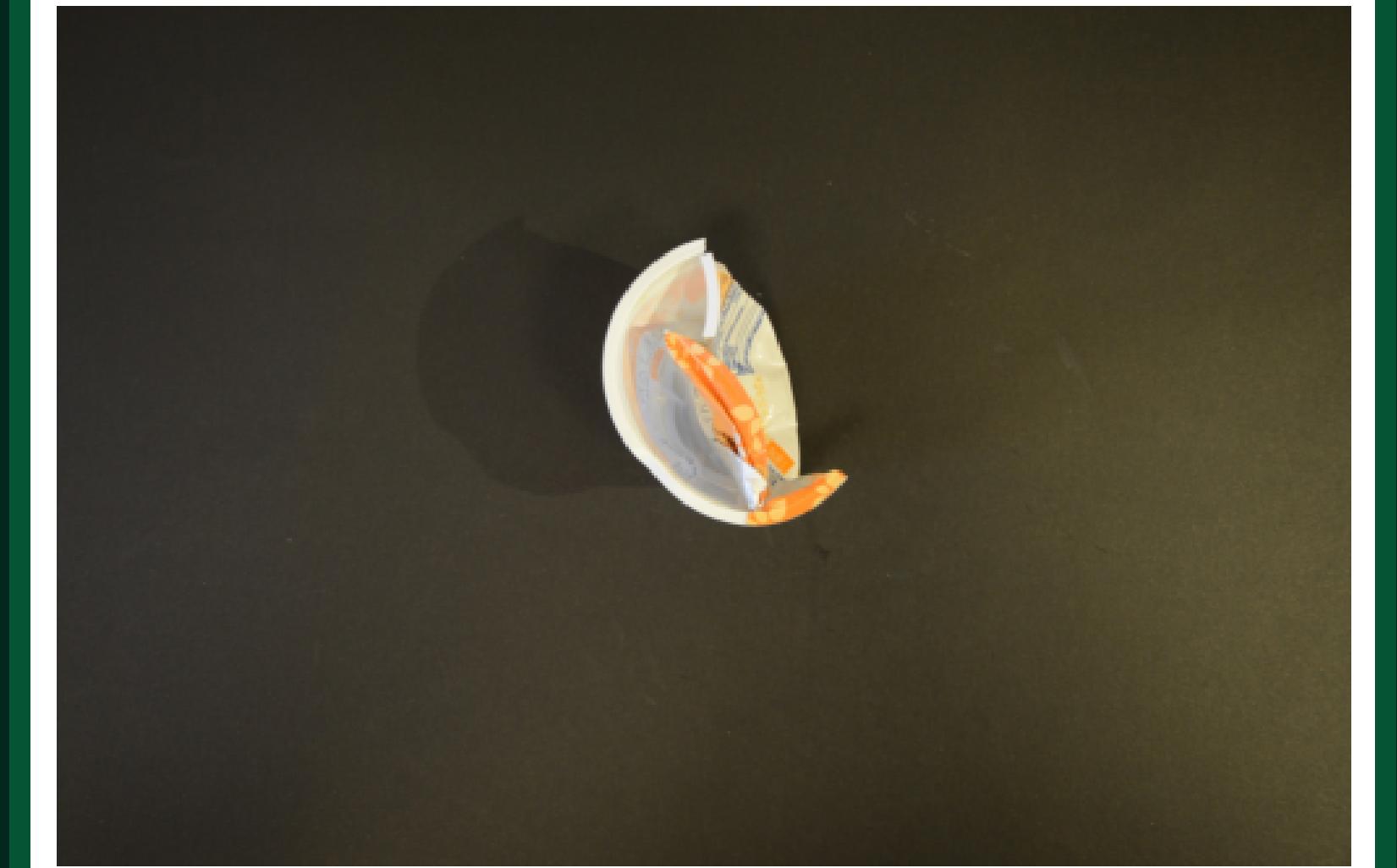
2. Model to detect objects

- Created models with different architectures.
- Most successful model achieved 89% accuracy.

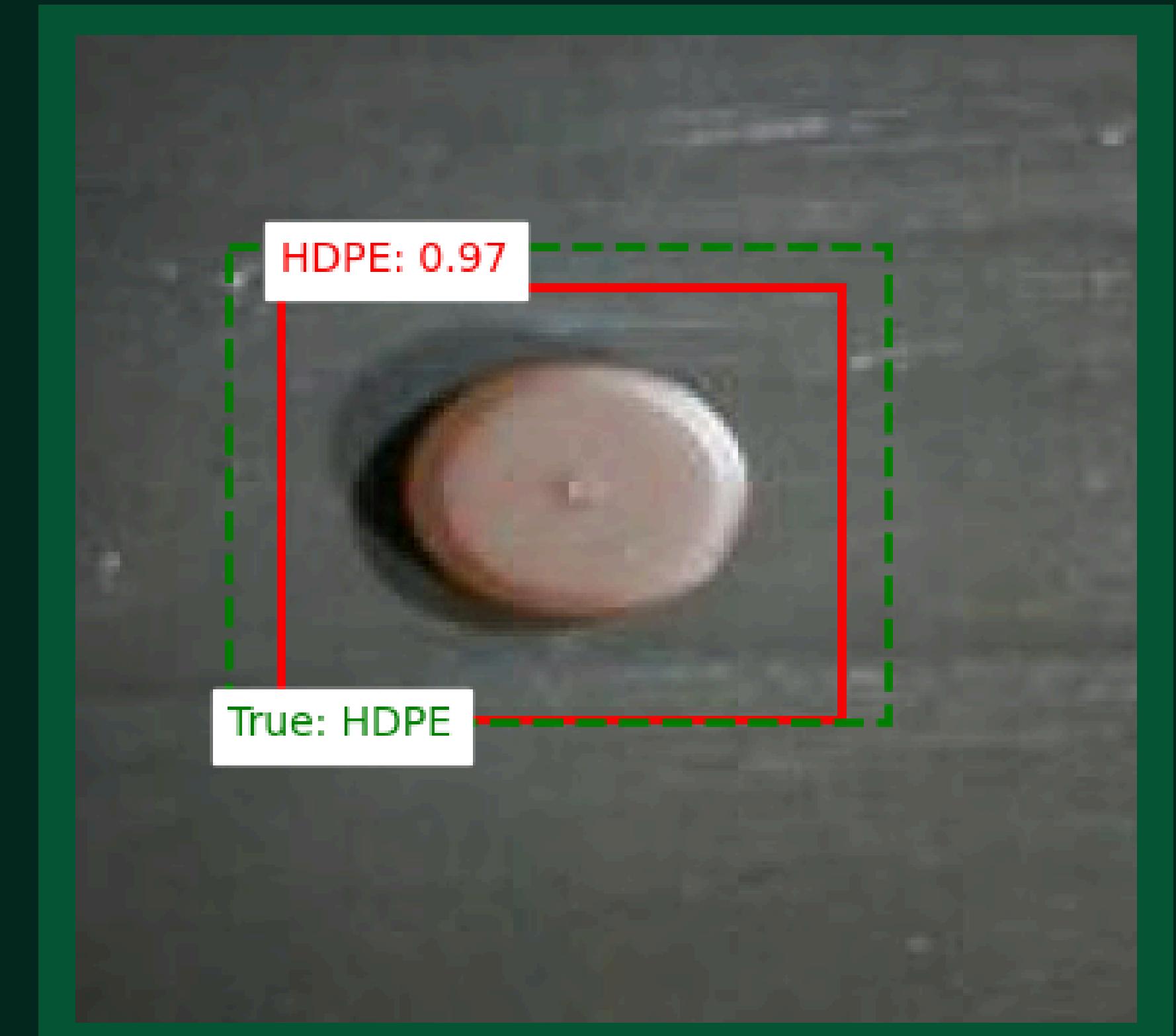
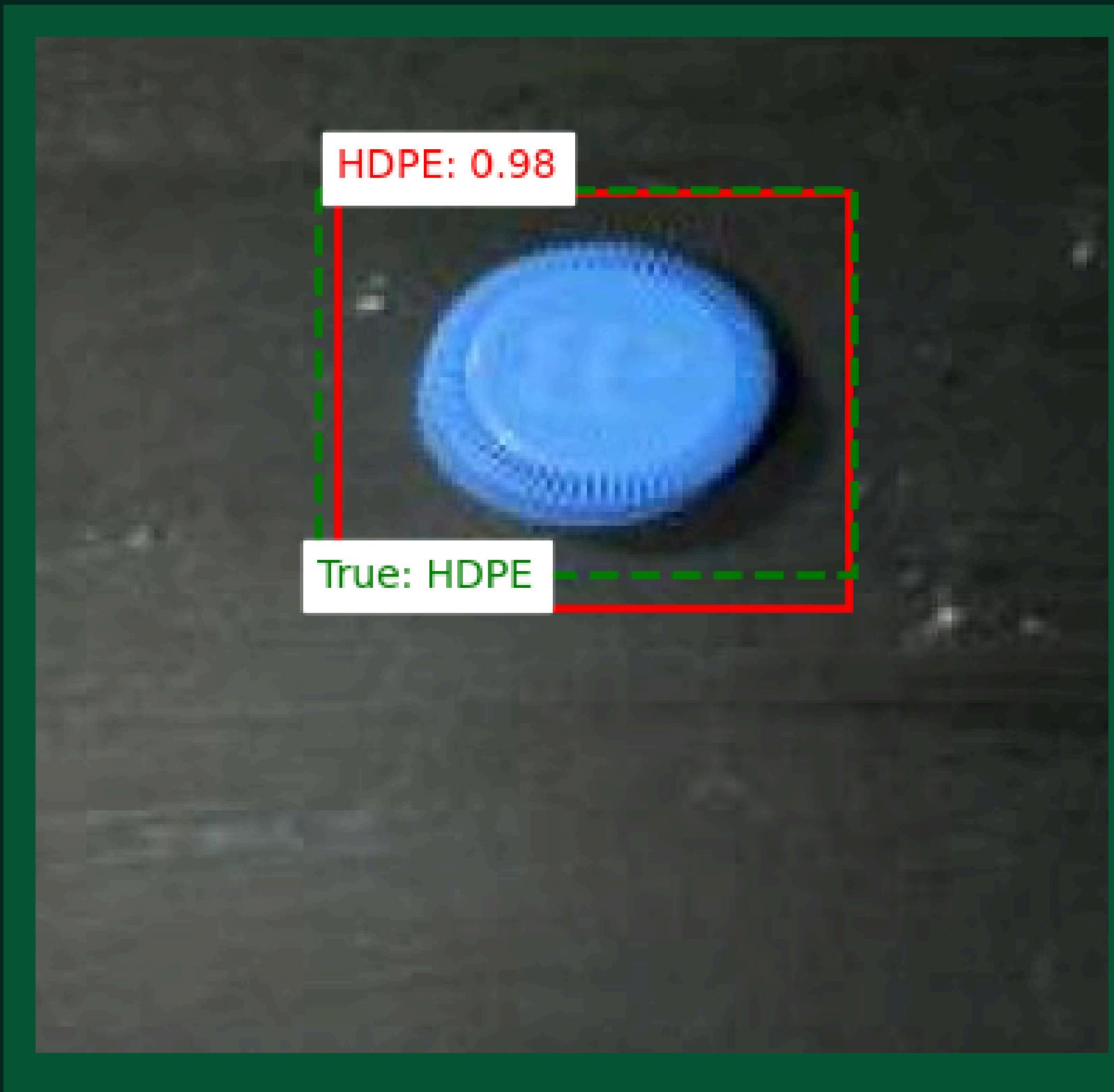
Predicted: PE-HD | Actual: PE-HD



Predicted: PS | Actual: PS

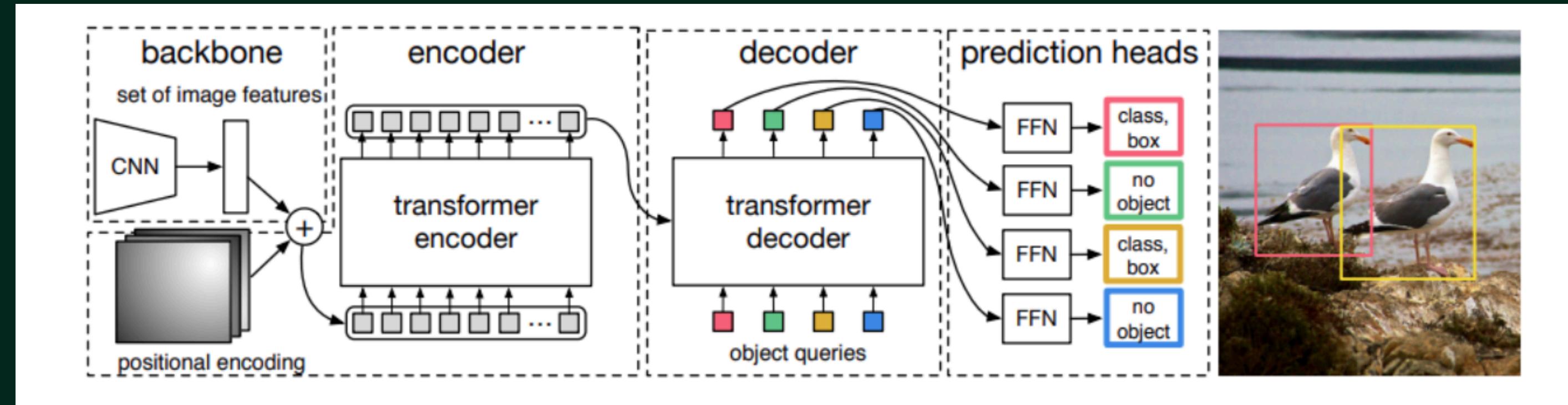


- Used FasterCNN\_MobileNet\_v3
- Achieved ~85% accuracy on detection



# Group 3: Yurim & Jasmine





Model to detect object & classify image at once - DETR

DETR (DEtection TRansformer) is an end-to-end object detection model that uses a transformer architecture to directly predict object bounding boxes and classes without requiring hand-designed components like anchor boxes or non-maximum suppression.



- **Data Preprocessing:**

Converts image files to **COCO** format

Extracts plastic type information from **filenames/paths**

Automatically generates **bounding boxes**

- **Model Architecture:**

Utilizes pretrained DETR model fine-tuned for plastic classification  
(facebook/detr-resnet-50)

Uses Mean Average Precision (mAP) as primary evaluation metric

- **Training Pipeline:**

Monitors validation loss and mAP

Implements checkpointing and **early stopping**

Includes learning rate scheduling

- **Inference & Visualization:**

Detects plastic objects in **new images**

Visualizes **bounding boxes, class labels, and confidence scores**

Evaluates performance on validation dataset





Thank You  
FOR YOUR ATTENTION