

1. Problem statement

I completely changed my model compared to the one from deliverable 2, as I found a framework that included everything that I needed. Instead of a single CNN network to determine if an image contains a handgun or not, I am using a framework called Mask-RCNN (https://github.com/matterport/Mask_RCNN) that puts a bounding box and a mask on any detected handgun in an image. It works similarly to YOLO and uses 2 neural network stages. The first one generates regions where one of the input classes might be in the input image. Then, a second stage predicts the object's class, a bounding box and a mask. Both stages are connected to a backbone, which is a FPN style NN that consists of a bottom-up pathway (generates a feature pyramid map). For now, the model can predict handguns in a static image, but struggles in a video stream (OpenCV2). I have determined that the model is overfitting and it is mostly since I only trained it on 100 annotated images. However, I plan on adding more and hopefully improve the results.

2. Final Demonstration Proposal

Initially, I planned on using a website to showcase my product using a ReactJS frontend because I used the framework extensively during my internship this summer. However, I do not think that I have the time, as I must refine my model so that it could hopefully work on a video stream. I plan on simply turning on my laptop's webcam using open-cv2 to demo my project and use a PowerPoint presentation to explain my project, if possible. If I have the time, I will implement the application on a website.