Smart Bin Classifier

AI for faster validation, packaging, and inventory management

Motivation

The e-commerce industry is growing unprecedentedly, leading to an increased demand for efficient inventory management and order fulfillment processes. One critical aspect of this process is accurately identifying items within bins or baskets, determining their quantities, estimating packaging sizes, and updating inventory availability in real time. Manual identification of items, their size estimation and feeding this information into the inventory management system are time-consuming; leading to suboptimal packaging and inefficient inventory management. The rapid growth of e-commerce demands innovative solutions for optimizing logistics, reducing packaging waste, and enhancing operational efficiency and customer satisfaction. The automation of item identification, quantity validation, and packaging size estimation while simultaneously updating inventory and availability can significantly improve operational efficiency. This project aims to do this automation by leveraging computer vision technology. Students are expected to develop this product with intrinsic features like simple interactive UI with a very well-integrated computer vision model.

Dataset details

The Amazon Bin Image Dataset contains over 535K images and metadata from bins of a pod in an operating Amazon Fulfillment Center. The bin images in this dataset are captured as robot units carrying pods as part of normal Amazon Fulfillment Center operations.

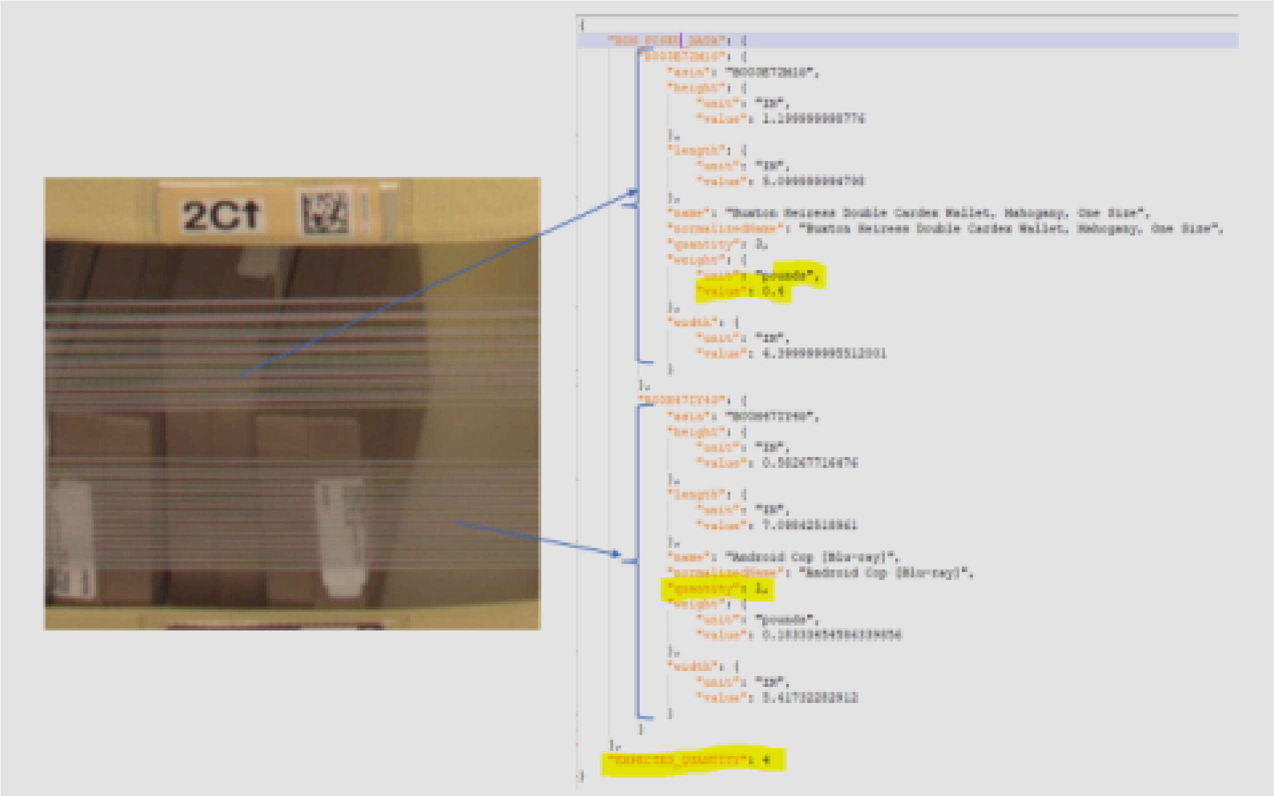
- You can find more details about the dataset here

Images:

Above are some typical images in the dataset S3 bucket ‘aft-vbi-pds’ and ‘/bin-images’ prefix.

A bin contains multiple object categories and various numbers of instances. The size of bins varies depending on the size of the objects in them. The tapes in front of the bins prevent the items from falling out of them; sometimes, this might make the objects unclear. Objects are sometimes heavily occluded by other objects or limited viewpoints of the images.

Metadata:



```
{
    "BIN_FCSKU_DATA": {
        "B00OE72M10": {
            "asin": "B00OE72M10",
            "height": {
                "unit": "IN",
                "value": 1.199999998776
            },
            "length": {
                "unit": "IN",
                "value": 5.099999994798
            },
            "name": "Buxton Heiress Double Cardex Wallet, Mahogany, One Size",
            "normalizedName": "Buxton Heiress Double Cardex Wallet, Mahogany, One Size",
            "quantity": 3,
            "weight": {
                "unit": "pounds",
                "value": 0.4
            },
            "width": {
                "unit": "IN",
                "value": 4.39999999551201
            }
        },
        "B00H472Y40": {
            "asin": "B00H472Y40",
            "height": {
                "unit": "IN",
                "value": 0.582677164876
            },
            "length": {
                "unit": "IN",
                "value": 7.08661418861
            },
            "name": "Android Cop [Blu-ray]",
            "normalizedName": "Android Cop [Blu-ray]",
            "quantity": 1,
            "weight": {
                "unit": "pounds",
                "value": 0.18333454586339856
            },
            "width": {
                "unit": "IN",
                "value": 5.41732282912
            }
        }
    },
    "EXPECTED_QUANTITY": 4
}
```

The corresponding metadata for each bin image includes the item identification (Amazon Standard Identification Number, ASIN), quantity, height, length, width, weights.

Understanding the Image and its metadata is crucial before you develop the solution. Some examples can be found here.

Dataset you should use for this project:

The dataset is a list of images and metadata file names without .jpg and .json extension from the AWS S3 bucket which you should download and use for this project. The link to download the dataset is attached with the Google Classroom post.

Objectives

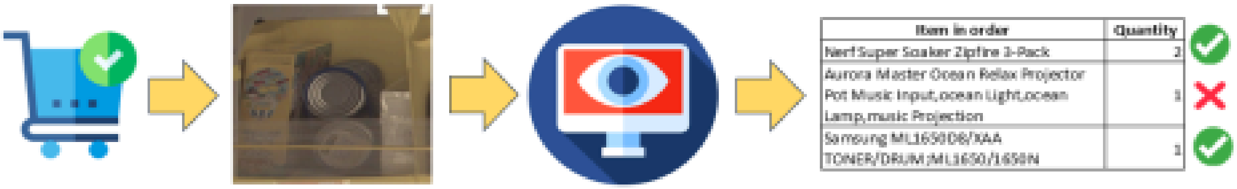
Core objectives of this project:

- Use the dataset (images + metadata) and develop a highly accurate and fast computer vision model to verify if the items with their respective quantities are present in the image of the bin.

  For example, You get an order for 3 items and their quantities as mentioned below.

Item in order	**Quantity**
Nerf Super Soaker Zipfire 3-Pack	2
Aurora Master Ocean Relax Projector Pot Music Input,ocean Light,ocean Lamp,music Projection	1
Samsung ML1650D8/XAA TONER/DRUM;ML1650/1650N	1

  Now, you get an image of the bin where items in the order are present. Your objective is to validate whether the items in the order are the ones in the bin.



Item in order	Quantity	
Nerf Super Soaker Zipfire 3-Pack	2	✓
Aurora Master Ocean Relax Projector Pot Music Input,ocean Light,ocean Lamp,music Projection	1	✗
Samsung ML1650D8/XAA TONER/DRUM;ML1650/1650N	1	✓

- Demonstrate creativity and innovative mindset while working on EDA, data preparation, model selection, application and evaluation, etc.

- Develop a simple UI using tools/frameworks of your choice for:

○ Allowing the user to select an item and its quantity he/she wants to order. ○ Selecting the most appropriate image from the dataset which corresponds to order and displaying in the UI.

○ Using the CV model for inferencing and displaying the results of the model to validate if items and its respective quantity in order exists in the bin image.

● Create a comprehensive presentation and documentation that explains the understanding of objectives, methodology, architecture, decisions, milestones, model selection, hyperparameter tuning and evaluation performed, UI functionality and considerations for MLOps (model deployment, model monitoring and re-training).

***Choose cost-effective options to meet the above objectives**

Submission Guidelines

Working prototype:

● Code for a fully functional and working prototype of a computer vision based system which performs item and quantity validation with high accuracy.

● The working prototype could be a web interface (HTML/react/streamlit/gradio/etc.), a mobile app, or a combination of both.

● A screen recording video which explains how the model is deployed, monitored and re-trained.

● Multi-user support (**Optional**): Allow multiple users to login and use the product at the same time.

Model, EDA and MLOps artifacts:

● Saved model file for keras/tf models and state dictionaries for pytorch models. ● Step by step code and visualization which explains the EDA done and decisions taken before data preparation for modeling.

● Detailed step by step code and comments which explain selection of CV models, hyper parameter tuning done for various models, evaluation performed, etc.

● Code with explanation for each step in MLOps - model deployment, monitoring, logging and re-training.

● Annotated data **(Optional)**: Annotated image data if annotation is done for applying object identification and localization models like YOLO, Faster-RCNN, etc.

Documentation:

- A comprehensive presentation and documentation that explains the understanding of objectives, methodology, architecture, decisions, milestones, model selection, hyperparameter tuning and evaluation performed, UI functionality and considerations for MLOps (model deployment, model monitoring and re-training).
- Stretch goal documentation **(Optional)**: User manuals, FAQ and guides for explaining different functionalities of UI, MLOps, etc.

Evaluation criteria

- **Understanding of objectives (Weightage: 20%)**
  Demonstrate a clear and thorough understanding of the objectives and emphasizes on building an highly accurate, efficient and productionized CV based system for detecting if the item with its respective quantity is present in the bin image.
- **Selection and application of CV models (Weightage: 20%)**
  Perform EDA and research on the CV models which are most suitable for this problem. Demonstrate a well-designed experimental plan to select and apply various CV models on the data.

- **Evaluation of CV models (Weightage: 20%)**
  Select and use the classification/regression metrics judiciously. Justify the metric choices and ensure that they are insightful and align with the objectives and characteristics of the dataset and expected results.

- **Innovation and creativity (Weightage: 8%)**
  Demonstrate highly innovative approaches and showcase creativity in integrating existing technologies and proposing an innovative solution, breaking away from traditional methods. Suggest thoughtful and highly feasible enhancements, considering the project's context.

- **Presentation and Documentation (Weightage: 7%)**
  Clear and well-organized presentation of the project, including detailed documentation, code comments, and visual aids (charts, graphs) for understanding results. Comments and documentation should be comprehensive and crystal clear, facilitating easy understanding for the reader/user.

- **MLOps (Weightage: 15%)**
  Develop an efficient and automated deployment process with minimal downtime. The model monitoring should be comprehensive and should track the performance. Logging

should be done for predictions, errors, and system behavior.

- **UI/UX (Weightage: 10%)**
  Develop an intuitive, fast and easy to use interface which takes the inputs from the user and responds back with analysis/results and exceptions/errors which are easy for the user to understand.

Cloud and GPU usage:

For downloading the dataset and training CNN models, you need:

1. AWS account credentials to download dataset from S3 bucket,
2. at least 2 T4 GPU instances,
3. at least 30 GB RAM,
4. EC2 M4 instance for deployment

Stretch objectives of the project (Optional):

- **Inventory management:** Use the dataset (images + metadata) and create a database to store item dimensions - height, length, width, weight, etc. and build functionality for inventory management.

  Dataset for inventory availability, you can build your synthetic dataset using the items that are available in images in the dataset. For example, for the items available, you can assign initial availability as some random number between 20 to 100.

- **UI:** Develop an user-friendly, highly responsive, intuitive UI which is well integrated with inventory availability database and camera (optional) for:
  - o Capturing the information from images taken of the bin in real-time by integrated camera and storing them in a database.
  - o Displaying real-time inventory availability dashboard.

- **MLOps:** Integrate MLOps frameworks to automate the monitoring of the model in real time, and deployment of model at regular intervals when data drifts are detected.

- **Additional considerations:** Create a further more comprehensive documentation that explains the considerations for building a database, database schema, functionalities in UI, user manuals for UI, MLOps strategy, etc..

Future considerations:

● When you train the models, they will only be trained on certain items. In the real world, a new item may get added, or the packaging could change for an existing item. ● If the camera is integrated in the system, then calibration is needed for the camera to know the distance from the bin, which would then help calculate the items' size in real-time. This must be done when this system is deployed in fulfillment centers, etc.

Similar business cases in other domains:

Your objective here is to validate whether the items in the order are the ones in the bin. This is an important process at the fulfillment centers of ecommerce and retail businesses. The underlying technique can also be used to formulate solutions for some other business cases mentioned below in other domains.

- **E-commerce**: Counterfeit and tamper detection of listings on ecommerce sites. Recommending similar products to customers.
- **Medical/Healthcare**: Validate if the medicines prescribed for patients are the ones that are being actually consumed.
- **Manufacturing**: Product inspection to identify quality issues in products during manufacturing.
- **Surveillance and monitoring**: Tracking changes in conditions of objects under surveillance.