Fast and Accurate Threat Identification with Object Detection

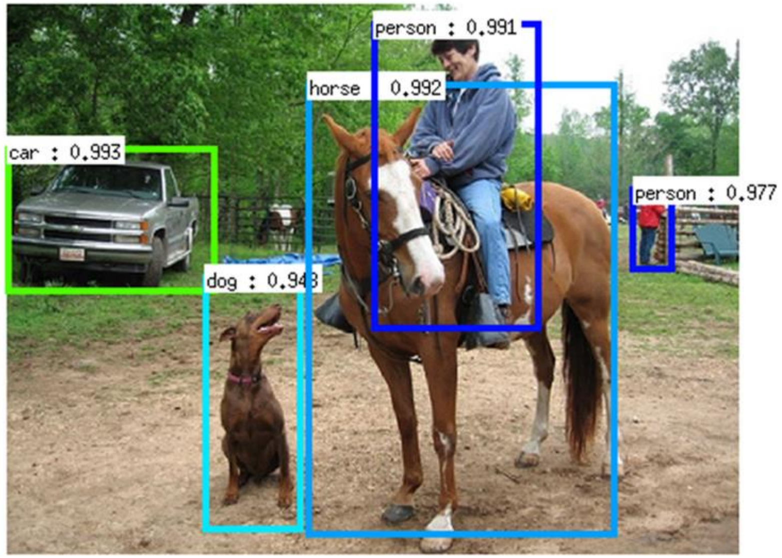
Passenger Screening Algorithm Challenge (Kaggle Competition)

Zachary Teed
(740) 816-8901
zteed@Princeton.edu

Overview

- My approach trains a neural network based object detector to locate and label threats from millimeter wave scanner data.
- Results: 4^{th} place solution (prize winning solution)
- Why did I participate: Summer before starting grad school, this seemed like a very exciting competition. I was interested to see of current object detectors could be applied to this type of data.
- Benefits of participating: I really enjoyed working on a problem which could improve airport safety and efficiency.
- I'm currently a PhD student at Princeton University working in the Vision & Learning lab

Object Detection: Given an input image, locate and classify objects of interest



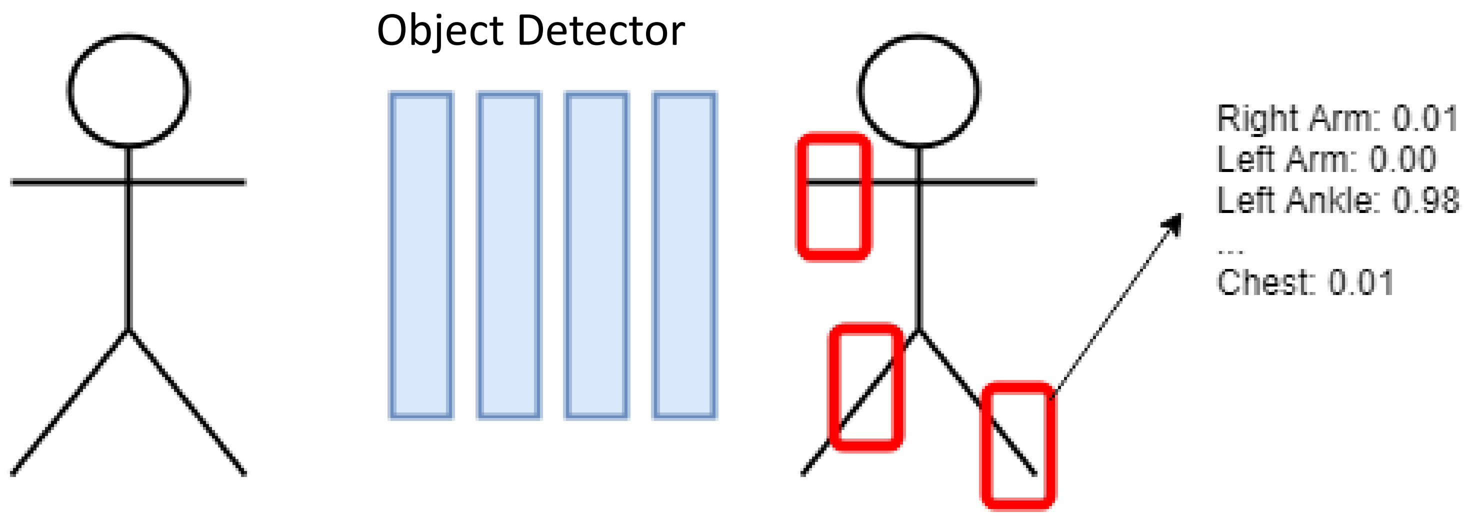
Faster-RCNN, Ren et al.

Motivation:

- CNNs are prone to overfitting, especially given the difficulty to collect this type of data. By providing the additional bounding box of the threat, the model was able to learn from a richer source of supervision.

- Forces the network to focus on regions of interest. This is a strong regularizer and provides some interpretability

Object Detection for Threat Identification

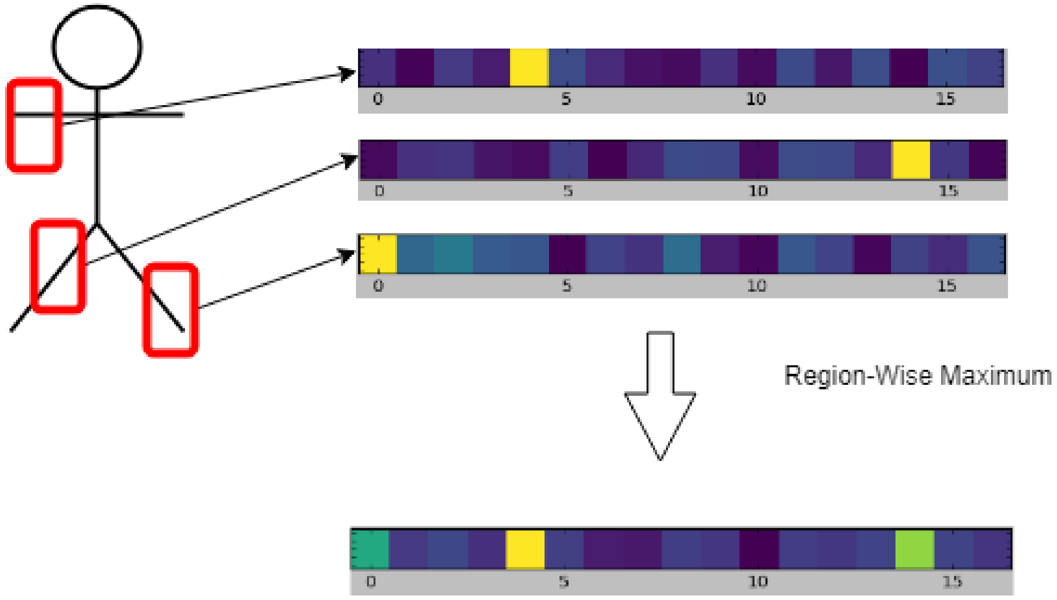


Threats cannot be divided into well defined classes. Instead, I define the object class by one of the 17 locations on the body (i.e. left ankle, right ankle, chest)

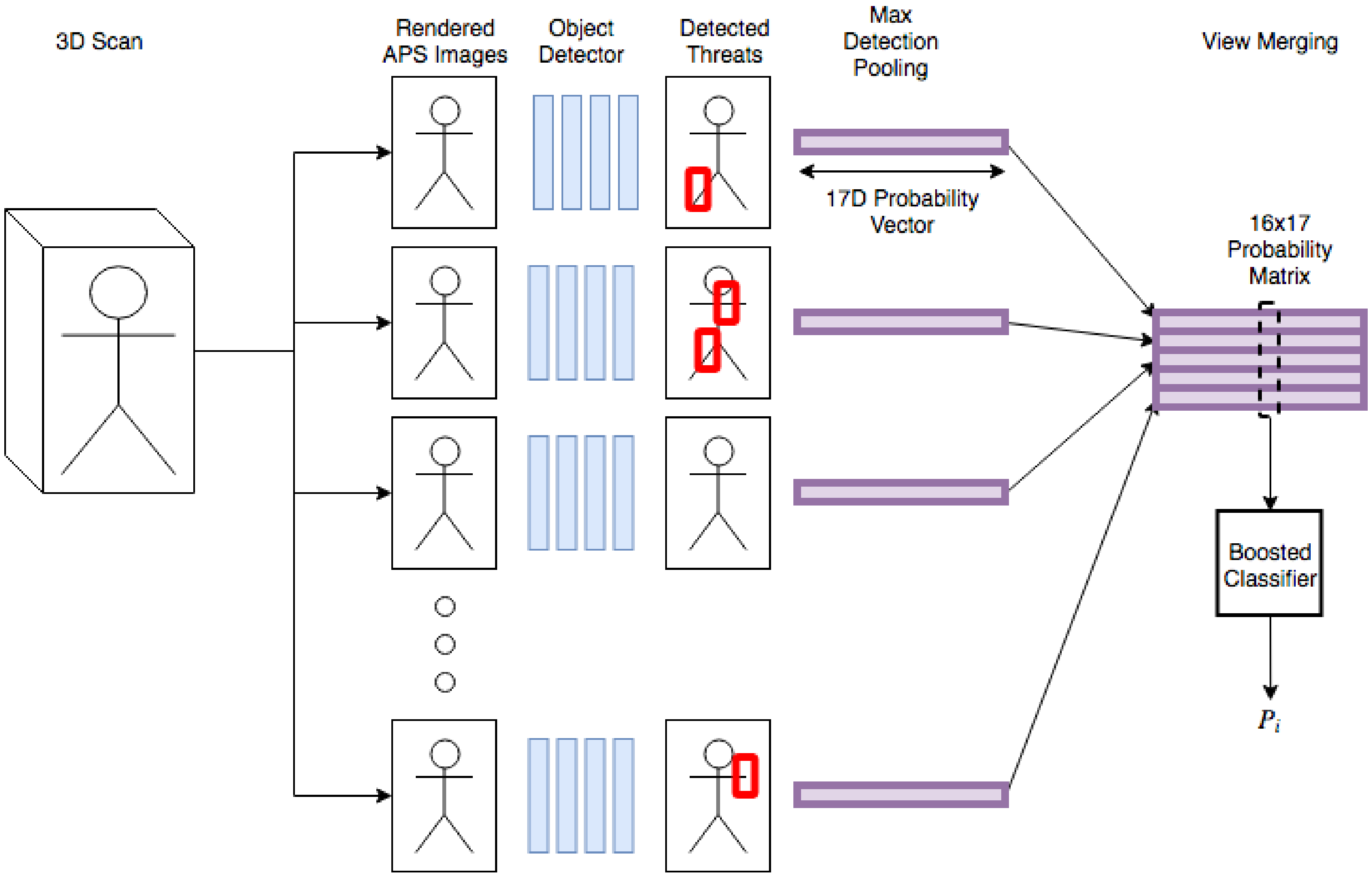
The detector predicts both threat bounding boxes and labels in a single forward pass—eliminating the need for multistage processing.

Post Processing

- Each rendered image (termed APS image) produces a set of detections. Each detection provides a 17D score for each body region.
- For each body region, I take the maximum probability score over all detected regions
- Repeat for 16 different rendered viewpoints. Take the 16 scores for each body region as the input features to a boosted classifier to produce a well calibrated probability estimate



System Overview



Results (4th Place)

(one of the 8 prize winning solution)

#	△pub	Team Name	Kernel	Team Members	Score
1	▲134	idle_speculation			0.02417
2	▲71	serg14			0.02659
3	▲69	David O. | Thomas A.			0.03042
4	▲1	teedrz			0.04211
5	▲40	Oleg Trott			0.04236
6	▲116	CNN is fake model			0.05501
7	▲64	suchir			0.05838
8	▲41	kaggle446			0.05970

Strengths:

- This approach works very well right out of the box. I applied a standard object detector with little tuning and no data augmentation and immediately got good results
- Object detection forces the model to identify regions of the image which contain potential threats
- New object detectors can be easily plugged into my model to improve results as the field advances

Weaknesses:

- Requires additional labeling of the bounding boxes
- Rendered 2D images instead of directly operating on the 3D scans

Important Findings

Formulating threat identification as an object detection problem substantially improves results by forcing the network to learn to localize threats in the image.

- Room for Improvement
    - More training data: my models improved greatly when using more data.
    - Better detectors: Faster-RCNN was published in 2015, a lot of progress in object detection has been made in the last 3 years
    - More viewpoints
    - Replace bounding boxes with segmentation masks

Transition to Field

- My approach is well suited for the field. It is fast enough to run in real-time on a modern GPU and doesn’t use any additional information other than the scan itself.
- Deployment will require validating the model on additional data and human oversight.
- Mechanisms to avoid adversarial attacks will need to be implemented before complete automation.