MAIN CHALLENGE

The goal the main challenge was to create an AI solution for localizing child seats in the passenger vehicle. In order to accomplish this, the problem was broken into the following pieces:

1. Localizing the passenger vehicle.
2. Localizing the child seats. (NOTE: this algorithm was developed for identifying only child seats, but not infant seats as well)
3. Checking if the child seat found was inside the area of the passenger vehicle.

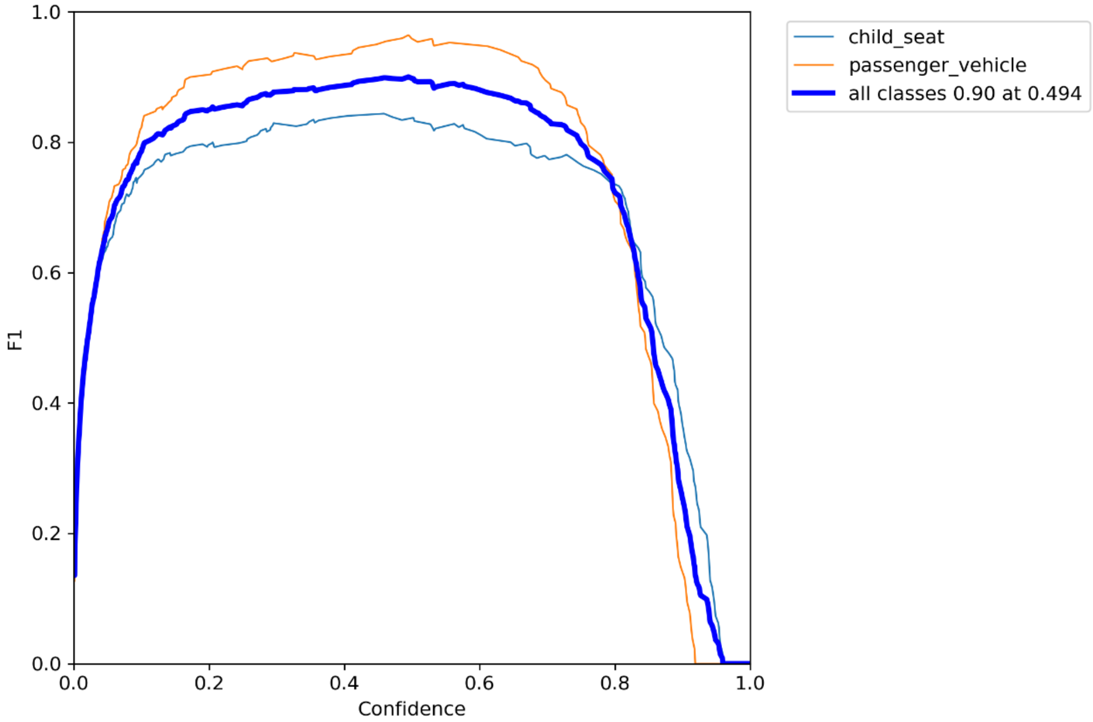
About the dataset, it was downloaded images from the SVIRO dataset (https://sviro.kl.dfki.de/), and only grayscale images were used. The main motivation of choosing grayscale images (which imitated the infrared system) is because some objects in the RGB version, depending on the light, could not be fully seen. The 2 images below exemplifies this, which In the first picture it cannot be seen the infant seat, while in the second picture it is possible.

So, for the creation of the dataset, it was downloaded the grayscale images from all the 10 car classes available. For each car class, it was manually chosen about 50 images which contained 1 or more child seats in it. The sum of all classes resulted in 500 images, which the images was then labeled manually one by one (using labelImg - https://github.com/HumanSignal/labelImg), and finally the dataset was splitted into train, test and validation data, following the proportions of 60% to train (300 images), 20% to test (100 images) and the other 20% for validation (100 images). The images were not randomly splitted, but instead different car classes were used in each portion, which means that the cars used for the training part are not present inside the validation nor the testing datasets, and vice versa, in order to get a more realistic metric and avoiding overfitting.

To build the AI solution for this problem, it was selected the yolov7 algorithm (https://github.com/WongKinYiu/yolov7), which is by the date of this writing considered the most accurate algorithm for computer vision (accordingly to https://dataphoenix.info/a-guide-to-the-yolo-family-of-computer-vision-models/), even with the release of yolov8 (which is not a release from the same authors of yolov7).

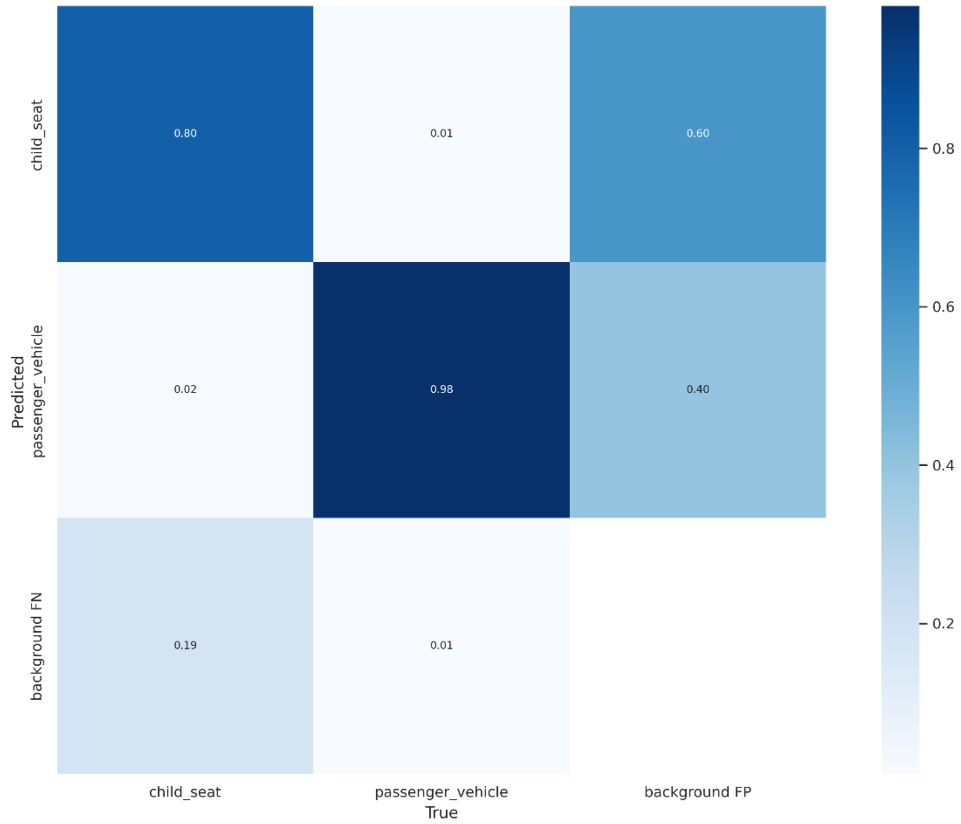
Yolov7 architecture is based on deep learning, and across the Years it has been improved to a point that it handles itself automatically with data resizing and even generating synthetic data (resizing, cropping, padding, etc.) of the input data. There are several models that we can fine tune at the official repository, and it was chosen the fastest model (YOLOv7.pt), not only because it runs faster, but also trains faster as well, which due to the limited time for developing this Project, was an important factor to be considered of.

About the training results, some of the most important metrics can be found in the images below, but more metrics can be accessed in the folder "yolov7/runs/train/yolov7-custom".

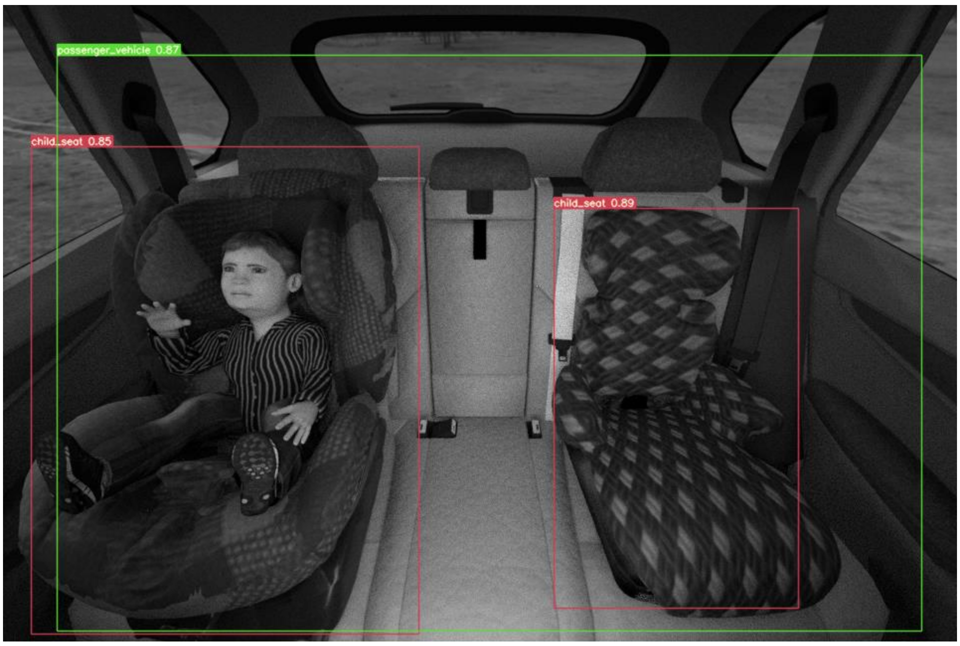


This F1 score is showing some excellent metrics, achieving 0.9 (out of 1.0), which means that the algorithm was able to learn what was a passenger vehicle (seat) and a child seat with a high confidence for most of the images presented.

After developing the algorithm with 100 epochs and batch size 16 using google colab, the best training weights was chosen as the model (file "best.pt", inside the folder “yolov7\runs\train\yolov7-custom\weights”).



The confusion matrix for this challenge indicates that 98% of the inferences, the passenger Vehicle could be correctly identified, and the child seats identification stands correct for 80% of the inferences. After the training was completed, it was validated with 2 cars not previously seen, and the inferences with the bounding boxes along with the confidence of such bounding boxes can be checked at “yolov7/runs/detect/exp”. The below image represents one of these inferences.



Finally, it was developed a python code that checks if the child seat identified is inside or not in the passenger Vehicle (code "bbox_overlap_calculation.py") by calculating the percentage of the overlapping area.

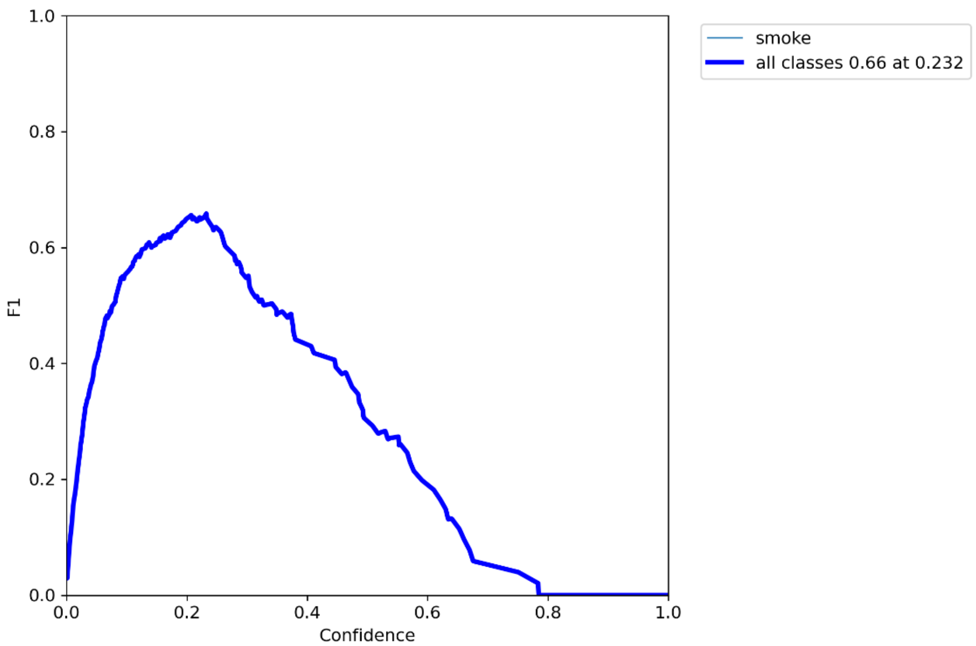
BONUS CHALLENGE

The main goal of this challenge was to generate synthetic data by inserting smoke inside the vehicle, and then implementing a smoke detection system based in an AI algorithm.

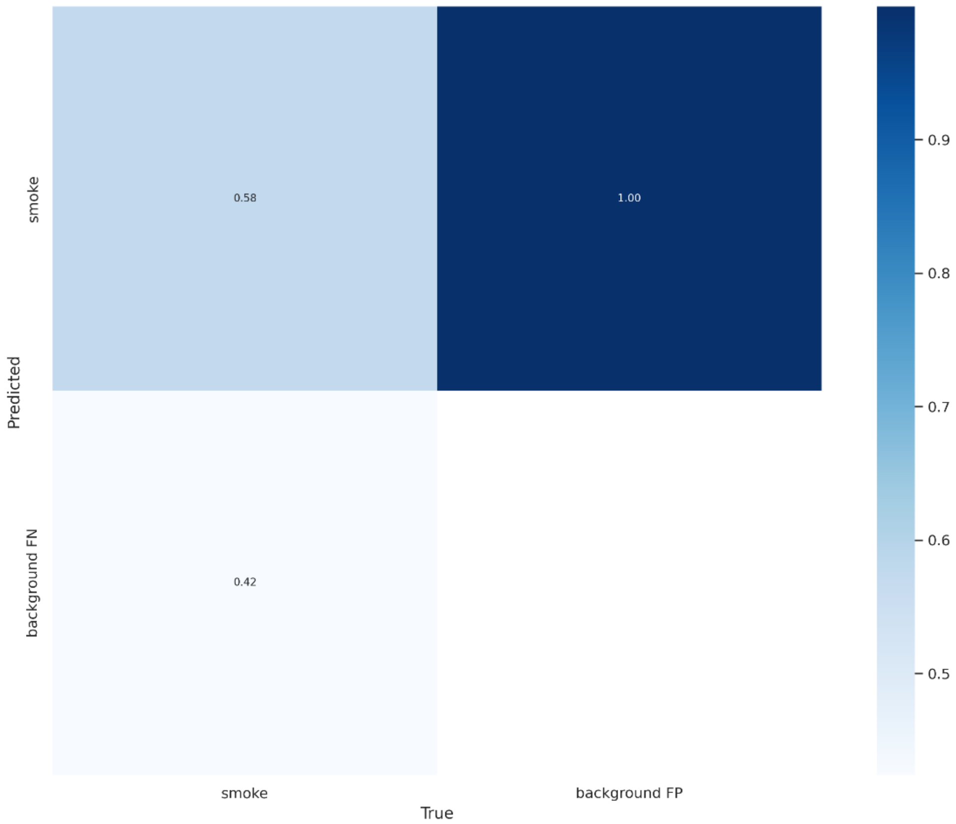
In order to solve this problem, it was first downloaded some smoke pictures without background in the ".png" extension (these images were downloaded from the website https://www.pngegg.com/pt). The main motivation for downloading pictures without background is that they are easier to insert into another picture and create a more fluid and realistic mix. Considering that the background (interior of the vehicles) that are being used are grayscale (simulating an infrared camera), it was also necessary to convert those png downloaded files into grayscale as well.

After getting some smoke grayscale pictures (32 images to be more precise), it was implemented with the aid of the AugLy library (https://github.com/facebookresearch/AugLy) a data augmentation algorithm, in order to generate some synthetic data with smoke inside the interior of the Vehicle. At this step, it was generated about 500 images for the custom training. Of these 500 images, 300 was used for training, 100 was used for testing and the other 100 was used for validating, same way it was done in the main challenge. The script for the creation of the dataset can be found inside the "Data Augmentation" folder, which also contains all images used for this process.

After splitting all the images accordingly to what was described previously, it was fine-tuned the model using yolov7 (same architecture used for the main challenge) in order to identify the presence of the smoke inside the vehicle. The main results are shown below here, but more metrics can be found inside the folder "yolov7/runs/train/yolov7-custom".

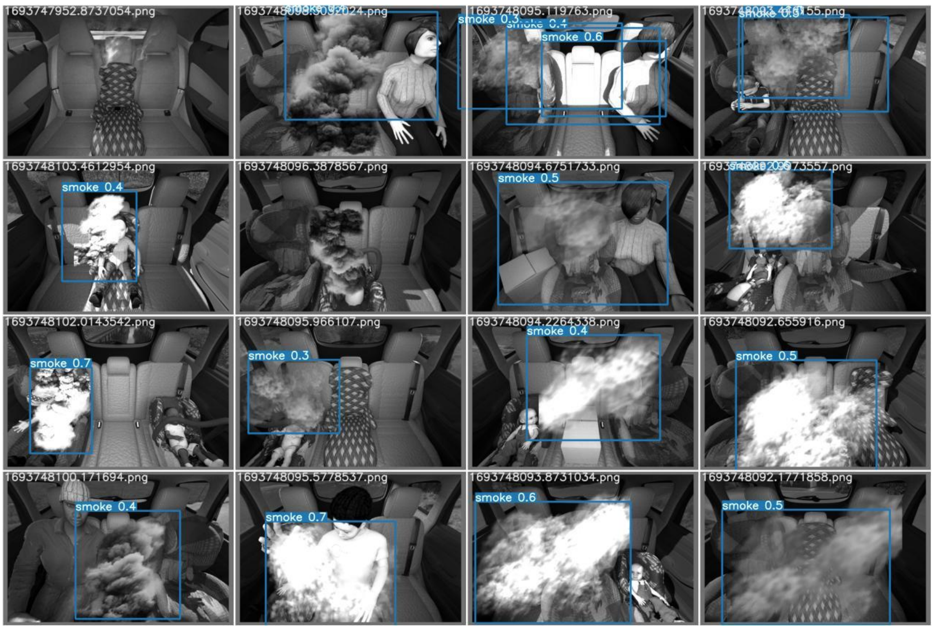


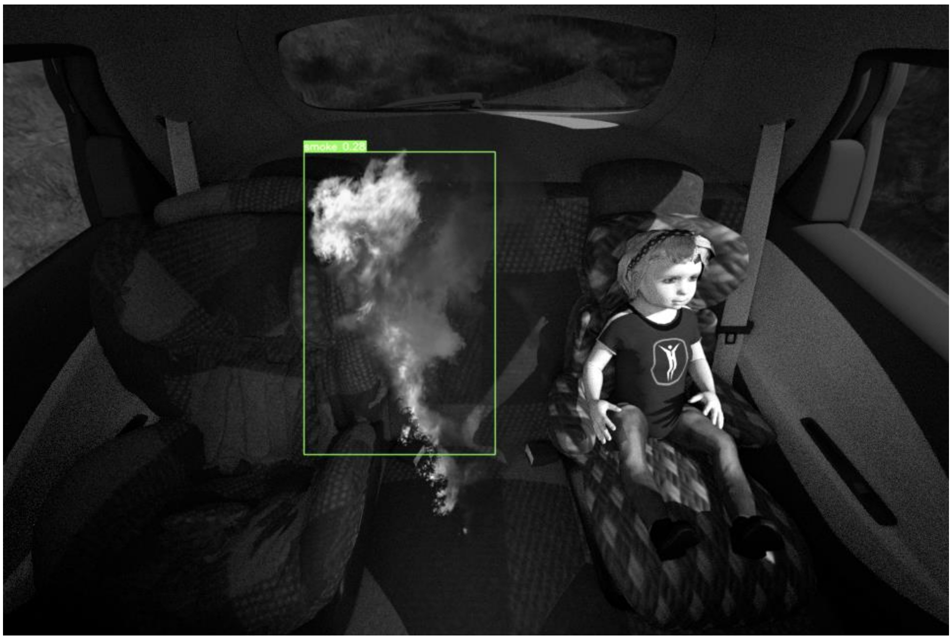
This F1 metric indicates that the algorithm has started to learn how to identify smoke, but it is not performing great, as the best F1 score (0.66) happened with a low confidence interval (0.232).



The F1 metric results are also supported by the confusion matrix, where it can be seen values around 0.5, where excellent value metrics would be close to 1.

Despite not presenting the best metrics, the fine-tuned model has shown to be capable of learning some smoke patterns, as can be seen at the below pictures.





CONCLUSION

The developed models, specially the smoke identifier, indicates that there is still room for some major improvements, yet, specially due to the reduced time to develop this project, this challenge could be considered as successful. The next steps for these refinements could be described as:

- grabbing more pictures (png smoke images for the bonus challenge and adding more angles of the pictures inside the car), which makes the algorithm more robust, as deep learning model based algorithms tends to learn better when there are large pools of data.
- Implementing even more data augmentation techniques.
- Training with different parameters, such as the learning rate, epochs, batch size, etc.
- Training more accurate models, like the YOLOv7-E6E.