YOLOv8-based Continuous Learning Furniture Detection System for Retail Recommendation

1. SUMMARY

The furniture retail sector faces the challenge of personalizing recommendations based on the customer's real context. Current systems lack the ability to automatically interpret the user's environment to suggest complementary products. Therefore, an architecture based on Deep Learning is proposed that uses YOLOv8 for the detection of objects (Sofas, Carpets, Cushions) and implements a continuous learning cycle (Active Learning) through user feedback. A public dataset present in roboflow was used and performance was evaluated using Mean Average Accuracy (mAP50 and mAP50-95) using MLflow to track metrics. The possible integration of a visual similarity engine is contemplated to recommend specific products from the catalog based on the detected cutouts.

2. PROPOSED METHOD

The implemented solution consists of a REST API developed in FastAPI that orchestrates the model lifecycle. The retraining request is received from the web application, and the application begins the relearning cycle, where the model will receive the user's corrections and will use a set of images (50) of the base dataset to perform its retraining, once the process is finished, the web page shows the new version of the model.

Figure 1 illustrates the flow of data from the system. The user interacts with the REST API to get predictions. When an error is detected, the user sends the correction, which is stored in the *Feedback Dataset*. The retraining process (retrain_service) combines a subset of the original dataset with the new feedback data, trains a new version of the YOLOv8 model, registers it in MLflow, and dynamically updates the model in production without stopping the service.

Figure 1. Flowchart

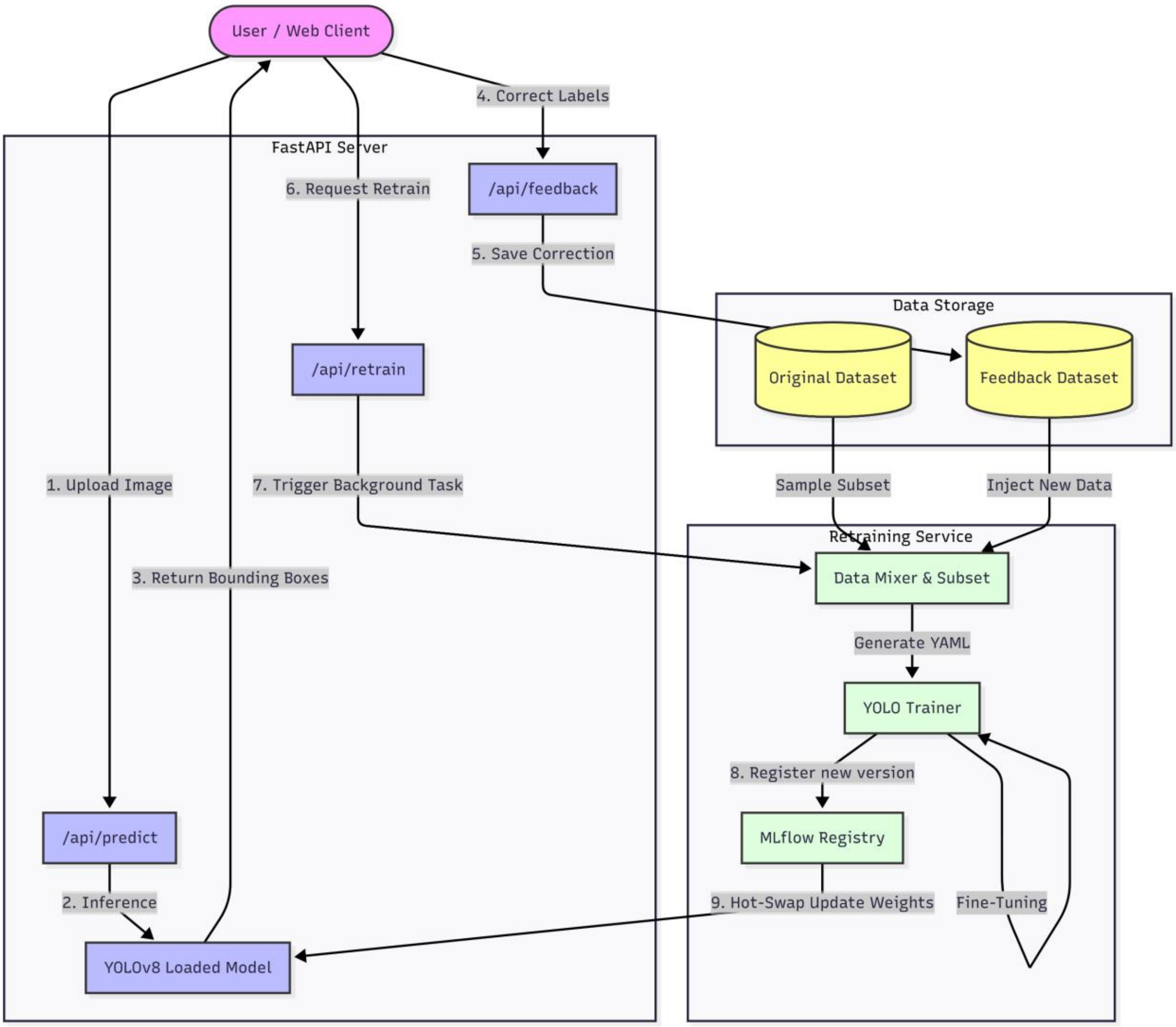


Table 1. Parameters of the proposed method

Name	Description
Class	Class ID detected
Box coordinates	Position of the box in the image

3. EXPERIMENT DESIGN

This section details the characteristics of the data and the configuration of the experiments performed to validate the adaptability of the system.

1) Dataset characteristics

The " Living Room Computer Vision Model " dataset hosted in the Roboflow Universe was used. This dataset presents significant challenges such as occlusions (tables covering sofas) and light variability. Because only 3 classes will be taken into account, the analysis is performed only for those classes.

Table 2. Dataset Distribution

Class	Description	Approx.
Sofa	Main element, great variability in shape and color.	~3000
Rug	Carpets, usually occluded by other elements.	~3000
Pillows	Cushions, small objects that are present multiple times in a single image.	~9000

2) Method optimization parameters

Two experimental scenarios were designed to demonstrate the evolution of the system:

- **Scenario A:** Simulated training with scarce data (50 images) and few periods, which provided an average base that allows visualizing learning in the future use of the application.
- **Scenario B (Production Retrain):** Re-training using the dataset, the same base (50 images) was used in addition to simulated correction data (taken from validation), applying augmentation techniques such as *Mosaic* to improve robustness.

Table 3. Training Hyperparameters

Hyperparameter	Base Model	Retrained model
Epochs	5	2
Batch size	8	16
Image Size	640	640

Hyperparameter	Base Model	Retrained model
Learning Rate	-	1e-4
Data Augmentation	-	Mosaic 0.5

4. RESULTS AND DISCUSSION

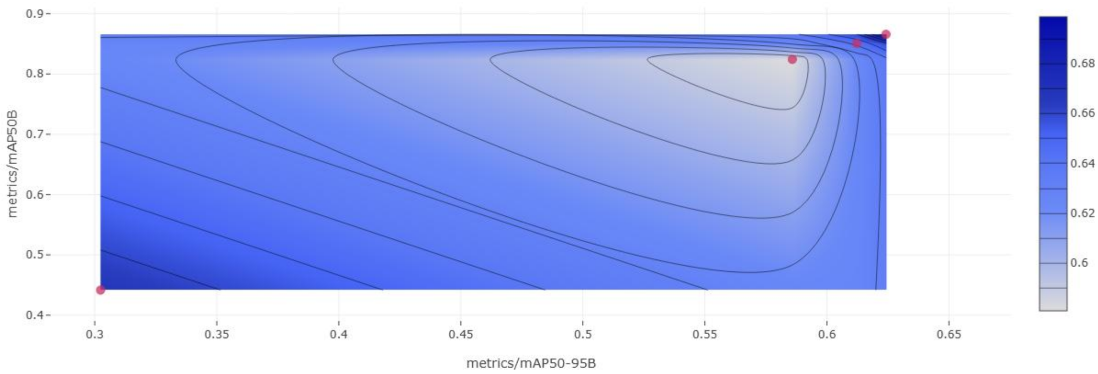
After the execution of the retraining cycle, a significant improvement in the generalization capacity of the model was observed.

The Base Model presented difficulties in distinguishing objects, with an mAP of 44% it failed to detect objects in the selected images.

After the retraining phase, the Refined Model (v2) showed rapid convergence thanks to Transfer Learning, doubling its mAP value, due to this, the training parameters were edited to achieve a smoother and more horizontal learning curve.

In the following graph you can see how the mAP50 and map50-95 increase with each version, the Z variable was defined as the recall, and you can see how it points to version 4 of the model.

Figure 1. Evolution of mAP50-95



To test the model's learning, it was tested against a new image using version 1 and then version 4.

Image 1. Version 1 Prediction

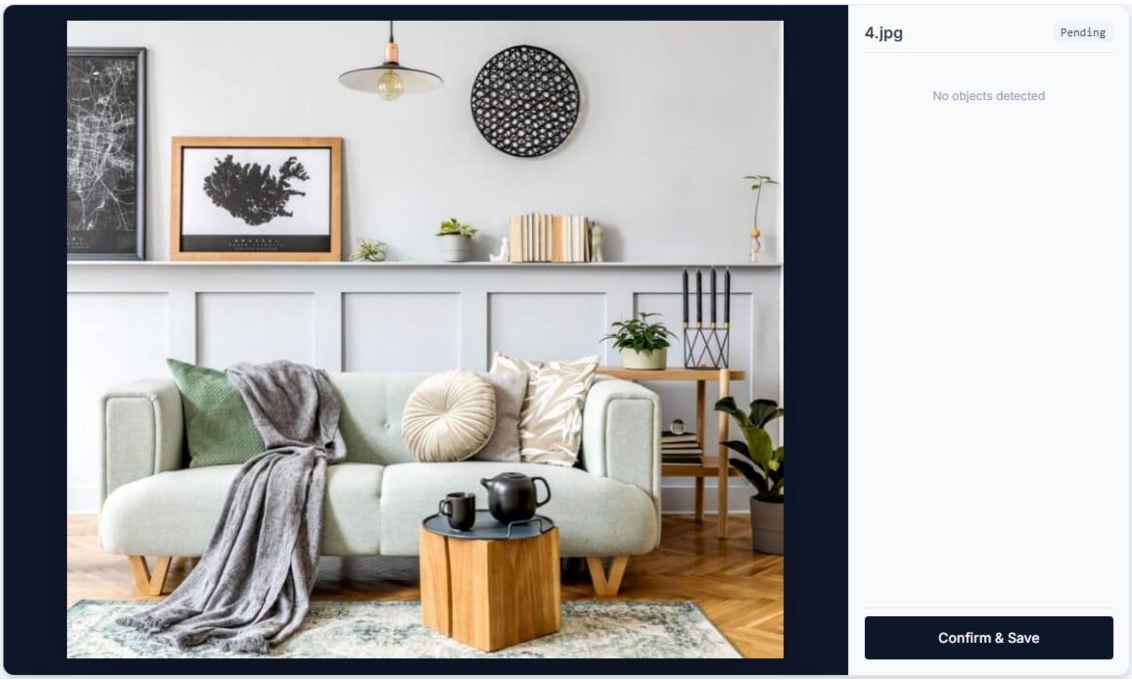
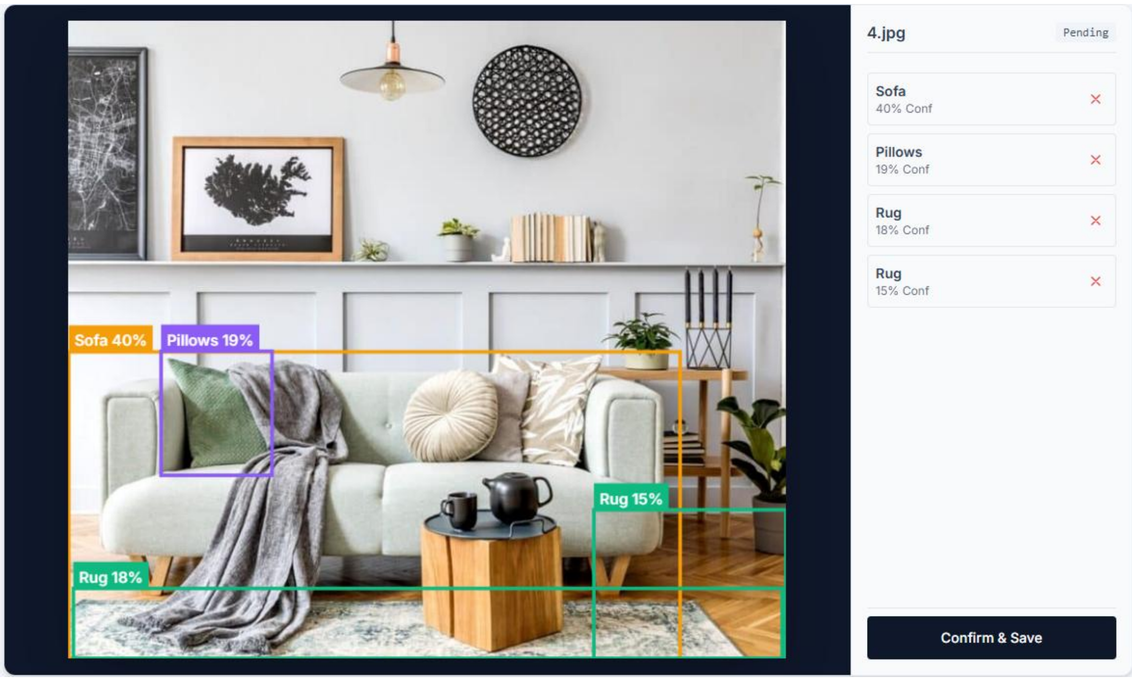


Image 2. Version 4 prediction.



As we can see, the model has learned to distinguish objects correctly and adapts perfectly to new examples.

The results indicate that the strategy of adjusting the hyper parameters gave an excellent improvement in terms of learning, parameters such as the low learning rate or the Mosaic helped the model to be more robust and not suffer from overfitting.

5. CONCLUSIONS

Implementing an MLOps pipeline integrated with an end-user application can dramatically reduce the gap between model development and its actual utility. It was shown that YOLOv8 is able to adapt to new domains (user-specific images) with a reduced dataset if the appropriate hyperparameter techniques are applied. As future work, it is recommended to implement a comparison system to detect objects similar to those detected.

BIBLIOGRAPHIC REFERENCES

1. Roboflow Universe. (2024). Living Room Object Detection Dataset. Retrieved from https://universe.roboflow.com/living-room/living-room-hn7cw

2. Zaharia, M.A., Chen, A., Davidson, A., Ghodsi, A., Hong, S.A., Konwinski, A., Murching, S., Nykodym, T., Ogilvie, P., Parkhe, M., Xie, F., & Zumar, C. (2018). Accelerating the Machine Learning Lifecycle with MLflow. *IEEE Data Eng. Bull., 41*, 39-45.