



DETR for fire detection

Evaluation using a transformer model

31/05/2023

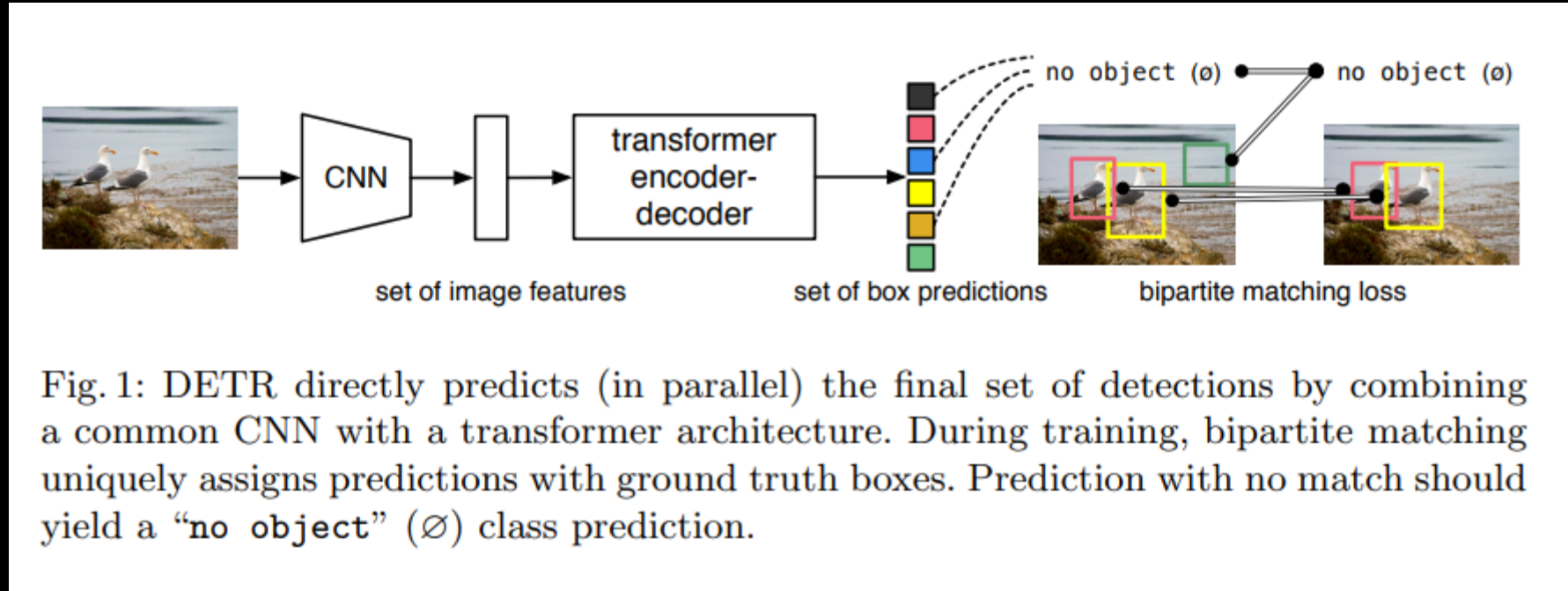
J.F. Alférez Muñoz



Fire detection project

- DETR transformer: <https://doi.org/10.48550/arXiv.2005.12872>
- Dataset: COCO format for DETR for default
- Useful Reference for DETR object detection:
https://huggingface.co/docs/transformers/model_doc/detr

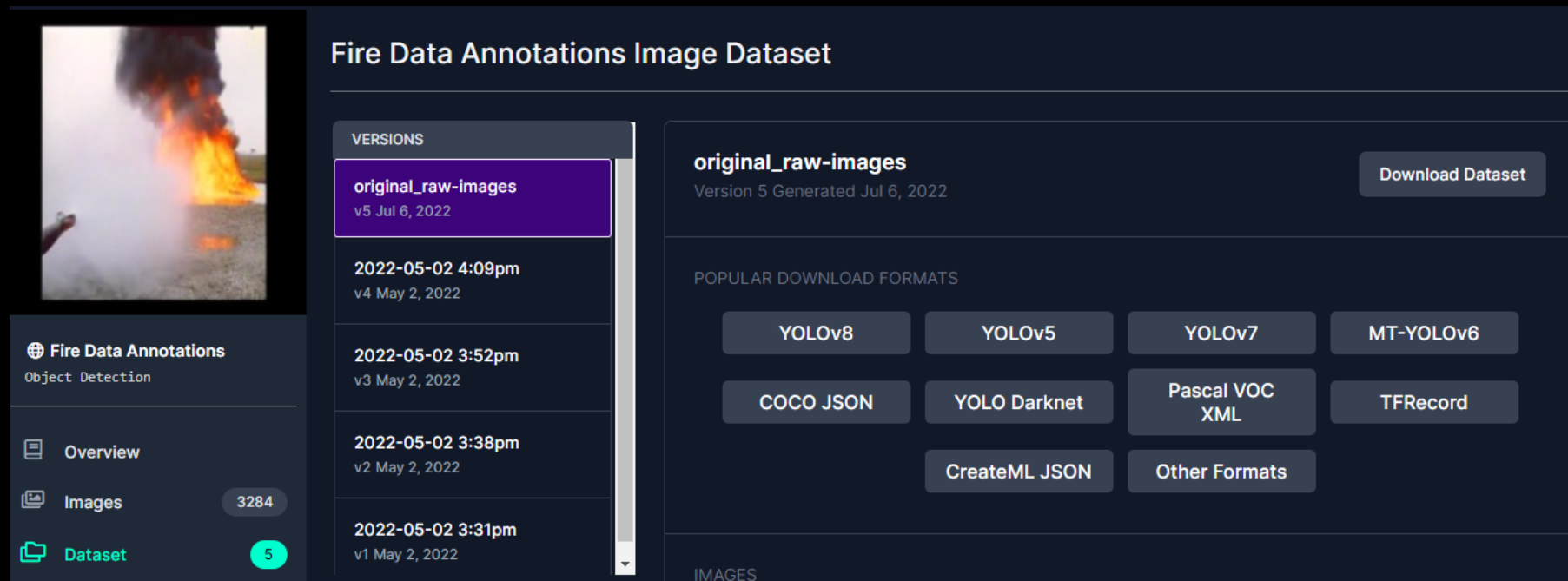
DETR transformer model



Carion, N., Massa, F., Synnaeve, G., Usunier, N., Kirillov, A., & Zagoruyko, S. (2020). End-to-End Object Detection with Transformers. *Lecture Notes in Computer Science (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 12346 LNCS, 213–229. https://doi.org/10.1007/978-3-030-58452-8_13

COCO dataset from Roboflow

- <https://universe.roboflow.com/fire-detection/fire-data-annotations/dataset/5>



The screenshot displays the Roboflow interface for the 'Fire Data Annotations Image Dataset'. On the left, a sidebar shows navigation options: 'Overview', 'Images' (with 3284 items), and 'Dataset' (with 5 items). The main content area is titled 'Fire Data Annotations Image Dataset' and features a 'VERSIONS' section with a list of dataset versions. The current version, 'original_raw-images v5', is highlighted in purple and shows it was generated on Jul 6, 2022. Below the versions, there are buttons for downloading the dataset in various formats, including YOLOv8, YOLOv5, YOLOv7, MT-YOLOv6, COCO JSON, YOLO Darknet, Pascal VOC XML, TFRecord, CreateML JSON, and Other Formats. A 'Download Dataset' button is also present. The bottom of the page shows a section for 'IMAGES'.

Fire Data Annotations Image Dataset

VERSIONS

- original_raw-images**
v5 Jul 6, 2022
- 2022-05-02 4:09pm
v4 May 2, 2022
- 2022-05-02 3:52pm
v3 May 2, 2022
- 2022-05-02 3:38pm
v2 May 2, 2022
- 2022-05-02 3:31pm
v1 May 2, 2022

original_raw-images
Version 5 Generated Jul 6, 2022

POPULAR DOWNLOAD FORMATS

- YOLOv8
- YOLOv5
- YOLOv7
- MT-YOLOv6
- COCO JSON
- YOLO Darknet
- Pascal VOC XML
- TFRecord
- CreateML JSON
- Other Formats

Download Dataset

IMAGES

Training phase

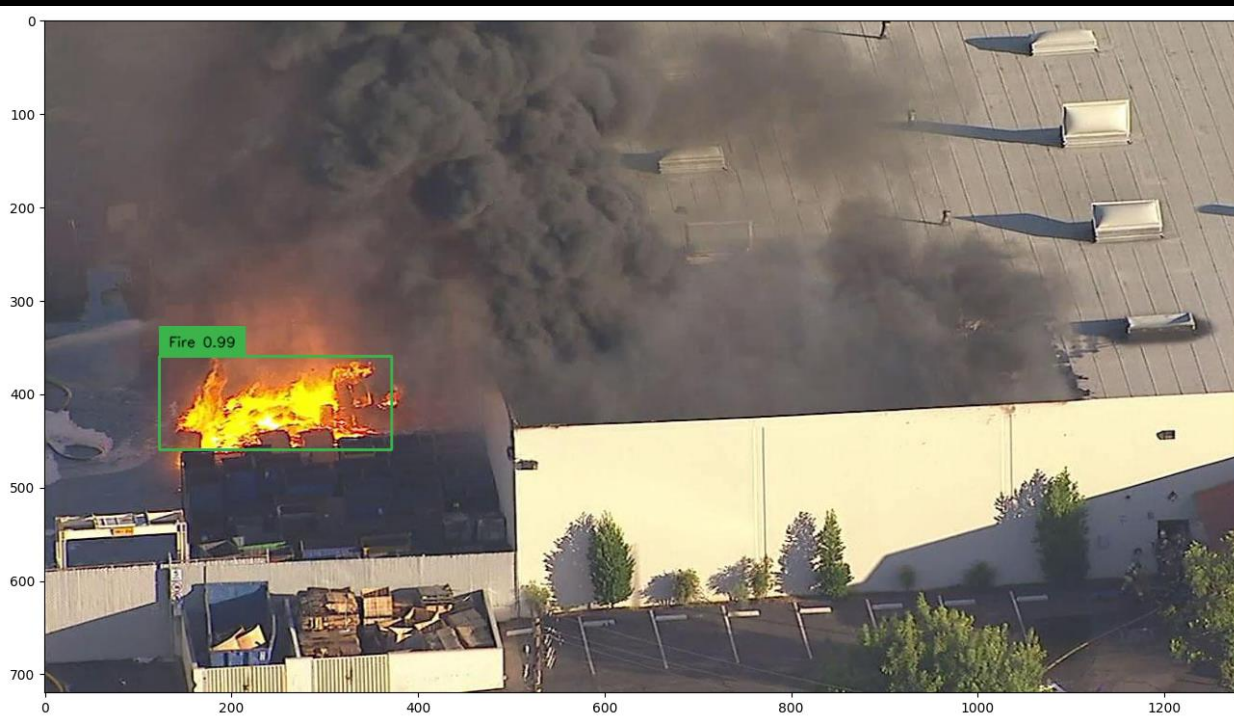
- Google Colab needed for memory resources needed in GPU

| | | | | | | | | | |
|---|----------|---------------|---------------|------------------|-----------------|----------|-------------|--------|--|
| NVIDIA-SMI 525.85.12 Driver Version: 525.85.12 CUDA Version: 12.0 | | | | | | | | | |
| GPU | Name | Persistence-M | | Bus-Id | Disp.A | Volatile | Uncorr. ECC | | |
| Fan | Temp | Perf | Pwr:Usage/Cap | | Memory-Usage | GPU-Util | Compute M. | MIG M. | |
| ===== | | | | | | | | | |
| 0 | Tesla T4 | | Off | 00000000:00:04.0 | Off | | 0 | | |
| N/A | 50C | P8 | 10W / 70W | | 0MiB / 15360MiB | 0% | Default | | |
| ===== | | | | | | | | | |

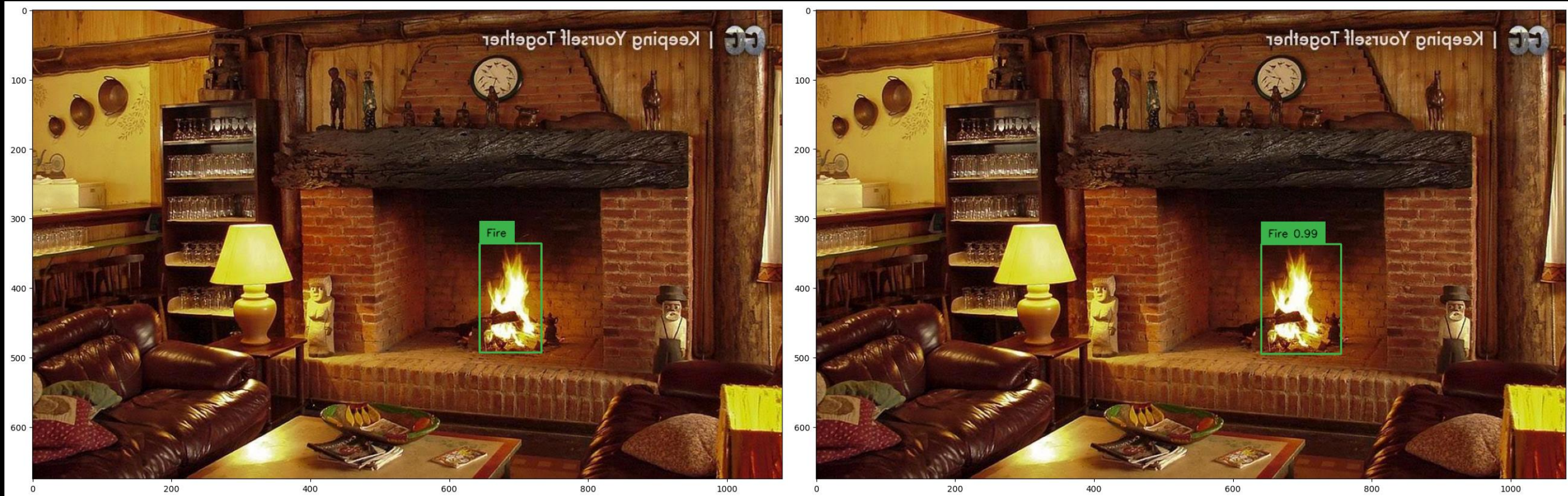
- 50 epochs \approx 4 hours
- 41.5M params

| | Name | Type | Params |
|---------|-------|--|--------|
| 0 | model | DetrForObjectDetection | 41.5 M |
| 41.3 M | | Trainable params | |
| 222 K | | Non-trainable params | |
| 41.5 M | | Total params | |
| 166.038 | | Total estimated model params size (MB) | |

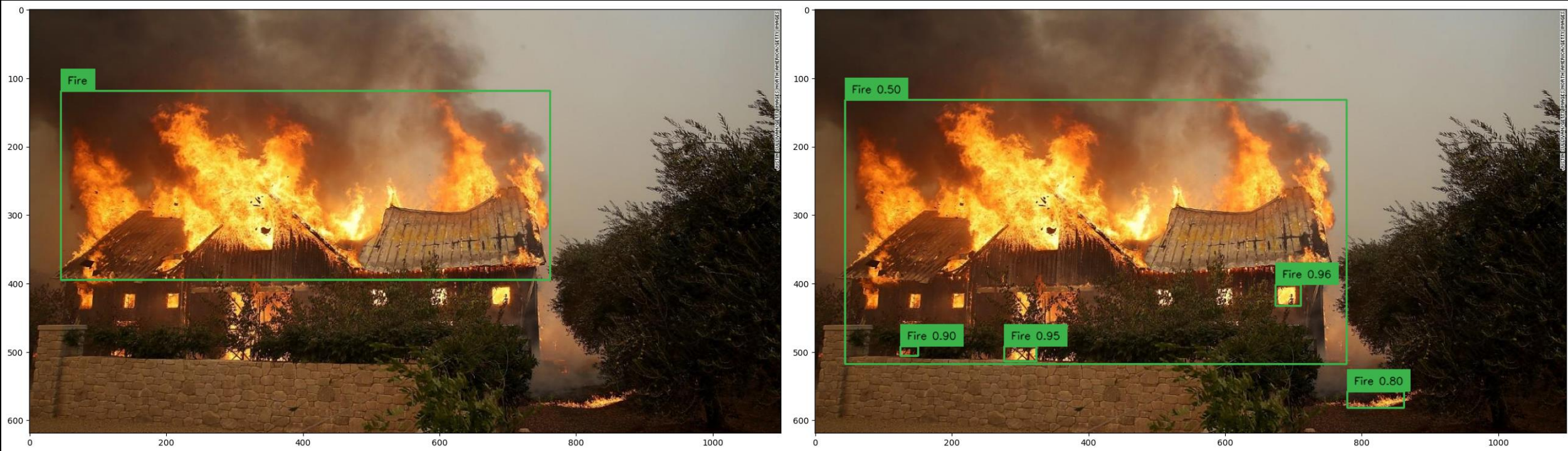
Inference: aerial images, so good!



Inference: clean indoor images, so good!



Inference: Several sources, not bad!



Inference: Several sources, not bad!



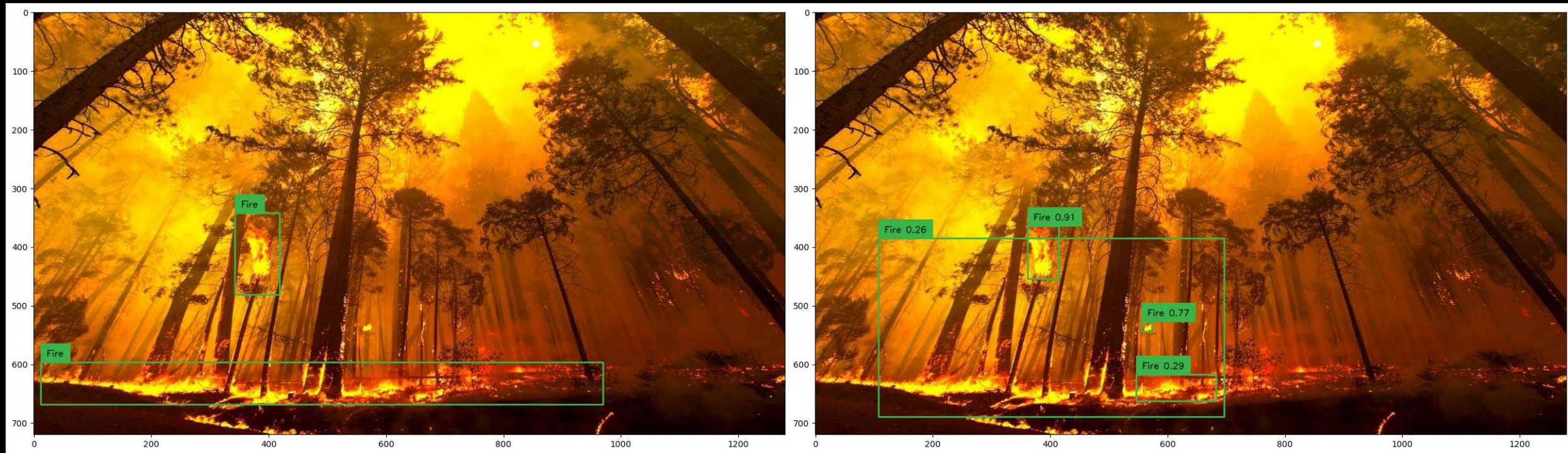
Inference: Several sources, not bad!



Inference: Several sources, not bad!



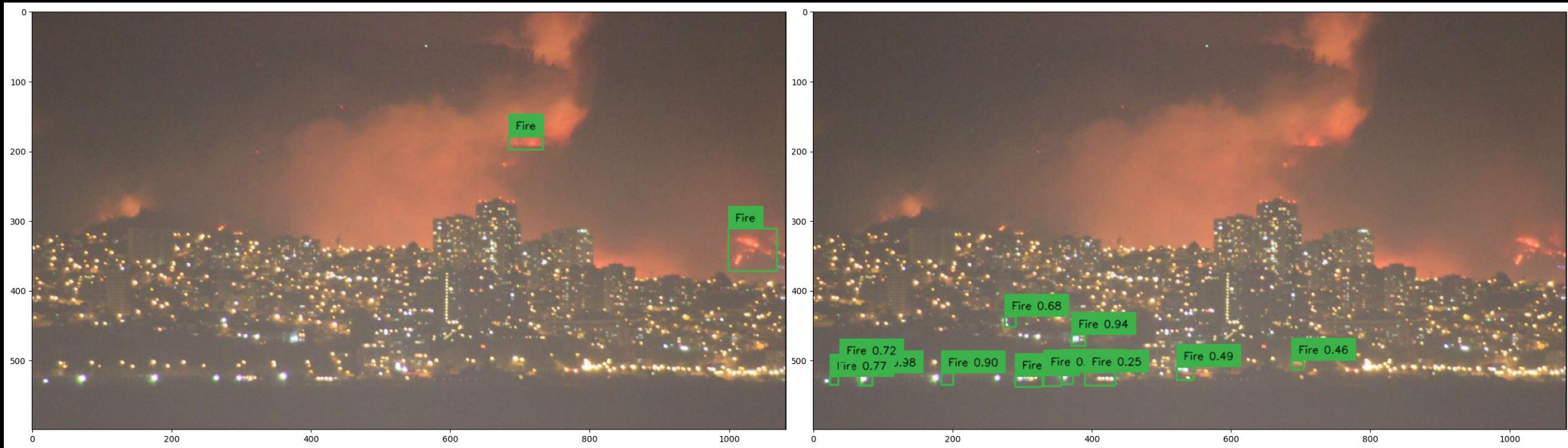
Inference: Into a wildfire, fire vs smoke and brightness



Inference: Close to the fire, not bad!



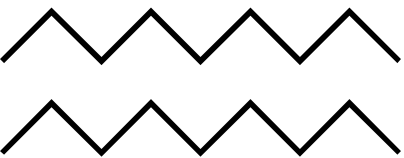
Inference: cities at night.
lights! -> worst results 😞





Thoughts..

- Difficult to train for so different conditions at once.
- Sometimes it is difficult to decide how to label “what is fire?” in pictures.
- Very good with “clean” images (contrast between fire and background).
- Not so good with shines in images, mostly cities at night.



Possible improvements



More training?



Better labeled images?



Focus on a specific type of environment, like aerial images?



Better hyperparameter configuration?

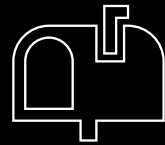


The fire nature is difficult to define by photos

Conclusions... for now

- Give a try if needed -> there is a lot of information about transformers
- There are heavy models with many parameters. Prepare it well before training (resources and time).
- Dataset could be the cause of better or worse results.
- NMS trial and error determination.

Appreciable any comment or suggestion



<https://www.linkedin.com/in/juan-francisco-alf%C3%A9rez-mu%C3%B1oz-559955142/>