IMPLEMENTAZIONE YOLOV5

Diagram, schematic

Description automatically generated

VISUALISATION: <https://netron.app>

* CSPNet (<https://arxiv.org/pdf/1911.11929v1.pdf>)
* Spatial Pyramid Pooling Layer (<https://github.com/ultralytics/yolov5/issues/5541>, https://www.youtube.com/watch?v=2IoHC\_fhrFU)
* PANET (FPN x 2)
* YOLOV5 (<https://github.com/ultralytics/yolov5/issues/6998>)
* HEAD WTF!! (<https://github.com/ultralytics/yolov5/blob/master/models/yolo.py>, <https://github.com/zhiqwang/yolov5-rt-stack/blob/afe8d1009d17356723e45d26af94a38485e20d44/yolort/models/box_head.py#L14-L23>)

MULTI-SCALE TRAINING: <https://github.com/ultralytics/yolov5/issues/1299>

RECTANGULAR INFERENCE: <https://github.com/ultralytics/yolov3/issues/232>

AUTO-SHAPE: <https://github.com/ultralytics/yolov5/issues/2722>

DYNAMIC INPUT SIZE: <https://github.com/ultralytics/yolov5/issues/6755>

YOLOV5 LOSS: <https://medium.com/visionwizard/understanding-diou-loss-a-quick-read-a4a0fbcbf0f0>

YAML: <https://github.comee/ultralytics/yolov5/issues/6142>

YAML width & depth explanation: <https://chowdera.com/2022/03/202203210632261925.html>

LAYERS: https://github.com/ultralytics/yolov5/issues/280 <https://www.researchgate.net/figure/YOLOv5-architecture-The-YOLO-network-consists-of-three-main-parts-Backbone-Neck-and_fig5_355962110>

TODOS:

double\_check e cancellare c3\_neck

PROBLEMI:

1. Implementare head🡪quasi
2. Capire anchors 🡪quasi
3. Capire fusting finale (post-training?) 🡪not yet

DETECT inference explanation: <https://github.com/ultralytwics/yolov5/issues/471>

ANCHORS EXPLANATION: <https://github.com/ultralytics/yolov5/issues/447>

KMEANS: <https://towardsdatascience.com/training-yolo-select-anchor-boxes-like-this-3226cb8d7f0b>

Diagram

Description automatically generated

Graphical user interface, text

Description automatically generated

DEPLOYMENT:

PRUNING AND QUANTIZATION: <https://www.youtube.com/watch?v=HHmYiwYL2Tc>

PYTORCH PRUNING: <https://leimao.github.io/blog/PyTorch-Pruning/>

Cosa non ho capito:

* Output shape: in che modo le immagini del dataset vengono poi trasformate in label? Cioè come fa un’immagine 3x640x640 a diventare target:1,3,80,80,85/

1,3,40,40,85/1,3,20,20,85? 🡪 20 🡪 640/32, 40 🡪640/16, 80🡪640/8, il che implica che quelle nx e ny rappresentano strided-porzioni di immagine.

Ergo: Capito

* Come fa il modello a generalizzare su input shape diverse? Come spiegato da Jenn Gosher la input shape non è rilevante purché height e width siano divisibili per 32.

Ergo: piuttosto capito

* Anchor boxes🡪bx, by, bw, bh explanation: [https://github.com/ultralytics/yolfov5/issues/471](https://github.com/ultralytics/yolov5/issues/471)+ screenshots sotto piu’ screenshot in cartella yolo🡪 Ergo: capito

:giou and IOU explanation

<https://giou.stanford.edu>

DATASET:

1. Check IOU between gt\_boxes and boxes given by the formula