INTER IIT TECH MEET 11.0

CLOUDPHYSICIAN

TEAM ID - 29

CONTENTS

PROBLEM STATEMENT

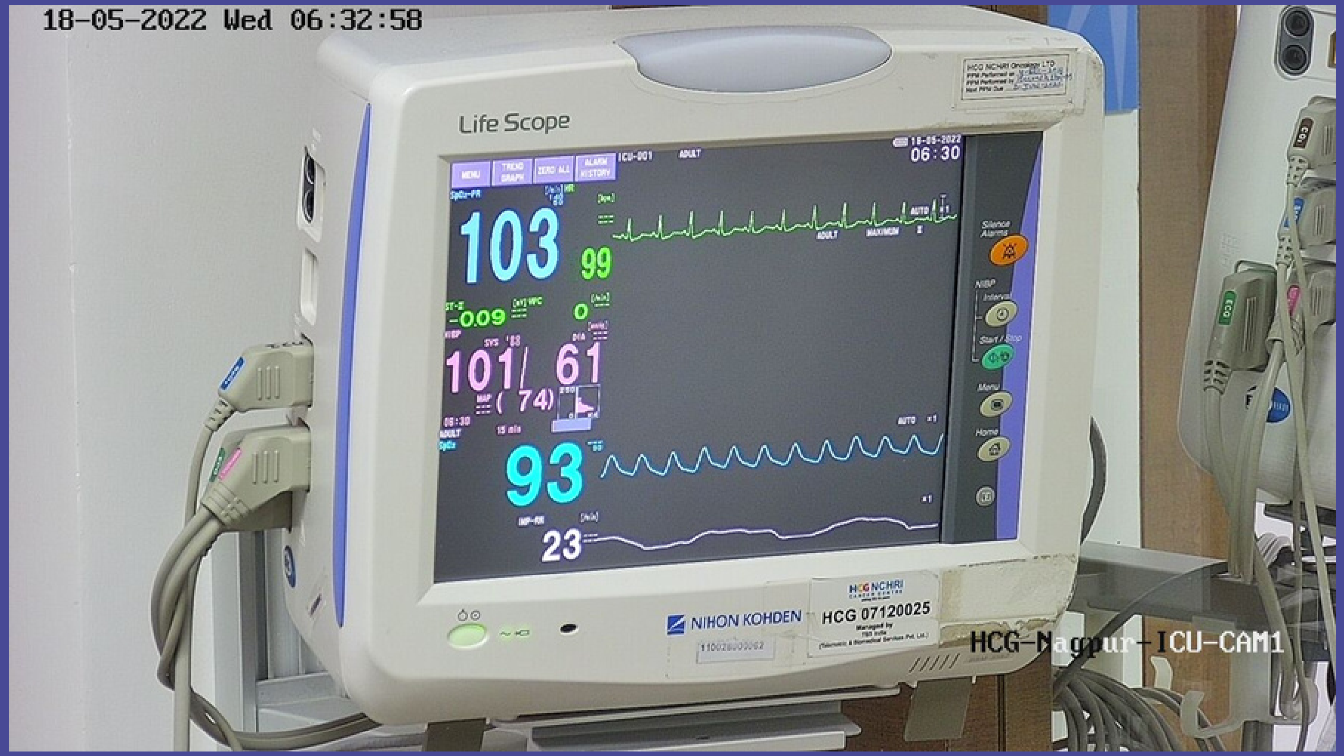
To extract Heart Rate, SpO2, RR, Systolic Blood Pressure, Diastolic Blood Pressure, and MAP from the provided images of monitors.

THE DATASETS

Three types of datasets were provided and used in our project.

LOCALIZATION DATASET

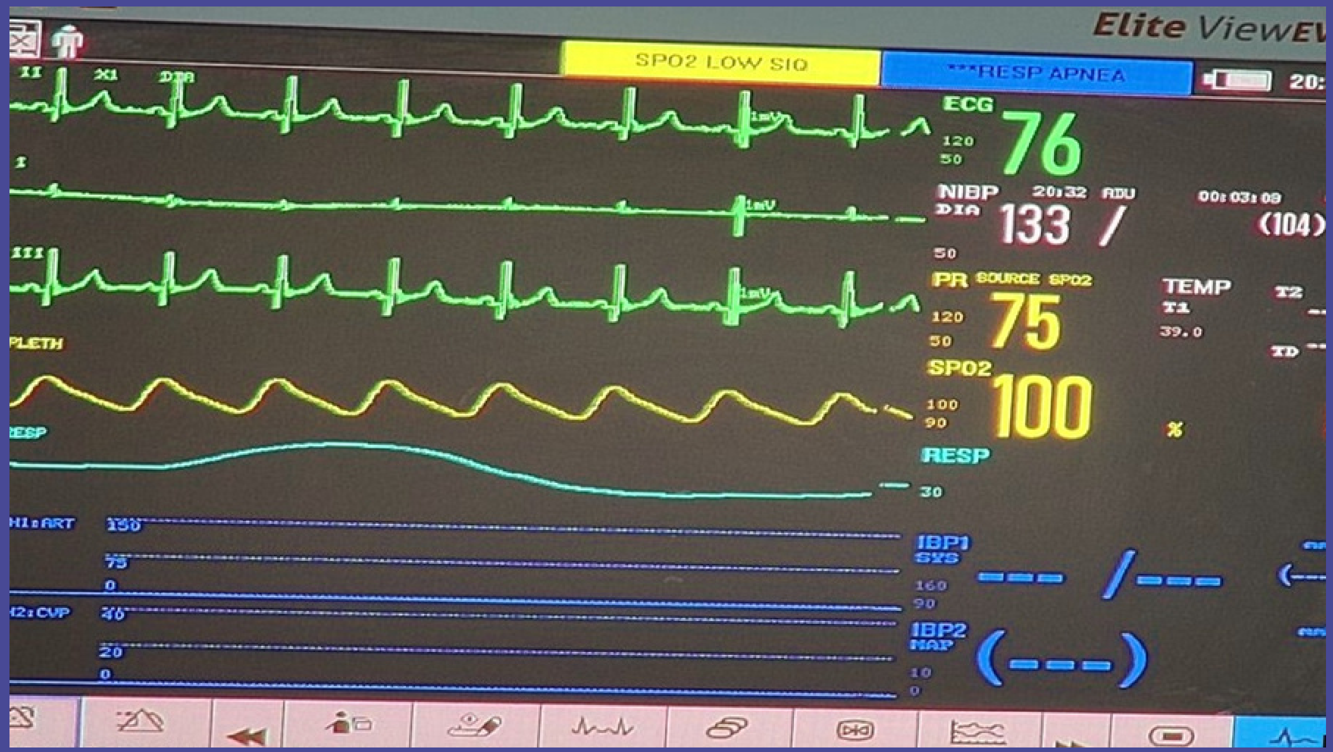
This dataset comprises the images of different monitors. It has 2000 images, the images are used for localising the position of the monitors using a CSV file containing the location of the bounding box of different images.



SAMPLE IMAGE

CLASSIFICATION DATASET

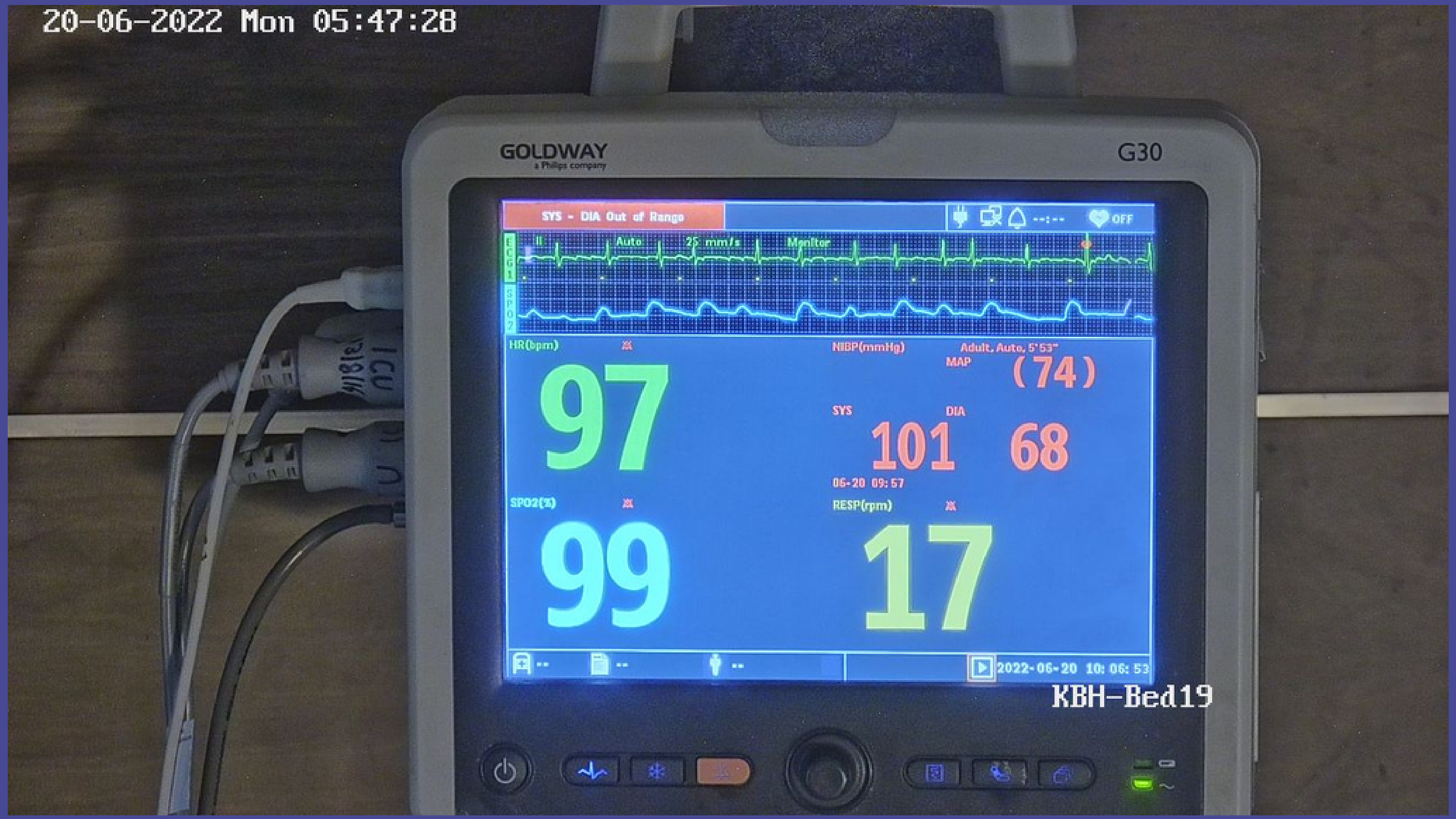
This dataset comprises cropped images of monitors, it has 1000 images belonging to four classes, the images are used for classifying the type of monitors based on the relative position of the vitals present on the screen, it also has a CSV file containing the location of the bounding box of different vitals.



SAMPLE IMAGE

UNLABELLED DATASET

This dataset comprises 7000 unlabelled images of monitors. The images were used for training semi-supervised model in collaboration with the localization dataset which was used for fine-tuning the model.



SAMPLE IMAGE

APPROACH

YoloV7 for getting a zoomed-in image of the screen (object detection), and classifying images (unlabeled) via MobileNetV3 Small. After classifying used pattern-based logic and pre-computed bounding boxes for vital detection with EasyOCR. The graph is digitized by converting the image into binary format and then smoothened using matplotlib.

STEP 01

LOCALIZE, SCALE, REORIENT THE IMAGE

STEP 02

CLASSIFICATION, APPLY OCR, VITALS MAPPING

STEP 03

GRAPH EXTRACTION, DIGITIZATION, PRODUCING OUTPUT

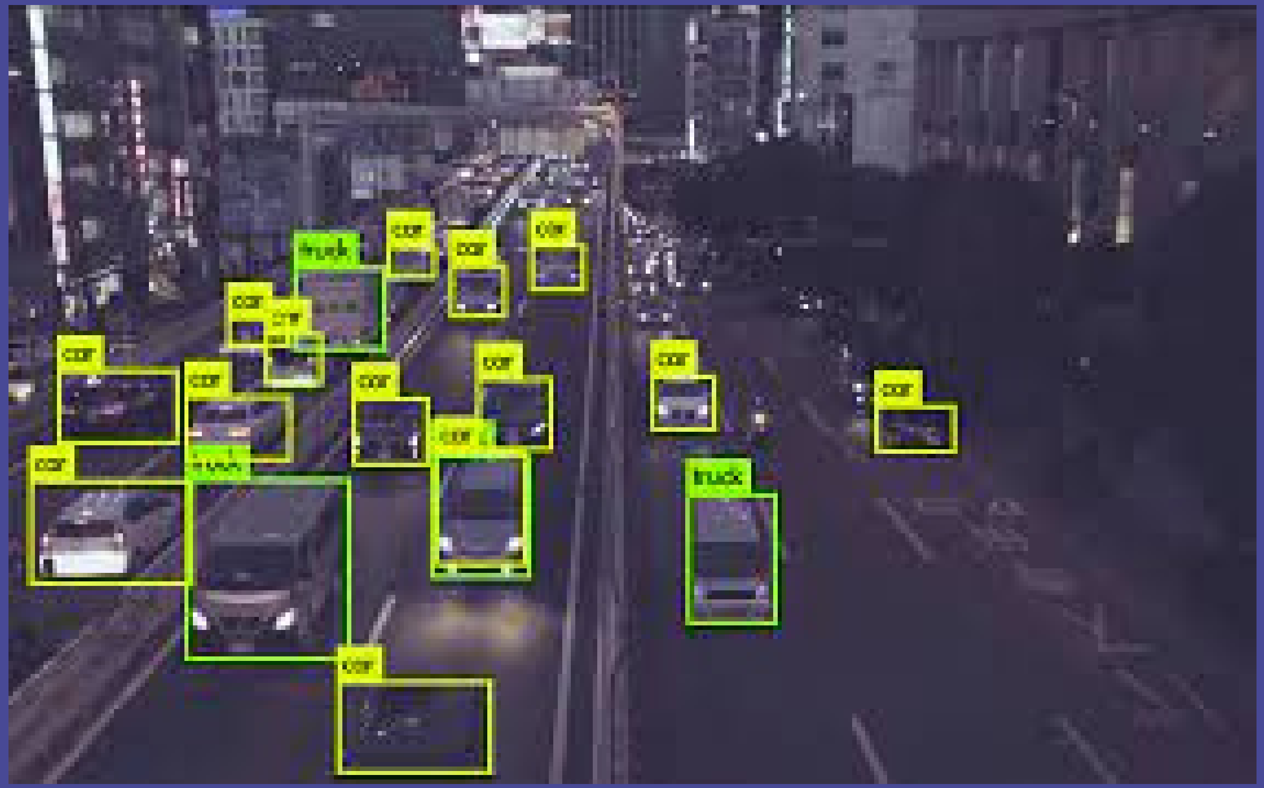
MODELS

Different models used in the pipeline

YOLO V7	MOBILE NET V3 SMALL	EASY OCR	YOLO V5
USED FOR GETTING ZOOMED AND CROPPED IMAGES FOR MONITOR DETECTION	USED FOR CLASSIFYING THE IMAGE IN ONE OF THE TEN CLASSES DEPENDING ON THE POSITION OF VITALS	USED FOR VITAL DETECTION AFTER UTILIZING PRE COMPUTED BOUNDING BOXES	USED FOR GETTING ZOOMED AND CROPPED IMAGES FOR GRAPH DETECTION

YOLO V7

- A real-time object detection model which improves upon previous models by cutting down 50% of inference time.
- Uses "Bag-of-freebies" and "Compound Scaling" to effectively utilize parameters and computation resources.

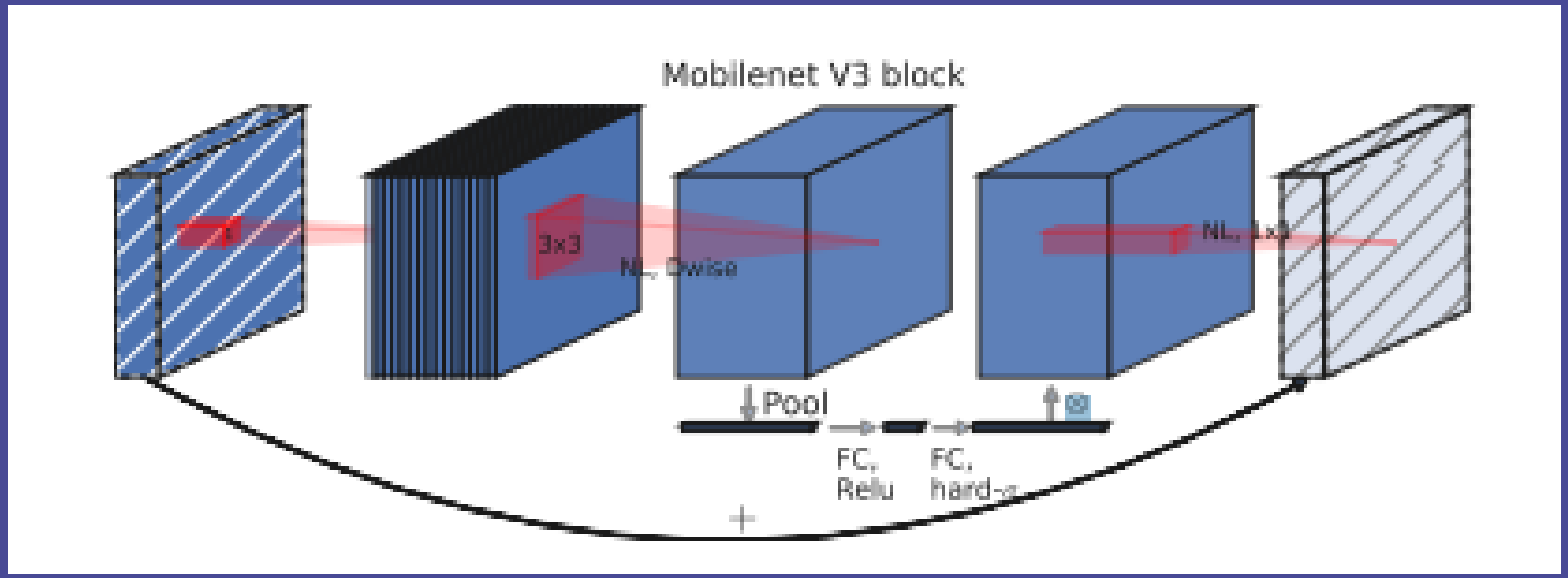


ACTIVATION : SiLU
OPTIMIZER : SGD w/ momentum 0.937
LOSS FUNCTION : BINARY CE LOSS
EPOCHS : 10
BATCH SIZE : 4
IMAGE SIZE : 416

MOBILE NET V3 SMALL

- MobileNetV3 is a convolutional neural network that is tuned to mobile phone CPUs through a combination of hardware-aware network architecture search (NAS) , complemented by the NetAdapt algorithm.

- The network design uses "squeeze-and-excitation" modules in the MBConv blocks to reduce the computation time.

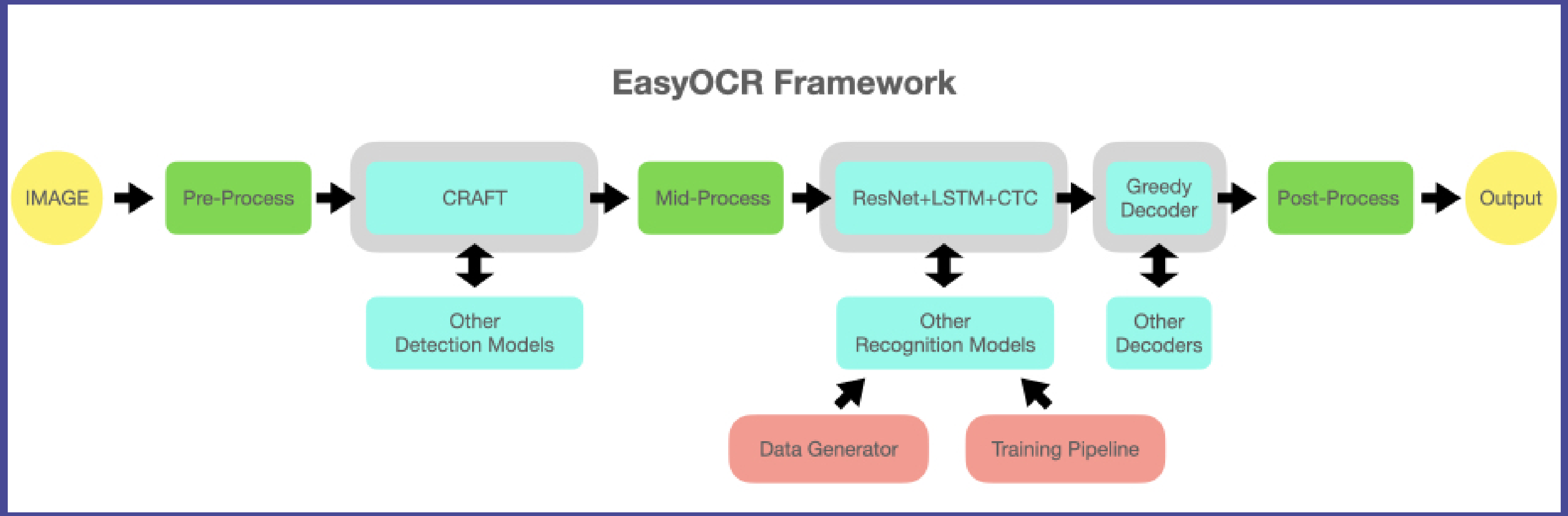


ACTIVATION : Hard Swish
OPTIMIZER : SGD
LOSS FUNCTION : NLL LOSS
EPOCHS : 10
BATCH SIZE : 64
IMAGE SIZE : 256

EASY OCR

- EasyOCR is a python module that can convert different types of documents, such as scanned paper documents, PDF files, or images into editable and searchable data by outputting the text detected in the document.

- Detection execution uses the CRAFT algorithm. The recognition model is a CRNNIt is composed of 3 main components: feature extraction (we are currently using Resnet) and VGG, sequence labeling (LSTM) and decoding (CTC).



PRE PROCESSING :
Gray-scaling the image

YOLO V5

- YOLOv5 is a family of compound-scaled object detection models trained on the COCO dataset.
- A one shot detector model which reduces the computation time.

ACTIVATION : Swish
OPTIMIZER : SGD w/ momentum 0.937
LOSS FUNCTION : BINARY CE LOSS
EPOCHS : 10
BATCH SIZE : 4
IMAGE SIZE : 416

THE PIPELINE

Description of the workflow

Roadmap

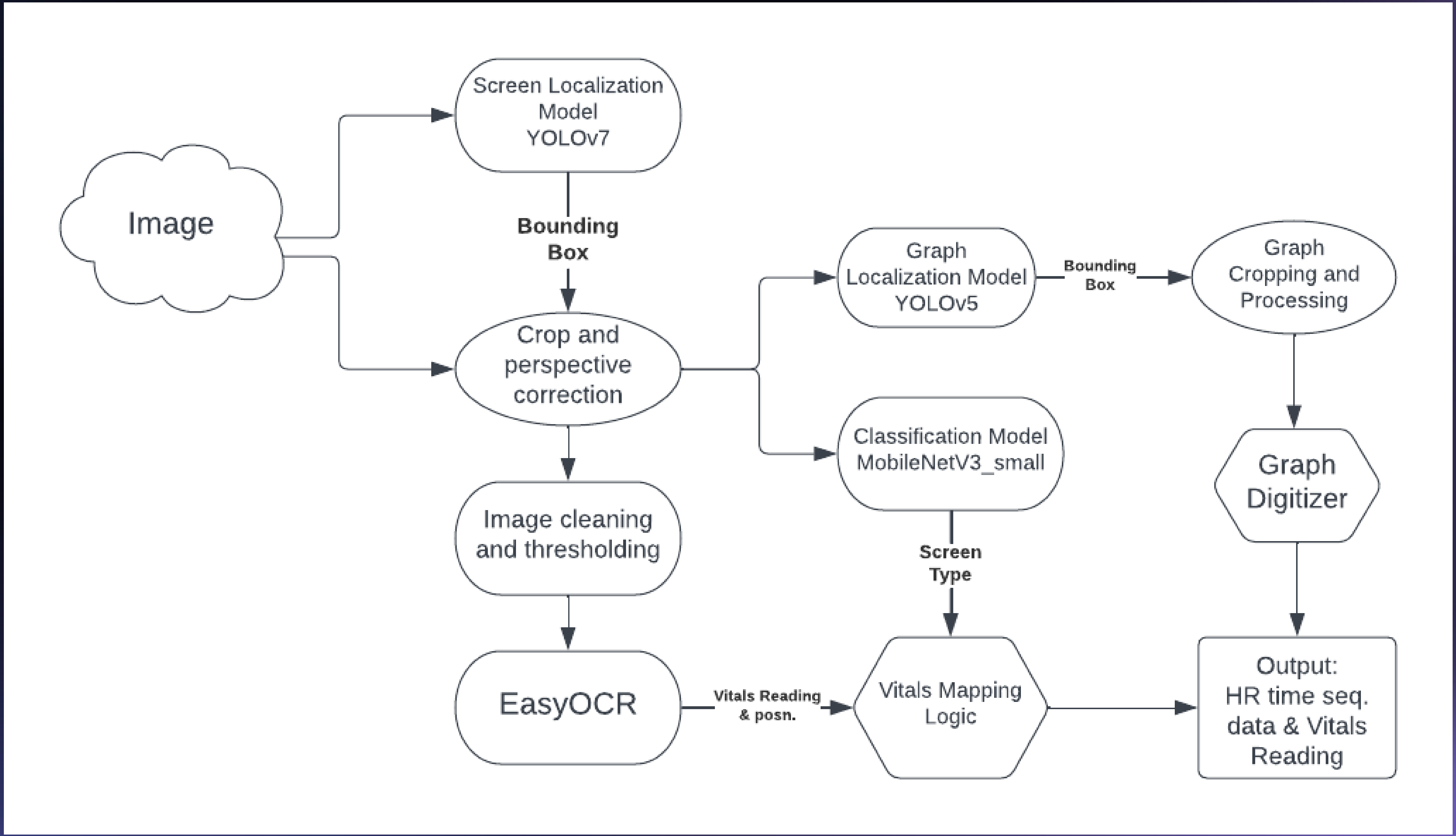
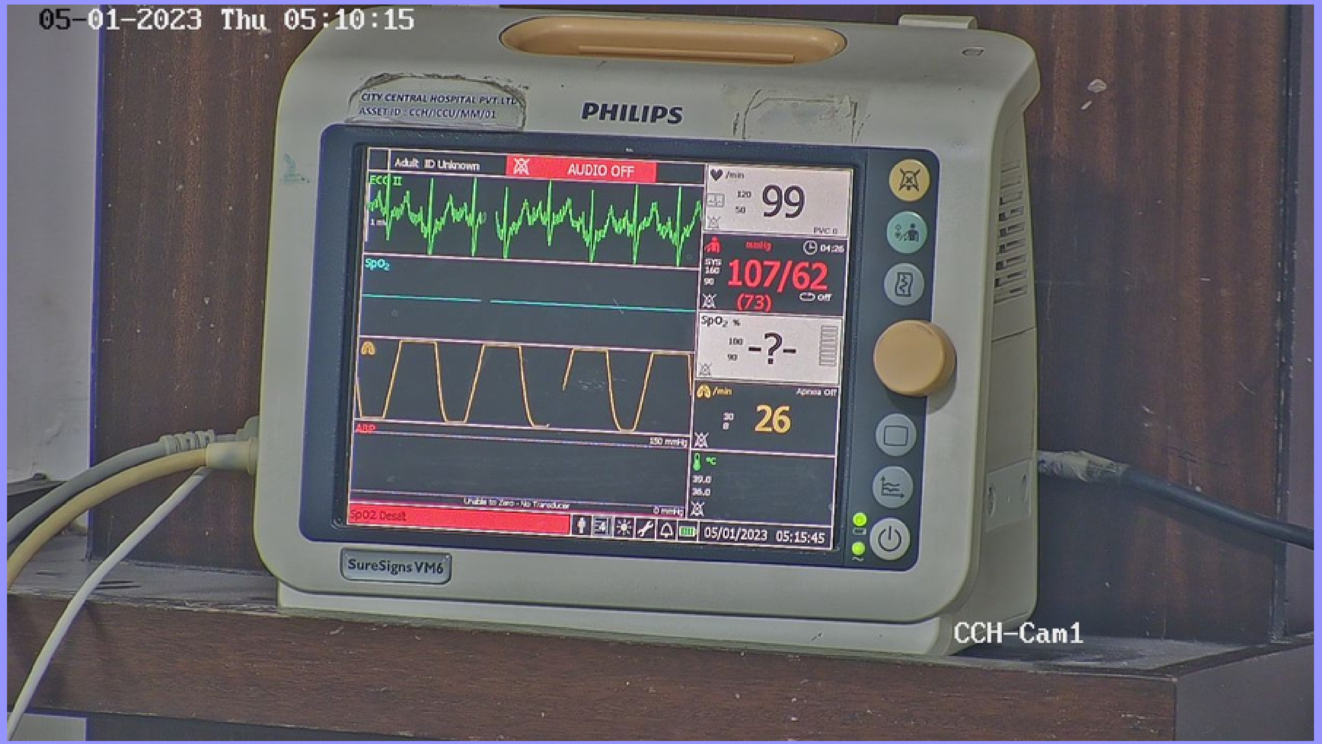


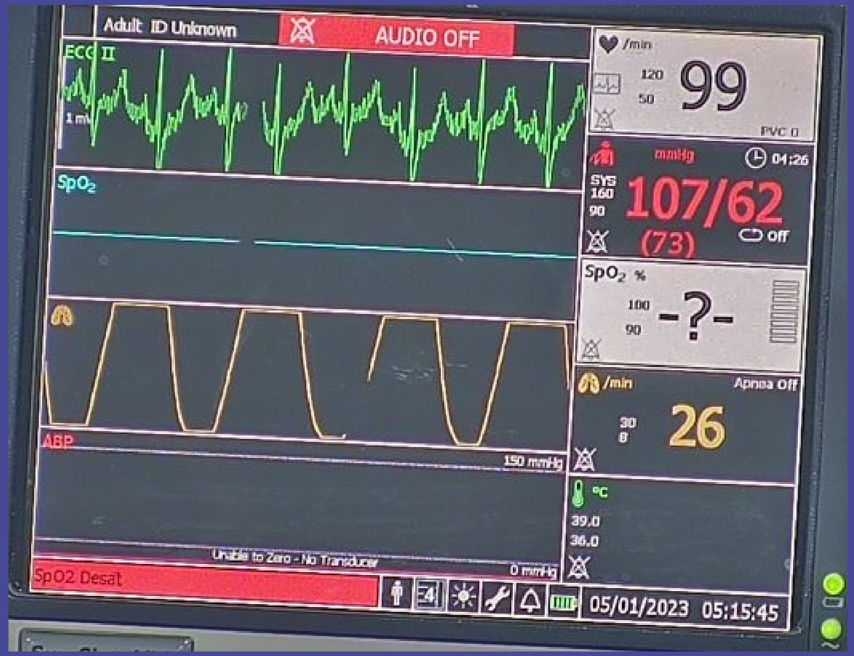
Image
Screen Localization Model YOLOv7
Bounding Box
Crop and perspective correction
Image cleaning and thresholding
EasyOCR
Vitals Reading & posn.
Graph Localization Model YOLOv5
Bounding Box
Graph Cropping and Processing
Classification Model MobileNetV3_small
Screen Type
Vitals Mapping Logic
Graph Digitizer
Output: HR time seq. data & Vitals Reading

Getting Cropped Image

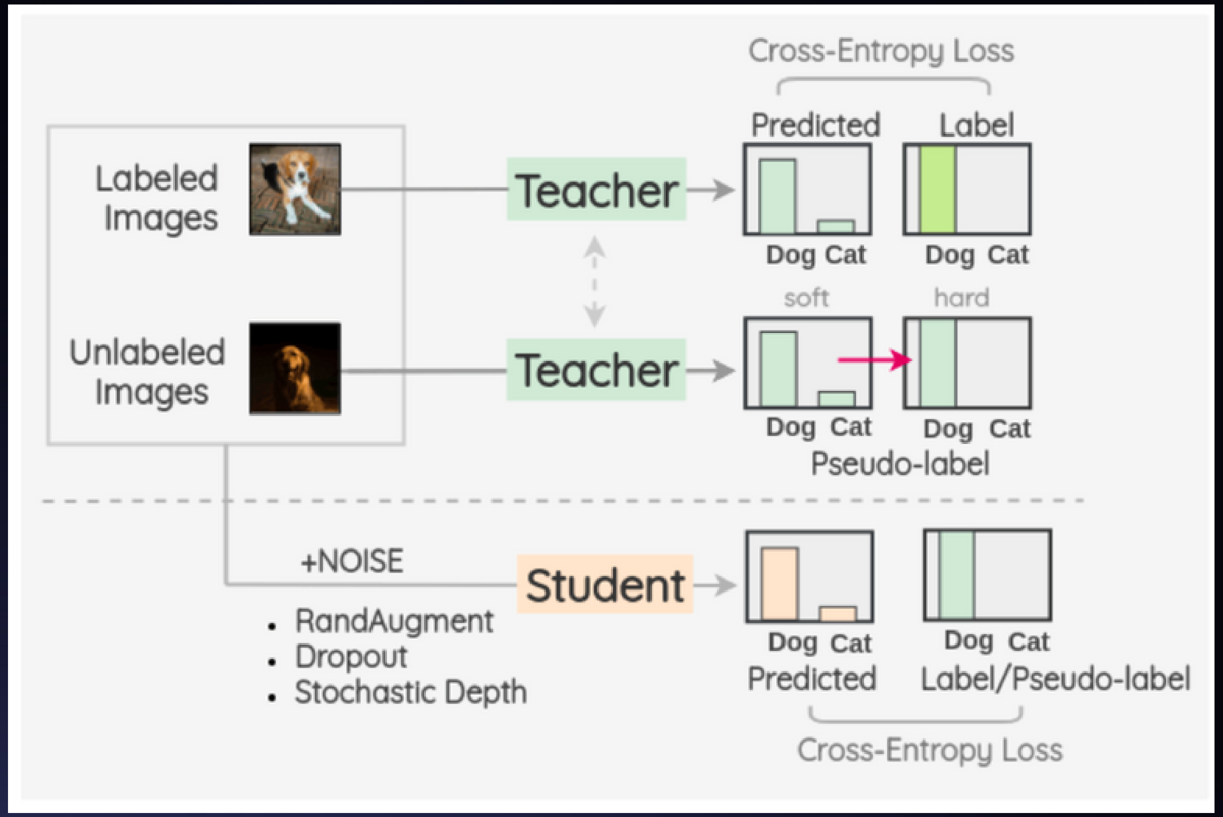
The image was passed through the YOLOv7 detection model to get the screen's bounding box. The screens were skewed in most cases so there was a need to get the four corner points for perspective correction.

We experimented with instance segmentation and polygon object detection (YOLOv3 quad) but abandoned the approach due to low accuracy and relatively high latency.





Classification



The Given Dataset

We found the given dataset lacking since it was not an accurate representation of the actual data that we were provided. The four classes provided did not prove themselves useful to us or our approach.

Some Data Analysis

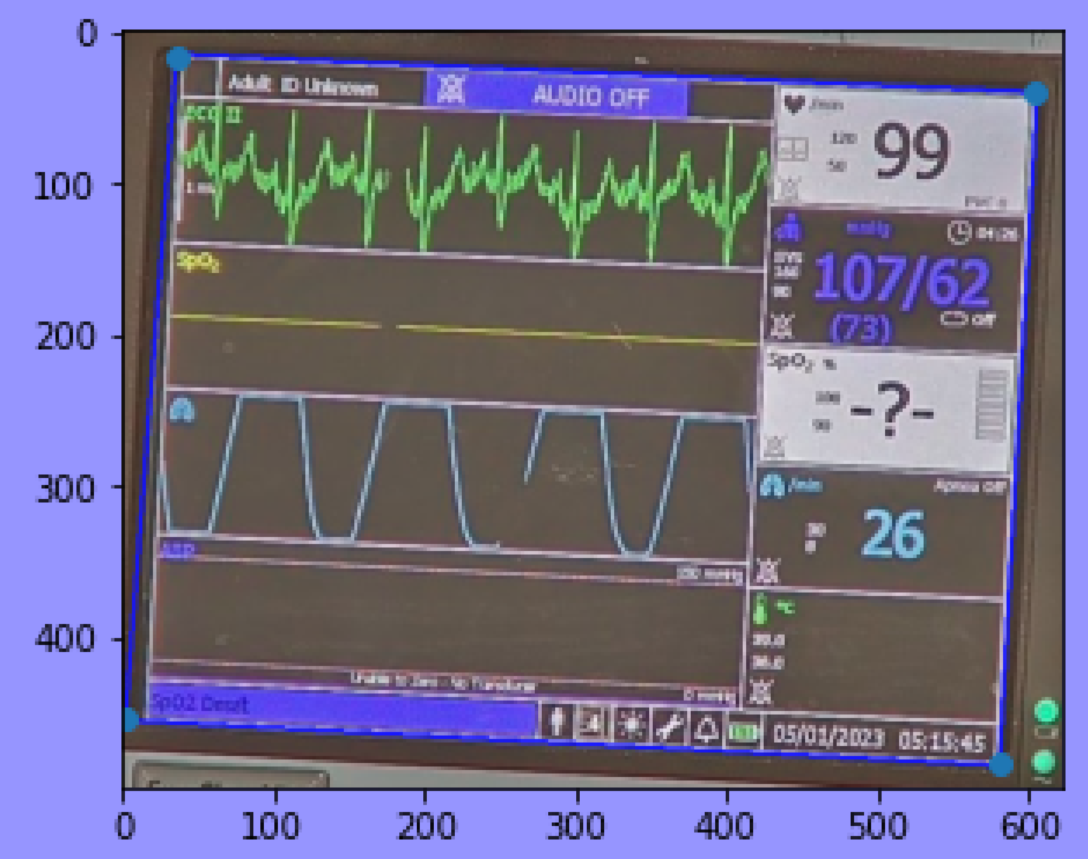
We performed some data analysis over the unlabelled dataset. We found that we required at least 10 different classes to represent the dataset satisfactorily.

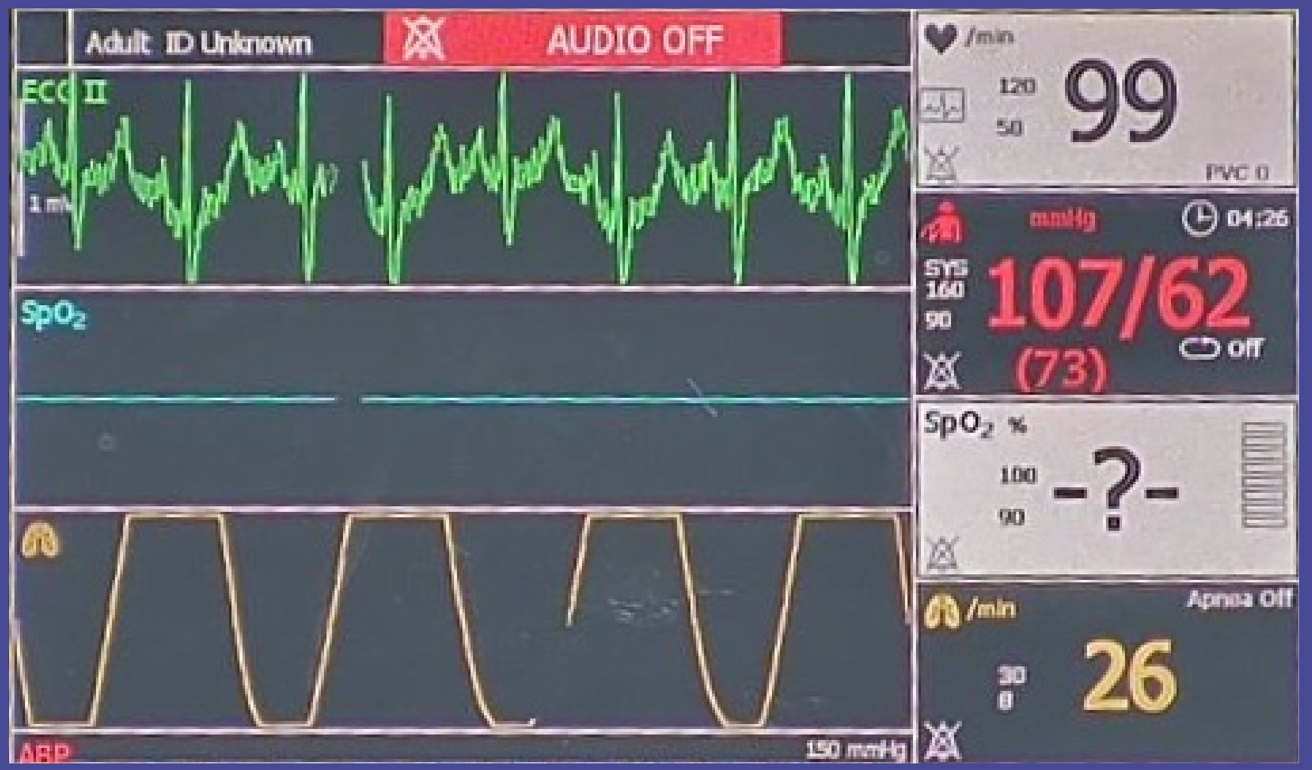
A Semi-Supervised Approach

A semi-supervised deep learning method using a "Teacher-Student" approach. The "Teacher" model is trained on labelled data, used to generate pseudo-labels for unlabeled data, and the combined data is used to train a "Student" model with techniques like Dropout and Stochastic Depth. This process is repeated three times, with the final "Student" becoming the new "Teacher". A light model (Mobilenetv3 small) is then trained on the complete data.

Perspective Correction

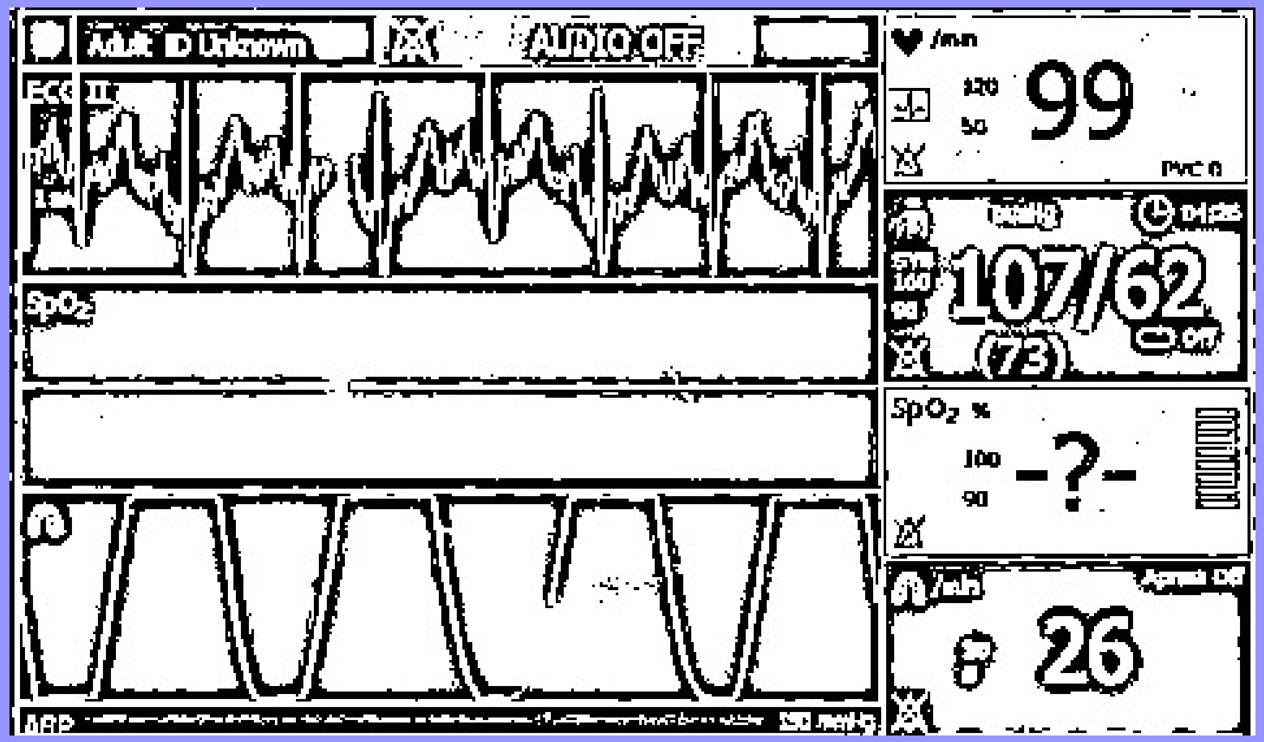
Canny edge detector was used to get the edges. The horizontal and vertical lines were separated and thickened. A connected component was run on each part to get the bounding box for each component and the overlap was considered as a candidate for corners. Heuristics were used to choose the four most likely candidates and passed to the four-point perspective transform function.

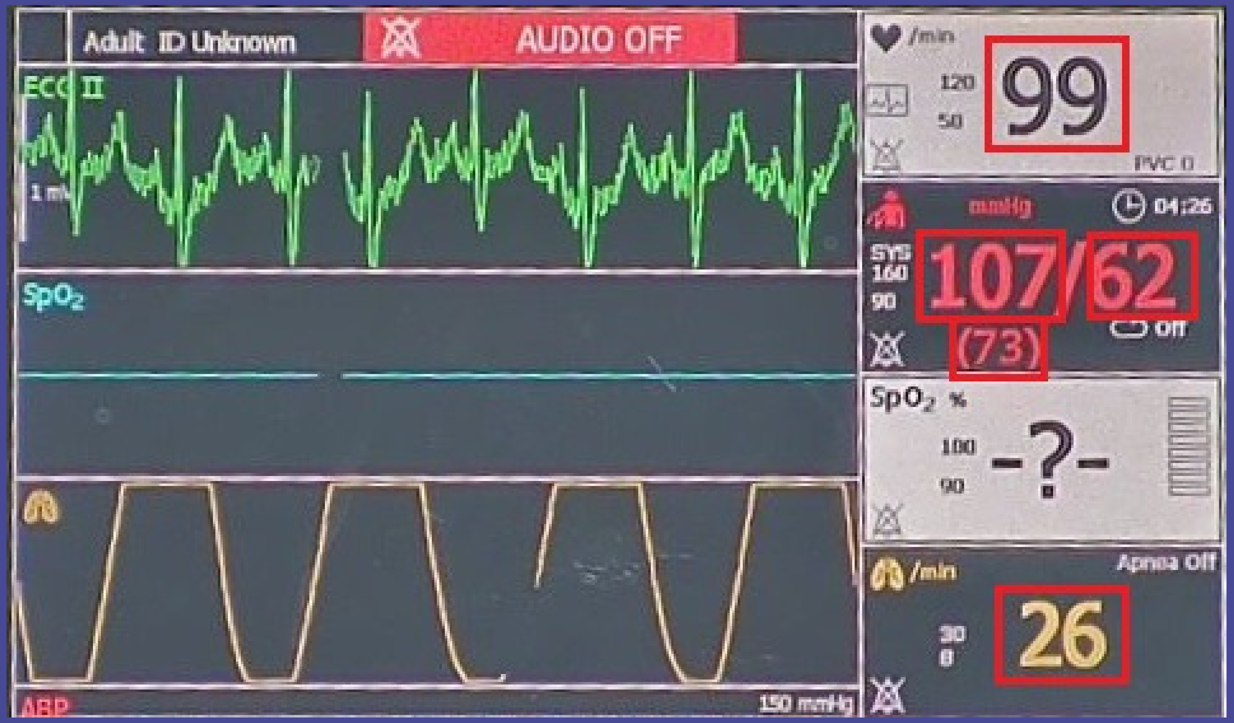




Optical Character Recognition

EasyOCR was used to read the texts from the screen and unnecessary strings were filtered out. Pattern-based logic involving value range, font size, colour and proximity to key texts was used to map the readings to their respective vitals.

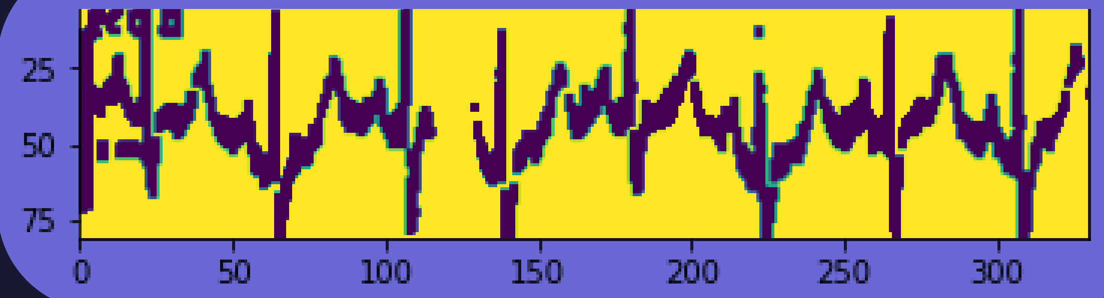


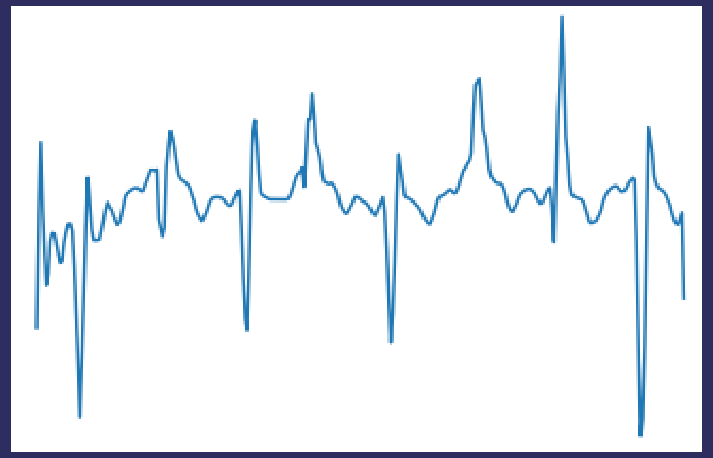


•••

Graph Digitization

For Graph Digitization, the cropped graph image obtained from the YOLOv5 model, was read in the grayscale format. After that, Gaussian Blurring with a kernel size of 5 was applied which was followed by thresholding on the inverted binary image.
The binary image underwent opening to get better details.
For each column, the first row from top and bottom was found where the pixel was white and the average was considered as the column value for the row. The series data so obtained was smoothened out with the help of the moving average.





...

FEATURES

The features and novelty in our project

Semi-supervised learning

We used a semi-supervised method inspired by Knowledge Distillation called 'Noisy Student'

Graph Extraction

We used a robust approach to extract and display the graphs present on the screens.

Quick vital mapping

Our low latency approach allows our pipeline and model to run on the go. Since we map the vitals using pattern rather than position it allows for a robust model.

Perspective change

With YoloV7 we were able to successfully extract vitals even when the detected monitor was skewed

TECH STACK

01	Python	OpenCV	02
03	PyTorch	EasyOCR	04
05	MobileNetV3	YOLO	06

CITATIONS

PAPERS USED IN THE DEVELOPMENT OF THE MODEL

NOISY STUDENT

https://amitness.com/2020/07/semi-supervised-learning/

EASYOCR

https://github.com/JaidedAI/EasyOCR

MOBILENET V3

https://arxiv.org/abs/1905.02244

YOLO V5

https://arxiv.org/abs/2108.11539