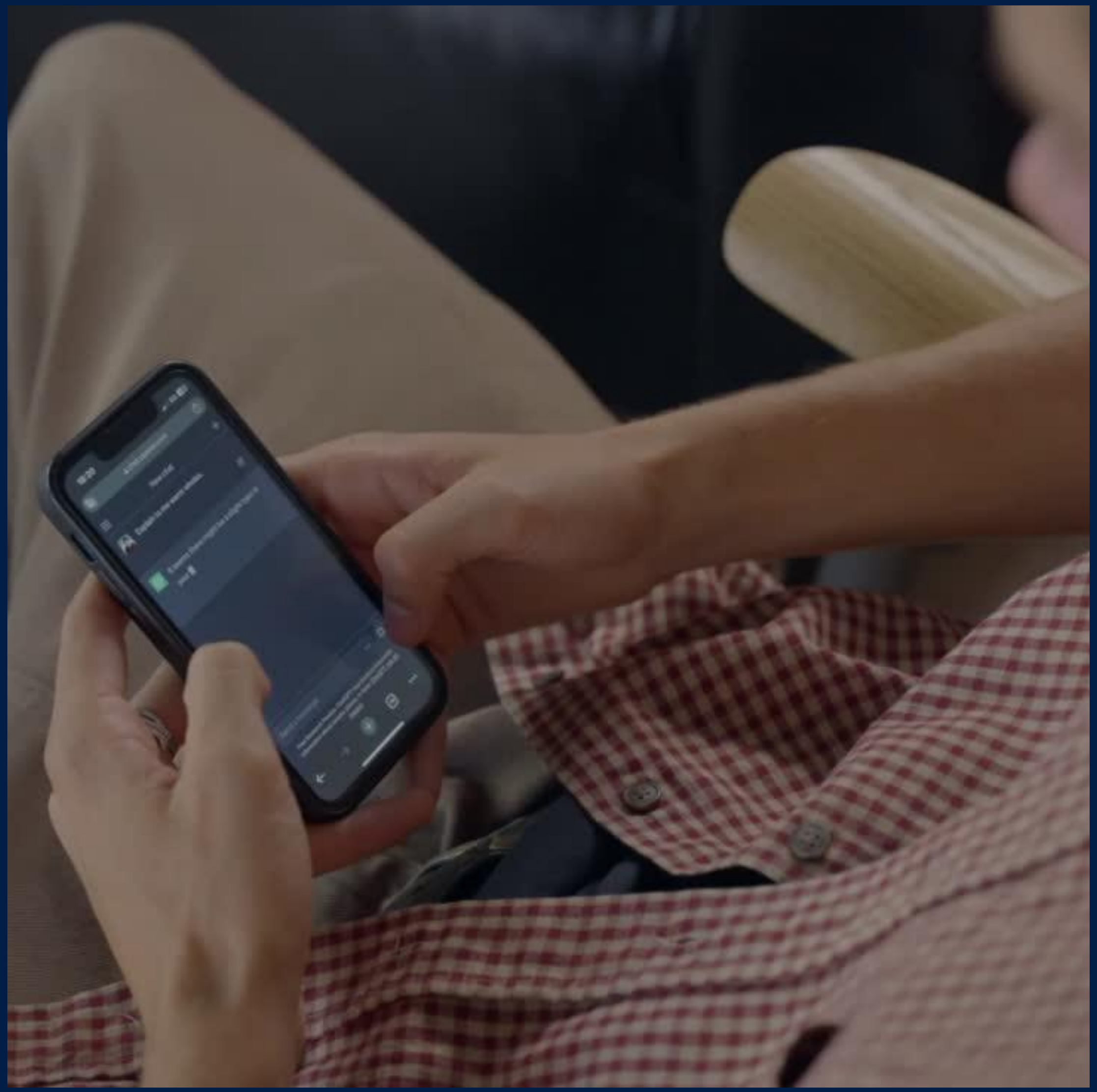
Optimizing Fall Detection Algorithms using Knowledge Distillation (Pixel52)

By: Salsabil Soliman,
Landry Tun, Niccolò Meniconi, Peter Mousses, Quy Hoang Nguyen

CONTENT

PROJECT OVERVIEW

Optimizing fall detection using knowledge distillation or fine tuning



PROBLEM STATEMENT

Many existing medical object detection models are designed to run on cloud-based systems, which introduce latency, require stable internet connections, and raise concerns about data privacy. In resource-constrained environments, such as rural clinics or emergency situations, reliance on cloud computing is impractical.

There is a critical need for a lightweight yet accurate medical object detection model that can run efficiently on edge devices.

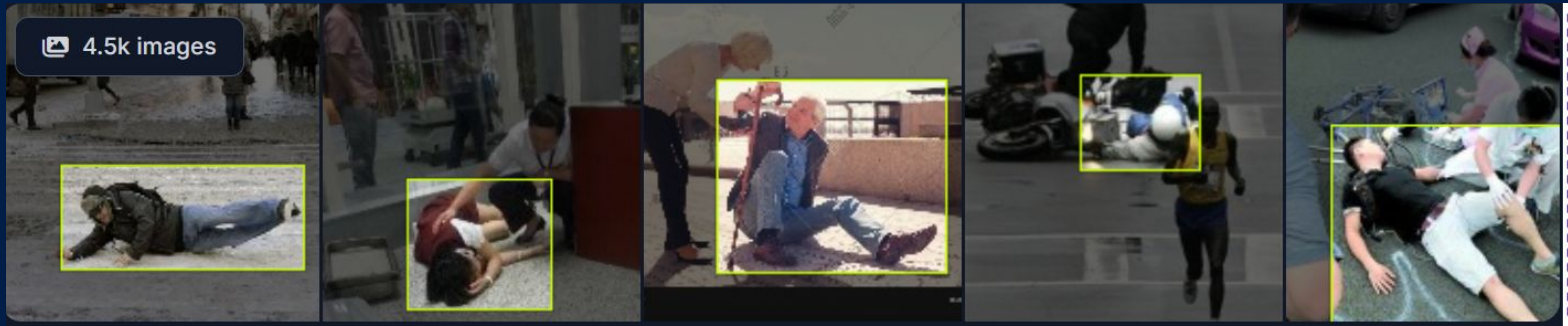
We propose an approach for developing lightweight fall detection model by fine tuning from general purpose object detection model on a fall detection dataset.

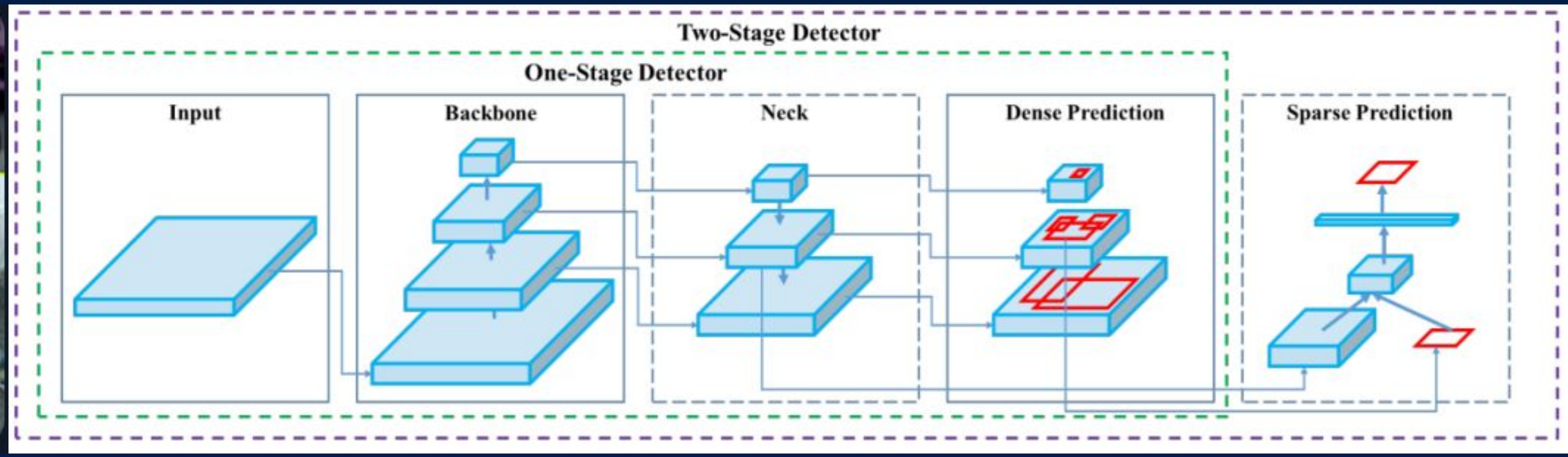
We also explore using knowledge distillation along with quantization to get and optimize rich fall detection embeddings from a generalized object detection model to make it easier for the edge device to run said model.

03

DATA AND METHODOLOGY

https://viso.ai/deep-learning/yolov8-guide/





Data Collection

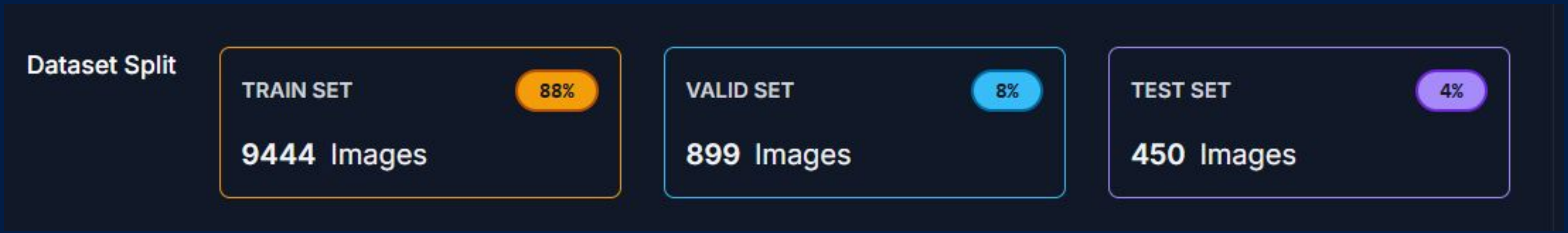
Roboflow: Fall Detection Dataset
Total: 10,388 Images

Pre processing

Data is augmented out of the box

Model Selection

YOLOv8



https://universe.roboflow.com/roboflow-universe-projects/fall-detection-ca3o8/dataset/4

MODEL DEVELOPMENT

Expert models exist for active motion detection from skeletal and video data. These can be distilled to a smaller model used only for fall detection, with richer embeddings. The distillation process is as follows.

1. The teacher weights are not used for backpropagation, while the student weights are. Both models are used without heads, such as a classifier or a detector head.
2. Input data is used for a forward pass on both models.
3. The loss is calculated using L2 or cosine similarity between the student and teacher embeddings
4. Backpropagation is run on the student model, aligning the student's embeddings with the teacher model.

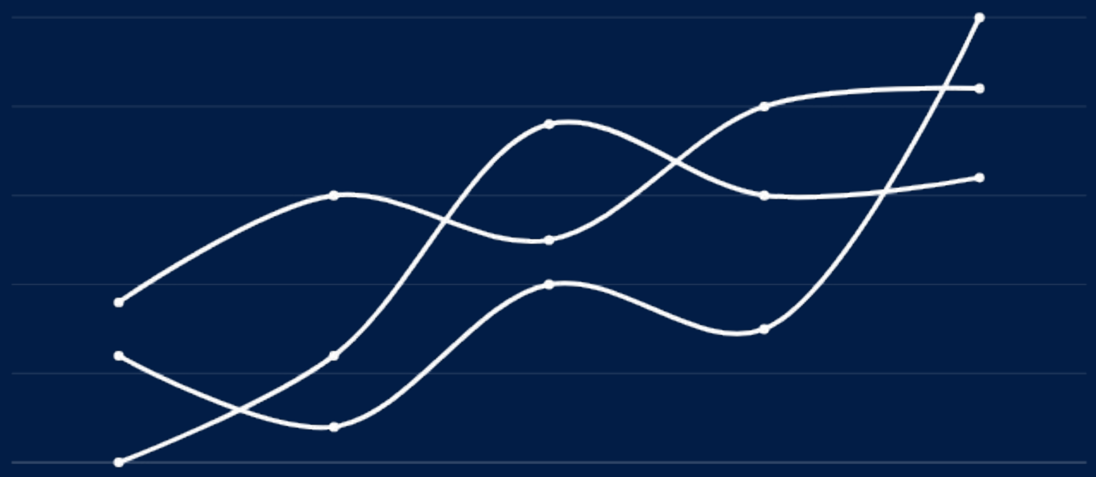
Desired end result:

- Richer embeddings for specialized use-case
- Student model can be more compact, as long as the embedding dimension matches the teacher

EVALUATION METRICS

We define appropriate metrics to assess the performance of our AI models, such as accuracy, precision, recall, F1 score, etc. These metrics help us quantify the effectiveness of our solutions.

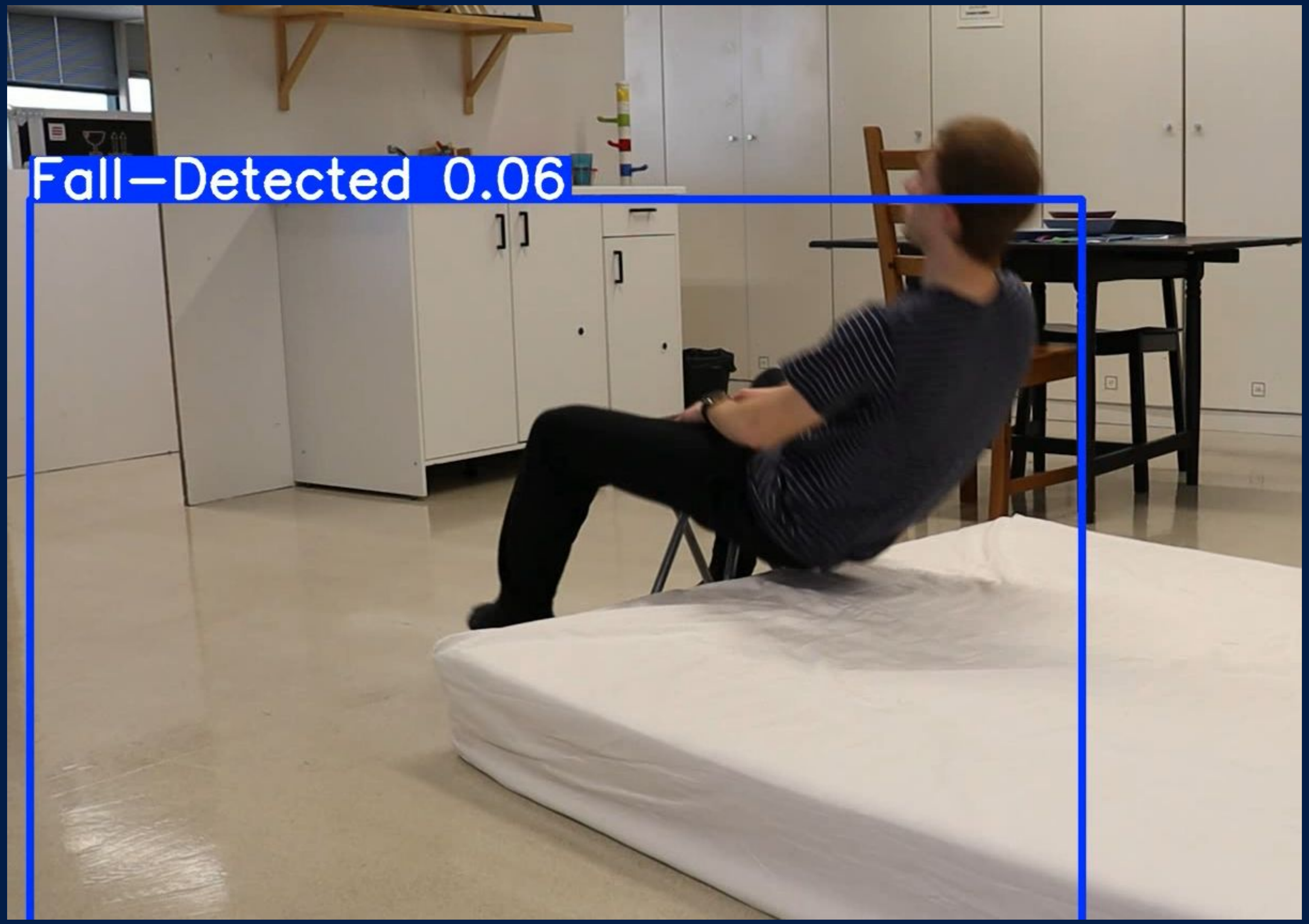
Evaluation Metrics	Training Accuracy	Test Accuracy	Precision	Recall	F-1 Score
Teacher Model	–%	–%	–	–	–
Student Model	–%	–%	–	–	–



RESULT AND IMPACT

Performs well on image dataset
Needs more further training to improve domain generalizability

https://universe.roboflow.com/roboflow-universe-projects/fall
-detection-ca3o8/dataset/4



https://github.com/joaojtmarques/WEDA-FALL/blob/main/README.md

FUTURE DIRECTIONS

- Next steps in the project would include a comprehensive evaluation of the model performance, across different kinds of fall datasets
    - Required for healthcare applications, since malfunctions can be dangerous
- Future research opportunities include the development of real time fall detection models that can be implemented in edge devices
- Apply KD to optimize a light YOLO model using a fine-tuned, more complex model

THANK YOU