End of Semester Final Deliverables Check List

The following documents must be included in the final submission by the due date.

Deliverables Check List	Submitted Yes/No/NA
Main Document	
• Cover Sheet	Yes
• Check List	Yes
• Project Report (Video link attached)	Yes
Appendices	
• Source code and executable	Yes
• Who did what?	Yes
• Presentation slides	Yes

Cover Sheet Information

Project Title: Reshaping the Retail Industry using AI vision		
Team Name: Group 9		
ID Number		*Name*
1	101233250	Munieb Awad Elsheikhidris Abdelrahman
2	102776293	Rafid AL JAWAD
3	102763758	Kelly Atieno DIANG'A
4	102762726	Farisya Kayrin Binti Fahzmie

Who Did What?

As you are to be individually assessed it is necessary to ensure your marker understands your individual contribution. This document is to demonstrate who was responsible for each piece or contribution to each piece of work in your project. The following is a template to present and **must be signed by all team members**.

Project Title	
Munieb Awad Elsheikhidris Abdelrahman	Annotated the Multi class data. Build and trained the multi class model for all three arcitcture. Evaluated all three multi class model. Build and disgned the gui. Image detection, video detection, live video detection. Writing the report PowerPoint and Video presentation
Rafid AL JAWAD	Annotating test dataset and unorganized training set Train unorganized one-class model Writing of code for Google Collab for unorganized class
Kelly Atieno DIANG'A	Annotation of empty shelves dataset in the multiclass training Refining the data annotation for the empty shelves dataset for one class training Training both models for the empty shelves one class training Writing the report
Farisya Kayrin Binti Fahzmie	Data collection and data annotation for one-class model, empty Report writing Training for empty class Writing of code for Google Collab for the training of empty class. PowerPoint and video presentation

I declare this is an accurate description of team contributions of the team members

Team Member Name	Signature	Date
Munieb Awad Elsheikhidris Abdelrahman		24/5/2024
Rafid AL JAWAD	Rafid	24/5/2024
Kelly Atieno DIANG'A		24/5/2024
Farisya Kayrin Binti Fahzmie	Farisya	24/5/2024

Table of Contents

1. Introduction

In the dynamic realm of retail, where consumer preferences evolve rapidly, the integration of cutting-edge technologies becomes imperative for staying ahead of the curve. This project embarks on a journey to harness the transformative potential of AI vision within the retail sector. Focusing on the implementation of object detection systems, specifically tailored to address the pressing challenges of out-of-stock detection and product management, our endeavor is to propel the industry towards enhanced efficiency and customer satisfaction. By leveraging meticulously annotated datasets and employing state-of-the-art object detection pipelines, we aim to unveil the comparative strengths and weaknesses of one-class and multi-class approaches. Through this exploration, we endeavour to provide actionable insights that will empower retailers to optimize store operations, streamline supply chain processes, and ultimately redefine the retail experience for the modern consumer.

2. Overall System Architecture

The section on overall system architecture provides a comprehensive overview of the various components and their interactions within the project framework. It outlines how different elements such as data collection, object detection models, graphical user interface (GUI) and evaluation methods are integrated to achieve the project's objectives effectively. This section serves as a roadmap, guiding readers through the design and implementation of the system, highlighting its modular structure, and emphasizing the collaborative effort required for its successful development.

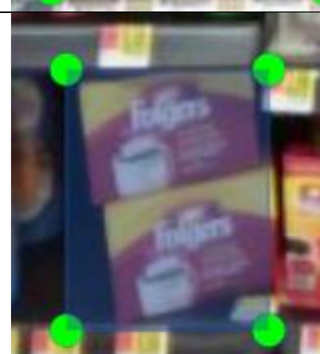
2.1 Data Collection and Annotation

Our primary goal is to curate high-quality datasets for training and testing our object detection models while ensuring the integrity of the evaluation process. To achieve this, we undertake several crucial steps. Firstly, we collect our own training and testing sets, meticulously sourcing images from reputable sources like Roboflow and Kaggle. Roboflow serves as a versatile Computer Vision developer framework, offering robust data collection, preprocessing, and model training techniques. Meanwhile, Kaggle is a prominent platform and online community designed for data scientists and machine learning practitioners. It offers a wide range of resources, including datasets, competitions, and collaborative tools, to help users enhance their skills and apply them to real-world problems. Through Roboflow and Kaggle, we gain access to public datasets. Furthermore, we employ a similarity checker to identify and exclude duplicate data, ensuring the uniqueness and diversity of our dataset.

Once the dataset is compiled, we meticulously annotate each image using annotation tools like LabelMe. LabelMe is a widely used annotation tool that facilitates precise bounding box annotations, enabling us to delineate regions of interest accurately. By leveraging LabelMe, we ensure that our annotations are consistent and comprehensive, laying the foundation for robust model training.

During the annotation process, maintaining annotation consistency is paramount. This involves ensuring uniformity in annotation style and methodology across all labeled images to accurately represent the ground truth. Clear delineation of out-of-stock items and unorganized shelf areas is essential for dataset integrity. Standardized labeling conventions and guidelines are employed to minimize discrepancies between annotators, promoting uniformity in dataset annotation. Additionally, stringent quality control measures are implemented. Regular quality checks are conducted to validate the accuracy and completeness of annotations, ensuring that all relevant instances of out-of-stock items and unorganized shelf areas are properly labeled. Any discrepancies or errors in annotations are promptly addressed to uphold the integrity of the dataset. Below are the standards we have established for scenarios involving out-of-stock items and unorganized shelf displays.

Table 1:Annotation standards

Out-of-stock items	Unorganized shelf displays
	
	
	

Additionally, we adhere to a rigorous 80% split for training and a 20% split for testing, ensuring sufficient data for model training while maintaining a separate set for unbiased evaluation. It's imperative to note that the same testing data are utilized for all three models, and the models remain unexposed to the testing data during training to uphold the integrity of the evaluation process. Through these meticulous data collection and annotation processes, we aim to equip our models with the requisite information to effectively tackle the challenges of out-of-stock detection and product management in the retail environment.

2.1. Object Detection Pipelines

To address the challenges of out-of-stock detection and product placement evaluation in the retail industry, we developed two distinct object detection pipelines: one-class object detection and multi-class object detection.

2.1.1 One-Class Object Detection

In the one-class object detection pipeline, we train separate models for each specific scenario: out-of-stock detection and unorganized shelves evaluation. This approach allows us to focus on the unique characteristics and requirements of each task independently. The out-of-stock detection model aims to identify instances where products are missing from their designated shelf spaces. It is trained exclusively in images annotated for out-of-stock scenarios, learning to recognize empty shelf areas where products are supposed to be stocked. The advantage of this approach lies in its high precision in detecting missing items due to the specialized training dataset and tailored model architecture. Similarly, the unorganized shelves evaluation model assesses the organization of products on shelves, ensuring compliance with planograms and identifying disorganized items. This model is trained in images annotated for product placement scenarios, learning to identify shelves with items that are improperly arranged, misplaced, or have unnecessary gaps. The advantage here is the enhanced capability to detect disorganized shelves and ensure planogram compliance due to focused training on relevant features.

2.1.2 Multi-Class Object Detection

The multi-class object detection pipeline involves developing a single, unified model capable of simultaneously detecting and classifying both out-of-stock and unorganised shelves scenarios. The unified model's objective is to detect and classify both out-of-stock items and unorganized product placements within the same framework. It is trained on a combined dataset that includes annotations for both scenarios, learning to differentiate and classify these scenarios concurrently. This streamlined approach allows a single model to handle multiple tasks, potentially reducing computational overhead and simplifying deployment. The unified model outputs bounding boxes for detected objects, alongside labels indicating whether the detection pertains to an out-of-stock situation or a unorganized shelve issue. By implementing and comparing these two distinct object detection pipelines, we aim to evaluate their respective strengths and weaknesses.

2.2 Model Training and Evaluation

Models are trained using annotated datasets, where each image is meticulously labeled to identify out-of-stock items and unorganized product placements. In this process, out-of-stock scenarios are labeled as 'empty,' while unorganized shelf scenarios are labeled as 'unorganized.' The training process involves feeding these annotated images into the object detection algorithms, allowing the models to learn and recognize patterns specific to each scenario. Once trained, the models are evaluated based on their performance using a variety of metrics.

Precision measures the accuracy of the positive predictions made by the model, indicating the proportion of correctly identified instances among all instances flagged as positive (Huilgol 2024). Recall, on the other hand, assesses the model's ability to identify all relevant instances within the dataset, showing the proportion of true positive instances that were correctly detected (Huilgol 2024). The F1-score combines precision and recall into a single metric by calculating their harmonic mean, providing a balanced measure of the model's performance, especially when there is an uneven class distribution (Huilgol 2024).

Mean Average Precision (mAP) is another crucial metric used to evaluate the models. mAP summarizes the precision-recall curve and provides a single value that reflects the average precision across different recall levels (Shah 2022). It is particularly useful for assessing the performance of object detection models, as it accounts for the precision and recall at various threshold levels for detection (Shah 2022).

By utilizing these metrics—precision, recall, F1-score, and mAP—we can comprehensively assess the performance of our object detection models, ensuring that they effectively address the challenges of out-of-stock detection and product placement evaluation in the retail industry. This thorough evaluation helps in identifying the strengths and weaknesses of each model, guiding further improvements and optimizations.

2.3 Graphical User Interface (GUI)

Our user-friendly GUI enables seamless interaction for object detection tasks. Users can upload images or videos and stream live video for real-time detection by selecting from different multi-class model types MobileNet 640x640, MobileNet 320x320 and ResNet50 640x640.

For image uploads, the GUI processes the image and displays results with bounding boxes around detected items. Video uploads are analyzed frame-by-frame, utilizing a process_fraction of 0.05 to expedite processing by examining only 5% of the frames. The processed video is then available for download.

For live video streaming, the webcam feed is processed in real-time, with detection results displayed directly on the GUI. This interface streamlines the application of advanced detection models, enhancing efficiency in retail environments.

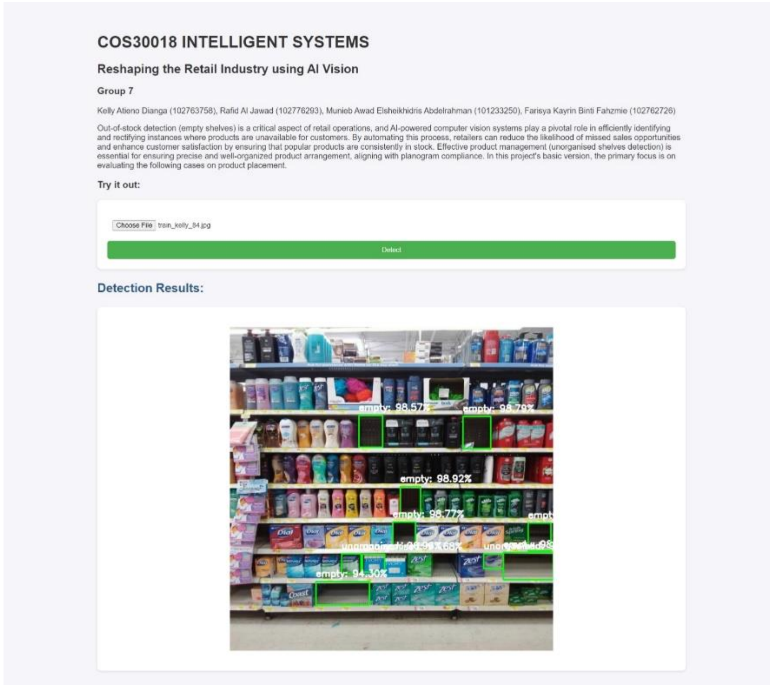


Figure 1: GUI

3. Implemented Machine Learning Techniques

This section outlines the machine learning techniques utilized for developing and evaluating object detection models. The process involved data preprocessing, augmentation, model configuration, training, and evaluation. Through parameter adjustments and augmentation strategies, the models were trained to detect objects accurately. Evaluation metrics such as mAP provided quantitative insights into model performance, facilitating refinement and optimization.

3.1 Model Training

Step 1: Data Preprocessing

- **Resizing**: Images were resized to a fixed size of 640x640 pixels to maintain consistency across the dataset.

Step 2: Data Augmentation

Augmentation techniques such as random cropping was applied to increase the diversity of the training data and improve model generalization.

- **Cropping**: Randomly cropped regions of the images to introduce variations in object sizes and positions.
- **Horizontal flipping**: technique that creates a mirror image, enhancing dataset diversity and model robustness.

Step 3: Model Configuration

- **Batch Size**: A batch size of 8 to 16 was chosen to balance memory constraints and computational efficiency.
- Activation function: RELU_6 is used during training and SIGMOID for post-processing.

Step 4: Model Training

- The training process consisted of multiple epochs (200), with the model updating its weights based on the computed loss between predicted and ground truth bounding boxes.
- During training, both classification and localization losses were minimized to improve detection accuracy.
- Cosine decay learning rate: Gradually reduces learning rate, improving training stability and convergence over 20,000 steps.

3.2 Model Evaluation

Step 1: Evaluation Metrics

- Mean Average Precision (mAP) was used as the primary evaluation metric to assess the model's performance. This metric compares precision, which measures the accuracy of the positive predictions, with recall, which measures the ability to retrieve all relevant instances.

Step 2: Validation

- Performance metrics were computed based on the model's predictions compared to ground truth annotations.
- Non-maximum suppression (NMS): Reduces overlapping bounding boxes, keeping only the most relevant ones for accurate object detection.

4. Examples to Demonstrate System Functionality

How the system's functionality is showcased is discussed in this section.

4.1 Out-of-Stock Detection

The model identifies empty shelves and promptly alerts store managers to restock items.

4.2 Product Management

The model detects disorganized products and ensures they adhere to planograms for proper arrangement.

5. Critical Analysis of the Implementation

Our project's objective was to compare the effectiveness of different object detection models and approaches in the context of out-of-stock detection and product management in the retail sector. Specifically, we aimed to evaluate the performance of one-class and multi-class object detection pipelines utilizing models such as SSD MobileNet V2 FPNLite and SSD ResNet50 V1 FPN. Through rigorous experimentation and analysis, we have drawn several key insights regarding the strengths and limitations of these models.

6.1 One-Class Object Detection Performance

The one-class object detection approach involved training separate models for detecting empty shelves and unorganized products. Our findings indicate that the SSD MobileNet V2 FPNLite models showed significant improvements in detection accuracy when using higher resolution inputs (640x640), as shown in the table. The unorganized shelf detection achieved a substantial mAP of 57.4% with the SSD MobileNet V2 FPNLite 640x640 model, demonstrating its capability to handle complex visual clutter effectively.

6.2 Multi-ClassObject Detection Performance

The multi-class object detection approach involved training a single model to simultaneously detect both empty and unorganized shelves. The performance metrics, detailed in the table, show varying degrees of effectiveness across different models and configurations. Notably, the SSD MobileNet V2 FPNLite 640x640 model achieved the highest overall mAP of 14.18%, while the SSD ResNet50 V1 FPN 640x640 model demonstrated superior performance in detecting unorganized shelves with an mAP of 6.79%.

Table 2: Comparative Analysis and Recommendations

<table>
<tr><th>Class Type</th><th>Model</th><th colspan="2">mAP</th></tr>
<tr><td rowspan="2">Empty</td><td>SSD MobileNet V2 FPNLite 320x320</td><td colspan="2">22.38%</td></tr>
<tr><td>SSD MobileNet V2 FPNLite 640x640</td><td colspan="2">29.93%</td></tr>
<tr><td rowspan="2">Unorganised</td><td>SSD MobileNet V2 FPNLite 320x320</td><td colspan="2">34.2%</td></tr>
<tr><td>SSD MobileNet V2 FPNLite 640x640</td><td colspan="2">57.4%</td></tr>
<tr><td rowspan="6">Multi-class</td><td rowspan="2">SSD MobileNet V2 FPNLite 320x320</td><td>Unorganised: 3.00%</td><td rowspan="2">Overall: 8.07%</td></tr>
<tr><td>Empty: 13.15%</td></tr>
<tr><td rowspan="2">SSD MobileNet V2 FPNLite 640x640 (batch size: 16, data size increased to 800)</td><td>Unorganised: 3.69%</td><td rowspan="2">Overall: 14.18%</td></tr>
<tr><td>Empty: 24.66%</td></tr>
<tr><td rowspan="2">SSD ResNet50 V1 FPN 640x640 (RetinaNet50)</td><td>Unorganised: 6.79%</td><td rowspan="2">Overall: 14.89%</td></tr>
<tr><td>Empty: 22.99%</td></tr>
</table>

Upon comparing the one-class and multi-class object detection approaches, several critical observations emerge:

1. **Model Complexity and Performance:** One-class models outperformed multi-class models in individual class detection accuracy. This can be attributed to the specialized

nature of one-class models, which allows them to focus solely on specific detection tasks without the additional complexity of multi-class management.

2. **Resolution Impact:** Higher resolution inputs (640x640) consistently improved model performance across both one-class and multi-class approaches. This underscores the importance of high-quality image inputs in enhancing detection accuracy, particularly for detailed and cluttered scenes such as unorganized shelves.
3. **Model Architecture:** The SSD MobileNet V2 FPNLite models demonstrated robust performance, particularly at higher resolutions. However, the SSD ResNet50 V1 FPN model showcased superior detection capabilities for unorganized shelves, indicating that more complex architectures like ResNet50 may offer advantages in handling intricate detection tasks.
4. **Practical Considerations:** For real-world retail applications, the choice between one-class and multi-class models should consider the specific operational requirements. One-class models may be preferable for tasks requiring high precision in detecting specific scenarios, while multi-class models offer a more streamlined and integrated solution for environments where simultaneous detection of multiple classes is needed.

In conclusion, our comparative analysis reveals that while one-class object detection models deliver higher accuracy for specialized tasks, multi-class models provide a more unified approach with acceptable performance trade-offs. Future work should explore hybrid models and further optimization techniques to enhance the robustness and versatility of object detection systems in the retail sector. By leveraging the insights gained from this study, retailers can make informed decisions to optimize their inventory and shelf management processes, ultimately enhancing operational efficiency and customer satisfaction.

6. Practical Application Description

The system developed in this project offers valuable practical applications within retail settings:

6.1 Inventory Management

Automated detection of out-of-stock items facilitates efficient inventory management, minimizing instances of missed sales opportunities. By promptly identifying product shortages, the system aids in maintaining optimal inventory levels.

6.2 Shelf Management

The system ensures precise product arrangement based on planograms, contributing to enhanced visual appeal and improved customer experience. Detecting and correcting misplacements or disorganized shelves helps maintain a neat and organized store layout.

6.3 Real-Time Monitoring

Deploying the system on edge devices enables real-time monitoring of shelf conditions and inventory status. This real-time monitoring capability allows for swift responses to inventory issues, ensuring timely restocking and optimizing store operations.

7. Conclusion

This project evaluated the effectiveness of different object detection models for out-of-stock detection and product management in retail, focusing on SSD MobileNet V2 FPNLite (320x320 and 640x640) and SSD ResNet50 V1 FPN. We compared one-class and multi-class detection approaches, finding that SSD MobileNet V2 FPNLite 640x640 outperformed the 320x320 variant due to higher resolution, while SSD ResNet50 V1 FPN excelled in detecting unorganized shelves, demonstrating the benefits of more complex architectures. One-class models showed higher accuracy in specific tasks due to their specialized training, whereas multi-class models provided a unified solution for detecting multiple scenarios, albeit with slightly lower accuracy.

The insights from this study highlight the importance of resolution and model complexity in improving detection precision. One-class models are preferable for high-precision tasks, while multi-class models offer streamlined deployment for detecting multiple scenarios. The developed system enhances inventory management, shelf arrangement, and real-time monitoring, significantly improving operational efficiency and customer satisfaction in retail environments.

Future research should explore hybrid models and further optimization techniques to enhance the robustness and versatility of object detection systems in retail. Expanding datasets and incorporating diverse scenarios will also improve model performance. In conclusion, our comparative analysis offers valuable guidance for optimizing retail operations, enabling retailers to enhance efficiency and deliver a superior customer experience.

8. References

Purva Huilgol 2024, ‘Precision and Recall | Essential Metrics for Machine Learning (2024 Update)’, *Analytics Vidhya*, viewed 10 May 2024, <https://www.analyticsvidhya.com/blog/2020/09/precision-recall-machine-learning/>.

Deval Shah 2022 ‘Mean Average Precision (mAP) Explained: Everything You Need to Know’ 2024, *V7labs.com*, viewed 10 May 2024, <https://www.v7labs.com/blog/mean-average-precision>.

9. Appendices

Github: https://github.com/MuniebA/COS30018-INTELLIGENT-SYSTEM

Video: COS30018 Group 7 Project Presenatation.mp4