AUTONOMOUS DETECTION AND SEGMENTATION OF PROHIBITED ITEMS IN LUGGAGE USING DEEP LEARNING TECHNIQUES

Digital Image Processing

Contents

Introduction

The goal of this project is to use deep learning techniques to develop a system capable of autonomously detecting and segmenting prohibited items within luggage images. This approach aims to enhance security screening processes by identifying unsafe images and highlighting objects that pose potential threats.

Project Overview

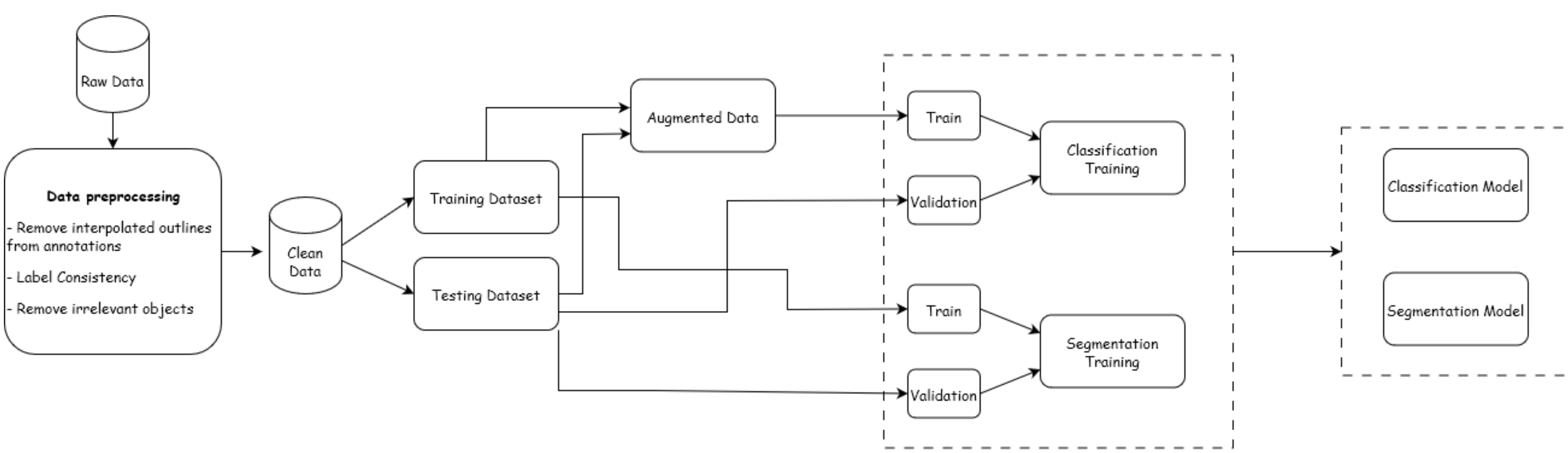
Objectives

- **Classification:** Classify images as safe or unsafe.
- **Segmentation:** Highlight and segment the detected prohibited items.

Methodology

This project involves several key steps

- **Data Cleaning and Preparation:** The dataset required cleaning to remove anomalies and reorganize it for effective model training.
- **Model Training:** Developed and trained a custom deep learning model for object detection and segmentation.
- **Model Evaluation:** Assessed the performance of the trained model using relevant metrics to ensure high accuracy and reliability.
- **Implementation:** Integrated the trained model into a system that can process luggage images, highlighting prohibited items for security personnel.



Data Preparation

Cleaning

The dataset had following issues

- The provided annotations were resized into 768x576, which resulted in objects getting a false outline due to interpolation.
- In some annotations irrelevant objects were highlighted (e.g. pliers).
- The labels were inconsistent throughout the dataset, varying across different batches of images.

Cleaning methodology

- To remove interpolated outlines, closing was applied on the outline label.
- Irrelevant objects were identified and removed manually.
- Ensured consistency in labeling.

Class	**Label**
Background	0
Gun	1
Knife	2

Reorganizing

For both the Binary Classification and Semantic Segmentation datasets, they were structured into two arrays

- One array holds all the images.
- The second array contains corresponding classifications
    - For Binary Classification, it denotes safe (0) or unsafe (1)
    - For Semantic Segmentation, it uses one-hot encoding for labels.

Binary Classification Model

Dataset Description

- Merged gun and knife data into unsafe category.
- Labeling scheme: 0 for safe, 1 for unsafe.

Model Architecture

```
Model: "sequential"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 conv2d (Conv2D)             (None, 256, 256, 32)      896

 max_pooling2d (MaxPooling2D  (None, 128, 128, 32)     0
 )

 dropout (Dropout)           (None, 128, 128, 32)      0

 conv2d_1 (Conv2D)           (None, 128, 128, 64)      18496

 max_pooling2d_1 (MaxPooling  (None, 64, 64, 64)       0
 2D)

 dropout_1 (Dropout)         (None, 64, 64, 64)        0

 conv2d_2 (Conv2D)           (None, 64, 64, 128)       73856

 max_pooling2d_2 (MaxPooling  (None, 32, 32, 128)      0
 2D)

 dropout_2 (Dropout)         (None, 32, 32, 128)       0

 global_average_pooling2d (G  (None, 128)              0
 lobalAveragePooling2D)

 dense (Dense)               (None, 64)                8256

 dropout_3 (Dropout)         (None, 64)                0

 dense_1 (Dense)             (None, 1)                 65

=================================================================
Total params: 101,569
Trainable params: 101,569
Non-trainable params: 0
_________________________________________________________________
```

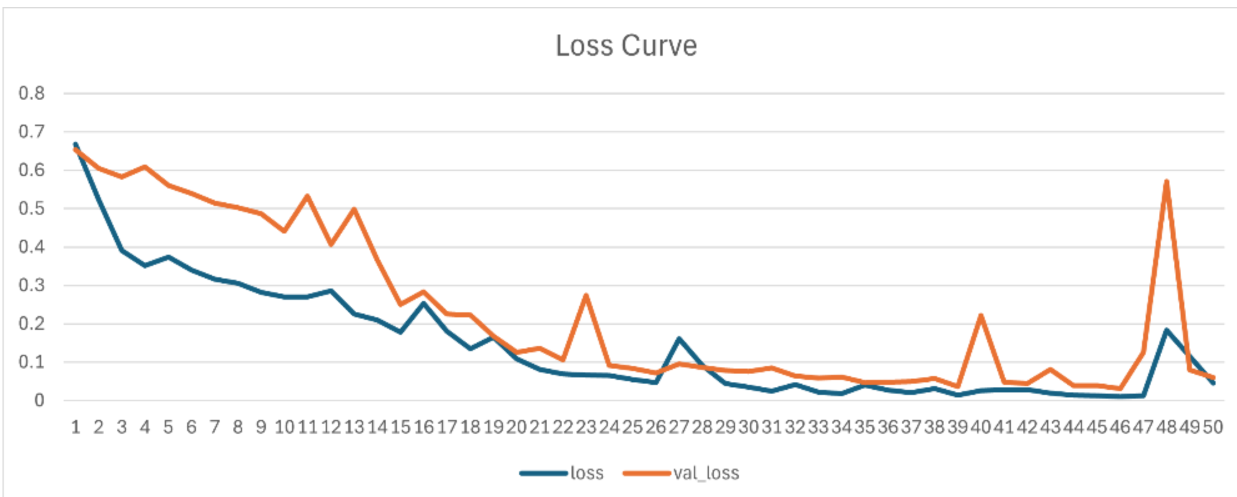
Challenges Faced

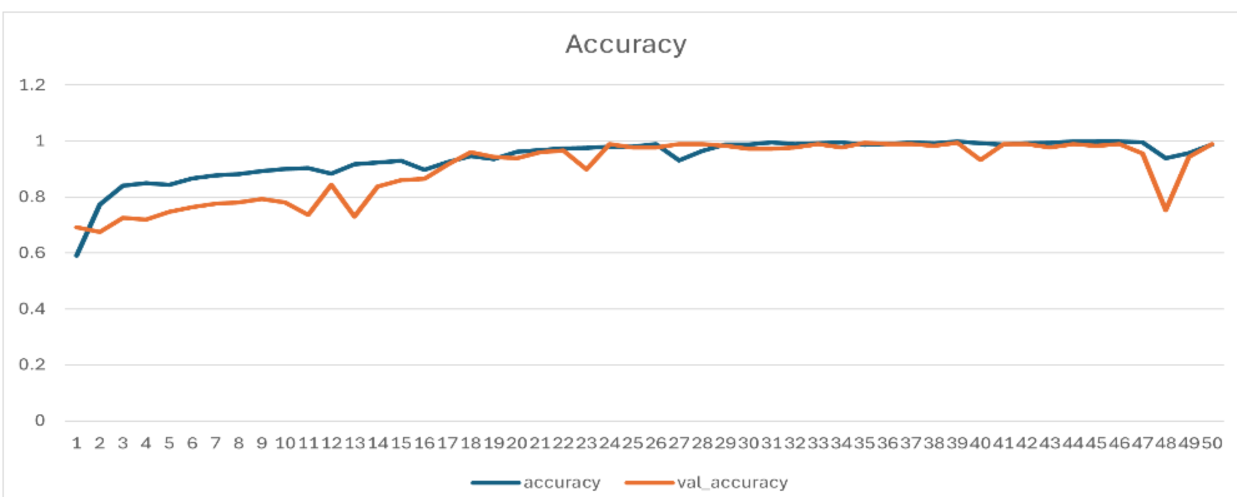
- **Class Imbalance**: Safe images are only 16% of the total dataset.
- **Training-Testing Data Discrepancy**: Different characteristics between training and testing safe images affecting model performance.

Solutions Implemented

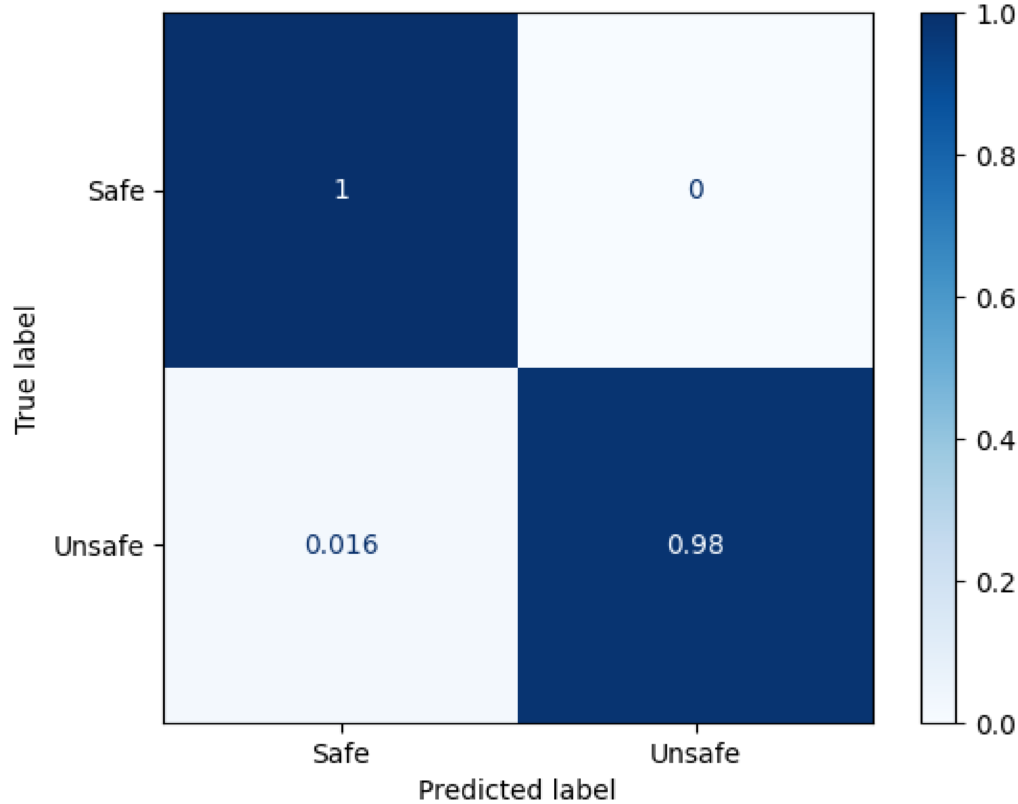
- **Data Augmentation**: Incorporated augmented images from both training and testing datasets to bridge the gap in image characteristics. Where 80% of the augmented data was generated via training dataset.
- **Regularization**: Implemented dropout layers to prevent overfitting.

Evaluation Metrics





- **Confusion Matrix:**



Semantic Segmentation Model

Dataset Description

- All images were merged into one dataset.
- Annotations followed the labeling scheme.

Class	Label
Background	0
Gun	1
Knife	2

Model Architecture

The U-Net architecture was chosen for this model due to its effectiveness in semantic segmentation tasks.

U-Net is designed to work well with relatively small training datasets, making it suitable for applications where annotated data is limited.

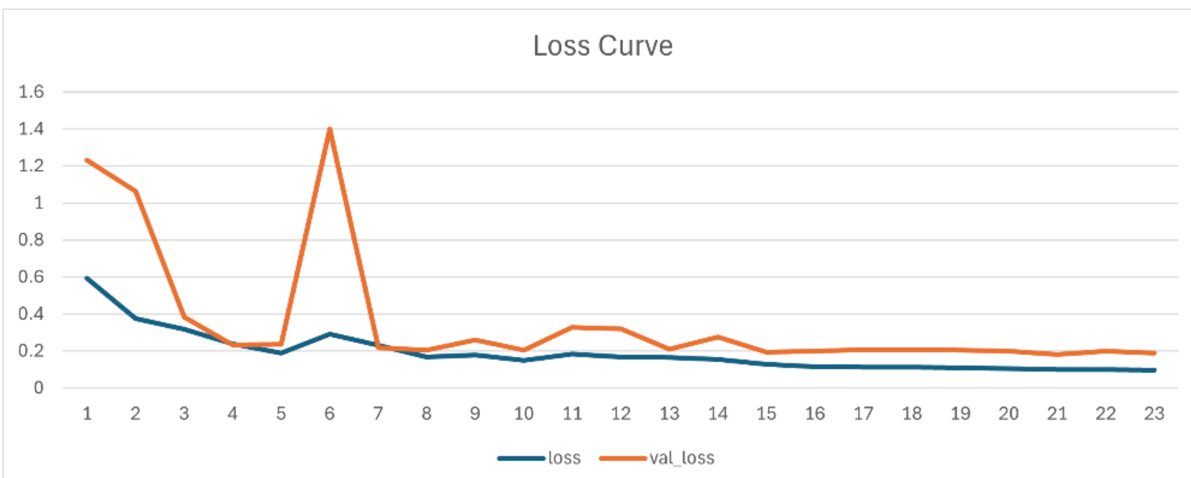
Challenges Faced

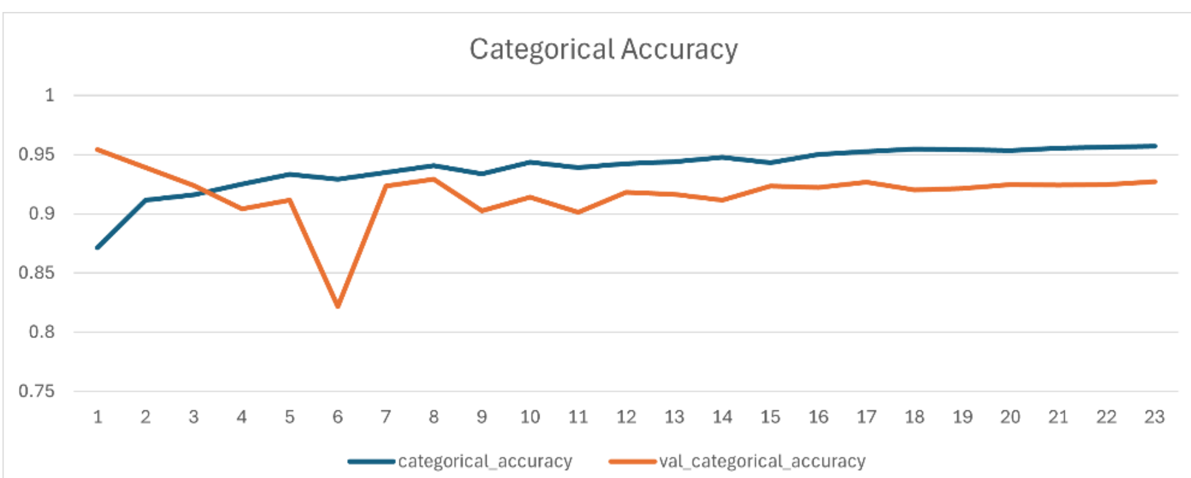
- **Class Imbalance:** The model predicted most pixels as background due to the higher proportion of background pixels.
- **Initial Solution**: Attempted to use class weights with model.fit but encountered an error as class weights are not supported for 3+ dimensional targets.
- **Subsequent Solution:** Switched to sample weights, but this approach was computationally expensive and caused VRAM limitations.

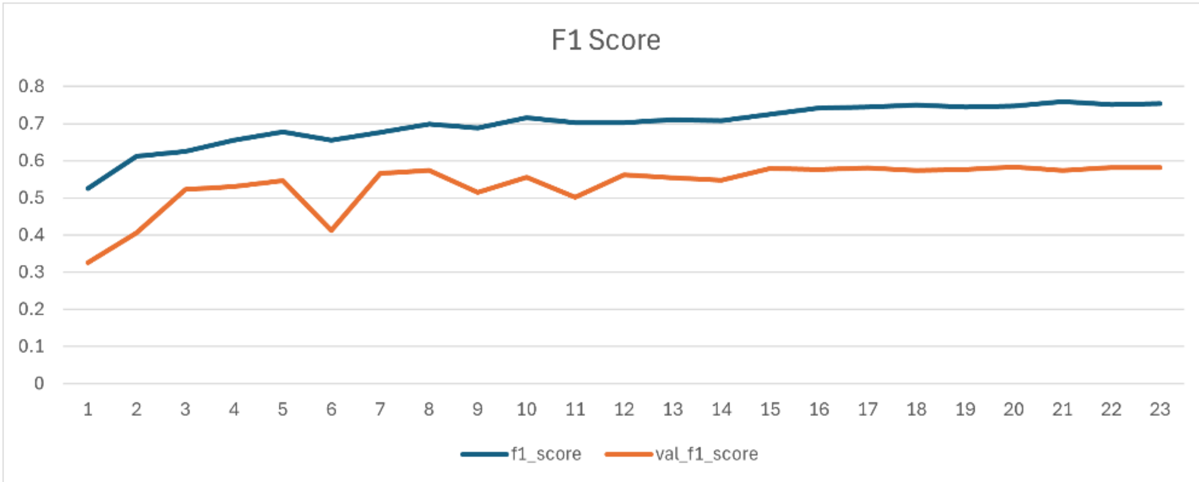
Solutions Implemented

- **Final Solution**: Developed a custom weighted categorical cross-entropy loss function. This function normalized the weights before calculating the loss, effectively handling the class imbalance without exceeding VRAM limits.

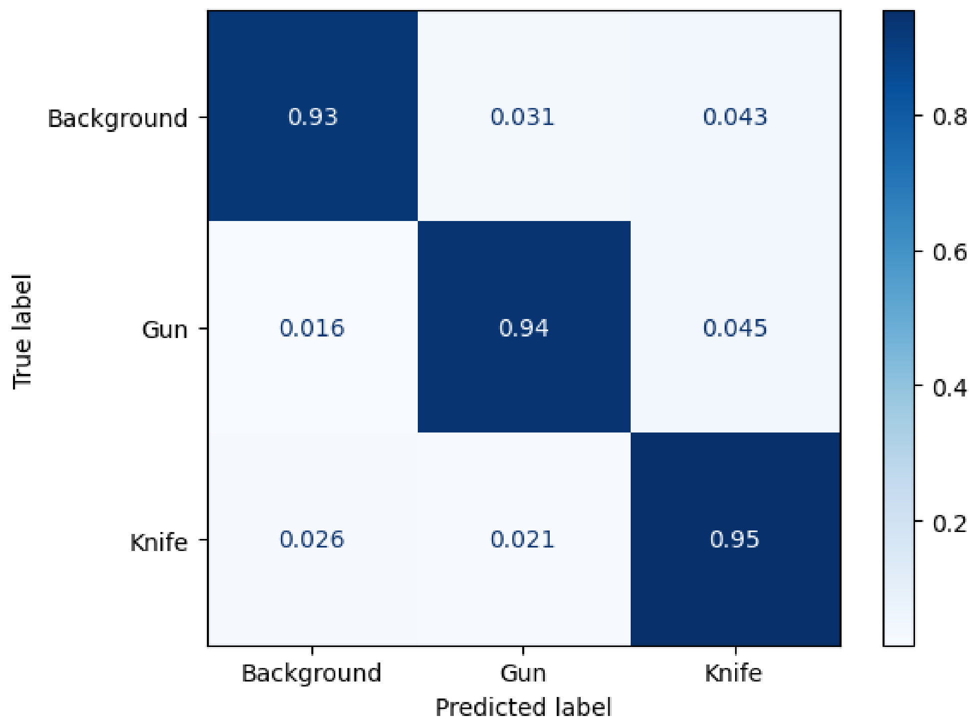
Evaluation Metrics







- **Confusion Matrix:**

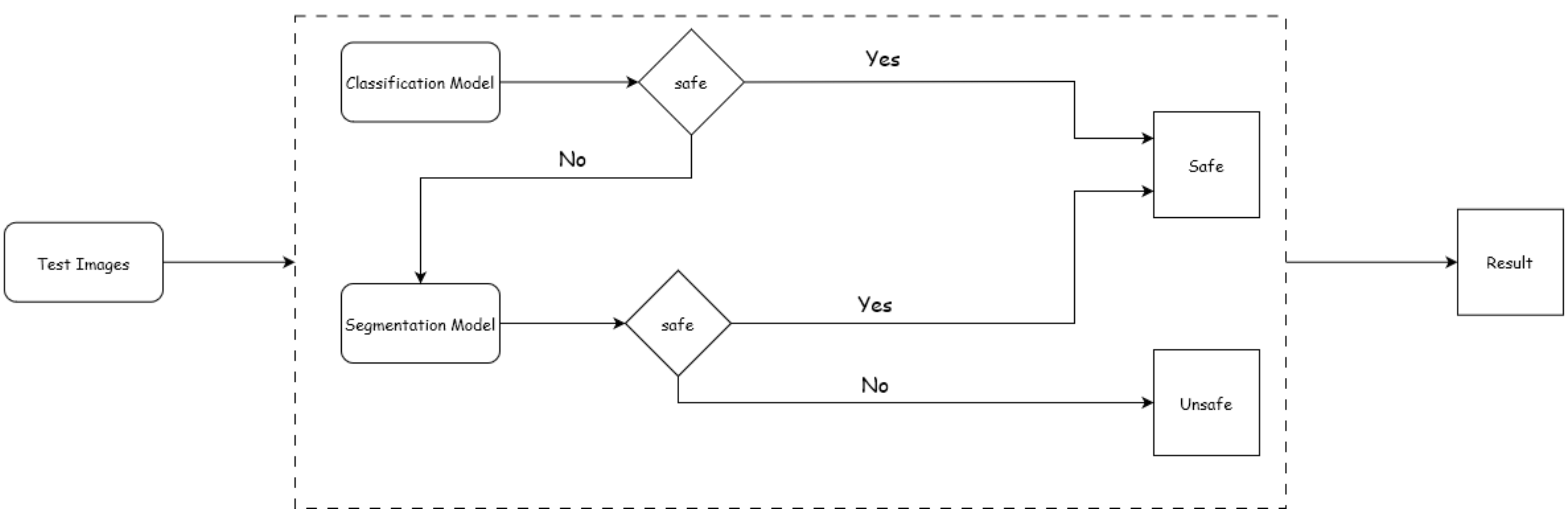


System Integration

Description of the integration process where the binary classification identifies potentially unsafe images and upon detection the semantic segmentation model highlights the prohibited items.

Images that need to be classified and segmented to determine their safety status are processed as follows:

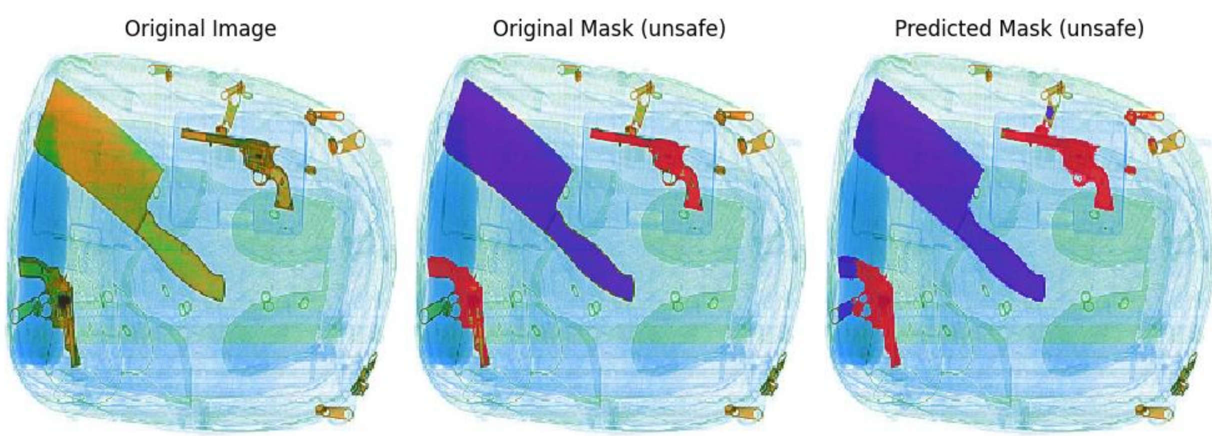
- **Classification Model**: images are first passed through the classification model to predict if they are safe or not.
- **Decision Node - Safe**:
  - Yes: If the classification model predicts the image as safe, it is classified as safe
  - No: If the classification model predicts the image as not safe, the image is passed to the segmentation model for further analysis.
- **Segmentation Model**: images predicted as not safe by the classification model are analyzed by the segmentation model to determine their safety status.
- **Decision Node - Safe**:
  - Yes: If the segmentation model does not find prohibited items in the image, it is classified as safe.
  - No: If the segmentation model finds prohibited items, it is classified as unsafe.
- **Result**: the result indicates whether the images are safe or unsafe. And appropriately generates the masks.



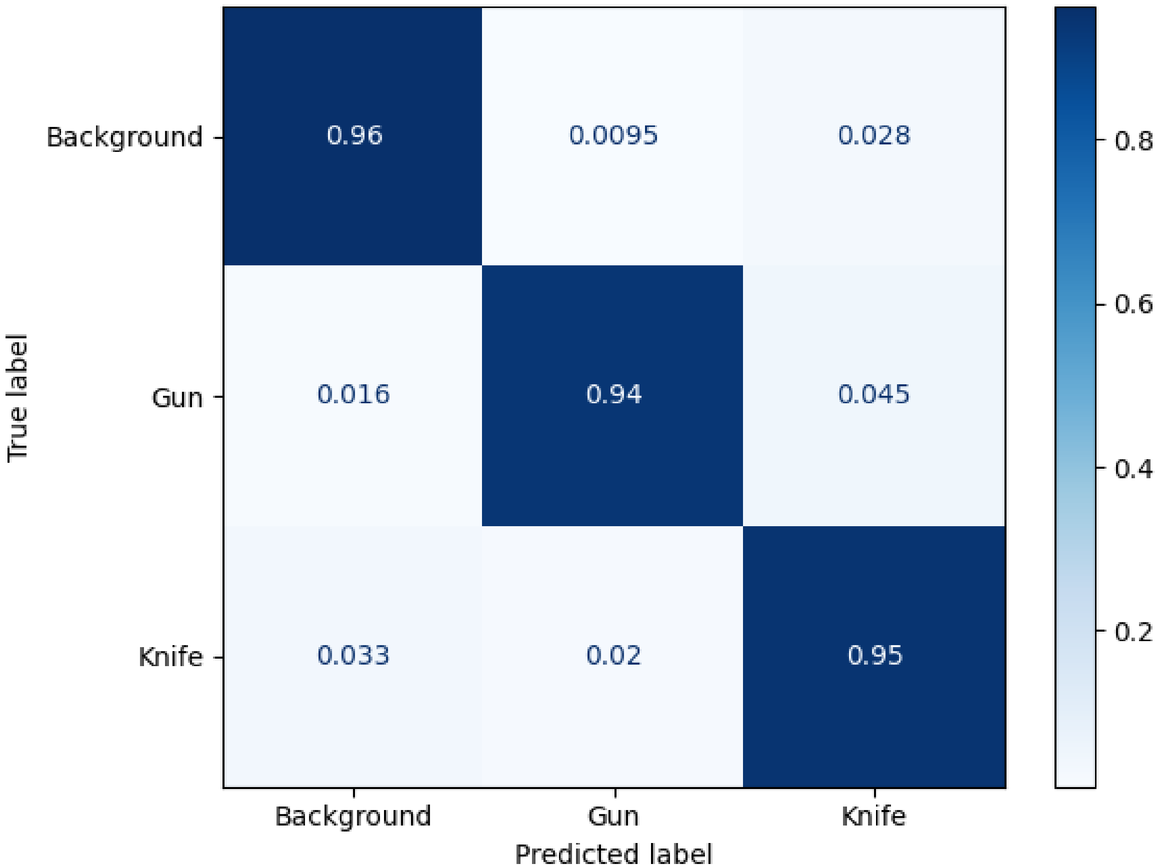
Results


Original Image
Original Mask (safe)
Predicted Mask (safe)


Original Image
Original Mask (unsafe)
Predicted Mask (unsafe)


Original Image
Original Mask (unsafe)
Predicted Mask (unsafe)

System Confusion Matrix



Conclusion

The project not only successfully developed a deep learning-based system for the autonomous detection and segmentation of prohibited items in luggage images but also served as a valuable learning experience in several key areas.

Addressing the challenges faced in various aspects of the project such as data preparation, binary classification and semantic segmentation provided invaluable insights into the complexities of deploying deep learning models for real-world applications. Overcoming issues like annotation inconsistencies and class imbalance through systematic approaches like data cleaning, augmentation, and custom loss function development was critical to achieving accurate and reliable results.