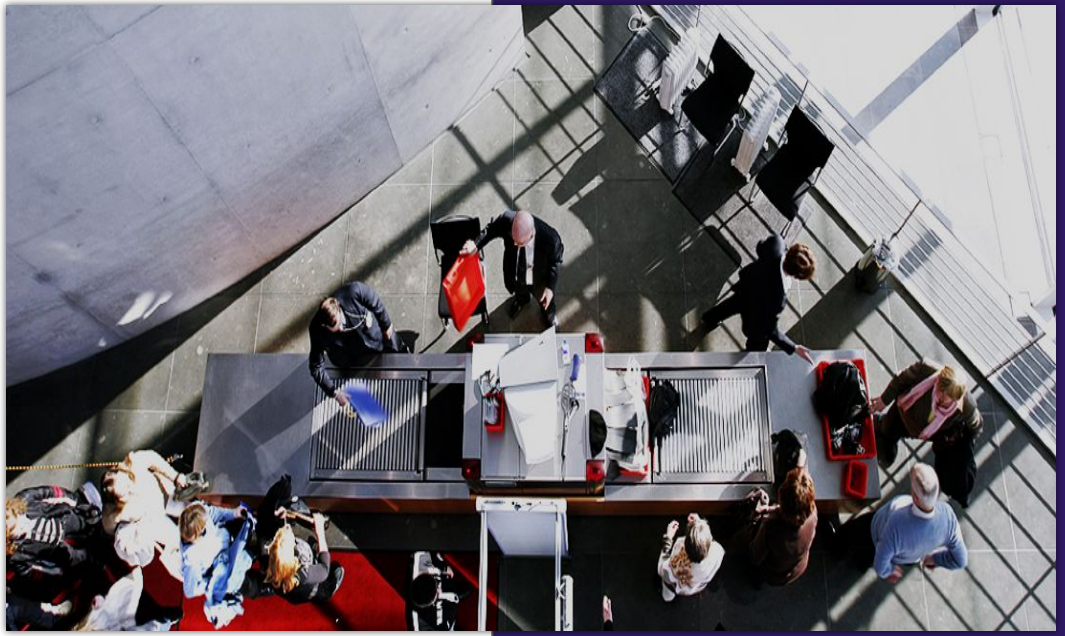
Potentially suspicious items Detection in X-ray Scans with Faster R-CNN



Project by:

- Nikolaos Prevolis
- Stefanos Tzaferis
- Eleftheria Galiatsatou
- Peter Kayiwa

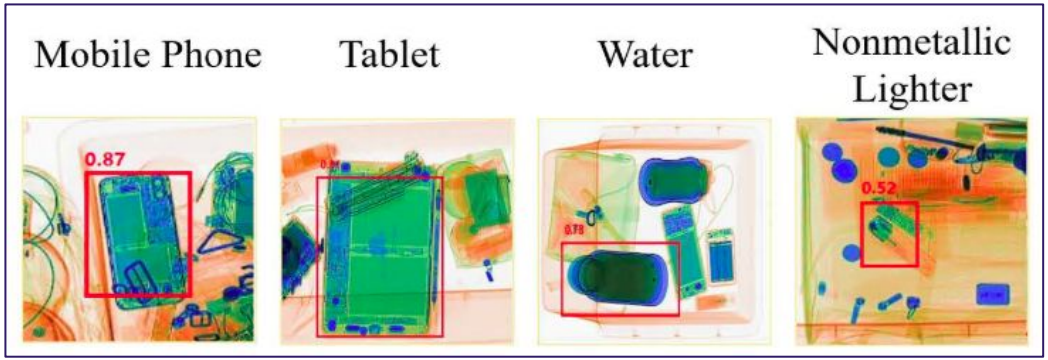
About the Project

This project aims to help airport security by:

- **Detecting potentially suspicious items in luggage X-ray scans using a deep learning model.** We trained a Faster R-CNN model to recognize 8 kinds of suspicious items.
- **Highlighting the location of items clearly on the X-ray image.** The model draws a box around any detected object and labels it, so that a human operator can quickly spot it.
- **Making the inspection process faster and more consistent.** By helping detect items automatically, the system reduces the chances of human error and speeds up decision-making during security checks.

Dataset

- Dataset: Hi-Xray – 45,364 annotated X-ray images (36,295 train / 9,069 test).
- Classes: Portable Charger (1 & 2), Mobile Phone, Laptop, Tablet, Cosmetic, Water, Non-metallic Lighter



The Process

1

Data Preparation

Hi-Xray dataset with 45,000+ annotated X-ray images

2

Pre-Processing

-Resized images to a standard shape.
-Normalized pixel values and converted images to tensors.

3

Model architecture

Built a custom Faster R-CNN object detection model using VGG16,RPN,ROI pooling

4

Training

-Used Stochastic Gradient Descent (SGD) optimizer

5

Evaluation

-using Mean Average Precision (mAP)
-Calculated **AP scores** for each class

6

Interface

simple interface using **OpenCV** and **Gradio**

7

Final output

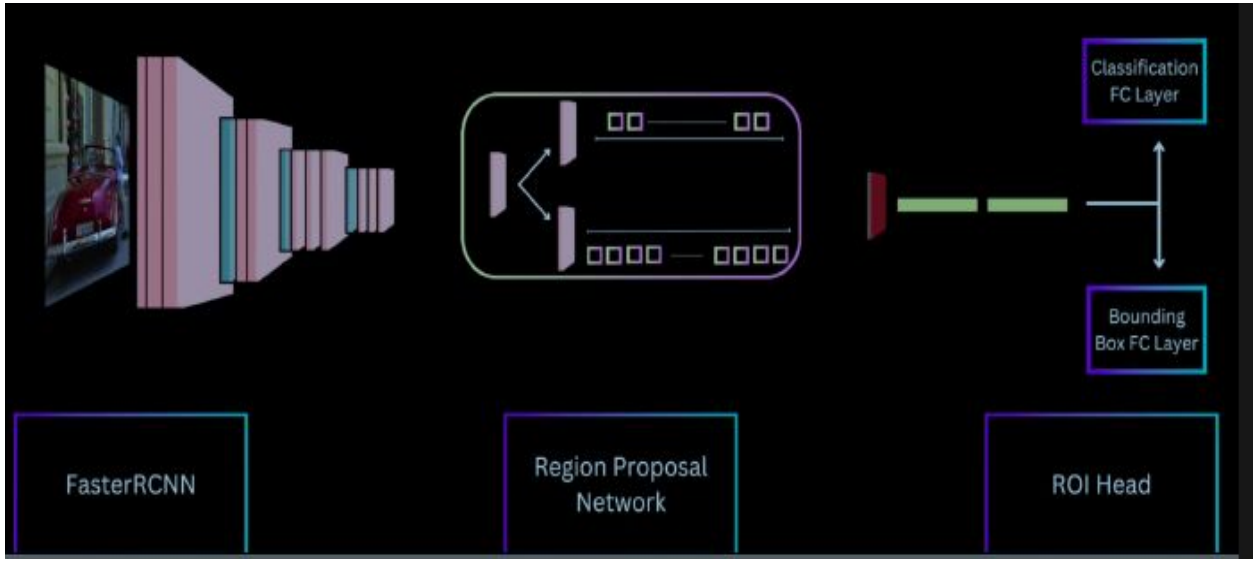
Complete system that:
-Processes new images
-Flags potentially suspicious items

Model Architecture

Based on Faster R-CNN which uses:

- VGG16 Convolutional Backbone
- Region Proposal Network (RPN) for anchor-based object proposals
- RoI Pooling for fixed-size region features
- Classifier & Regressor heads for object class & bounding boxes

Non-Maximum Suppression (NMS) applied to refine overlapping detections



Model take in an image:

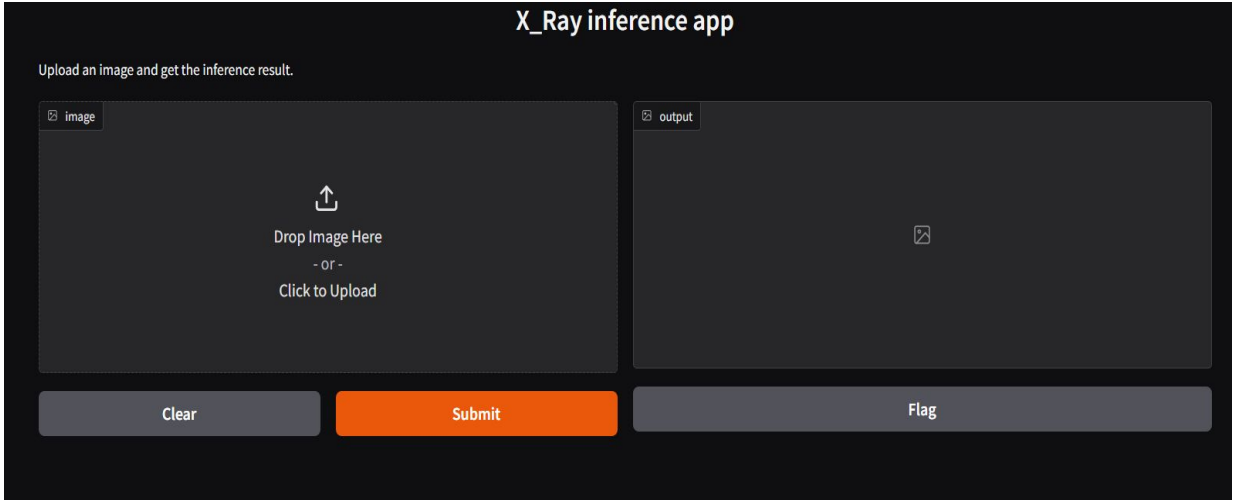
1.The image passes through a CNN (VGG16) that turns it into feature maps.

2.The **Region Proposal Network** (RPN) scans those features and suggests regions that might contain objects.

3.These regions are processed by the **RoI Head**, which decides *what* the object is and *where* it is.

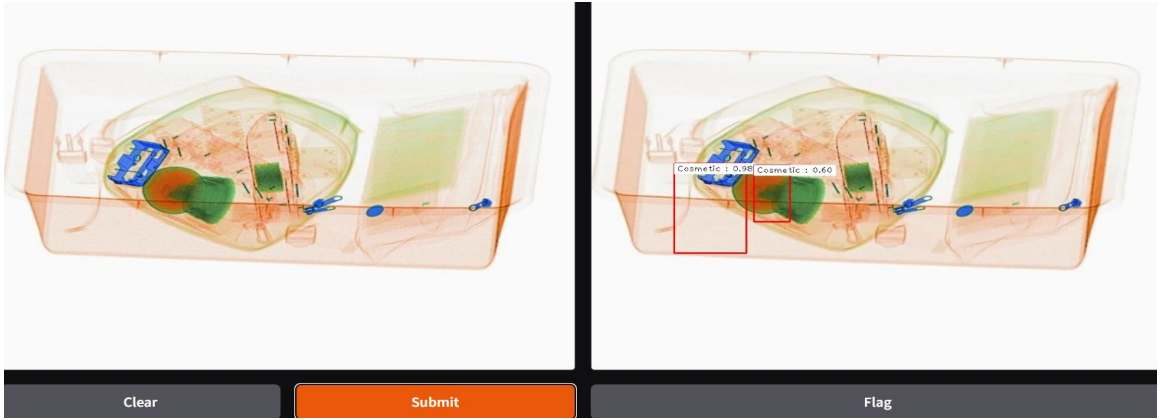
4.The model then draws a box and a label on the image.

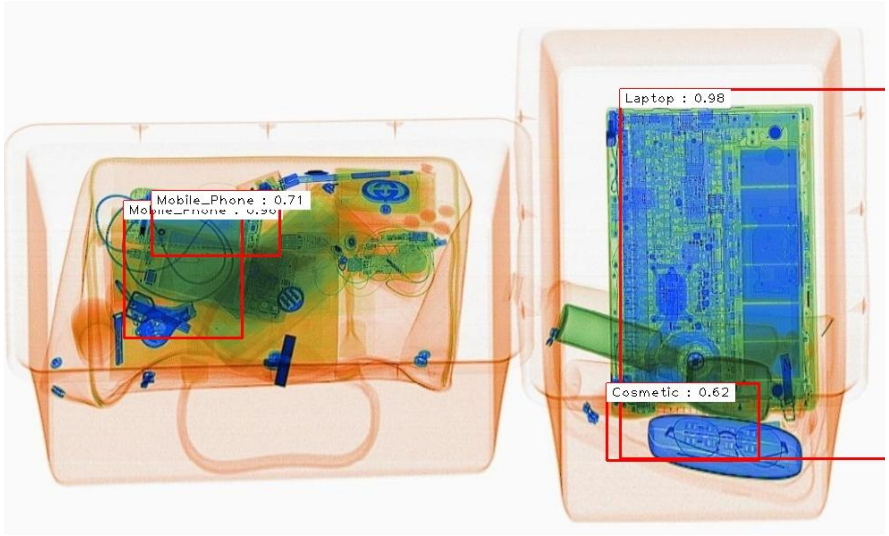
User Interface and Output



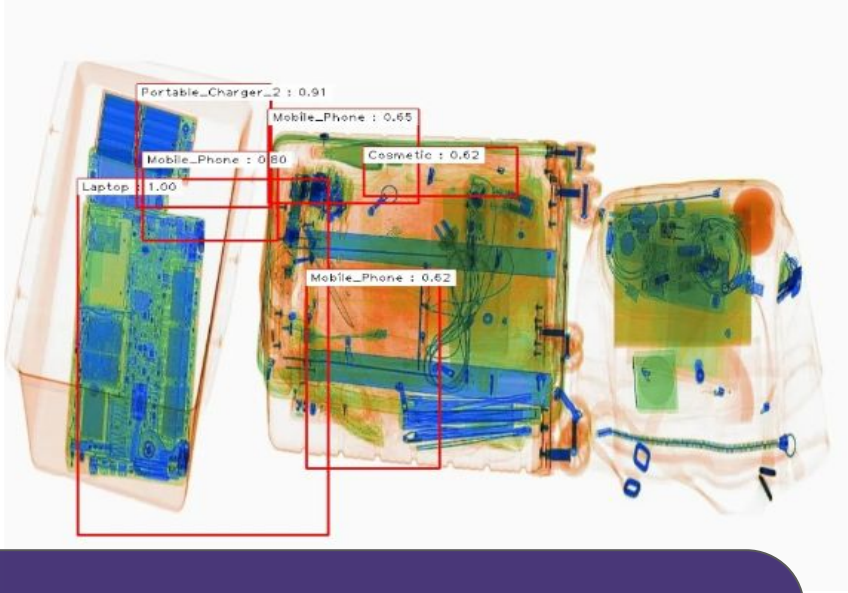
Built a simple interface using **OpenCV** and **Gradio**. User can upload an X-ray image, and the model:

- Predicts items
- Draws boxes with labels and confidence scores
- Saves or displays results in real time


Cosmetic : 0.98
Cosmetic : 0.60
Clear
Submit
Flag


Mobile_Phone : 0.71
Laptop : 0.98
Cosmetic : 0.62

Evaluation



Evaluated the model using **Mean Average Precision (mAP)**

Calculated **AP scores** for each class (Laptop, Phone, etc.)

Best performance on large, distinct objects like **Laptop (AP: 1.0)** and **Portable Charger (0.91)**.


THANK
YOU