

# Rapid development of object detection model at the edge

Adrian Chang/ Mentor: Shilpa George

# Goal/Solution

Goal: Exploring how to do automated training of object detection on the edge for faster development.

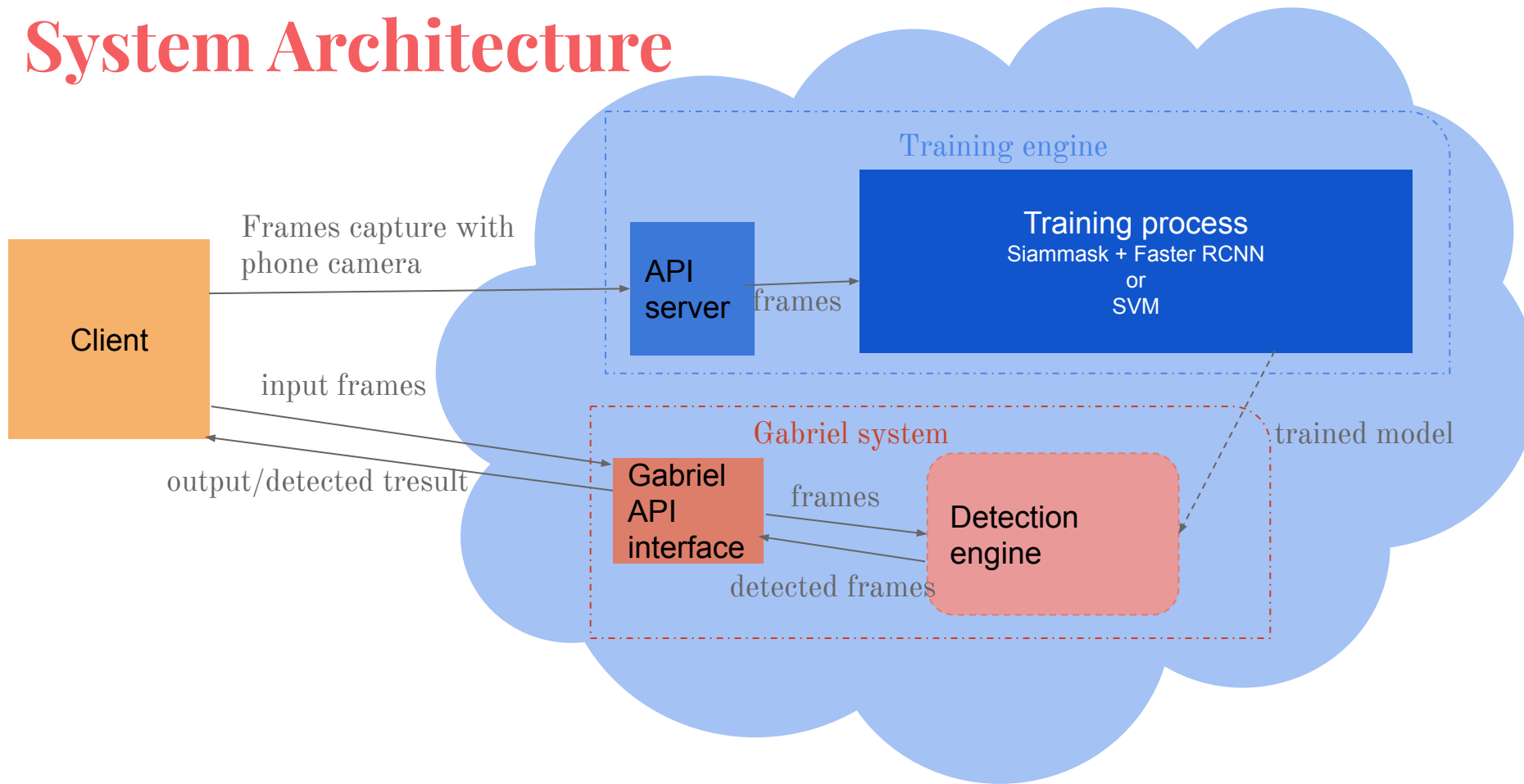
Solutions:

We provided two variants

1. Siammask + Faster RCNN
2. Edge detector + SVM

General workflow: Android app sends video to the cloudlet, cloudlet then starts the training process(the process depends on two variants). After the training is done, the Android app can then use the trained model for inferencing.

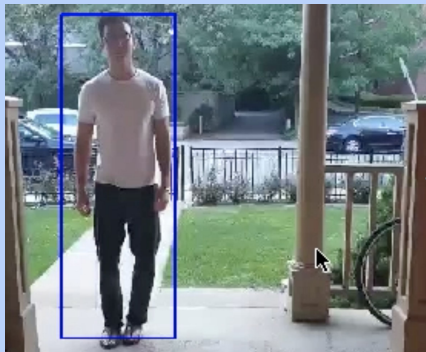
# System Architecture



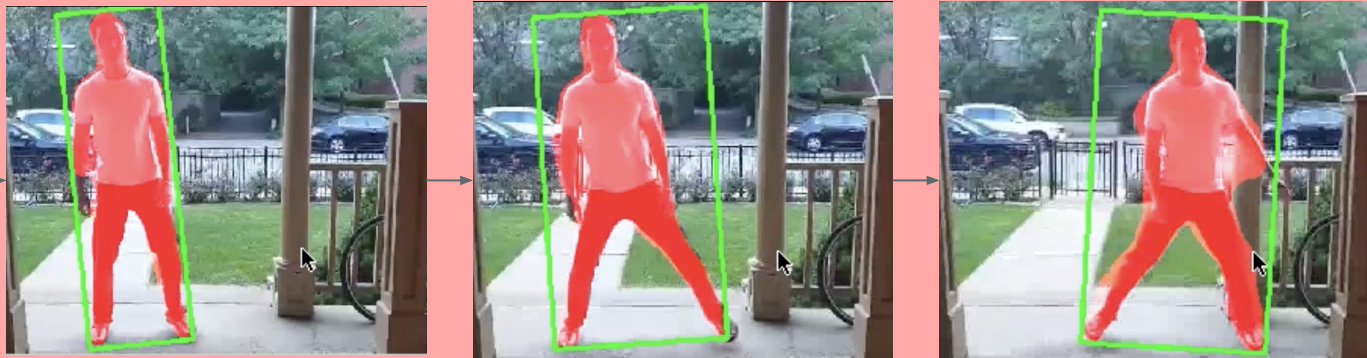
# Solution 1: Siammask + Faster RCNN

Siammask: a class agnostic object detection model which can help annotating video and reduce the labor work

Annotate the object once



Siammask tracks the object for the whole video



# Solution 1: Siammask + Faster RCNN

Faster RCNN: a fairly accurate object detection model

Pros: Good precision, pretrained with 330k images

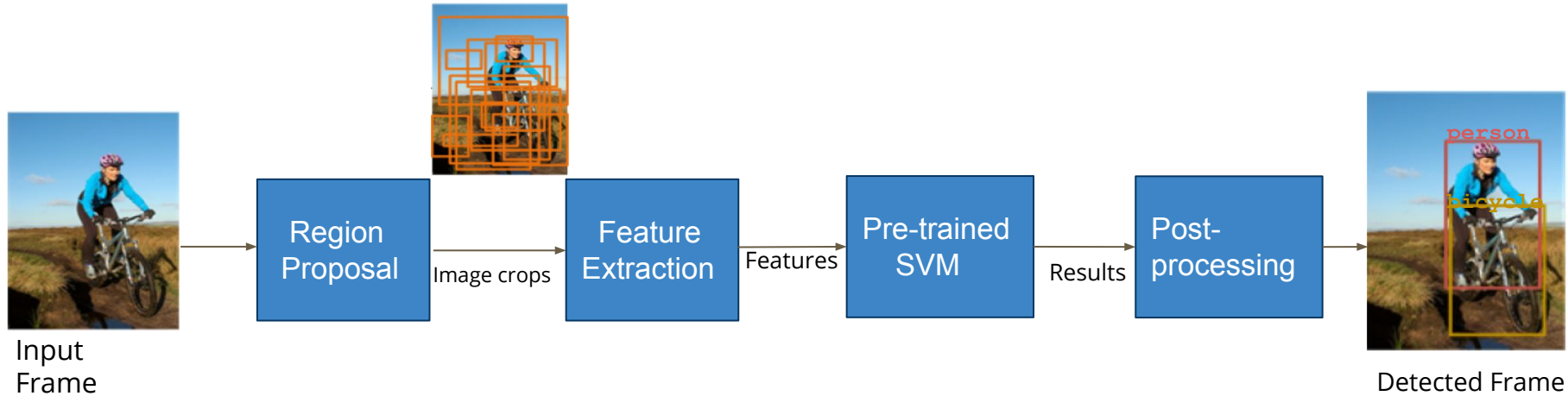
Cons: Slow to train(3hr on one GeForce GTX 1080), bad recall

Result on data produced by Siammask:

Precision: 0.81, Recall: 0.58, F1-score:0.68

Summary: It can recognize object well but tend to miss classified

# Solution 2: Edge Detector + SVM



Pros: Fast training(5sec)

Cons: Not very good precision

Precision: 0.67, Recall: 0.9, F1-score:0.74, fps: 3 frames/seconds

# Final result/Lesson learned:

Siammask + FasterRcnn is too slow to be a “fast development” model. Also, the recall is not that good. Thus, we decided to use SVM version.

Takeaway:

1. Off loading is crucial for ML related application on mobile phone because mobile phone doesn't have a lot of resources. Especially for training a model on mobile phone, edge computing is an important component.
2. Siammask can help with video annotation. However, in speeding up the object detection model training, the time to train the model is the bottleneck.