Developing Elevator Recognition Software for Indoor Autonomous Driving Robot

Participating Researchers : Minhyek Jeon, Seung Yoon Shin

Advisor : Sungjun Choi

July 18, 2023

The need for research

ROBOTIS

ROBOT INTELLIGENCE LAB

- Elevator state-awareness model for boarding sequences of indoor autonomous robots
    - To be able to react flexibly to special situations
      + door closes in the process of boarding after seeing the door open
      + button not pressed properly after entering the elevator normally
      , etc.
  - to recognize the status of elevators
  - recognition of various indicators (elevator switches, signs, license plates, etc.),
  - Need for a detector that is robust to motion blur caused by robot movement is required.

Table of Contents

ROBOTIS

ROBOT INTELLIGENCE LAB

1. Assignment Background and Development Environment

1.1 Assignment Goals

Different elevator indicator configurations, shapes, and looks in different buildings

- Developing a robust detector that can effectively detect different indicator shapes
  - Domain Adaptation: models motion blur due to robot movement and image changes due to different light conditions
  - Data Augmentation: Since we cannot consider all possible camera angles when training the detector, we will research/use data augmentation techniques utilizing Camera Perspective Can we be flexible enough to handle various situations?
- Design a response protocol for occlusions, which are relatively common inside elevators.

1.2 Assignment Requirements

Recognize elevator entries and **exits**

Recognize elevator exterior state

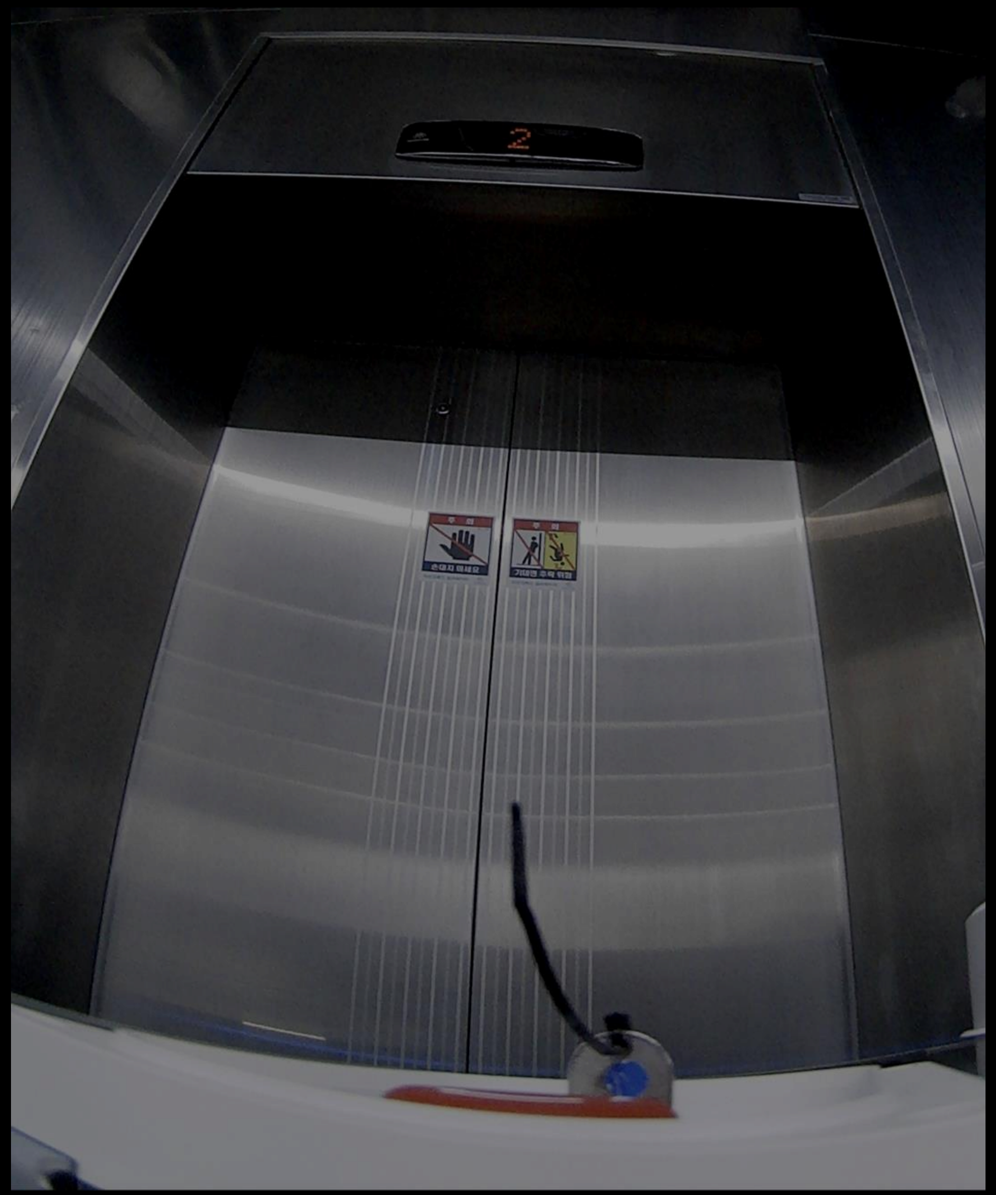
1. Elevator current floor
2. Direction of elevator travel
3. Elevator doors open/arrival

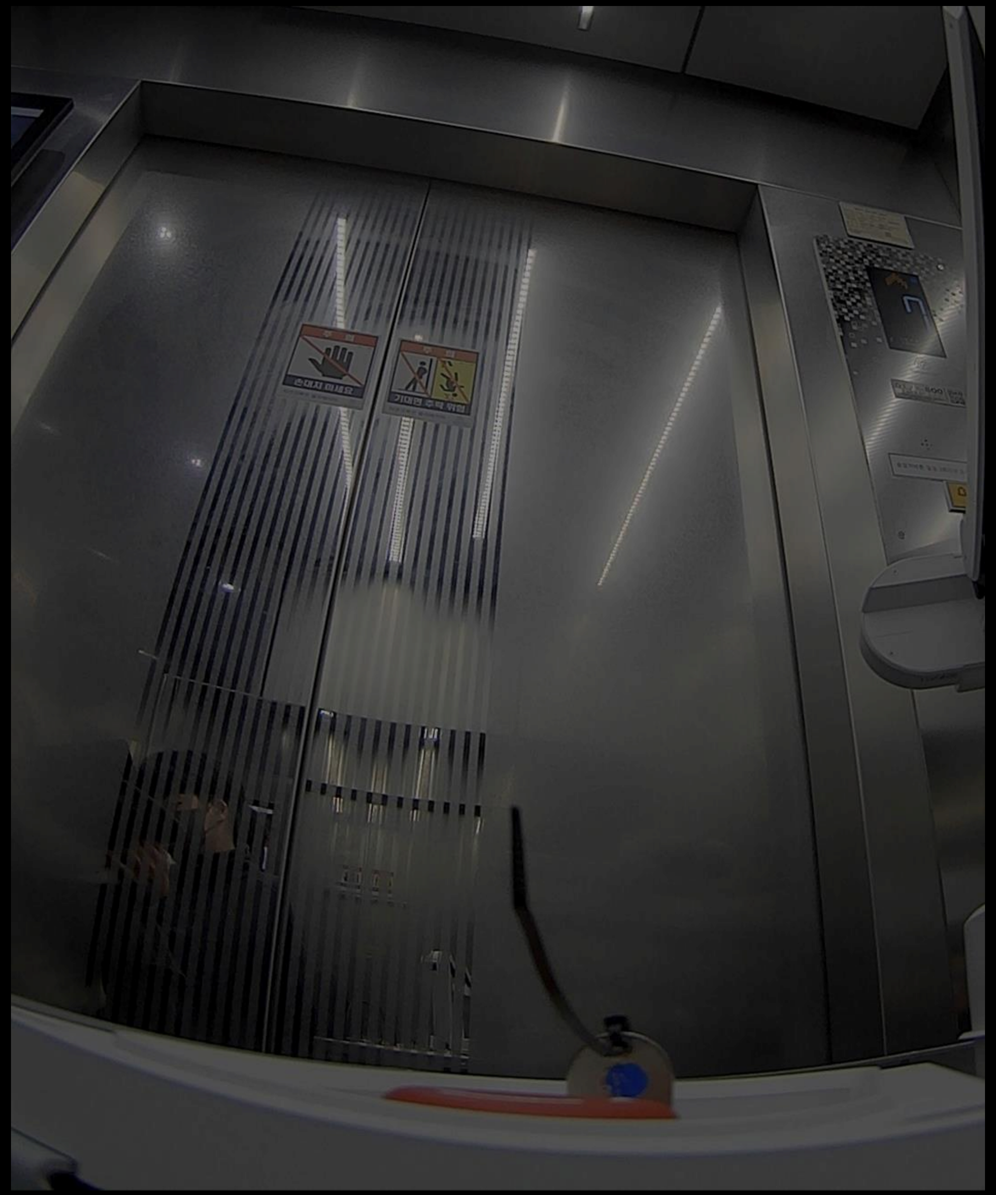
Recognize elevator interior state

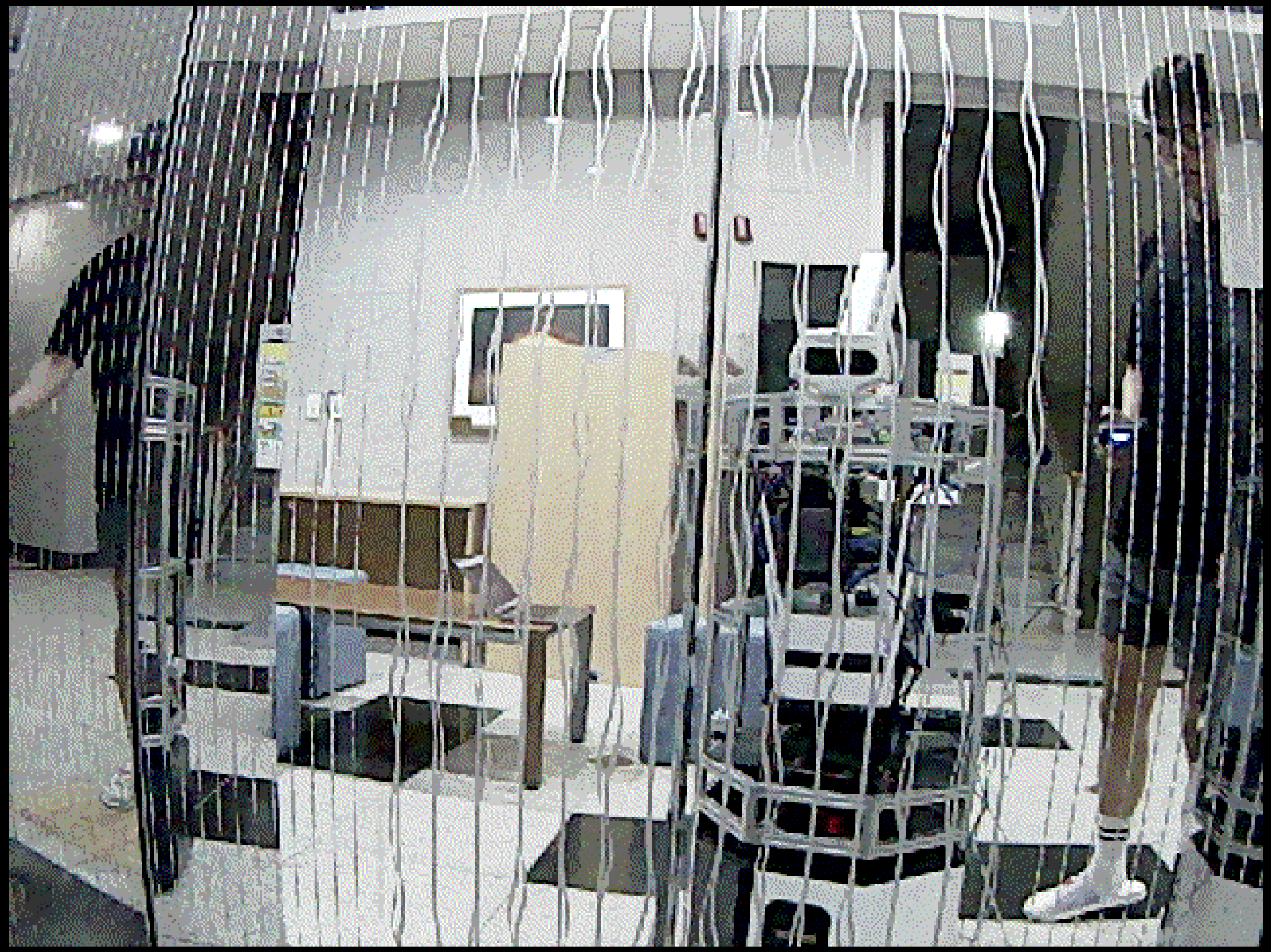
1. Elevator current floor
2. Direction of elevator travel
3. Elevator doors open/arrival

Recognize elevator load status

1. Door is fully closed/open
2. Door opening or closing







1.3 Challenge Environment and Constraints

Development Environment

- **Powered by Orin AGX**
  - **Jetpack 4.6: The latest version currently released**
  - **ROS2 Foxy**

Camera

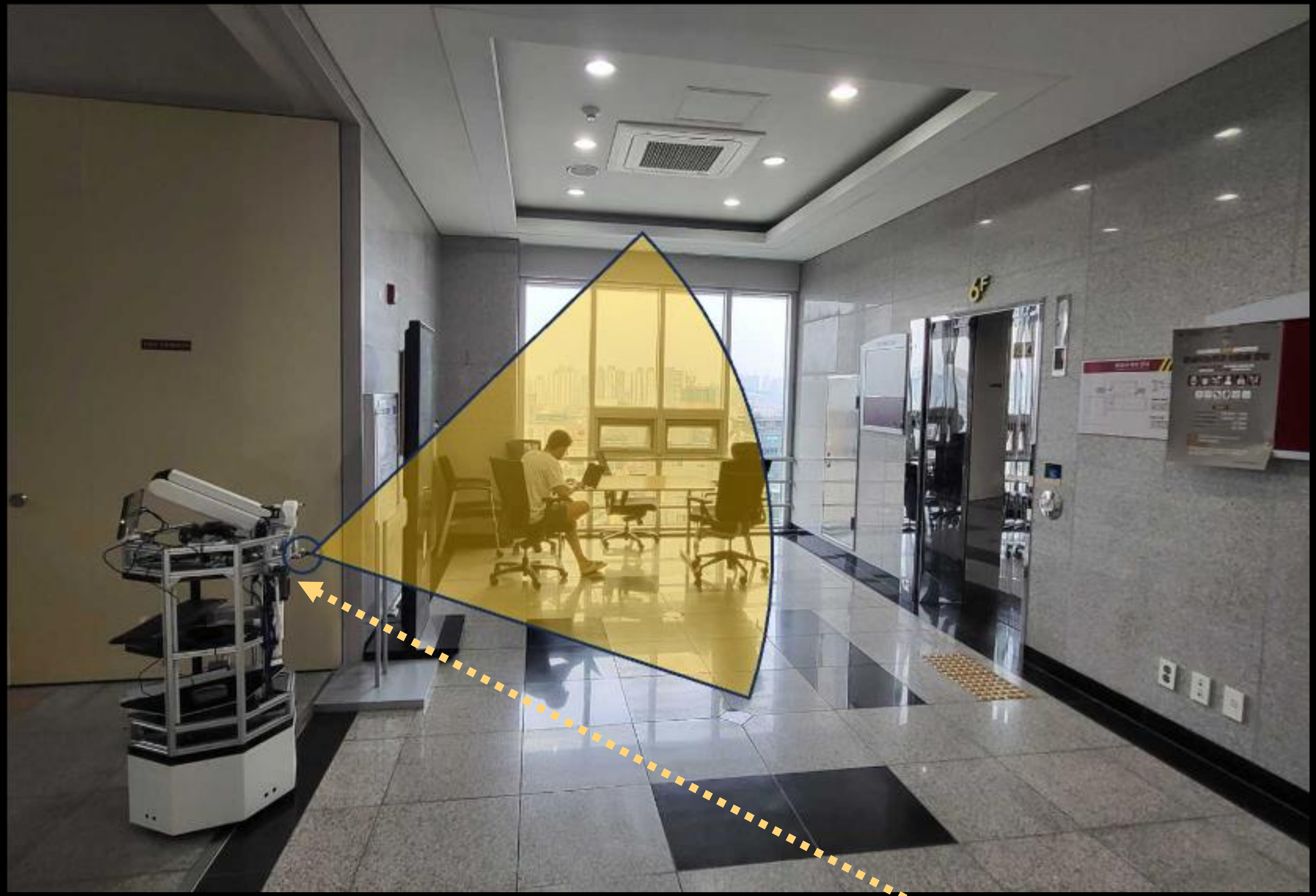
- **[The CSI]camera you want to use**
- **Resolution: FHD**
- **Viewing angle: wide angle greater than 120 degrees**

2. Setting up environment at Korea University

2.1 Angle of view issues

ROBOTIS

ROBOT INTELLIGENCE LAB



Initially, the robot's camera was located at the front of the robot,

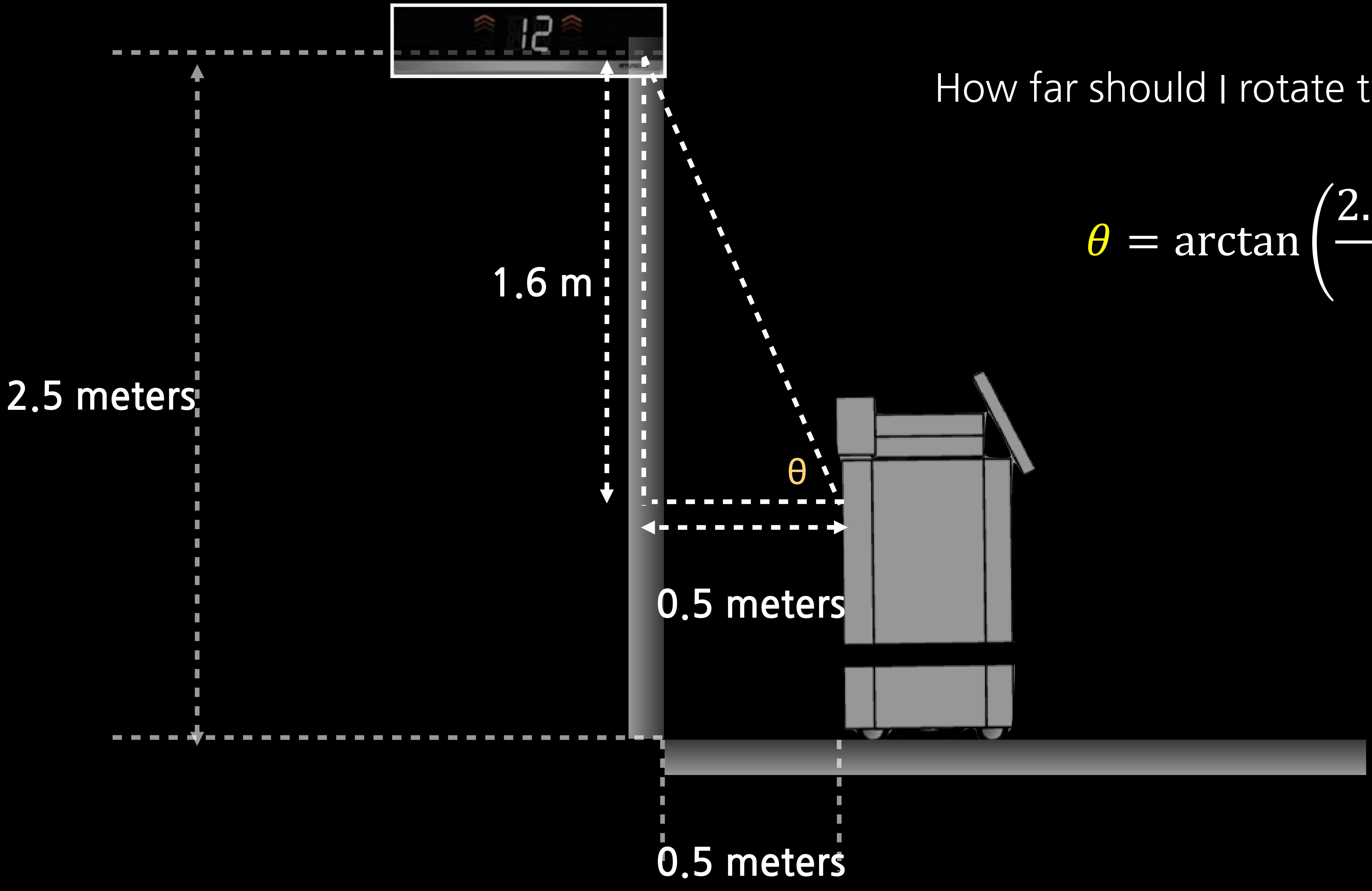
In this case, you'll need to move away from the boarding location to get to the

I could barely see all the indicators.

2.1 Angle of view issues

ROBOTIS

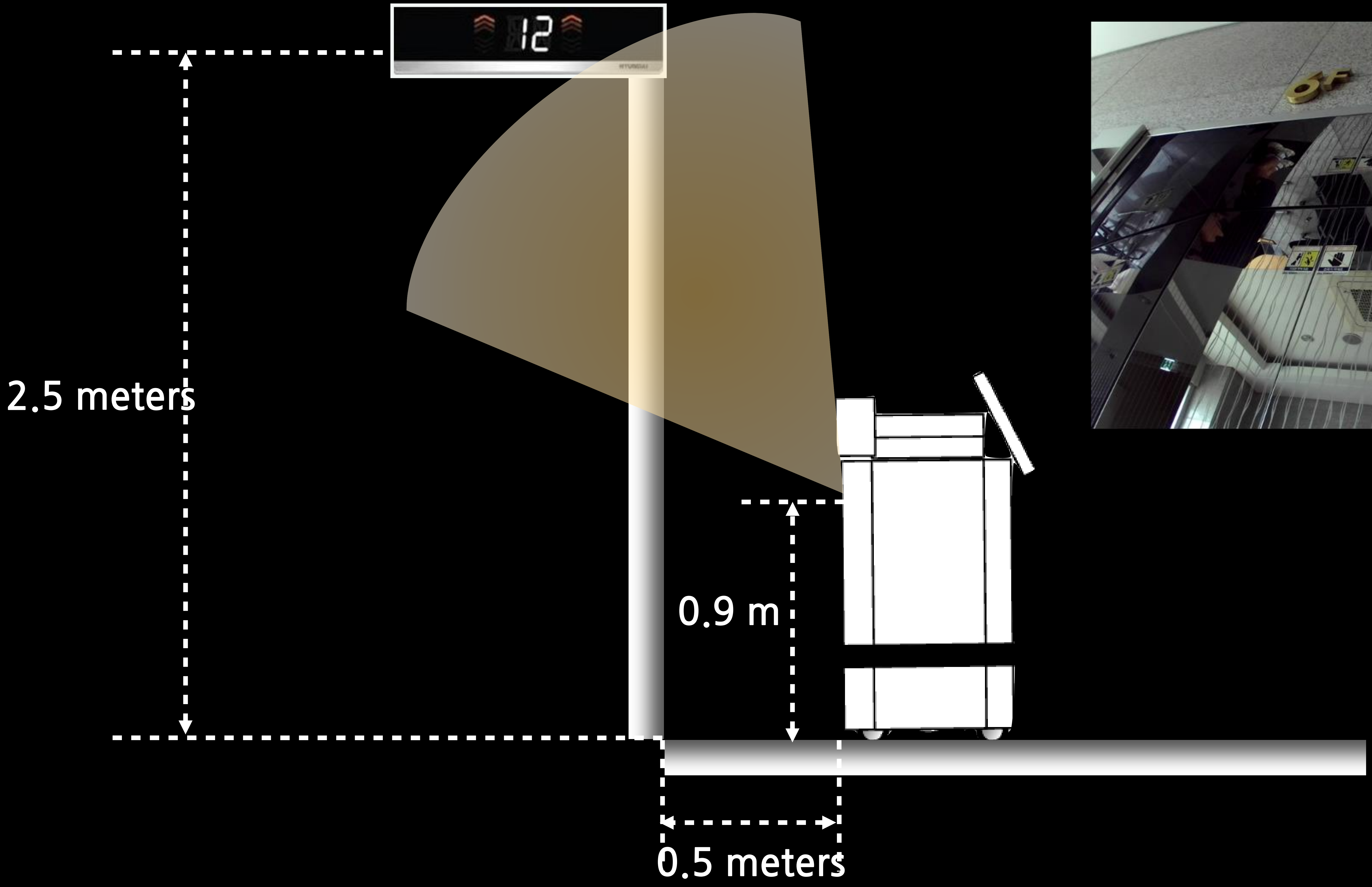
ROBOT INTELLIGENCE LAB



How far should I rotate the camera? Let's find out.

$$\theta = \arctan\left(\frac{2.5m - 0.9m}{0.5m}\right) \approx 72.65°$$

2.1 Angle of view issues



ROBOTIS

ROBOT INTELLIGENCE LAB

ROBOT INTELLIGENCE LAB

3. Hierarchical two Step Approach

3.1 Hierarchical Approach (September-November 22)

ROBOTIS

ROBOT INTELLIGENCE LAB

	Abbreviations	Templates	Definition	
Elevator Door	ED	ED_{Status}	Whether the door is open or closed	ED_OPEN
				ED_CLOSE
Floor Designator	FD	FD_{Floor Count}	Display the current number of floors	FD_B2
				⋮
				FD_7
Hall Lantern	HL	HL_{Direction}	Current direction of travel	HL_UP
				HL_DOWN
				HL_NONE
Landing Operator Pannel	LOP	LOP_{Direction}	Elevator button	LOP_UP
				LOP_DOWN
				LOP_NONE
Position Indicator	PI	PI_{Number of layers}	Current elevator location	PI_B2
				⋮
				PI_7

3.2 Two-Stage

YOLOv7 (Single Stage) + CNN (Second Stage)

Detect large classes like ED,FD,PI,HL,LOP first

Detecting state from large classes

3.2 Two-Stage

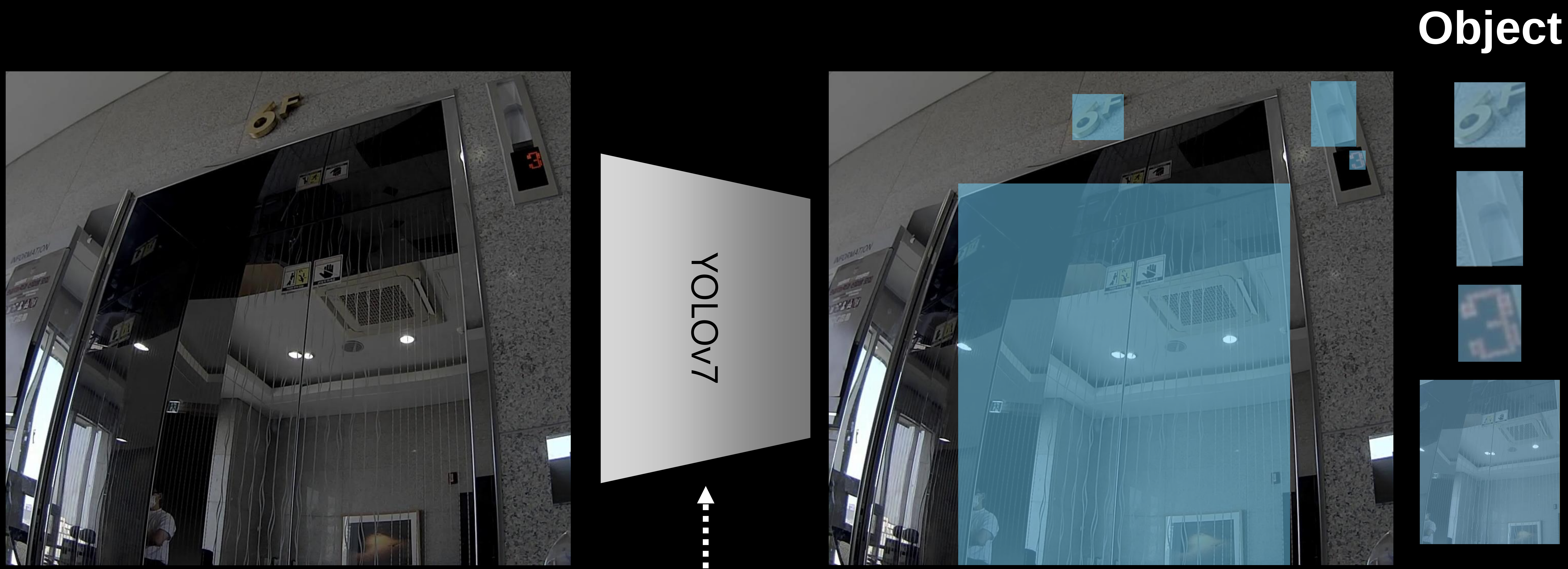
YOLOv7 (Single Stage) + 6xCNN (Second Stage)

Detect large classes like ED,FD,PI,HL,LOP first

Train different CNNs on different clas

3.2 Two-Stage

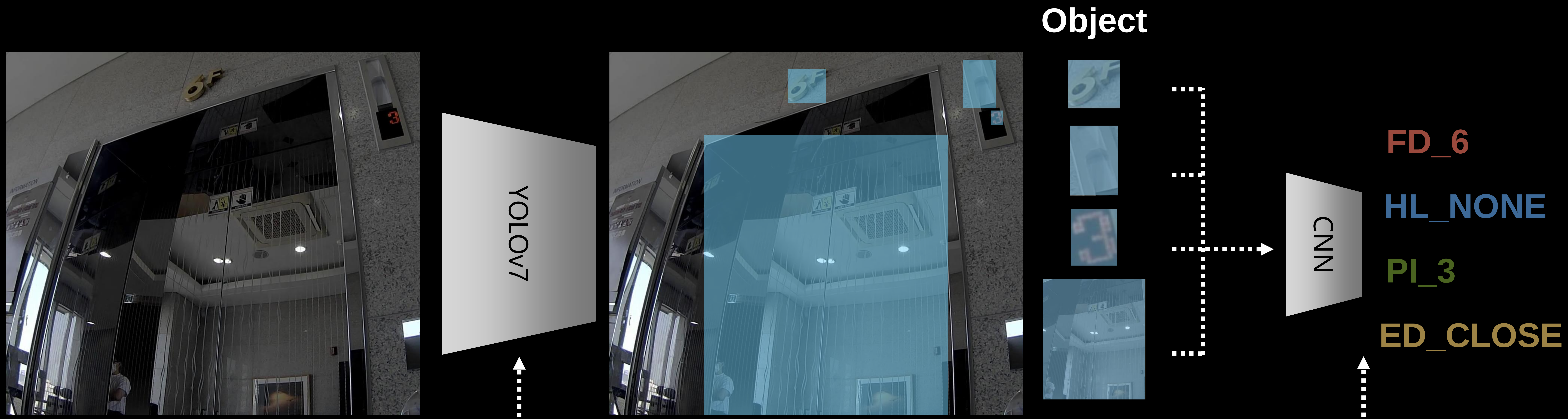
YOLOv7 (Objectness) + CNN (Second Stage)



Detect ED,FD,PI,HL,LOP by viewing them all as Object

3.2 Two-Stage

YOLOv7 (Objectness) + CNN (Second Stage)



Detect ED,FD,PI,HL,LOP by viewing them all as Object

If CNN is a large class and
Detect all states

3.3 Methodology Comparison

ROBOTIS ROBOT INTELLIGENCE LAB

	mAP(0.5)	mAP(0.95)
Single-Stage	0.2993	0.1414
YOLOv7 (SingleStage) + CNN (Second Stage)	0.1081	0.048
YOLOv7 (SingleStage) + 6xCNN (Second Stage)	0.1264	0.0557
YOLOv7 (Objectness) + CNN(Second Stage)	-.	-.

ROBOT INTELLIGENCE LAB

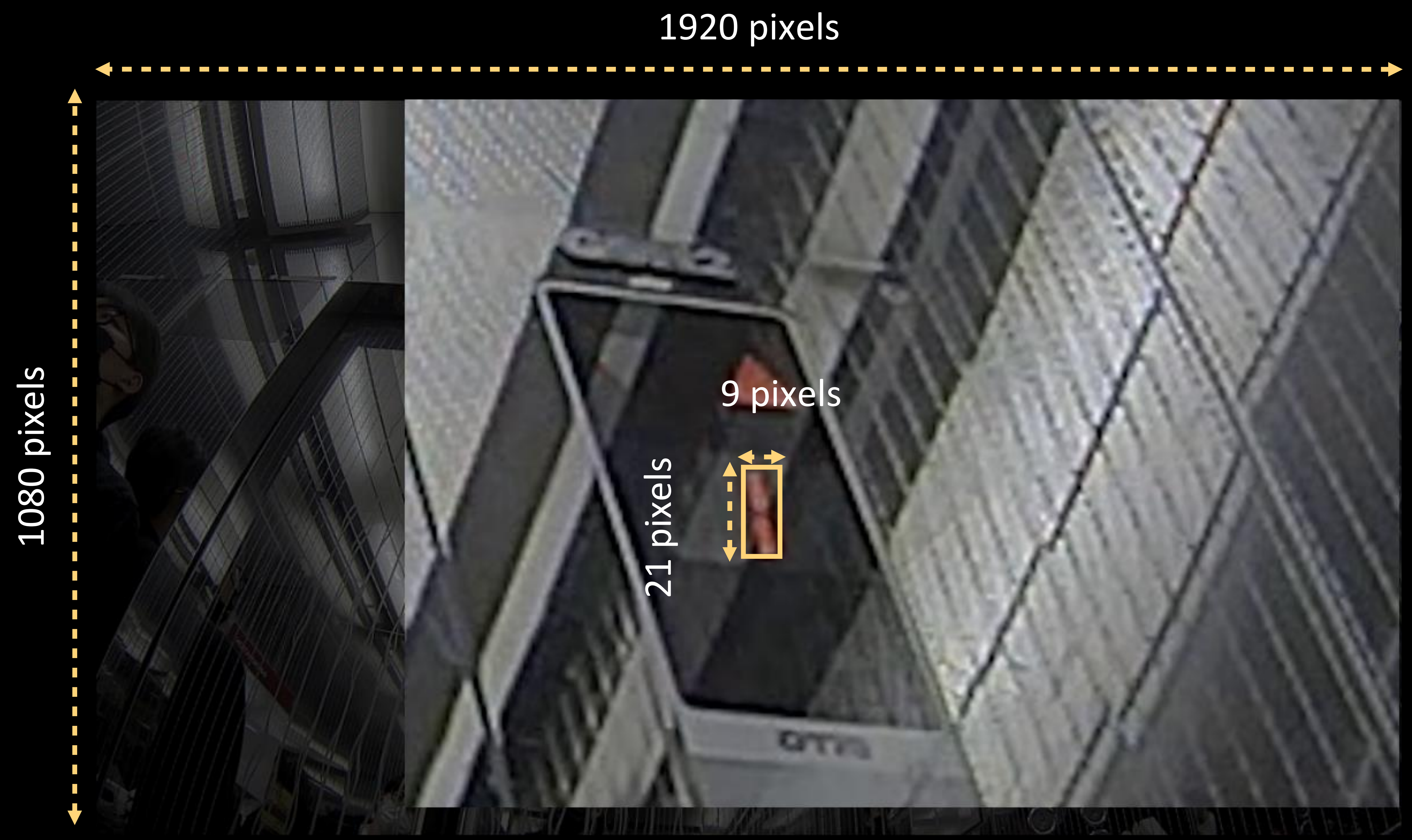
4. Patch Augmentation

4.1 Small Object Detection

ROBOTIS

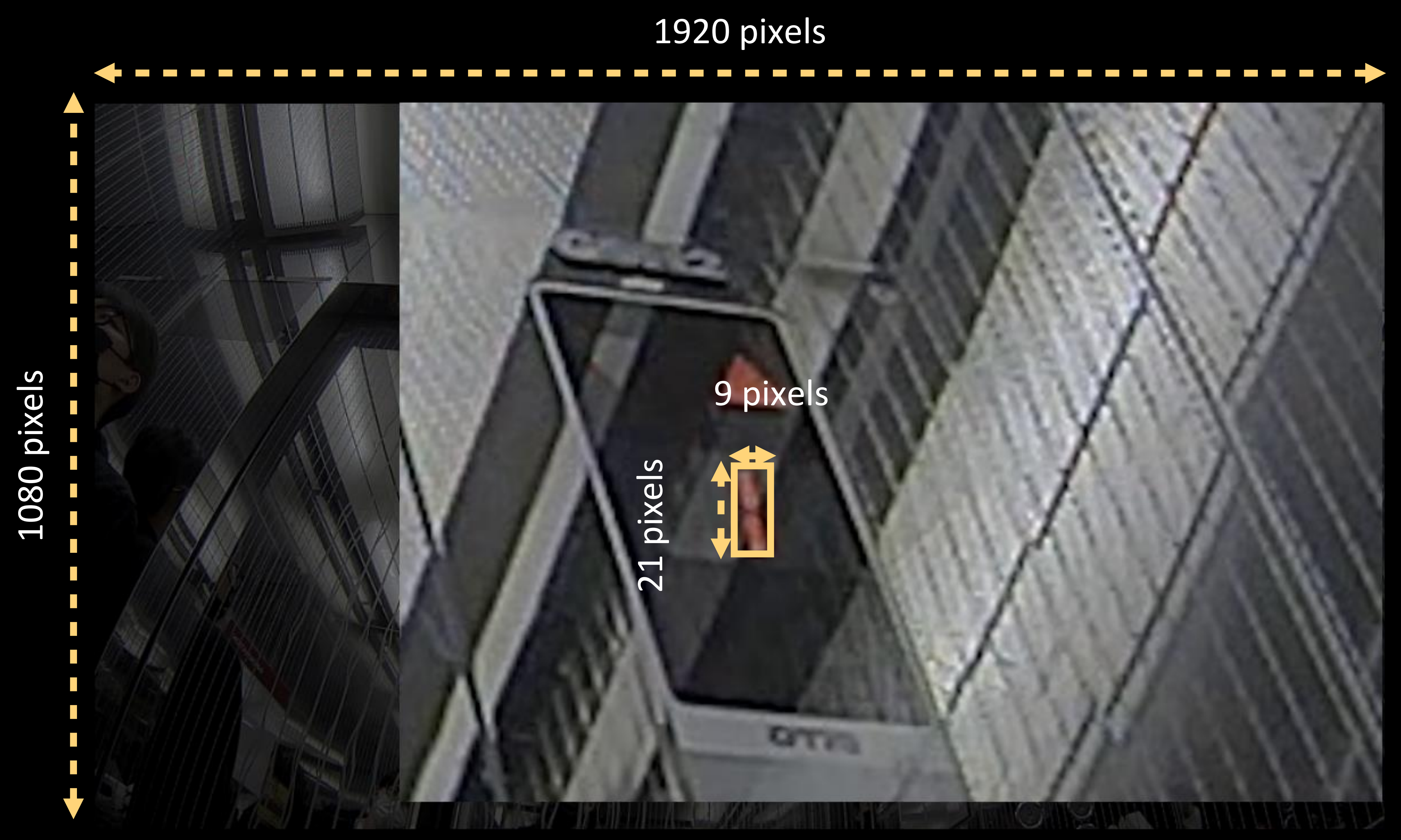
ROBOT INTELLIGENCE LAB

Resize that label to make it smaller

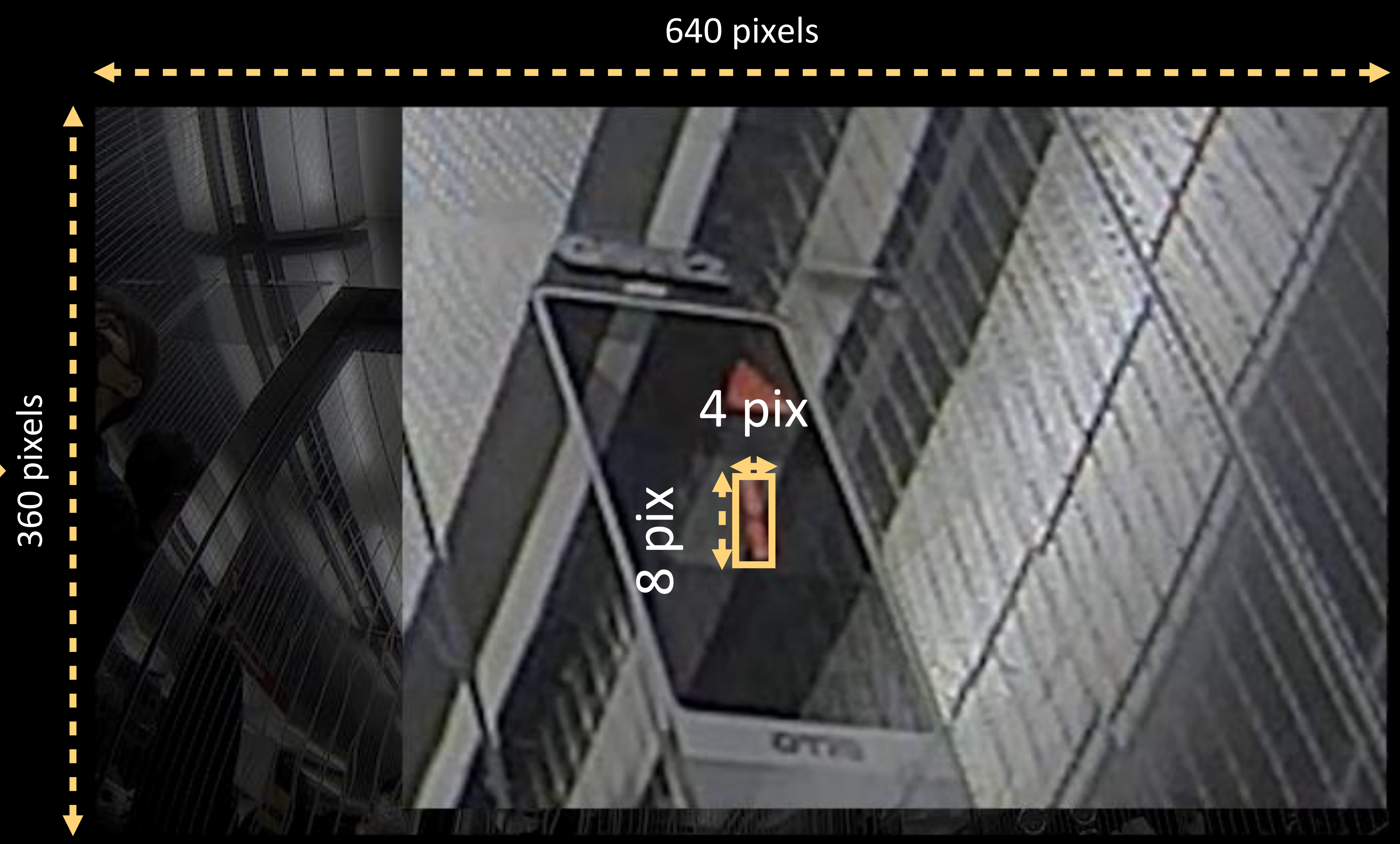


4.1 Small Object Detection

Lots of very small labels, especially numbers and letters



Resize to Inference Size
(1920,1080) → (640,360)



4.2 Patch Augmentation

Resize also breaks the resolution.

Patch at 1920x1080

Patch at 640x360

Increase your resolution when you train to train

4.2 Patch Augmentation
(Feb-Mar 23)

Let's start by cropping the patch to 640x640

Divided into 6 parts

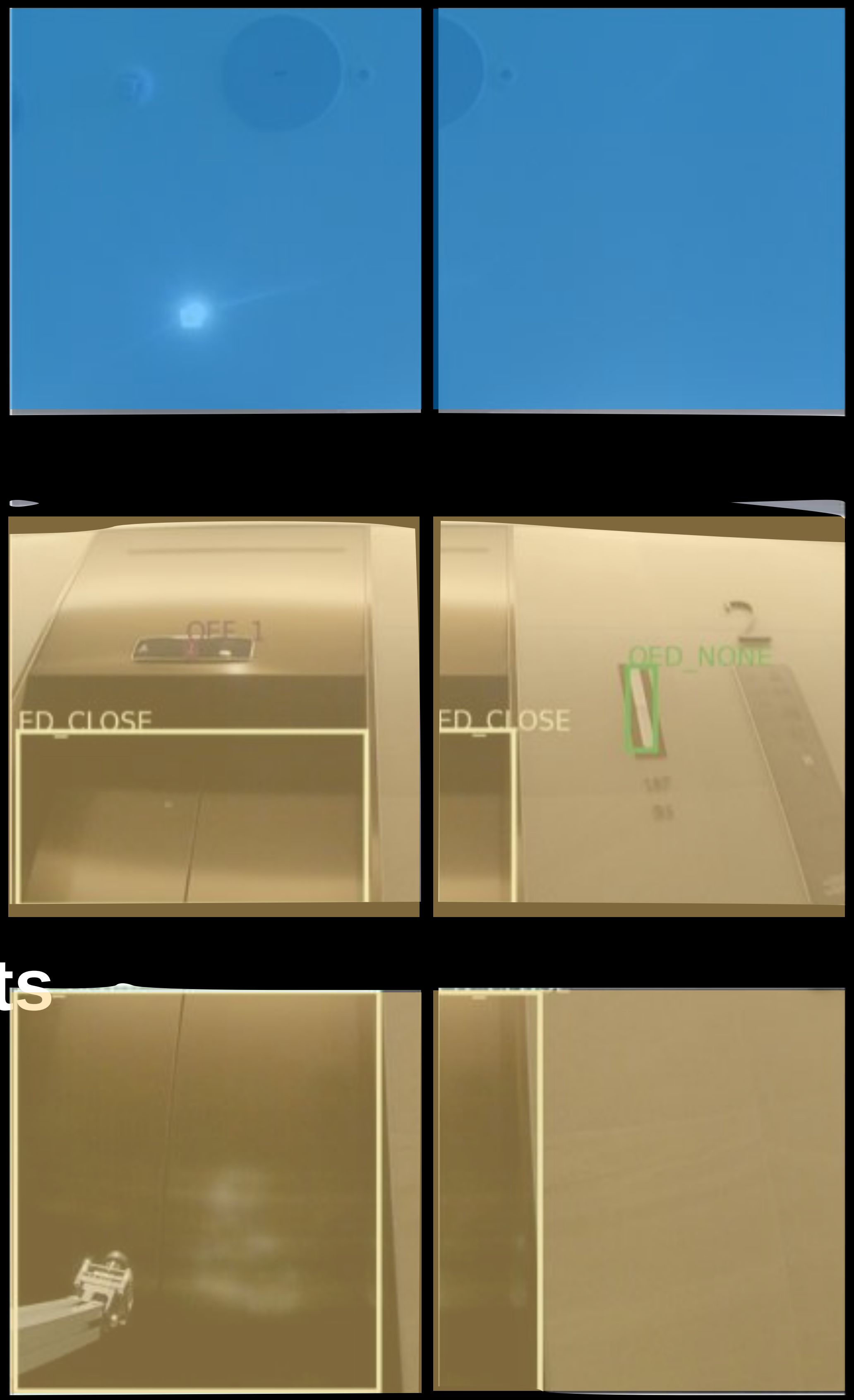
(1920 x 1080)

6 x (640 x 640)

4.2 Patch Augmentation

(1920 x 1080)

Divided into 6 parts



6 x (640 x 640)

that does not include a l

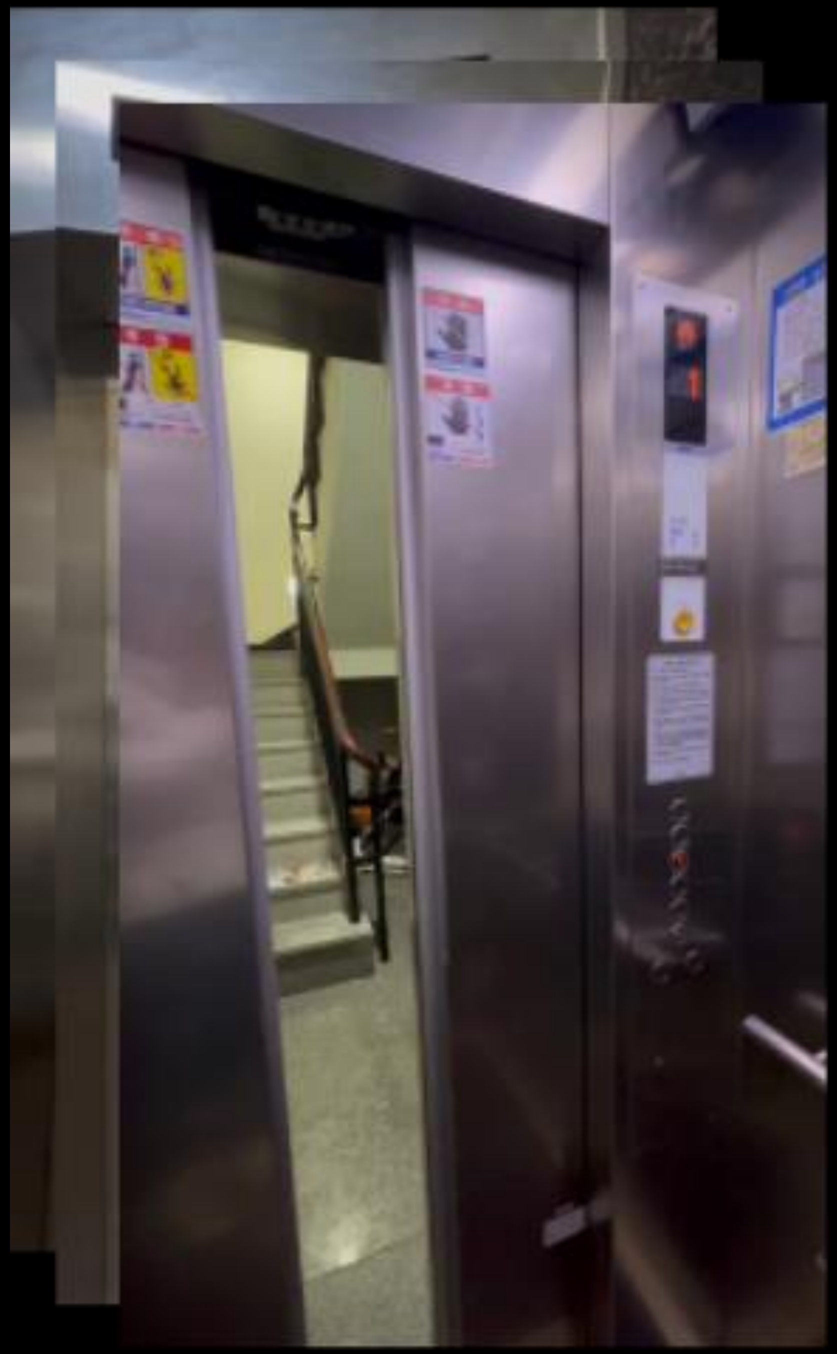
Patches

(Remove from Dataset)

containing a label

Patches

4.2 Patch Augmentation



X 5.6

Existing data : 5,000 photos

Existing data + Patch Augmentation : 28,000 sheets

4.4 Patch Augmentation

ROBOTIS

ROBOT INTELLIGENCE LAB

Improve performance by simply augmenting patches

Metric	mAP@0.5	Status Metric	
Test Time Augmentation			Test Time Augmentation + Patches + Full Image
Details		Precision/Recall/F1	
YOLOv7	0.730	0.813/0.881/0.843	0.602/0.934/0.736
YOLOv7 (+Patch Aug.)	0.784	0.878/0.792/0.833	0.767/0.961/0.853

ROBOT INTELLIGENCE LAB
ROBOT INTELLIGENCE LAB

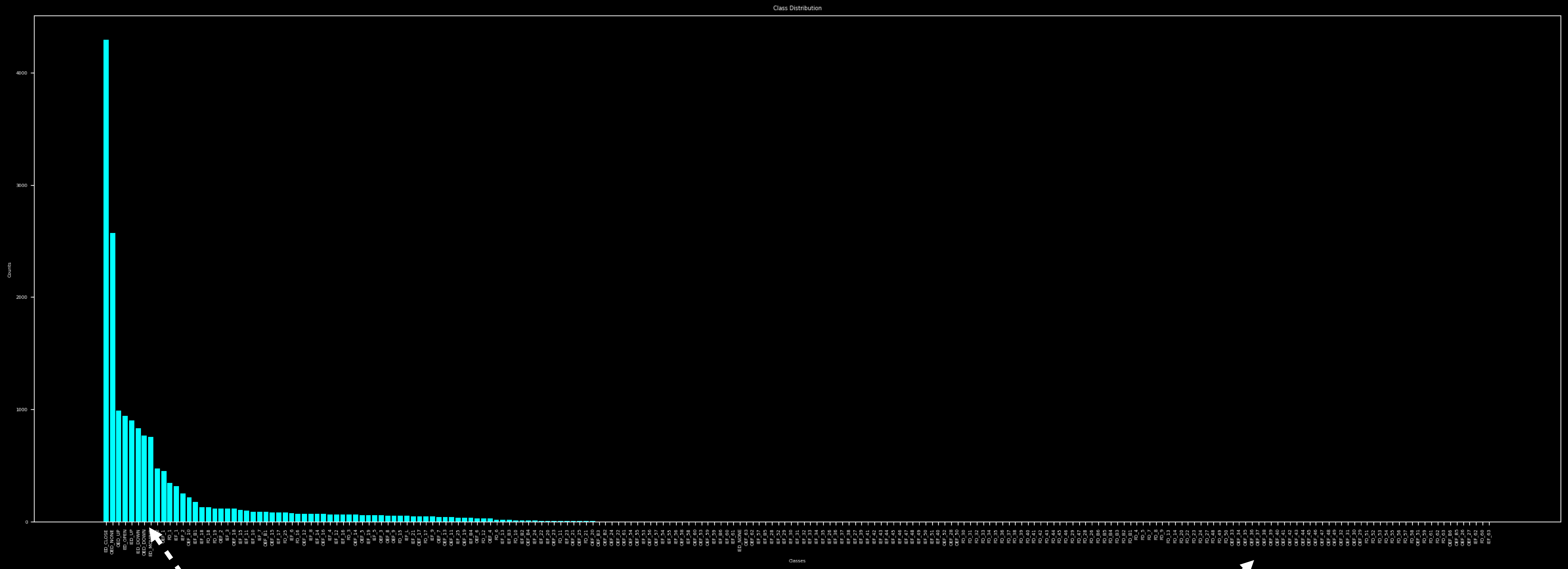
5. Diffusion Augmentation

5.1 Dataset statistics

ROBOTIS

ROBOT INTELLIGENCE LAB

Basic statistics for a dataset of seven locations



There are an overwhelming number of labels for elevators

The number of floors is overwhelmingly small

5.1 Dataset statistics

ROBOT INTELLIGENCE LAB

Basic statistics for a dataset of 7 places :: Label size and count.

In comparison, other labels are overwhelmingly few and far between.
It takes up fewer pixels in the image.

Elevator doors are overwhelming and
It takes up an large amount of pixels.

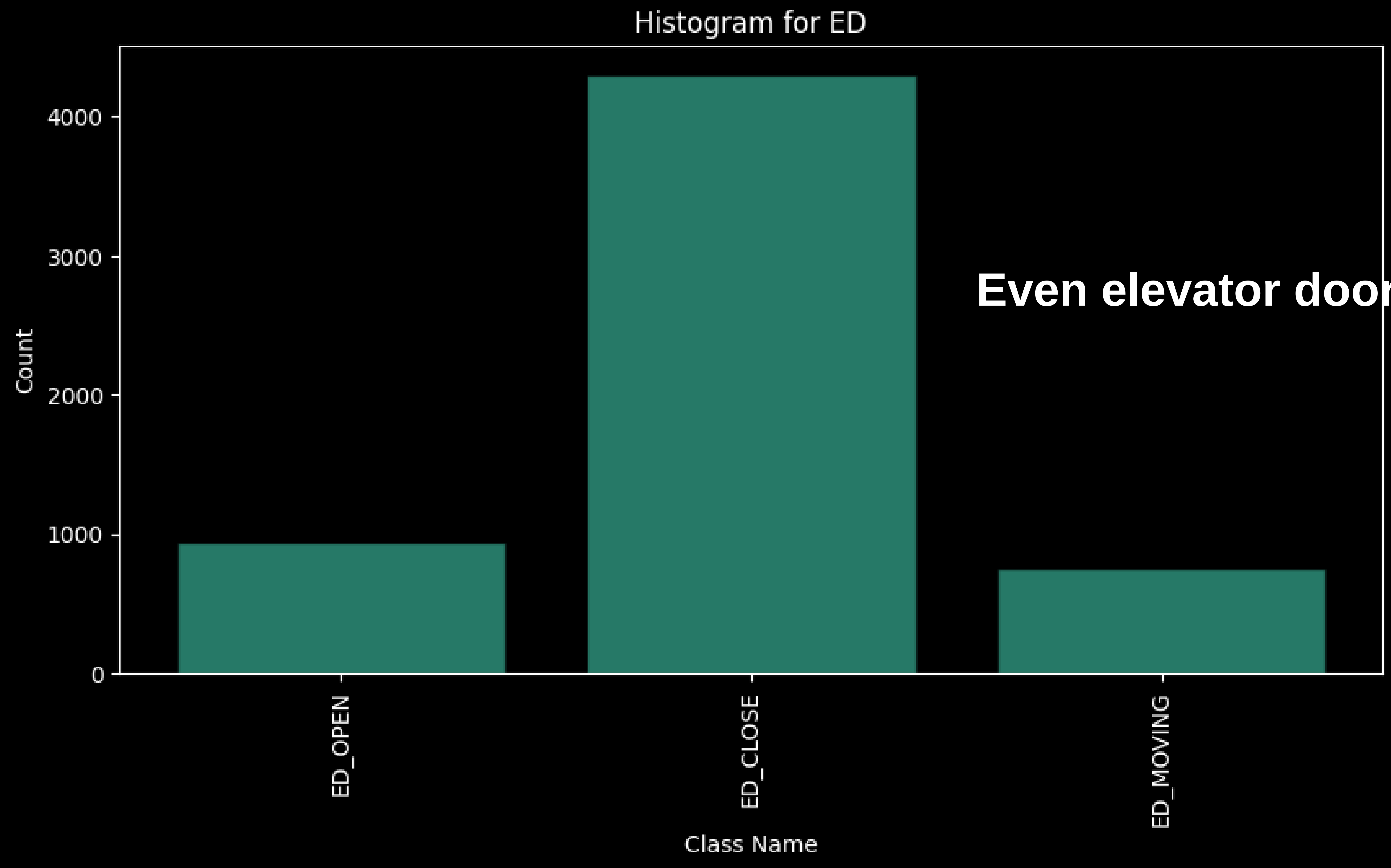
5.1 Dataset statistics

ROBOTIS

ROBOT INTELLIGENCE LAB

Basic statistics for a dataset of 7 places :: Highly unbalanced, even among the few classes.

Elevator doors

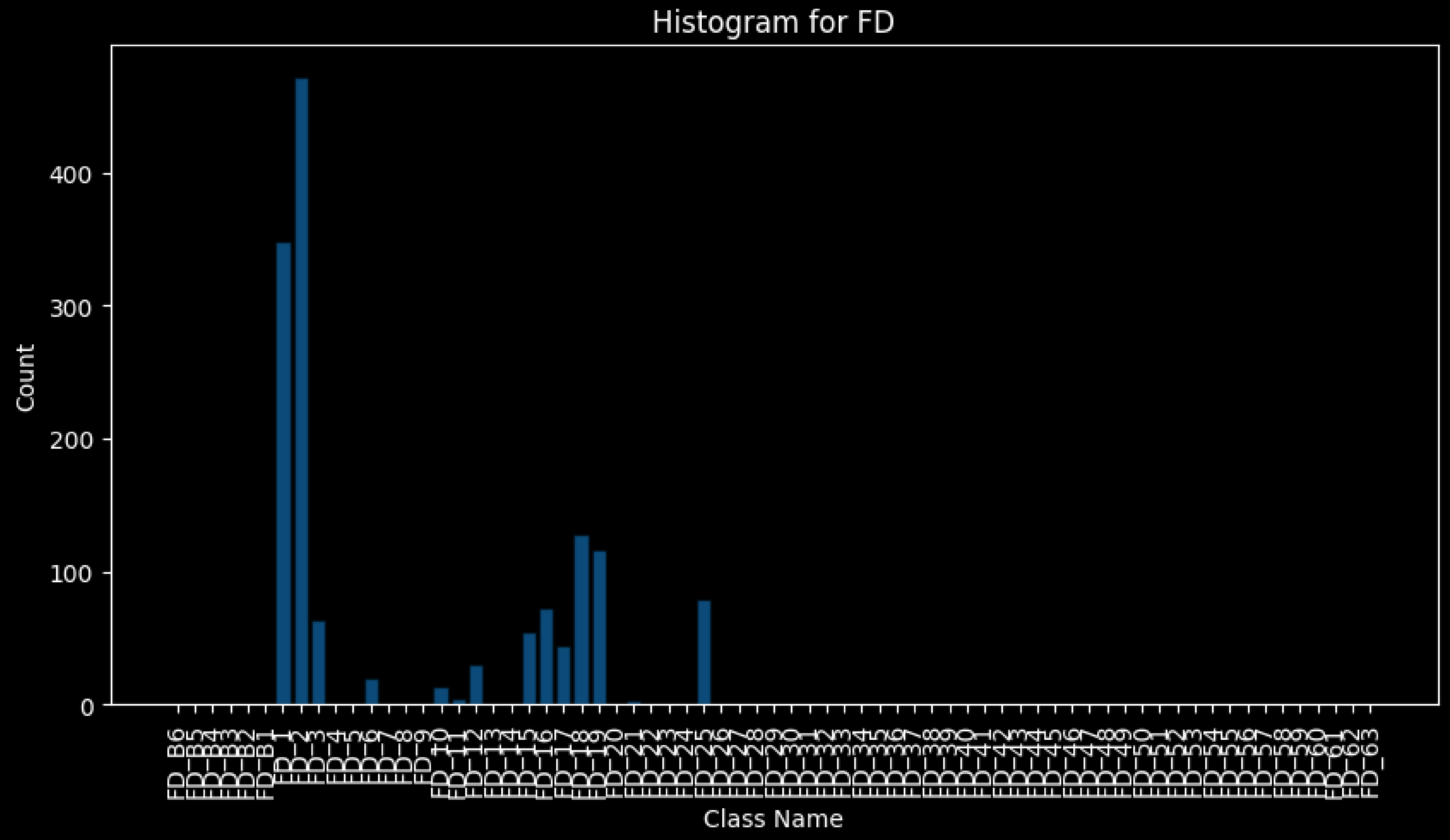


Even elevator doors have a lot more closes

5.1 Dataset statistics

Basic statistics for a dataset of 7 places :: Highly unbalanced, even among the few classe

The current floor of the house ant (Floor Designer)



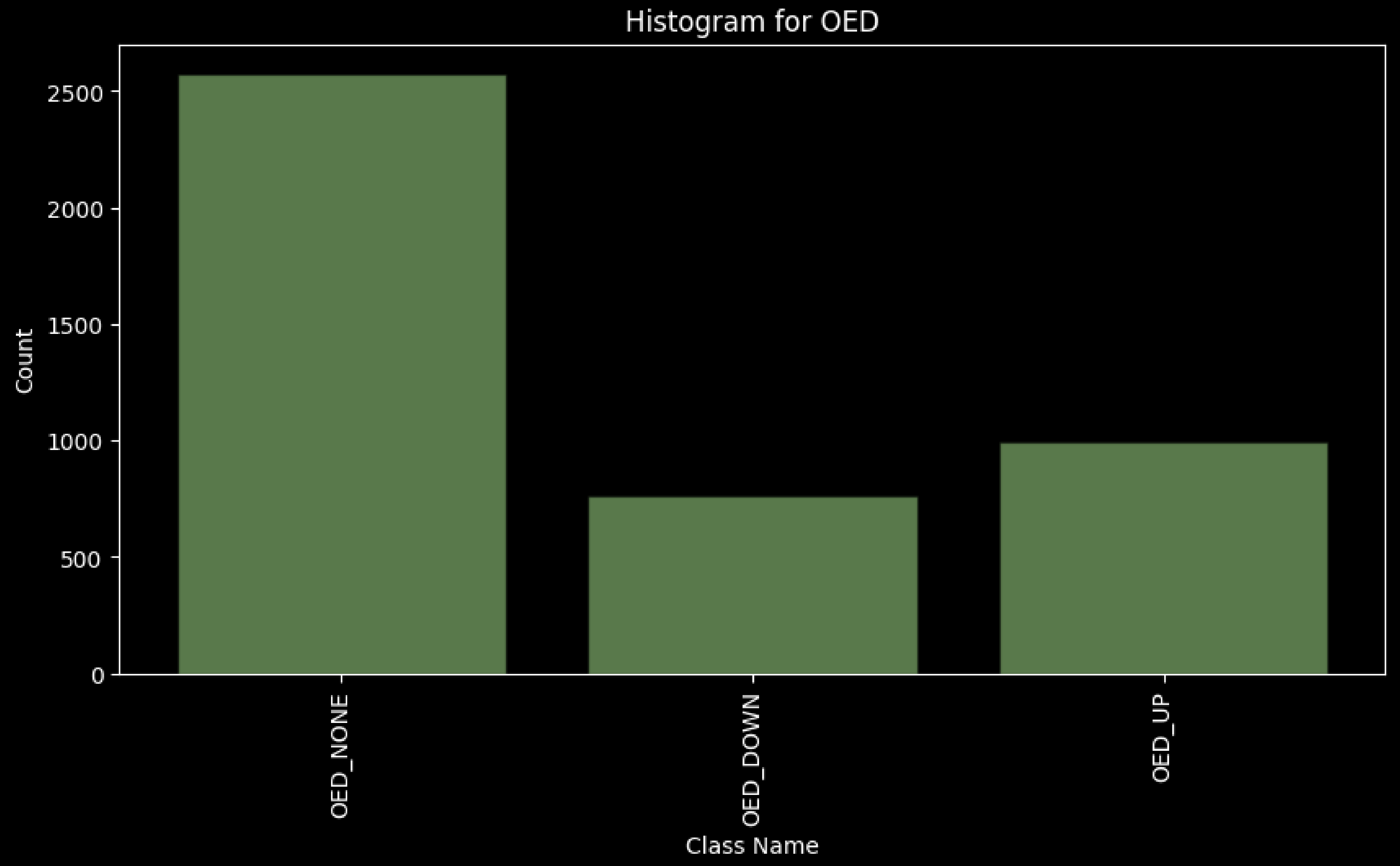
For external floor counts, they tend to cluster around a certain number of floors.
Significant impact on dataset unbalanced experience.

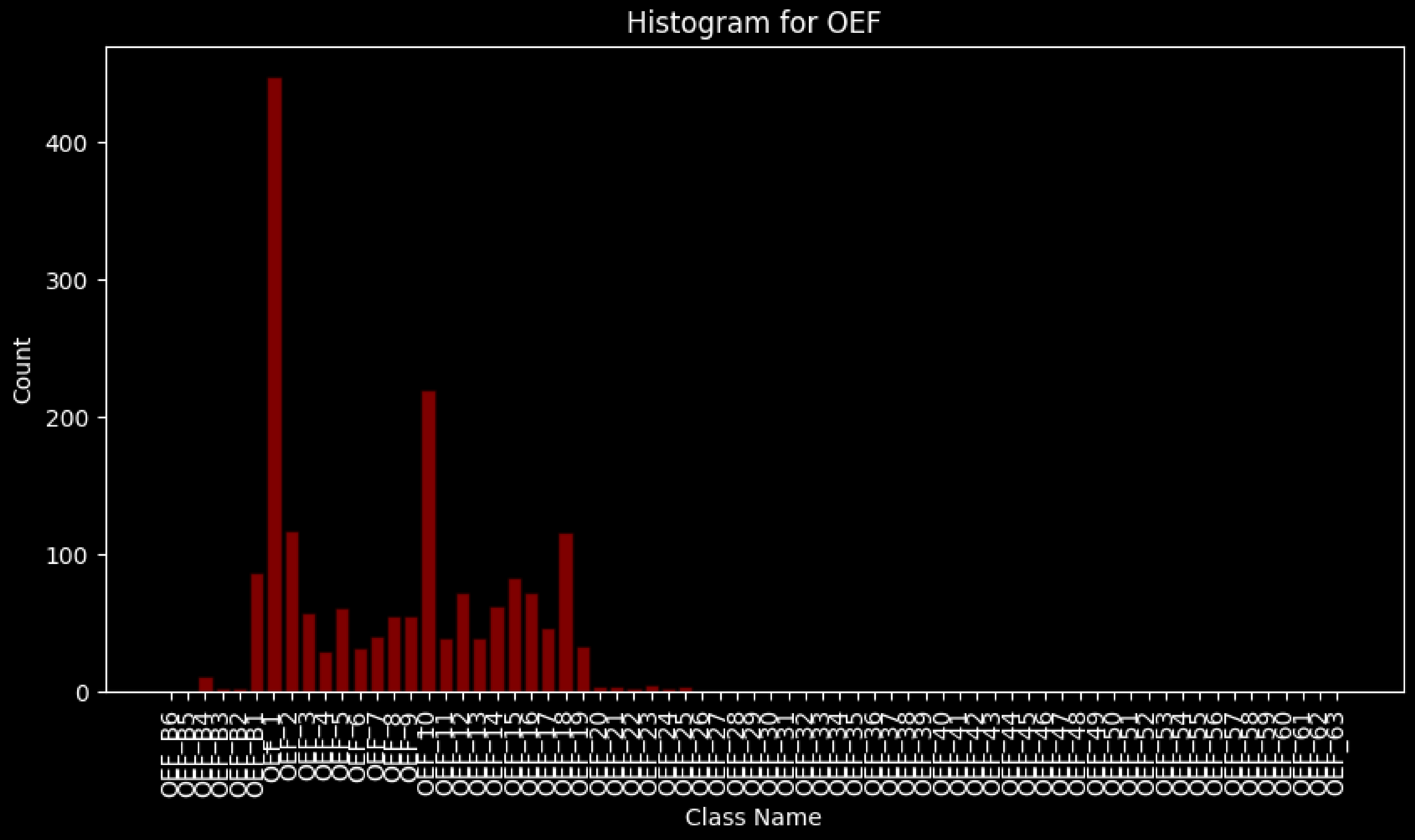
5.1 Dataset statistics

ROBOT INTELLIGENCE LAB

Basic statistics for a dataset of 7 places :: Highly unbalanced, even among the few classes.

Outdoor Elevator Direction/Floor



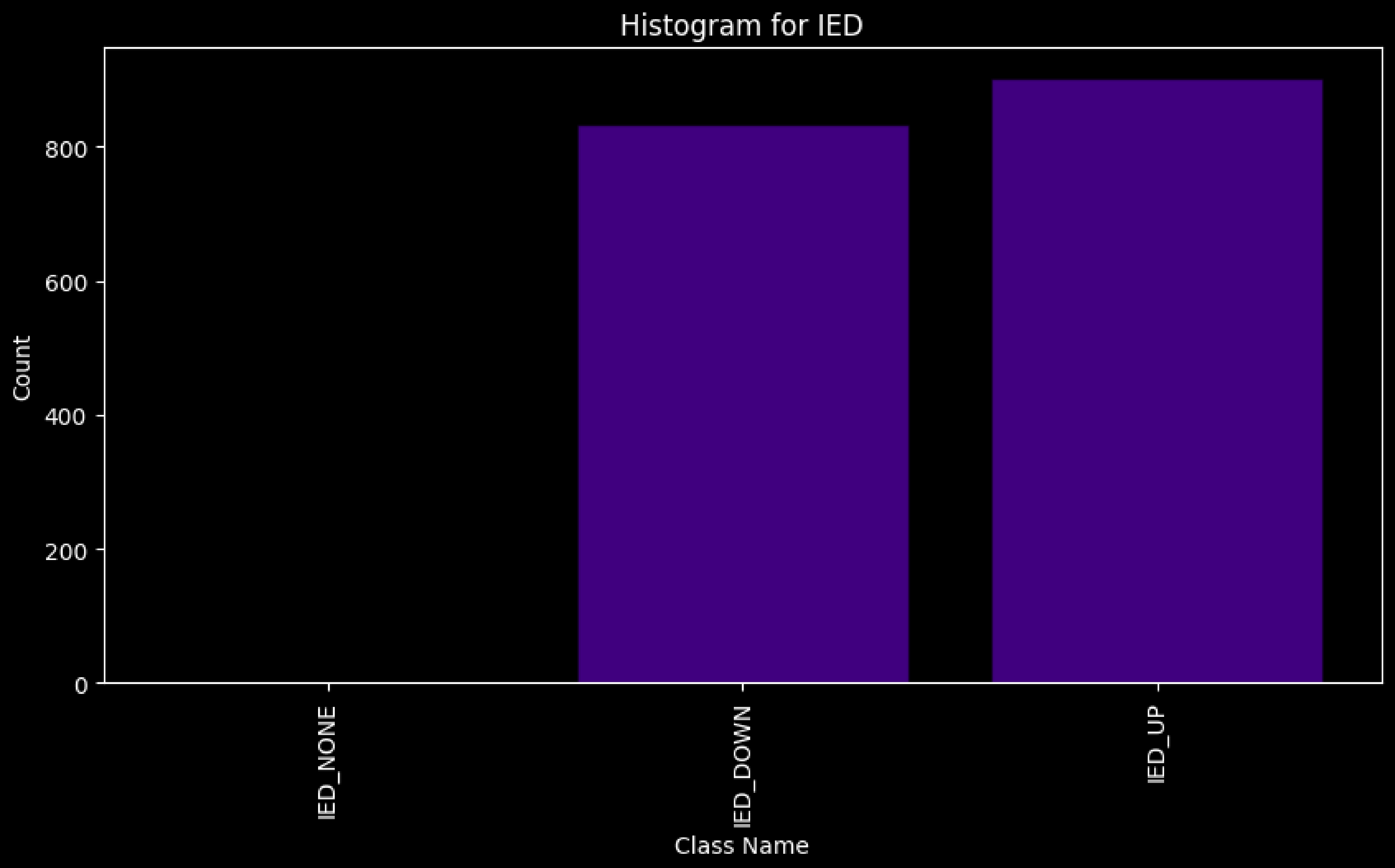


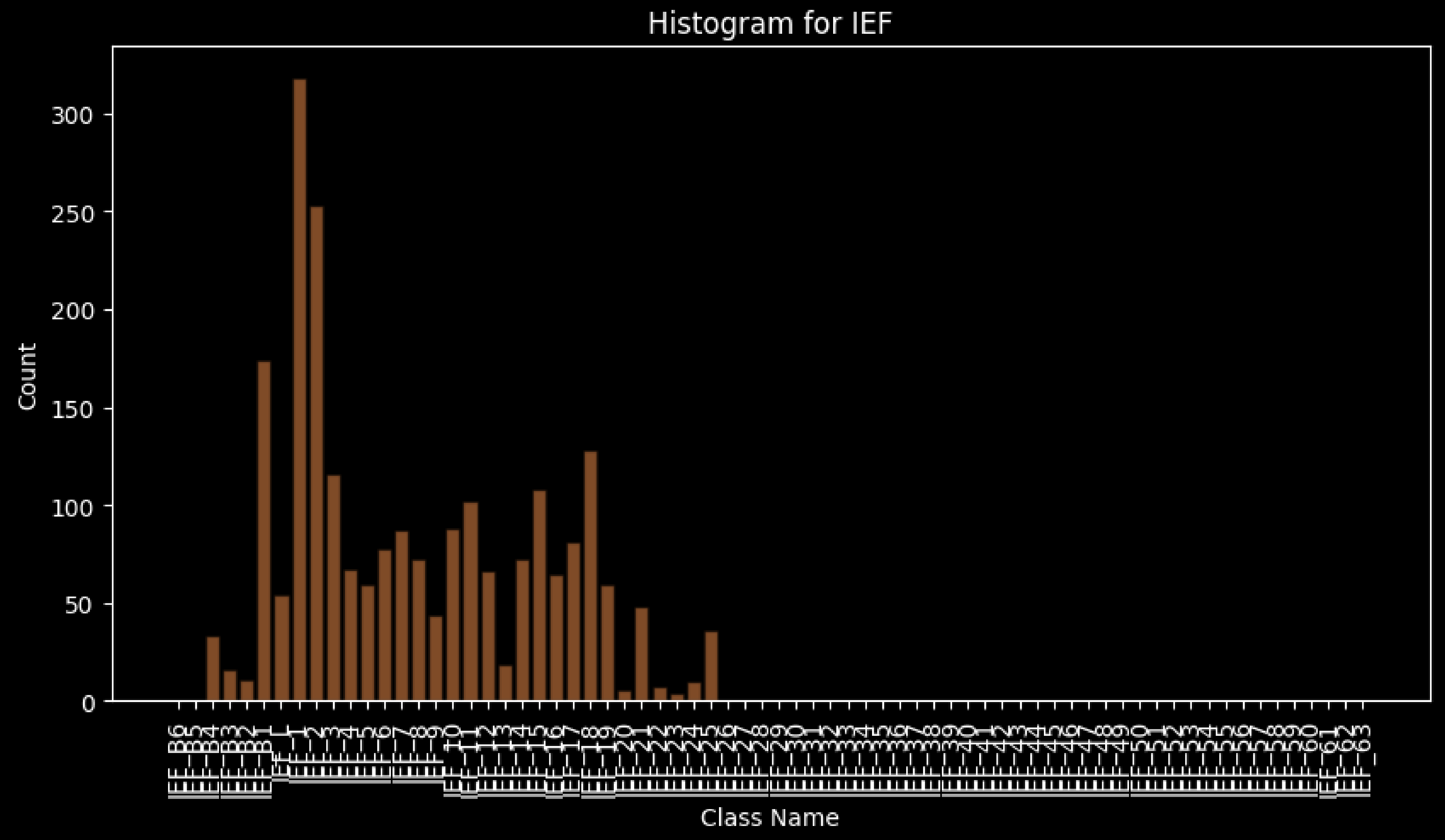
Outside the elevator, the indicator for the floor and direction of the elevator also has a severe label imbalance.

5.1 Dataset statistics

Basic statistics for a dataset of 7 places :: Highly unbalanced, even among the few classes.

Indoor Elevator Direction/Floor





In an elevator, the indicator for the floor and direction of the elevator also has a severe label imbalance.

5.2 Diffusion Inpainting

What about **inpainting** the class with a diffusion model?

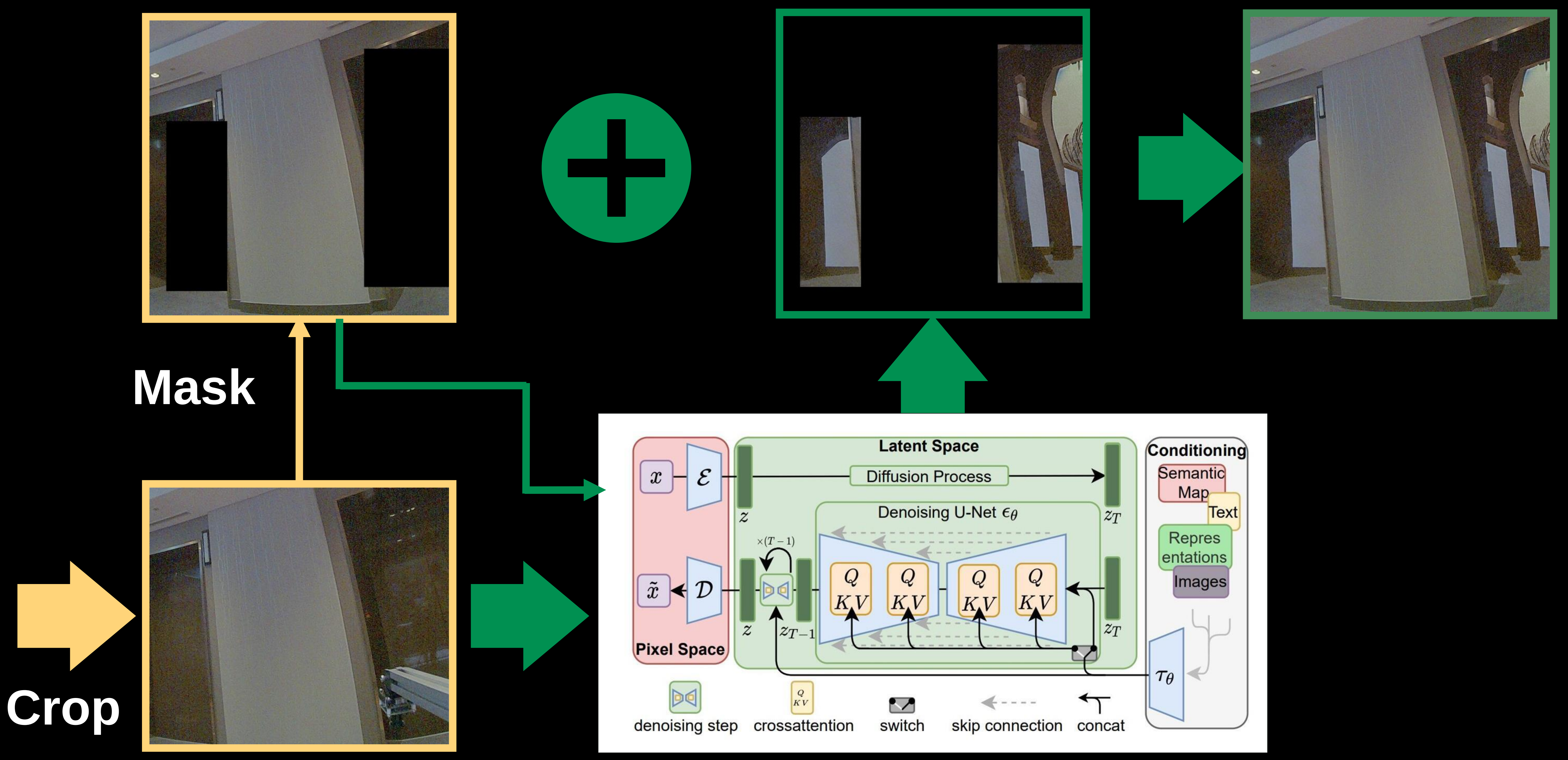
Select a target class and click
Get the Crop Image for that class

Crop

5.2 Diffusion Inpainting

What about **inpainting** the class with a diffusion model?

Configure the Mask as a Bounding Box



③ **Inpainting using the Diffusion Model**

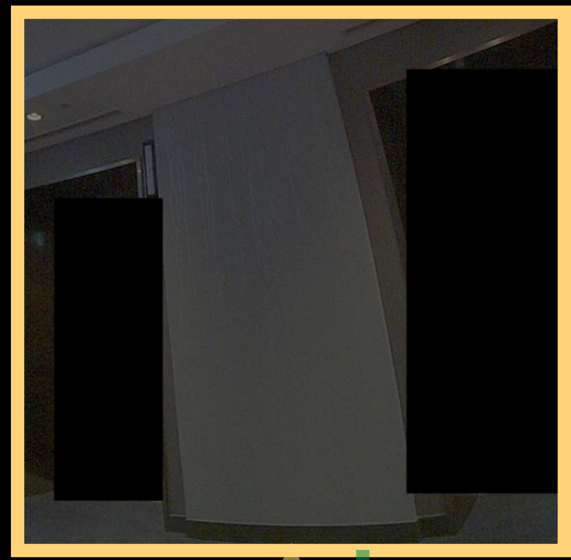
5.2 Diffusion Inpainting

ROBOTIS

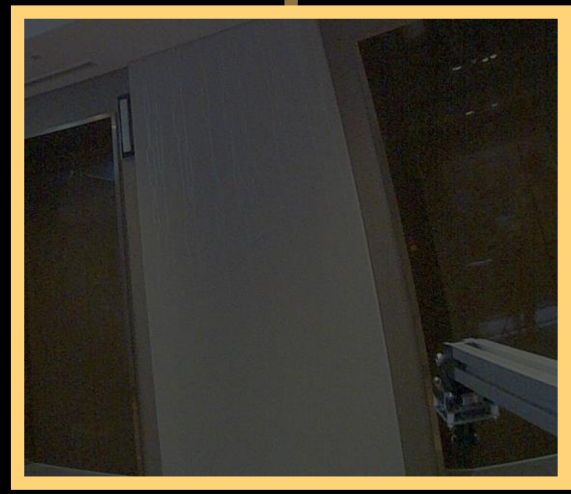
ROBOT INTELLIGENCE LAB

④ Complete the new imag

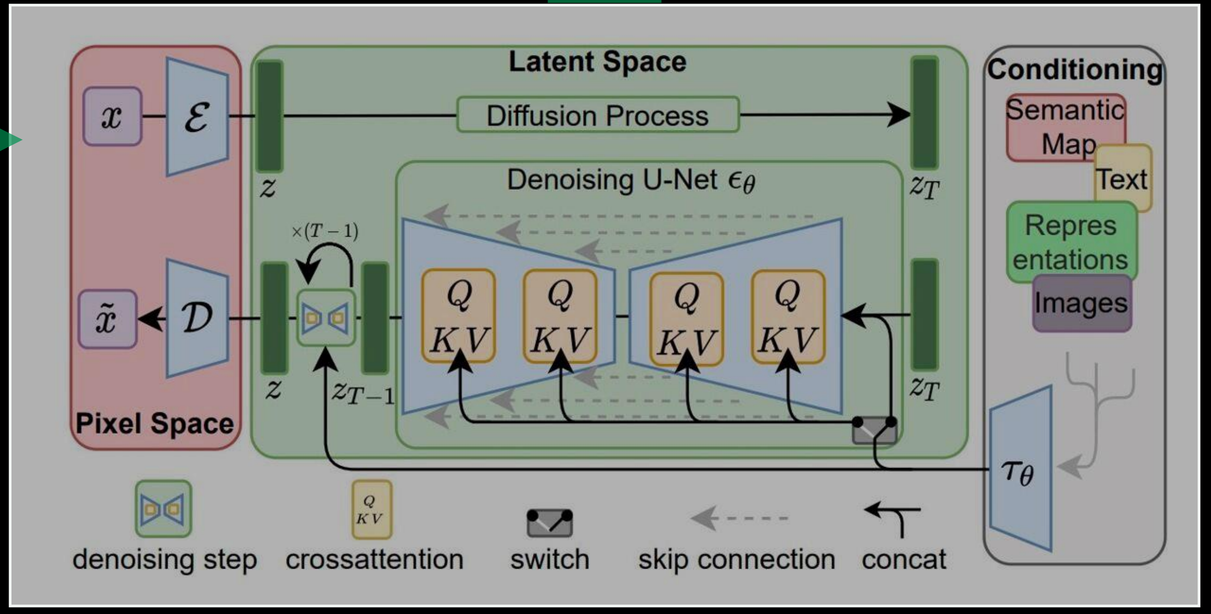
What about **inpainting** the class with a diffusion model?



Mask



Crop



5.2 Diffusion Inpainting

Can the floor plan generation model do a good job?

Prompt : "The photo of elevator floor the display indicating the elevator floor of 6 floor"

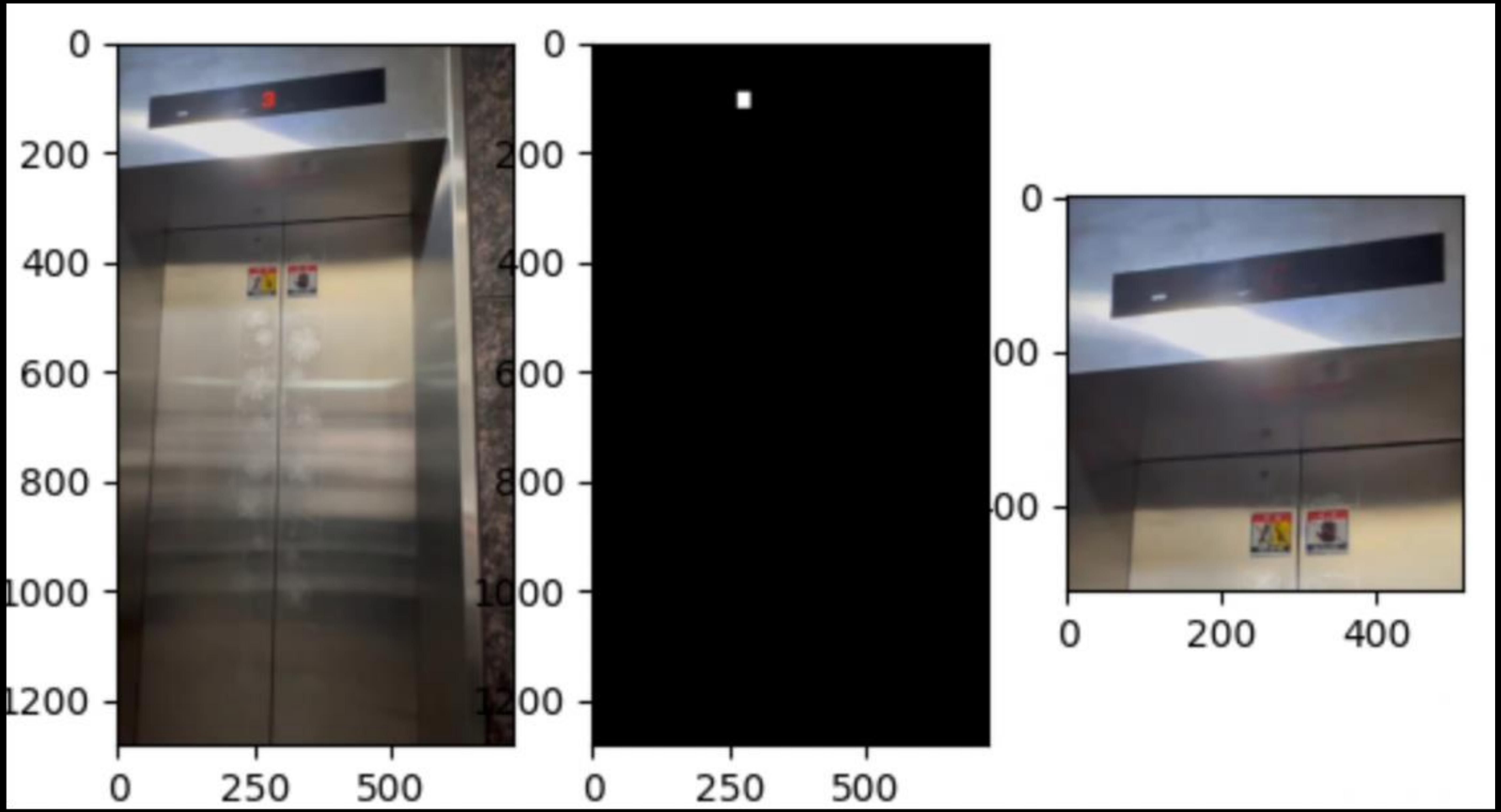


Image Mask inpainted

5.2 Diffusion Inpainting

ROBOTIS

ROBOT INTELLIGENCE LAB

Can the floor plan generation model do a good job?

Prompt : "elevator display digit with 3 floor"

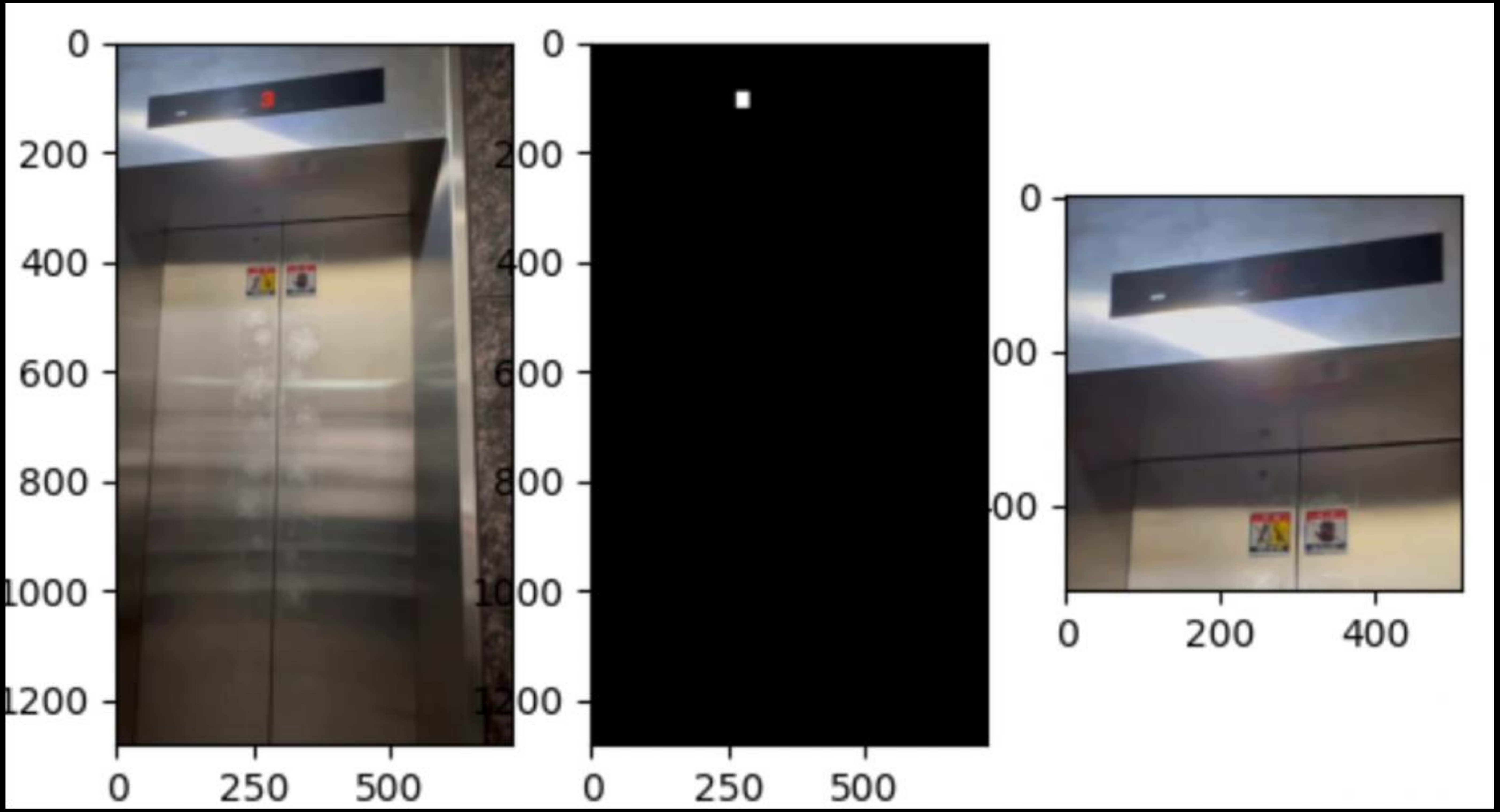


Image **Mask** **inpainted**

Generating letters and numbers is still a bit lacking

5.2 Diffusion Inpainting

ROBOTIS

ROBOT INTELLIGENCE LAB

Additional **performance gains** with Diffusion inpainting

Metric	mAP@0.5	Status Metric	
Test Time Augmentation		x	+Test Time Augmentation + Original Image
Details		Precision/Recall/F1	
YOLOv7	0.730	0.813/0.881/0.843	0.602/0.934/0.736
YOLOv7 (+Patch Aug.)	0.784	0.878/0.792/0.833	0.767/0.961/0.853
YOLOv7 (+Patch Aug. + Diffusion Inpainting)	0.779	0.898/0.771/0.827	0.792/0.956/0.867

ROBOTIS

ROBOT INTELLIGENCE LAB
ROBOT INTELLIGENCE LAB

6. Finetuning

6.1 Fine-tuning

ROBOTIS

ROBOT INTELLIGENCE LAB

Can a model trained on seven locations be effectively applied to other locations?

Pretraining	7 Place Model	**7 Place Model**	COCO	7 Place Model	**7 Place Model**	COCO
Fine-tuning		(Ananti, Henna, Mayfield)	(Ananti, Henna, Mayfield)		(Ananti, Henna, Mayfield)	(Ananti, Henna, Mayfield)
	Precision			Recall		
Ananti	0.531	0.534	0.123	0.509	**0.887**	0.156
Henna	0.581	0.626	0.146	0.749	**0.922**	0.276
Mayfield	0.438	0.515	0.208	0.670	**0.877**	0.366

6.1 Fine-tuning

ROBOTIS

ROBOT INTELLIGENCE LAB

Can a model trained on seven locations be effectively applied to other locations?

1. The model trained with 7 locations performs significantly better than COCO and
2. With 7 locations, training data performs better with new locations
3. Fine-tuning for new places improves performance. (1000 or so)

   3.1 Consistently better performance for all venues (Ananti, Henna, Mayfield)

6.1 Fine-tuning

ROBOTIS ROBOT INTELLIGENCE LAB

How much data should you use for fine-tuning?

There is no dramatic performance improvement from 200 to 1000 photos. Overall, there is a small increase in performance as more data is added.

Conclusion

ROBOT INTELLIGENCE LAB

1. Develop a recognition package for elevator status recognition
2. Developing to return to 80FPS to 100FPS in Orin
3. Suggestions for how data can be effectively augmented during training for state recognition in different locations
4. Shows that pretraining is essential for domain adaptation.
5. Developed a "Pretraining Model for Elevator Condition Recognition" that outperformed fine-tuning on public datasets such as COCO by an average of
   4x on public datasets.
6. (1-4) Provide a codebase to configure a system that performs better as more data is added