Machine Learning Engineer Capstone Proposal

Bruno Bitencourt Luiz

June 9th, 2019

Early Threat Detection and Object Recognition with Generative Adversarial Networks

Domain background

Airports security is tested each day by passengers from all around the world. Before entering a plane, the passenger and his luggage should be inspected to search for items that can harm other people or compromise flight security. This kind of inspection is made by scanning passengers body and luggage using x-ray techniques.

In most airports, each year the number of passengers increase and together, the number of illegal items that are apprehended. In 2018, the Guarulhos Airport in Brazil received almost 15 million[1] people from international flights and make 292 arrests. The majority of arrests are for drug traffic and other illegal items such as guns, knives, and other items. The luggage inspection is made when the baggage is being dispatched before the takeoff and on the arrival. Also, during terminals transportation.

Identifying threats is the most important thing to do. But object discrimination with x-ray can be very helpful. Especially, to reduce the estrangement of a passenger by having to open your luggage in front of the police. This can be used to identify if the passenger has a suspicious amount of items in his luggage, such as watches, shoes, glasses, etc.

The majority of systems in this field are closed and not share any dataset. Some of them, like ALERT[2], offers a dataset to sell, but if you buy it, you can't share with your project.

By working in this task we can improve the precision of object detection and also, provide inspiration for many objectives such as identifying threats in schools or small airports. Also, some aspects of the model can be applied to improve further works on x-ray interpretation for medical reasons, for example, wrong pathology diagnosis[3].

---

[1] Guarulhos International Airport - Operational Info
[2] ALERT Datasets
[3] Cross-Modal Machine Learning as a way to prevent improper pathology diagnostics

Problem statement

This problem can be framed with Deep Learning since we will use deep neural networks and imaging processing in many different ways.

In terms of imaging, we have an image classification with segmentation task, were in our dataset we have areas at the image where some items are illegal, others not. For example, baggage with normal stuff, like a phone, glasses, a wallet, and in a pocket may contain an illegal item like a fire weapon.

With the lack of data images showing dangerous items, we can apply Generative Adversarial Networks (GANs), in which we generate a set of images to improve the number of our samples and also increase our precision by using discriminative classifiers.
In summary, I will use:

- Deep Learning
- Image Classification
- Image Segmentation
- GANs - Discriminative and Generative

Datasets and inputs

As mention before, acquiring this kind of data is not an easy task. After I've done some research, I've found the dataset "*GDXray: X-ray images for X-ray testing and Computer Vision"* published by professor Domingo Mery[4]. The dataset has 2.9GB of baggage images containing dangerous items such as guns, knives, and non-dangerous items such as phones, keys, notebooks, and so on. This dataset contains more than 19 thousand black/white x-ray images, but only not all from baggage items. Another dataset I will be using is SIXray from University of Chinese Academy of Sciences (UCAS-PRISDL)[5]. This dataset contains more than 1 million images and almost 9 thousand of samples from dangerous items. This dataset has approximately 72GB. Both datasets can be used for educational purposes.

[4] GDXray Dataset
[5] SIXray Dataset

Solution statement

The main goal is to classify an image as containing a dangerous object or not. To achieve this, I gonna test neural networks with various architectures. For this kind of classification our main metric will be "*recall"* since its more important don't miss a dangerous object than triggering a false alarm. For working with imbalanced data, I've gonna need test different balancing between 2 classes.

Benchmark model

A good comparison with this project is the results from the paper: "*SIXray: A Large-scale Security Inspection X-ray Benchmark for Prohibited Item Discovery in Overlapping Images*[6]*".* This article produced the SIXray dataset and applies a technique called class-balanced hierarchical refinement (CHR) to evaluate the precision of identifying threats.

A good standard is to consider his final results using the proportion of 10 nondangerous items to one dangerous, which achieved an average precision of 77.2%.

Evaluation metrics

The metrics that I'm going to use are precision and recall. The success of the solution can be measured by achieving a precision greater than 77.2%. Personally, at the final model, I expect a recall for each object (guns, knives, wrenches, pliers, and scissors) with at least 70%.

Project design

The first thing on starting with the project will be data preprocessing and labeling. I will be using samples of dangerous items from both datasets. For the nondangerous, I will take only from SIXRay, since on GDXray these items are alone in the image. Thus, I can label all the samples as dangerous or not dangerous. By using these two datasets I will have samples

[6] *SIXray: A Large-scale Security Inspection X-ray Benchmark for Prohibited Item Discovery in Overlapping Images*

with shape well defined from GDXray and with overlapping from SIXray. During the benchmark, I will test with a hidden layer trained with just GDXray and check how it performs. Also, by using transfer learning, I will test with a layer trained with normal objects and see if we can improve our model.

After gathering an overview, I will proceed by optimizing the model. In the end, I will have the strengths and weakness of the model. I've presumed some image manipulation will be needed such as color filtering, rotation, etc. My intent is to document each experiment and results until the final result.

Resources I will use: Python 3, pandas, sci-kit learn, OpenCV, Tensorflow, Keras, Seaborn, Matplotlib, Jupyter Notebook.