YRIKKA 1-A
Bridging the Synthetic to Real Data Gap

AI Studio Final Project

December 6, 2025

Meet Our Team

Katherin Cuautli
Hunter College

Taanyaa Haridass Prasad
San Jose State University

Idranne Naike
Mount Holyoke College

Mildred Nwachukwu-Innocent
Elizabethtown College

Dedeepya Pidaparthi
UMASS Amherst

Md. Mashiur Rahman Khan
Vanderbilt University

Maha Shanawaz Syeda
UT Dallas

Our AI Studio Coach & Challenge Advisors

Kia Khezeli
YRRIKA
Co-Founder & CEO

John Kalantari
YRRIKA
Co-Founder & CTO

Maxim Clouser
YRRIKA
Founding ML Engineer

Swagath Babu
ML Engineer @ Humana

Presentation Agenda

AI Studio Project Overview	5 minutes
Data Understanding and Data Preparation	3 minutes
Modeling and Evaluation	7 minutes
Final Thoughts	5 minutes
Questions	5 minutes

AI Studio Project Overview

Project Description

Project focus: Bridging the synthetic-to-real (syn2real) data gap in object detection.

Context: Models trained on synthetic data often fail on real-world images. Our goal is to evaluate whether augmenting YOLO with high-quality synthetic data can improve performance.

What we're working on:

- Detecting five everyday objects: **potted plant, chair, cup, vase, book**
- Assessing YOLO's performance on synthetic data with 10% noisy or missing annotations
- Correcting labels using CVAT
- Collecting 200 challenging real-world test images
- Generating new synthetic data with YRIKKA's engine to refine the model

Our Goals & Motivations

Technical goal: Improve YOLO's mAP@50 by +0.10 over the baseline model.

Why this matters:

- Demonstrate whether synthetic data can meaningfully enhance real-world performance.
- Understand how annotation quality and targeted synthetic data generation impact detection accuracy.
- Build a robust pipeline for cleaning labels, fine-tuning models, and validating them on hard, real-world examples.

Business Impact

Why this work matters beyond this project:

1. Strengthens perception which is the most failure-prone layer in autonomous systems (drones, robots, vehicles).
2. Reduces the need for extensive, costly real-world data collection for rare or dangerous cases.
3. Shows how synthetic data can fill critical gaps in edge-case coverage.
4. Builds hands-on skills for creating deployable, real-world computer vision models using modern tools (YOLO, CVAT, synthetic generation engines).

Our Approach

September

Gaining a clear understanding of our project requirements and identifying the technologies needed for successful completion. We began by working with one dataset to get familiar with the workflow and learn how to use CVAT effectively.

October

Cleaned both datasets and merged them into a unified training set. Ran a pretrained YOLO model and iterated through multiple rounds of annotation corrections in CVAT. Trained our final YOLOv11 model after refining labels.

November

Collected 200 challenging real-world test images, annotated them using CVAT, and evaluated our trained model on this new dataset.

Resources We Leveraged

Data Understanding and Preparation

Data Set Overview

Two datasets: 496 images and 497 images

Each contains 10 object classes (e.g., pot, cup, potted plant, etc.)

Images include rich contextual metadata, such as:

- Scene (indoor, outdoor, kitchen, library)
- Lighting (bright, backlit, spotlight)
- Blur (motion blur, background blur)
- Occlusion (partially hidden or covered objects)
- Object classes detected per image

Data Set Cleaning

Data Preparation Process

- Mapped original classes to our 5 target classes
- Verified all annotations matched existing images
- Fixed and clipped invalid bounding boxes
- Merged both datasets (adjusted IDs to avoid duplicates)

Final Dataset

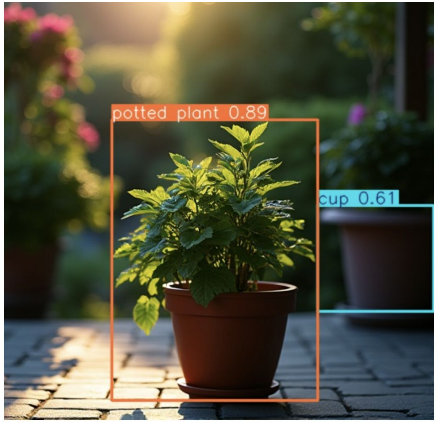
- 993 images
- 3,376 annotations

Modeling and Evaluation

Pre-trained YOLO model

Ran a pre trained YOLO11n model on our dataset: 304 annotations in 243 images were misclassified (out of total 3376 annotations) which is approximately 9% of the annotations

Most errors due to **missing annotations** with some **incorrect class labels.**



Wrong class

Missing annotation

Annotation Correction - CVAT

- Process in CVAT:
    - Uploaded the 243 misclassified images and their annotations
    - Reviewed and corrected labels and bounding boxes for each image
- Result:
    - Exported corrected annotations in COCO format
    - Replaced old misclassified annotations with updated CVAT exports
- Second baseline test:
    - After replacement → 91 annotations in 82 images were misclassified (out of 4599 total annotations) which is approximately 2% of the annotations

CVAT Annotations ~ Potted Plant ***(Ambiguous Case)***

POTTED PLANT

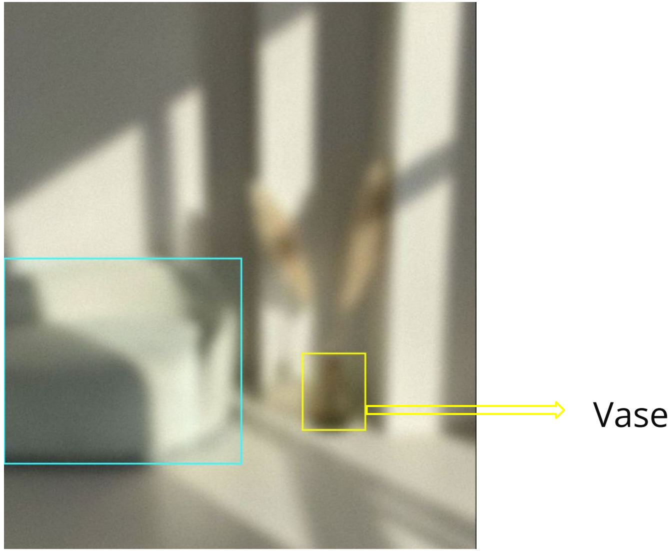
1. Bounding box around the entire potted plant -- labeled as ‘potted plant’
2. Bounding box around its vase only -- labeled as ‘vase’
3. Labeled indoor plants as ‘potted plant’ even if we cannot see its vase

Potted Plant

CVAT Annotations ~ Vase *(Ambiguous Case)*

VASE

1. If the plant in the vase did not look like a “full” plant, add bounding box around its vase only -- labeled as ‘vase’
2. If there is no plant inside the vase then draw bounding box around vase -- labeled as ‘vase’

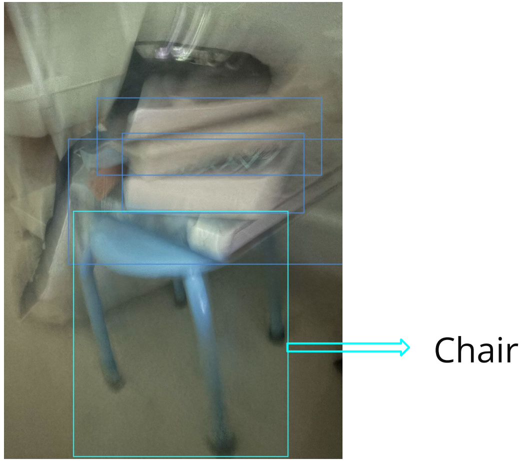


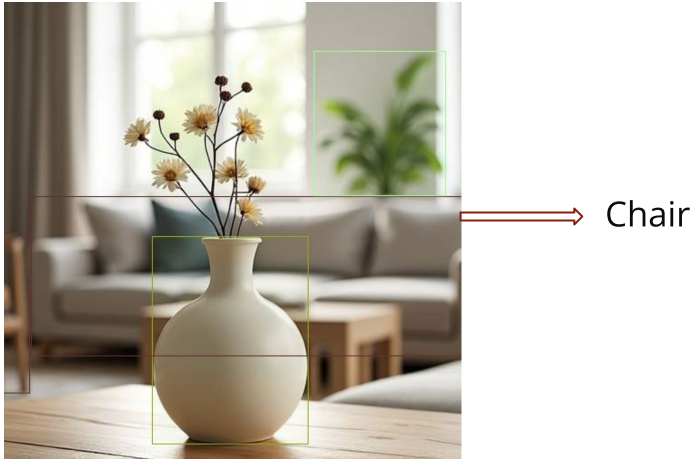
Vase

CVAT Annotations ~ Chair *(Ambiguous Case)*

CHAIR

1. Couches, stools, sofas all had bounding boxes drawn around them -- labeled as ‘chair’



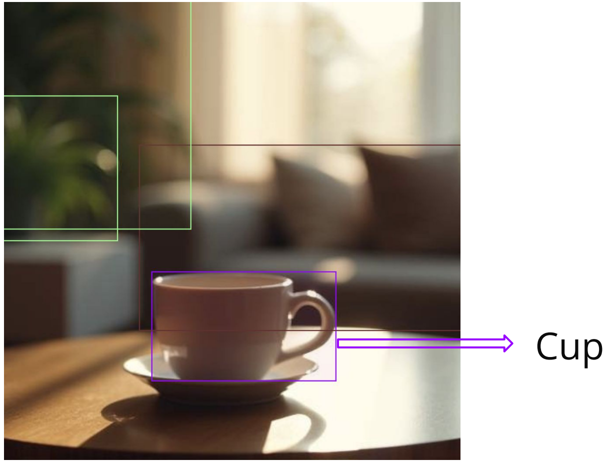


Chair

CVAT Annotations ~ Cup

CUP

1. Labeled object if it existed in the image -- labeled as ‘cup’



Cup

CVAT Annotations ~ Book

BOOK

1. Labeled object if it existed in the image -- labeled as ‘book’

Book

Book

Data Fixes and Conversions

Data issues fixed (converting coco to yolo):

- COCO JSON missing image width & height - added these fields (required by YOLO)
- Reindexed category ids from 1-5 to 0-4

- Conversion:
  - Converted COCO → YOLO format
  - Created yolo_dataset folder containing train & validation splits (80-20 split) for images and labels
  - Built dataset.yaml
    - Defines train/val paths
    - Lists class names

YOLO v. 11 Training

Model Performance on Validation Set (199 images, 638 objects)

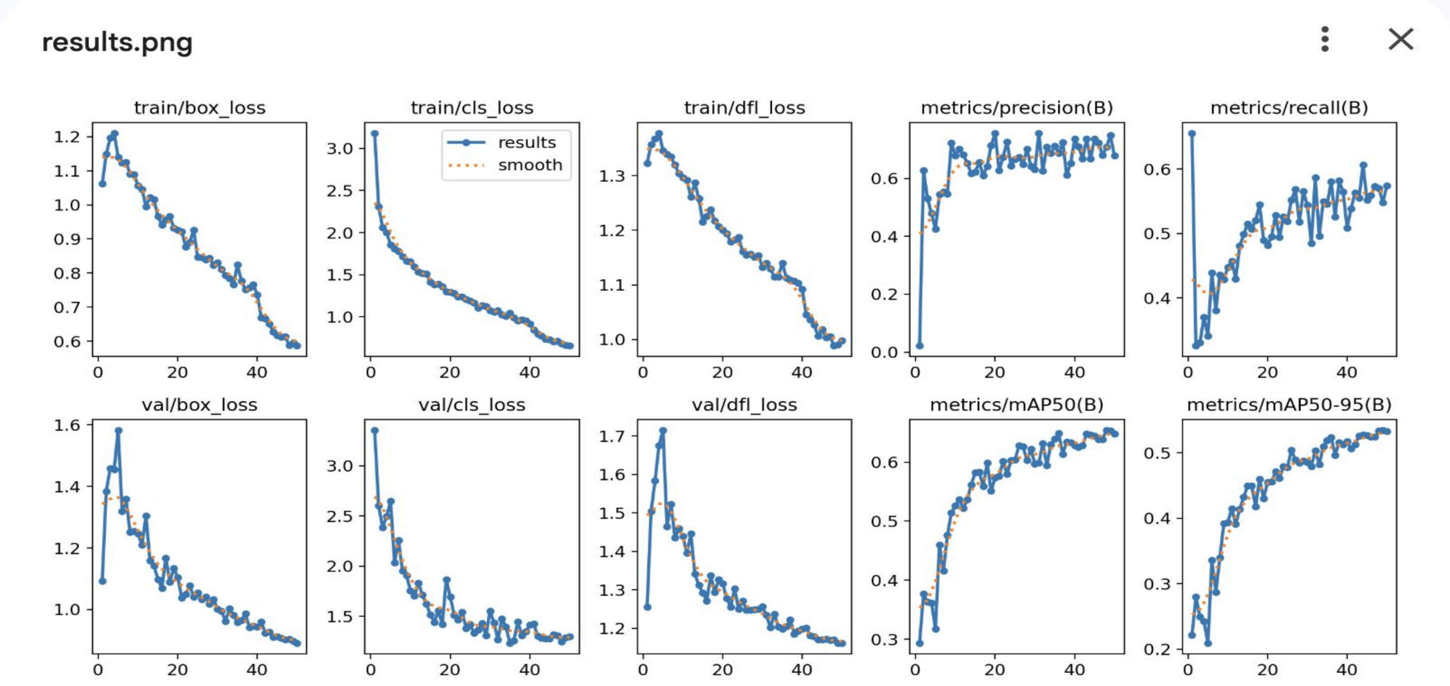
Overall Metrics:

- Precision: 75.3%
- Recall: 55.9%
- mAP@50: 64.9%
- mAP@50-95: 57.5%

YOLO v. 11 Training: Per-Class Performance

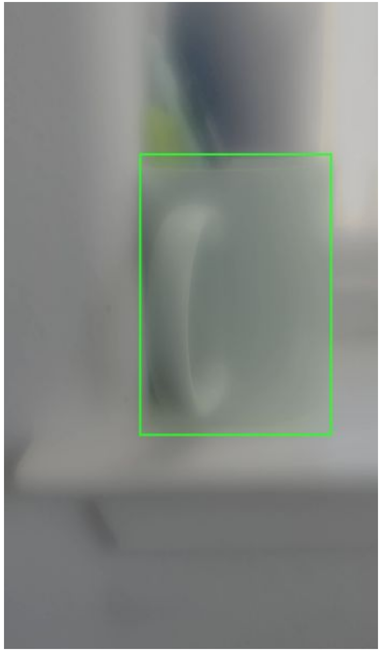
Class	Precision	Recall	mAP@50	mAP@50-95
Cup	86.4%	74.5%	81.7%	79.1%
Vase	80.4%	59.7%	65.6%	61.8%
Potted Plant	77.7%	53.3%	66.2%	52.0%
Chair	68.8%	42.9%	55.8%	46.1%
Book	63.4%	49.1%	55.1%	48.4%

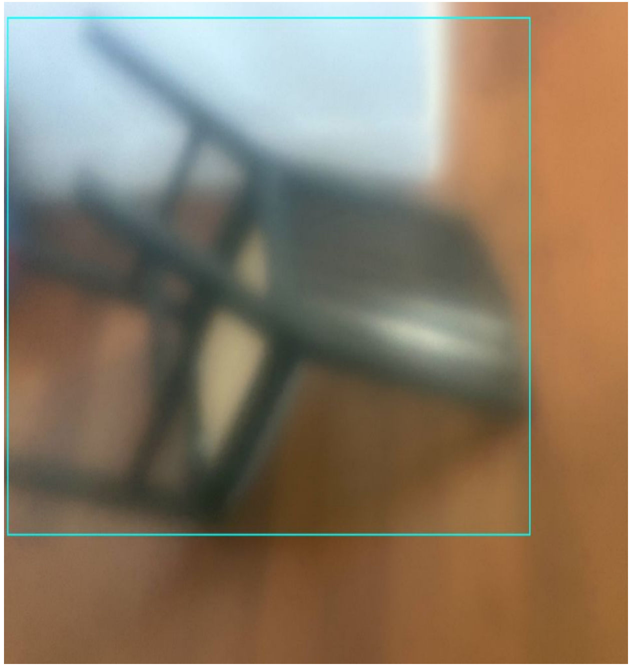
Results of training

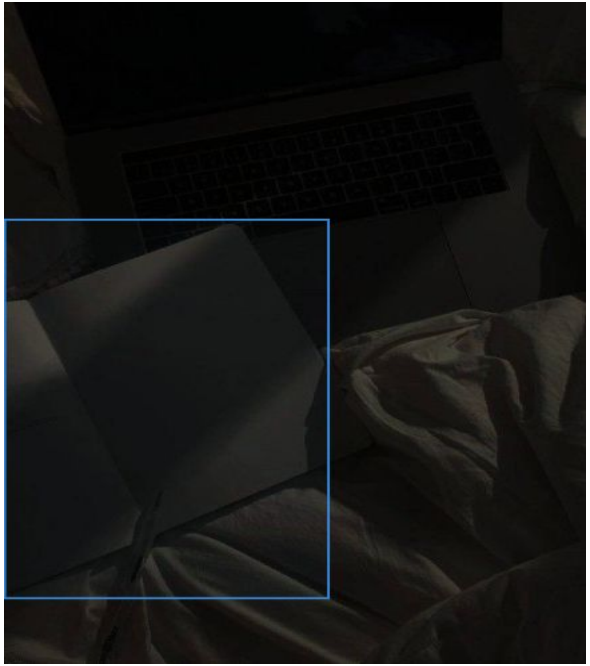


Collected test images

Collected 200 challenging real world images







Model Performances on Test set

Pre-trained YOLO model

Overall Metrics:

- Precision: 51.6%
- Recall: 35.1%
- mAP@50: 40.9%
- mAP@50-95: 25.3%

Fine-tuned model

Overall Metrics:

- Precision: 49.4%
- Recall: 33.7%
- mAP@50: 34.4%
- mAP@50-95: 19.6%

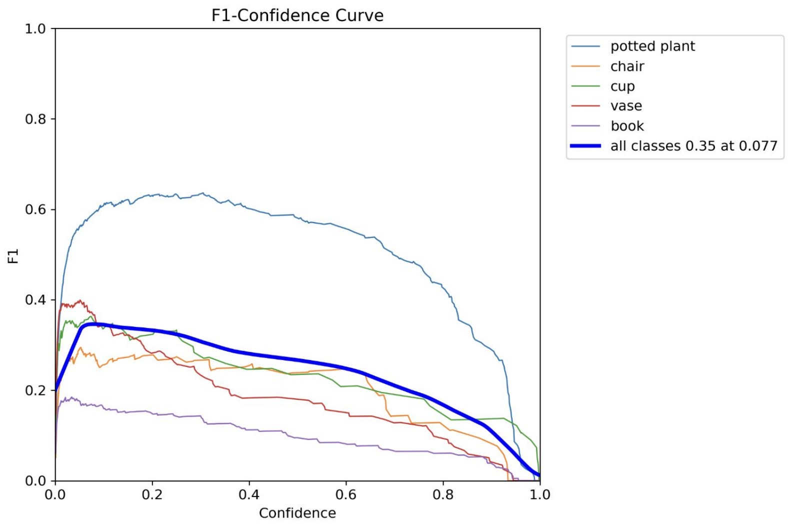
Per-Class Performance Comparison

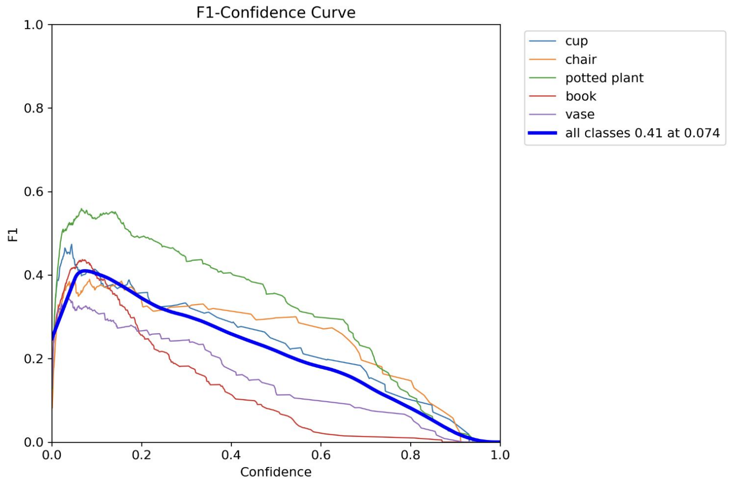
Pre-trained model

Class	Precision	Recall	mAP@50	mAP@50-95
Cup	52.1%	32.7%	43.3%	28.2%
Vase	45.6%	25.1%	29.2%	17.8%
Potted Plant	63%	48.5%	57.3%	38.4%
Chair	45.8%	31.7%	37.2%	24.4%
Book	51.7%	37.7%	37.4%	17.5%

Fine-tuned model

Class	Precision	Recall	mAP@50	mAP@50-95
Cup	62.9%	20.6%	28.3%	17.8%
Vase	57.8%	38.1%	40.2%	22%
Potted Plant	64%	62.7%	63.4%	36.8%
Chair	31.6%	32.7%	26.9%	14%
Book	30.6%	14.2%	13.3%	7.2%





Fine tuned model

Baseline model

Class Distribution Analysis

Class	Train Set	Validation set	Test set
Cup	6.29%	6.08%	10.8%
Vase	15.61%	10.8%	21.77%
Potted Plant	25.65%	19.51%	20.83%
Chair	18.79%	15.88%	37.79%
Book	33.65%	47.73%	9.52%

Potential reasons for decreased performance of Fine tuned model

Class imbalance

- Frequent classes dominated; rare classes under-learned.

Small dataset size

- 993 images still small for YOLO; limited learning capacity.

Overfitting to training data

- Model memorized training images instead of generalizing.

Training–test distribution mismatch

- Differences in lighting, angles, object sizes, or backgrounds.

Label quality issues

- Missing boxes, wrong class IDs, or annotation inconsistencies.

Suboptimal fine-tuning hyperparameters

- Learning rate, epochs, augmentations not well-matched.

Final Thoughts

What We Learned

- Data quality (incorrect or missing annotations) directly affects model performance
- Train, fine-tune, and evaluate YOLO models
- Manage a full ML workflow (business and data understanding → data preparation → model training → evaluation) to build an object detection pipeline

Potential Next Steps

- Correct the class imbalance found in our data
- Fine tune the model further by utilizing the synthetic images we got from YRIKKA data engine

Thank you!

Questions?