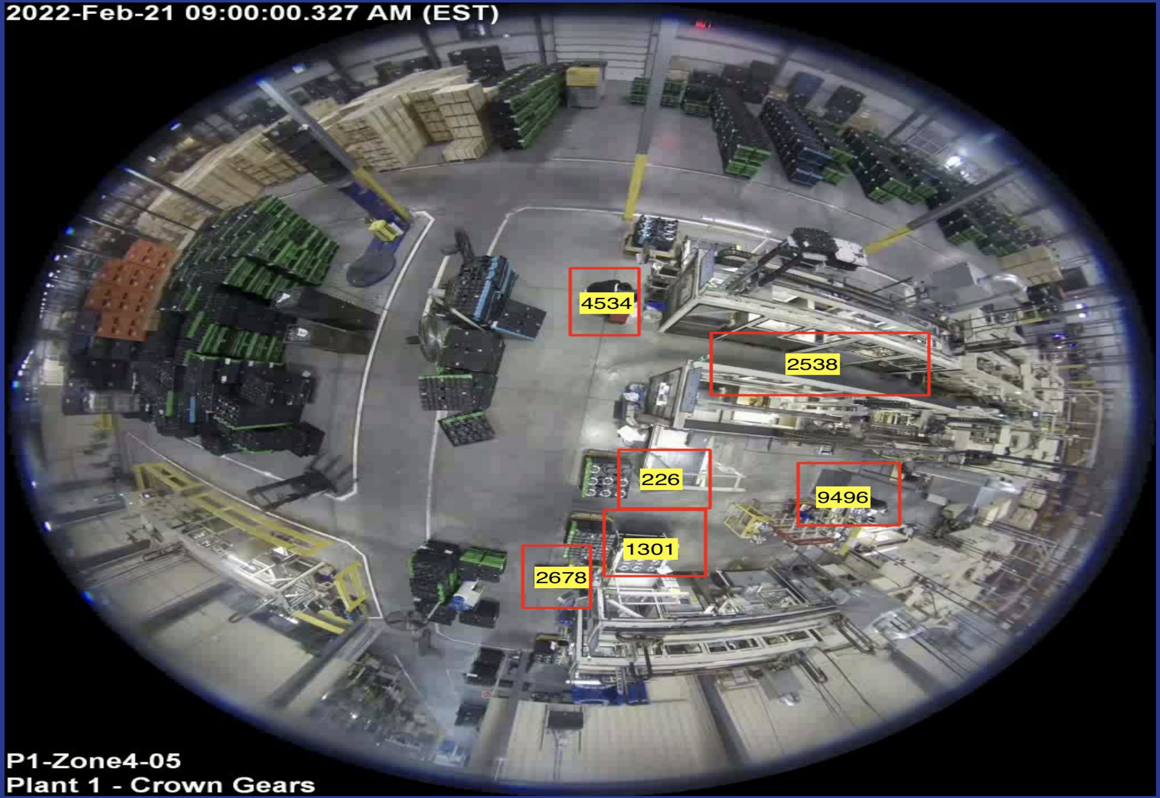
Data Challenge 2022

Lestr.ai

Ankur Phadke, Kartikey Sinha, Raghav Saboo, Shriram Holla

Summary of Results



Plan of Action

Try Everything! *But keep it simple*

Possible Solutions

- Use computer vision techniques to detect people within the region of interest.

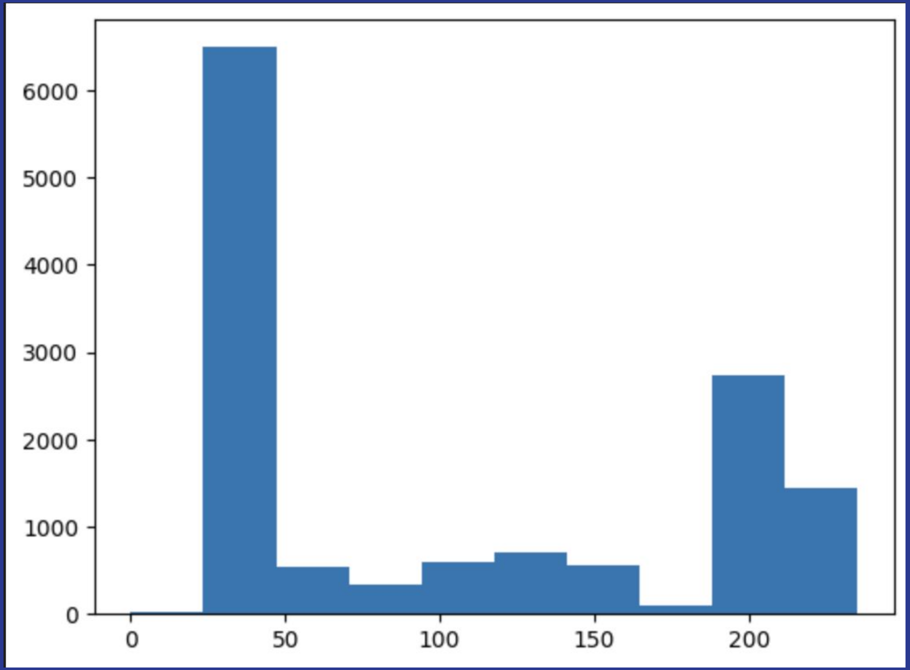
- Using pre-trained model like YoloV5.

- Train a custom model for each ROI.

Solution 1

- Use computer vision techniques such as thresholding to filter out when people exist in a region of interest.
- Use a base ROI with no people in it and compare with a frame.
- Build out a framework and iteratively improve from here

Errors - ROI 1



Results

- Some ROIs reported accurate values initially.
- Drastically dropped once lighting conditions changed.
- A LOT of false positives.
- Due to moving machines and crates, people that were initially detected could not be detected anymore

Improvements

- To avoid changes in lighting conditions, we need to use a robust thresholding algorithm such as Adaptive Gaussian Thresholding.

- Crop regions within the ROI where people cannot walk (machines, crates).

Transformations

Before

After

Results v2

ROI 1
Worked great due to contrasting colours!

ROI 2
Cropping the moving machine parts increased accuracy. A low number of false positives

ROI 5
Due to changes in lighting conditions, there were a lot of false positives

ROI 3
Worked sufficiently well with infrequent false negatives

ROI 4
Worked great due to lots of movement!

ROI 6
Due to items being placed and removed, there were a lot of false negatives

Solution 2

- Use well-researched model architectures (eg- VGG18)

Results

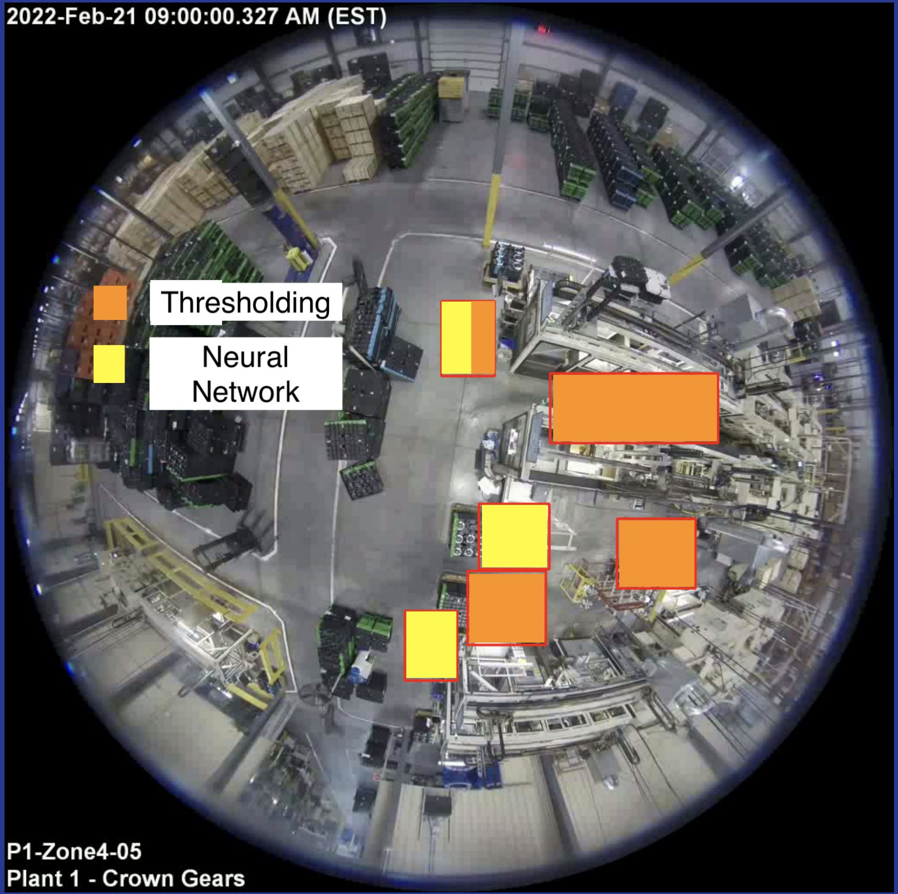
- Models showed great results when trained from scratch on localized ROIs. This allowed the model to learn how to account for radial distortion. More versatile as compared to solution 1.

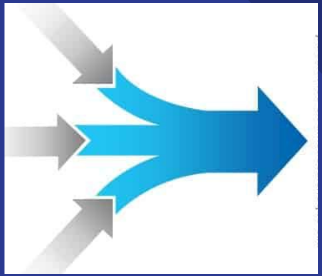
Drawbacks

Solution 2

- **Painful to label images-** Hard to label so many frames and thus more difficult to make algorithm more robust.

- **Computationally Heavy-** Not efficient to make 6 model predictions for each frame.





Combining our solutions

Best of both worlds

Challenges

- Working with fisheye images
- Figuring out the right threshold was hard due to changes in lighting conditions
- Distinguishing between humans and other moving objects.
- Finding the best neural network was difficult and time consuming
- Combining our approaches together to find the best solution

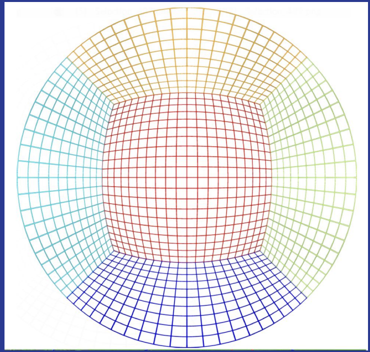
Advantages

- Ensemble-inspired method allow us to get as close to the real value as possible.
- Data-driven solution ensures robustness.
- Low computational requirements as some ROIs do not need a neural network

Poetry in Motion!

What could we do to further improve our accuracy?

- Since no method worked the best individually, we decided to combine our approaches based on their advantages and disadvantages
- What's next? Fisheye CNNs
- How to scale with new ROIs?
  - *Edge Detection*




That's all Folks!