Science Fair

By: Augustus Rollin & Marrio Jeyakumar

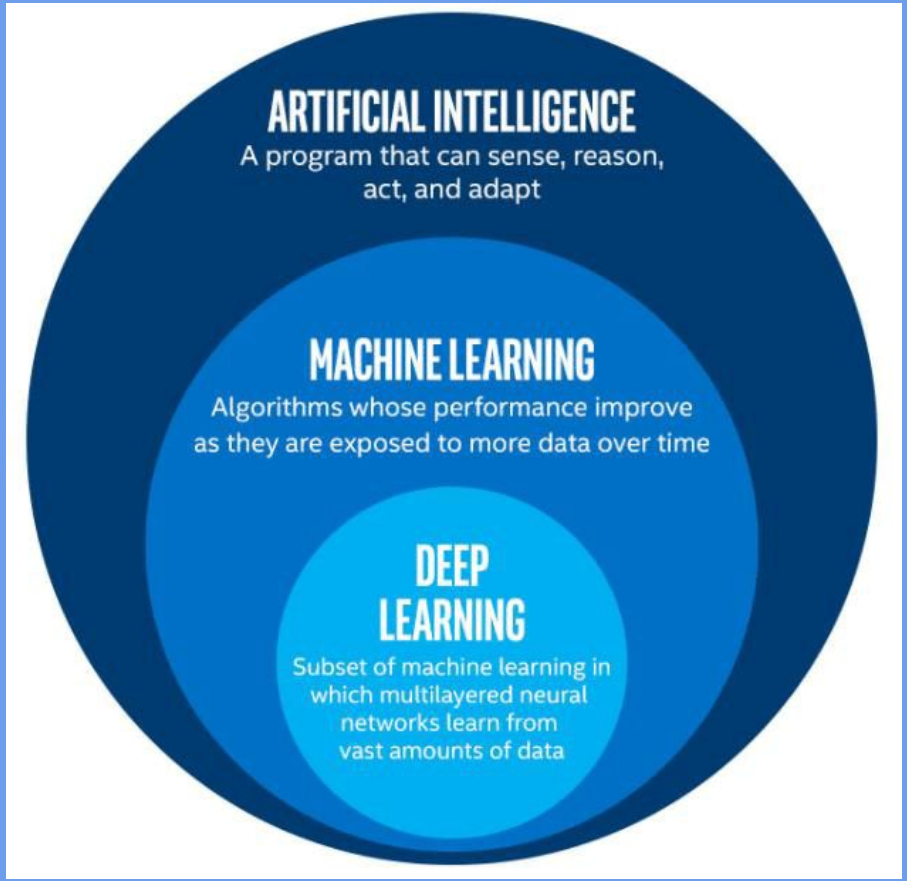
Background

Ai (Artificial Intelligence)

A computer that has the capability of what a human can do, and through machines it can often simulate human intelligence.

Machine Learning

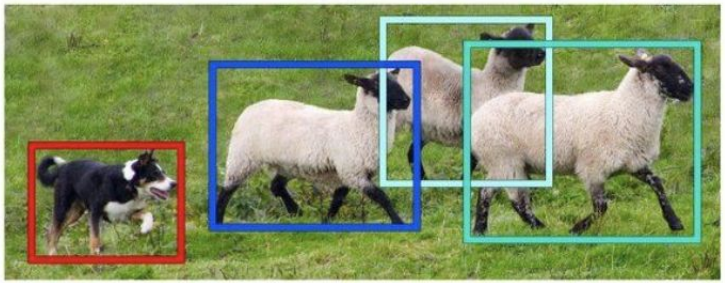
The improvement of performance over time without explicitly giving instruction to do so.



Background

Computer Vision Tasks

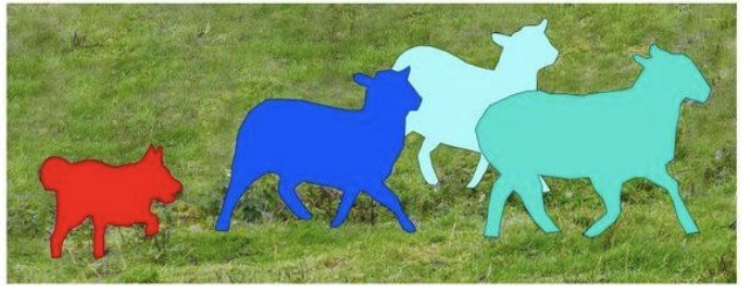
- Classification - The way in which an Ai determines what is in the image.
- Object Detection - Locating objects in the image, typically using a bounty box
- Segmentation - Highlights exactly what is detected by the Ai.



Object Detection

Detection Types

- Face Detection
- Person Detection
- Clothing Detection



Instance Segmentation

Determination of Membership via Uniform Compliance

Goal Statement

Identify strangers (non-members) in a population at a specific location (campus) without the sacrifice of personal information

- CRITICAL: Facial recognition jeopardizes member privacy and therefore we have designed a solution to **not use** facial recognition technology.
- We perform authorization via uniform detection using a passive camera based system.

Details

- System must run in real-time. Latency of operation is critical.
- Member quantity can be large, i.e. > 100 members in a single camera view.
- Cameras are constantly scanning the environment.

Use Cases

- Alert on presence of non-member
- Alert on uniform non-compliance
- Blur faces of members in video

Models/Data Sets

Object Models

- Histogram of Oriented Gradients (HOG)
- Fast R-CNN, Faster R-CNN
- Region-based Convolutional Neural Networks (R-CNN)
- Single Shot Detector (SSD)
- RetinaNet
- Spatial Pyramid Pooling (SPP-net)
- Detectron 2
- YOLOv5/7

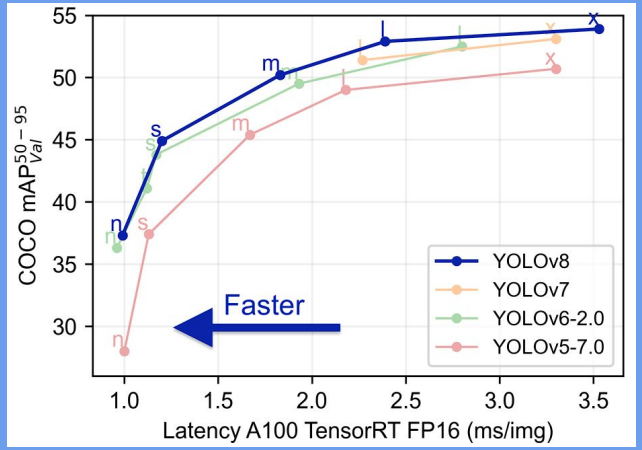
Data Sets

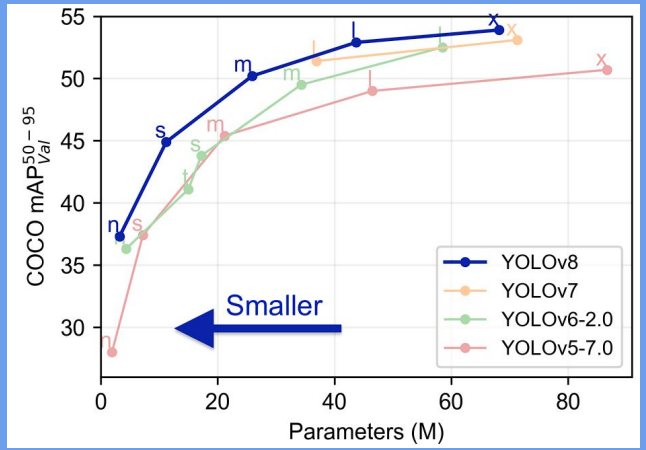
- COCO (Common objects in context)
  - Can use object segmentation
  - 80 object categories
- DeepFashion-MultiModal
- Deep Fashion
- Deep Fashion 2

Capability of YOLOv8 Object Detection Model

YOLOv8 (You Only Look Once)

- SOTA Object Detection Model (January 10 2023)
- Leading Competition in speed, accuracy, and generalization
  - Faster than any other model on the market with 91 fps for v7, expecting more for v8.
  - More accurate than YOLOv5 and other competitors with very minimal background errors.
    - Capability to adapt to new, unknown data based on the same distribution used to train/create the model.
- Open-source, easily accessible and modifiable for developers
- Multitude of applications such as Healthcare, Agriculture, Self-Driving cars, and Surveillance.

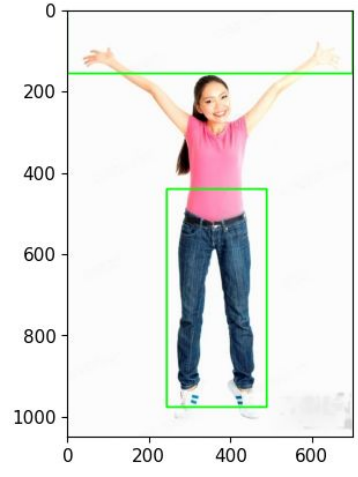


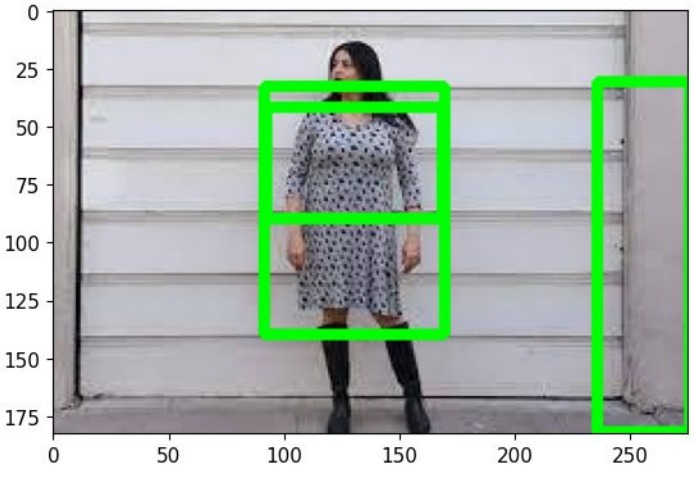


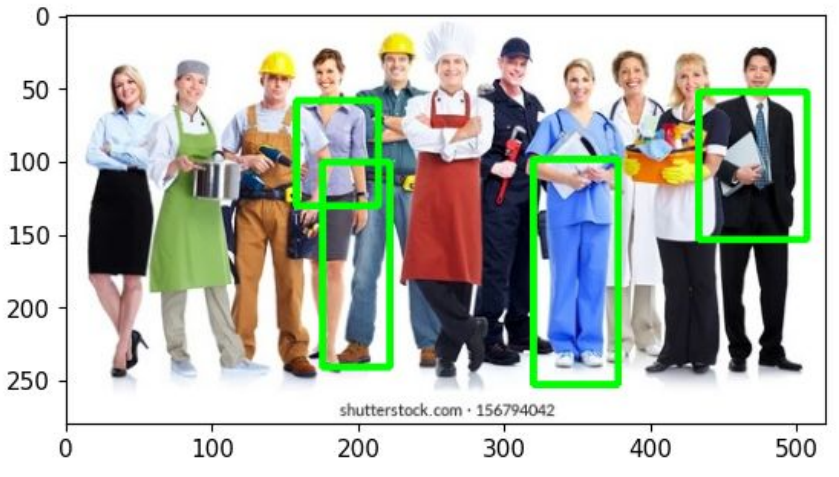
Great Image Examples

Mediocre Image Example

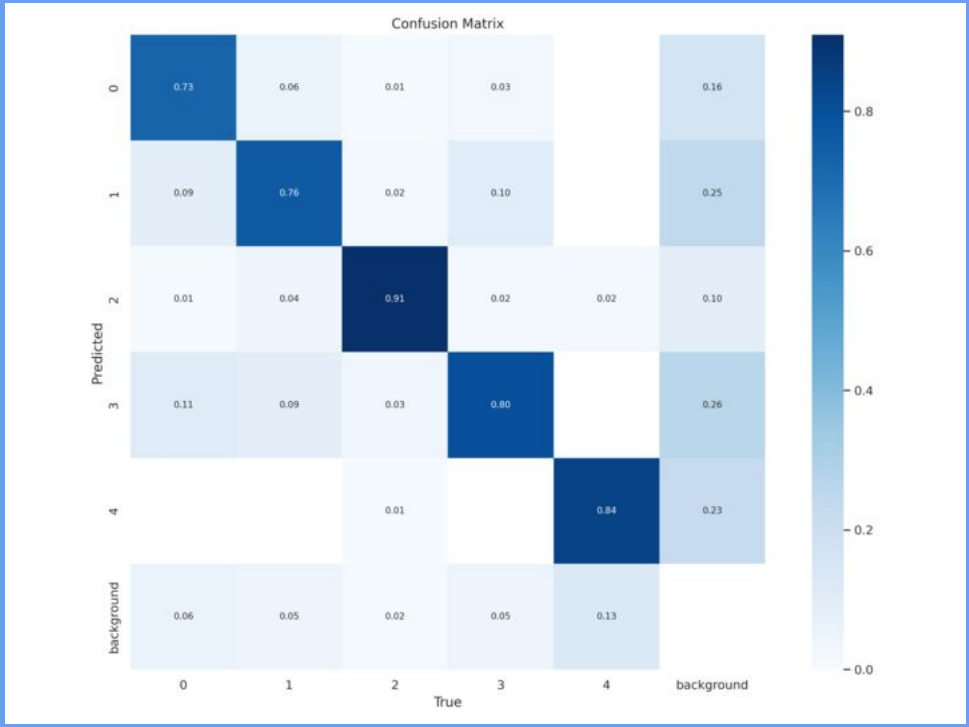
Bad Image Example

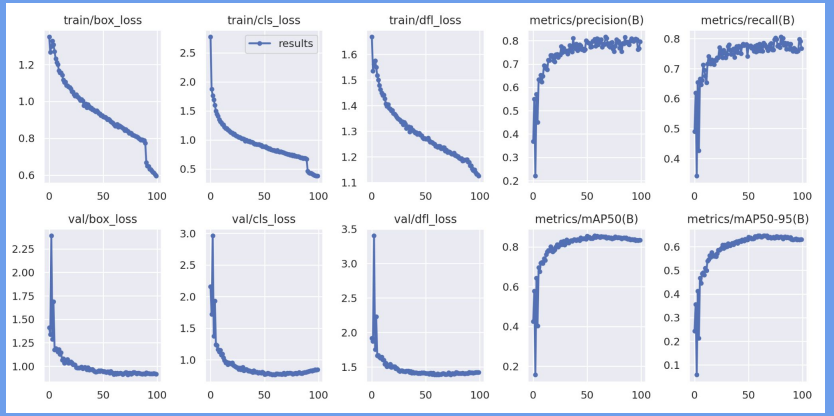


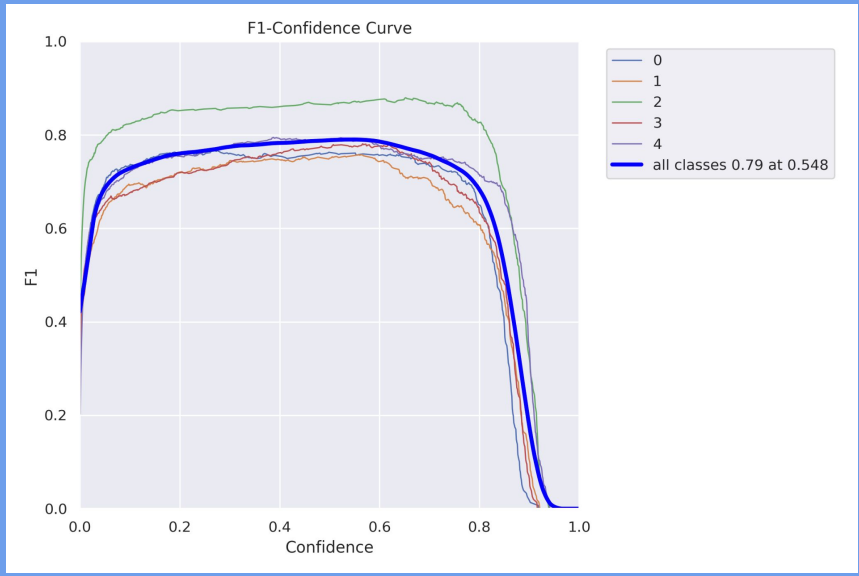




```
(fused): 168 layers, 3006623 parameters, 0 gradients, 8.1 GFLOPs
  Class     Images  Instances      Box(P          R      mAP50  mAP50-95):
  [00:10<00:00,  4.65it/s]
    all       1500       1500      0.815       0.77      0.851      0.646
      0       1500        300      0.862       0.68      0.835      0.595
      1       1500        300      0.786      0.723      0.815      0.585
      2       1500        300       0.85      0.888      0.929      0.761
      3       1500        300      0.771      0.784       0.83      0.618
      4       1500        300      0.804      0.773      0.848      0.674
```







Deep Fashion Model

Upper Body		Lower Body		Full Body	
Anorak	1	Capris	2	Caftan	3
Blazer	1	Chinos	2	Cape	3
Blouse	1	Culottes	2	Coat	3
Bomber	1	Cutoffs	2	Coverup	3
Button-Down	1	Gauchos	2	Dress	3
Cardigan	1	Jeans	2	Jumpsuit	3
Flannel	1	Jeggings	2	Kaftan	3
Halter	1	Jodhpurs	2	Kimono	3
Henley	1	Joggers	2	Nightdress	3
Hoodie	1	Leggings	2	Onesie	3
Jacket	1	Sarong	2	Robe	3
Jersey	1	Shorts	2	Romper	3
Parka	1	Skirt	2	Shirtdress	3
Peacoat	1	Sweatpants	2	Sundress	3
Poncho	1	Sweatshorts	2		
Sweater	1	Trunks	2		
Tank	1				
Tee	1				
Top	1				
Turtleneck	1				

Lower Body: 16
Upper Body: 20
Full Body: 14

YOLO

Jackets: (3)
Blazer
Hoodie
Jacket

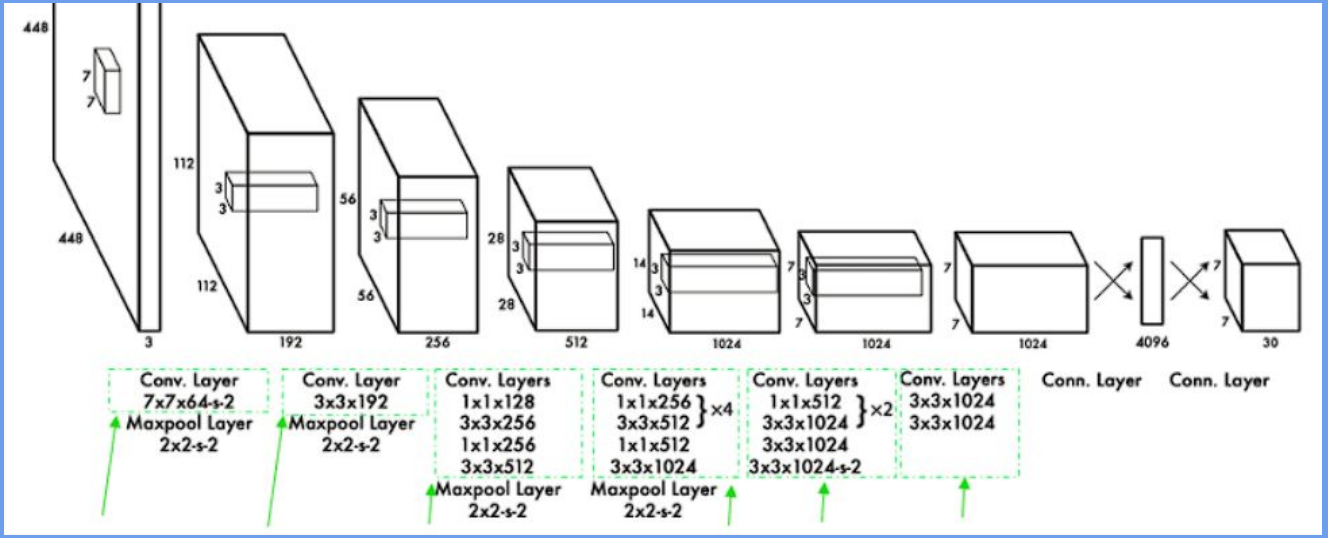
Dresses: (3)
Dress
Romper
Halter

Sweater: (2)
Cardigan
Sweater

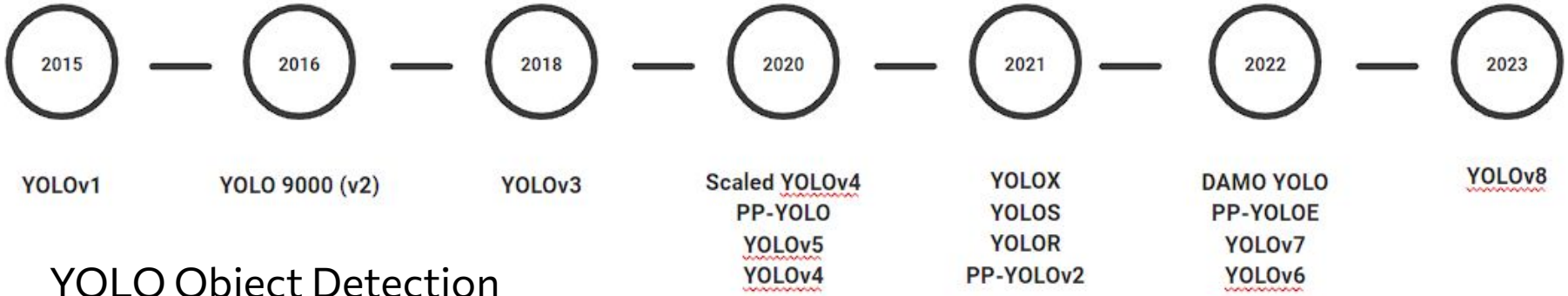
Pants: (10)
Capris
Chinos
Cutoffs
Jeans
Joggers
Leggings
Shorts
Skirt
Sweatpants
Trunks

Shirts: (6)
Blouse
Button down
Henley (long sleeve)
Tank
Tee
Top

0. Jackets - 22,010
1. Shirts - 87,997
2. Dresses - 79,583
3. Sweater - 26,434
4. Pants - 56,651



YOLO Architecture



YOLO Object Detection Models Timeline

Future Extensions & Applications

Future Extensions

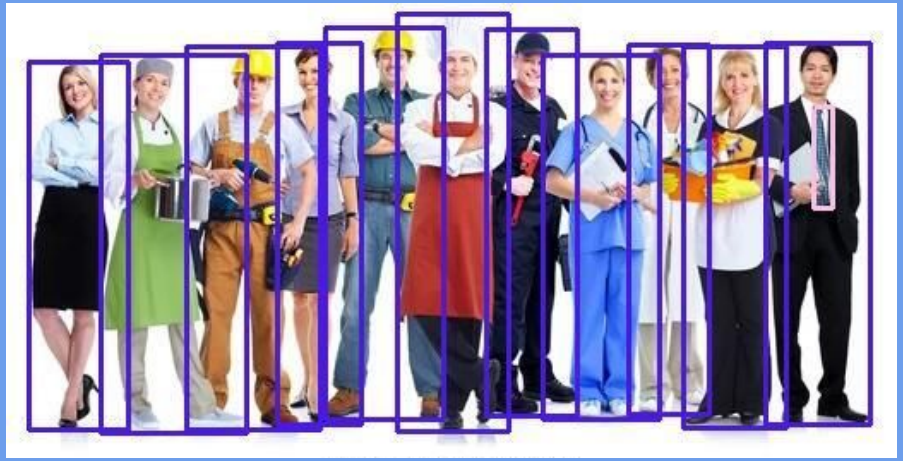
1. With DeepFashion, retrain clothing detection model
   - ➢ Use 3 classes instead of 5 to reduce error
   - ➢ Balance dataset to train each class with an equivalent amount of images
2. Look into using DeepFashion2
   - ➢ Has segmentation which would reduce error and help with identifying colors
   - ➢ Larger dataset leads to a more well trained model
   - ➢ Uses less classes, reduces clutter and background error
   - ➢ Performs better in less ideal situations
3. Advance into logo detection
   - ➢ Train our model with a custom-made data set for a specific company/schools logo eg. (FedEx, AMSA, Military, Walmart, etc.)
4. Put our prototype to the test in a real school/work environment
5. Collect more data on effectiveness of models in different conditions, clothes, and environments

Analysis

- Person detection work amazingly and had minimal error
- The model could detect a person even without a full body view
  - For example, a frame of just the legs of a person could be used to detect a person
    - Offers more flexibility with camera placement, does not necessarily need to show face for person detection
- Filtered classes for Object Detection Model (YOLOv8.) to 1 class, person
- Training the model with the DeepFashion dataset took a lot more time than expected.
  - Dataset was both noisy and imbalanced
  - Contained highly undertrained classes that were not relevant
  - Required manual merging of classes into more generic terms for clothing
  - Didn't have segmentation which made it much more difficult for color detection
- Clothing Model performed poorly when given less ideal situations (noise in image, view of clothing obstructed)
  - Performed better in images with people wearing dresses as opposed to shirts or sweaters

Conclusion

1. Person detection worked great.
2. Clothing detection was very difficult.
    a. More work needs to be done on it because it worked poorly it many scenarios. We believe image segmentation would greatly improve it. We skipped color because of the lackluster results of the clothing model.
3. Outfit detector still needs to be completed.
4. The project was to see if it has merit to it, and to see whether it can be a real working system, which we do.
5. Plan on furthering it, despite the risks.



Alternative Approach

Creating a Bespoke Model

- Collect data from existing security system
- Label data
  - Leverage tools like roboFlow (labeling tool)
  - Hire data engineers to label
- Train model
  - Hire data scientist to train the model
  - Test it to make sure it functions properly
- Recommend to use it on YOLOv8

Pros

- Specific to location
- Works with video system already in place (don't need to adapt them)
- Population of users would be known by the labelers
- Bias could be understood
- Data requirements are less than our pipeline solution

Cons

- Each location
- Updates on uniforms
- Simple model
- Only as good as the labeling and training (Make sure you have enough data)

QR Code