Weapon Detection

Biologically Inspired Artificial Intelligence

Dawid Kubów
Mateusz Paskuda
GKiO1

Repository link: https://github.com/mpaskuda/BIAI

1 Topic, Assumption and problem description

Everyday at the airport milions of luggage is scanned by x-ray scanners to watch if inside there are not any kind of weapon or other illegal item, but all time operator must watch at the screen to search this item.

Our idea is to create program to make it automatic without an operator to watch screen and search items, but including neural network we want learn it to recognize guns and sharp tools recognized as weapon. As input we would use ready x-ray image in grayscale which would be scan for identification weapon and return percent value what program detect (gun, sharp tool or no weapon found).

2 Technology

Program we based on open Source library Tensorflow which has many useful function which we implement in our program what made easier to process images and recognize items on it.

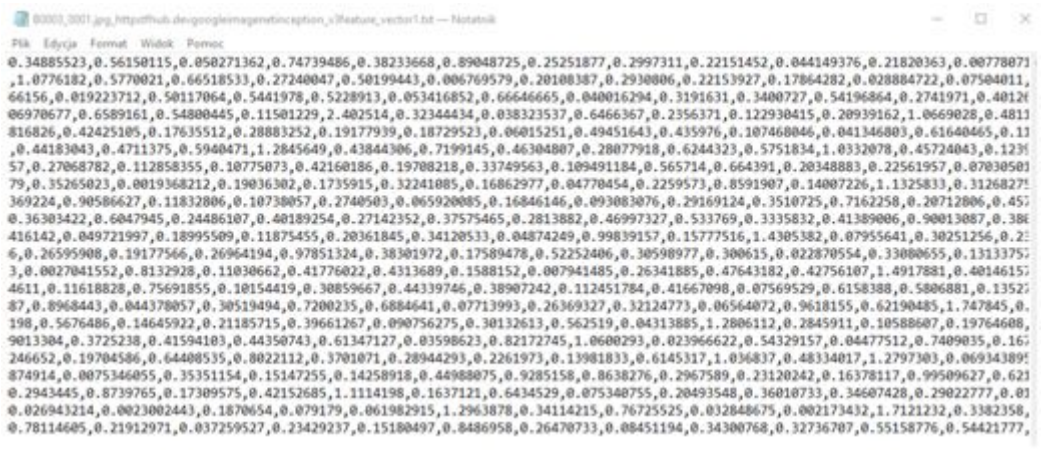
We used MobileNET which is ready convolutional neural network to recognize images so we only need to change few things to change and run transfer learning of neural network to teach recognize weapon.

3 Used data

For input data we use ready x-ray images which show luggage with items inside it in grayscale. Before run learning and validation of recognizing luggage we generate files to each image which contains representing image as vector data using in main program to compare data of input file with learned knowledge about shapes and display of illegal items. This step of generate file of vector data safes time and resources as well as make better accuracy.

Input image and look of image which is generating to vector data

Image of file with vector data of image



5 Implementation

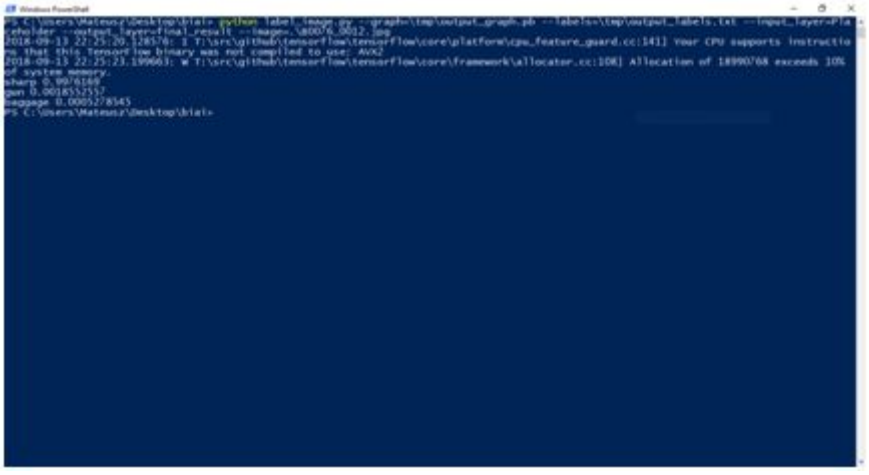
Program is using commands running from command line, but Tensorflow is created for Linux so on Windows we must spend additional time to make it runnable due to issue and problem with correct working.

To generate files we used Bottlenecks result file include vectors of image we used later in process of recognize items on images what make sit faster and used less resources.

Algorithm default segregate data into special groups for testing, validation and images are random assign to groups to avoid bad learning and repeat of images. The percent of assign we can change in settings but we used default.

6 Testing and results

To run algorithm we use command in command line



While testing we try different amount of steps while process image and we noticed some dependence namely low amount of steps cause fast processing and low usage of resources but has lower accuracy however using bigger amount of steps cause increase of accuracy but also higher resources consumption and takes much more time.

Additionally we noticed that our program better recognize guns rather that sharp tools recognize as weapon, tested on image with both items, however sharp tools was more diverse that guns (guns have more similar each other and have some part of shape identical which could be the reason).

However this situation could be due to small amount of different input data of luggage contains bigger range of multiple variants of guns and sharp tools recognize as weapon and luggage without weapon which is hard to find without contact to the airport and we could only base on data found on internet for non-commercial use.

Results from recognize

For 500 steps

sharp 0.96549517

gun 0.023937853

baggage 0.010567048

For 4000 steps

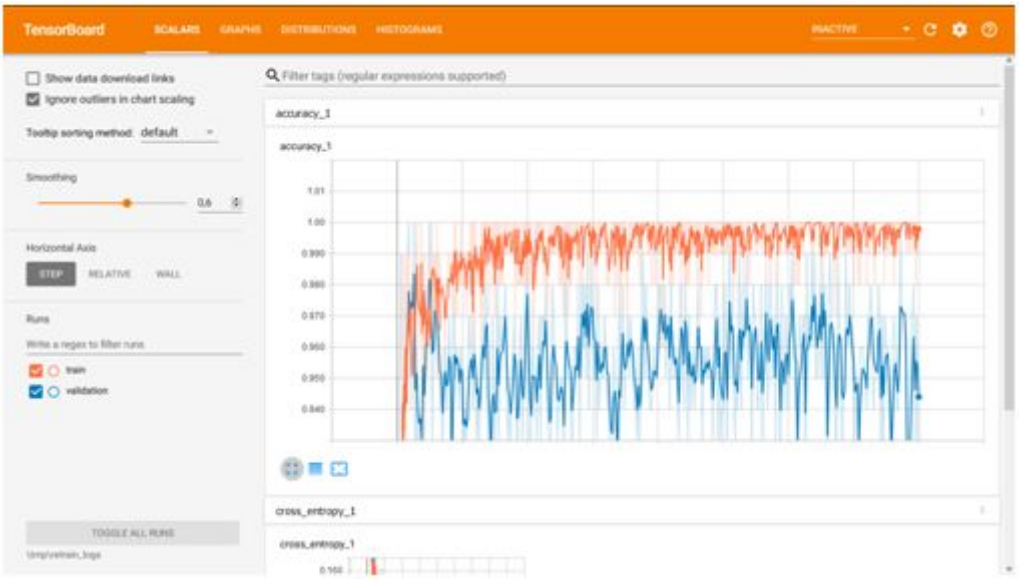
sharp 0.9976169

gun 0.0018552557

baggage 0.0005278545

Looking at a combination of two amount of steps we saw conclusion written above that the higher amount of steps the better accuracy we can achieve.

Graph showing accuracy on train phase and validation.



7 Conclusions

While learning neural network we noticed that it is easy and hassle free to create program a prediction program to recognize certain elements.

However the biggest problem is to get right set of data to property teach program to realize problem which we want to learn. Problem with not correct preparation of data we could noticed on testing image with items from different categories (gun and sharp tools) and some problems when there are not included weapons in luggage.

We have found out that it is easy to implement algorithms from learning to testing or validate neural network so we can easy create basic program, conducive to this is fact that in the internet there are many libraries so this encourages further work in this field and solve other problems.

8 Bibliography

https://www.tensorflow.org/hub/tutorials/image_retraining
https://codelabs.developers.google.com/codelabs/tensorflow-for-poets/#0
http://dmery.ing.puc.cl/index.php/material/gdxray/