Master in
Computer Vision
Barcelona

M5 Project: Cross-modal Retrieval

Week 2

Introduction to Object detection and Instance Segmentation with Detectron2

Rubèn Pérez Tito
rperez@cvc.uab.es

Ernest Valveny
ernest@cvc.uab.es

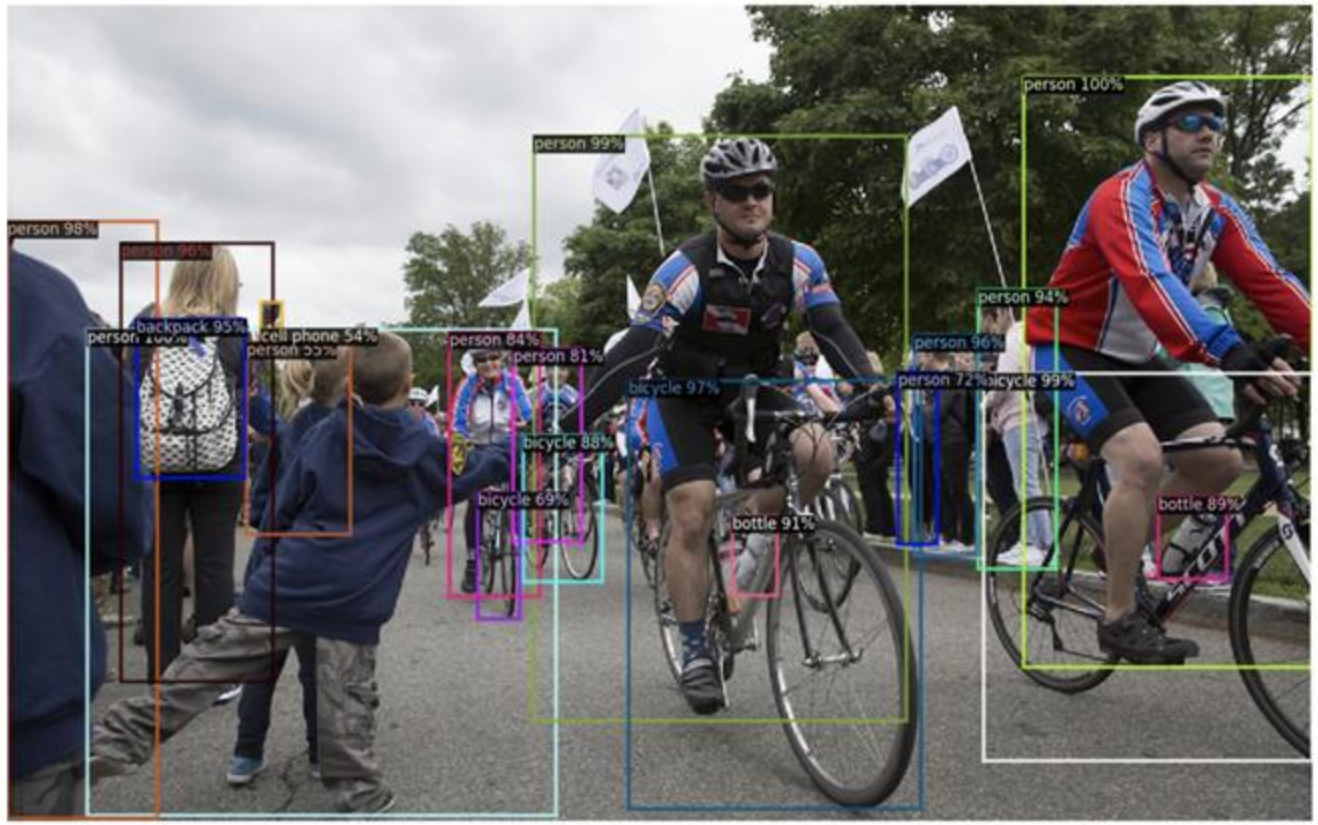
M5 – Introduction to Object Detection and Segmentation

In M5 project, we will use Detectron2 framework from Facebook Artificial Intelligence Research (FAIR), which is a research platform for object detection and segmentation in Pytorch.

- https://github.com/facebookresearch/detectron2

Detectron2

Detectron2 is Facebook AI Research's next generation software system that implements state-of-the-art object detection algorithms. It is a ground-up rewrite of the previous version, Detectron, and it originates from maskrcnn-benchmark.



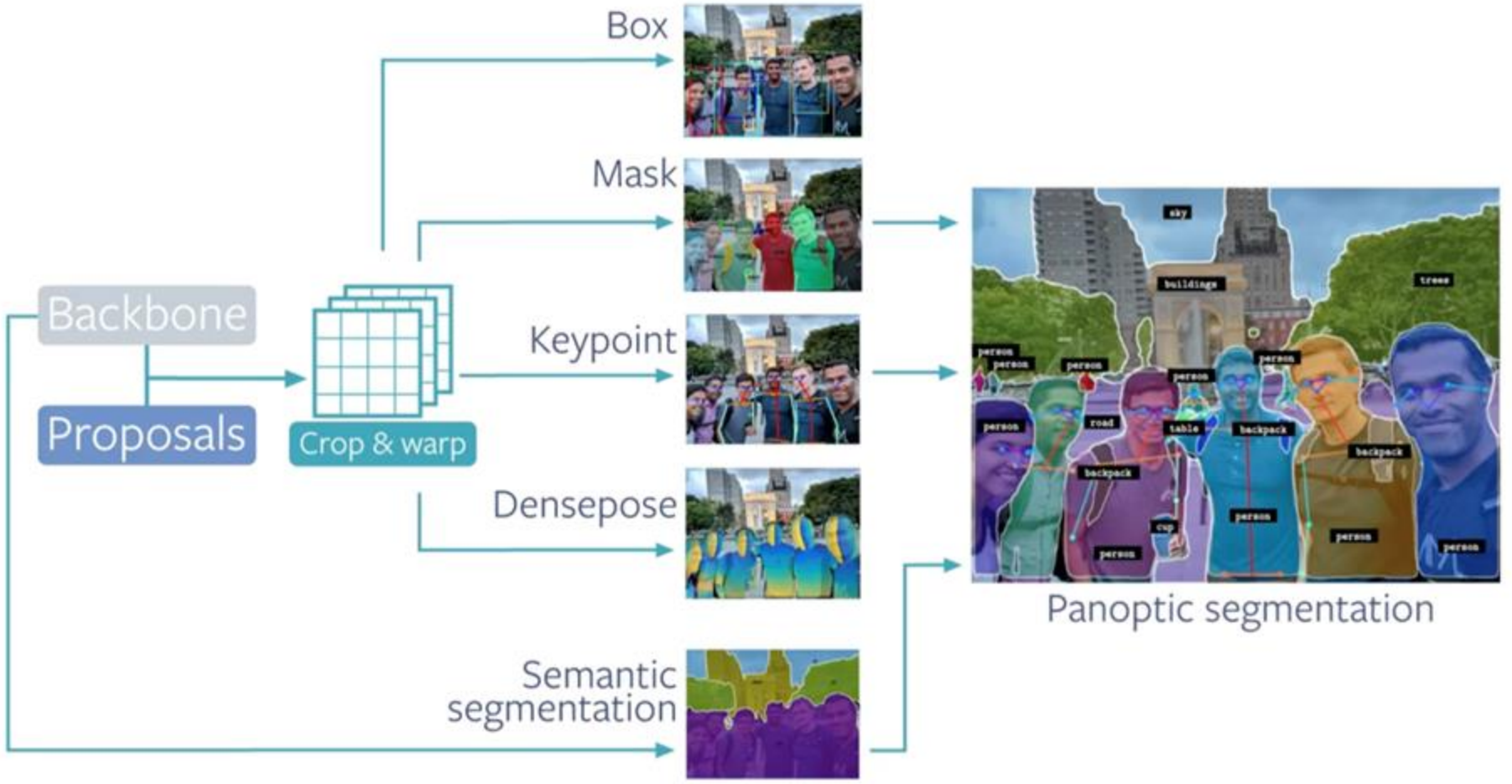
M5 – Introduction to Object Detection and Segmentation

What can we find at Detectron2?

- It supports a range of tasks related to object detection:
  - Powered by PyTorch deep learning framework.
  - **Object detection with boxes**
  - **Object detection with instance segmentation masks**
  - Human pose prediction
  - Semantic segmentation
  - Panoptic segmentation

M5 – Introduction to Object Detection and Segmentation

What can we find at Detectron2?



M5 – Introduction to Object Detection and Segmentation

What can we find at Detectron2?

- It includes many models:
  - **Faster R-CNN**
  - **Mask R-CNN**
  - RetinaNet
  - Dense Pose
  - TensorMask
  - Panoptic FPN
  - Cascade R-CNN

M5 – Introduction to Object Detection and Segmentation

What can we find at Detectron2?

- Check [model zoo and baselines]

COCO Object Detection Baselines

Faster R-CNN:

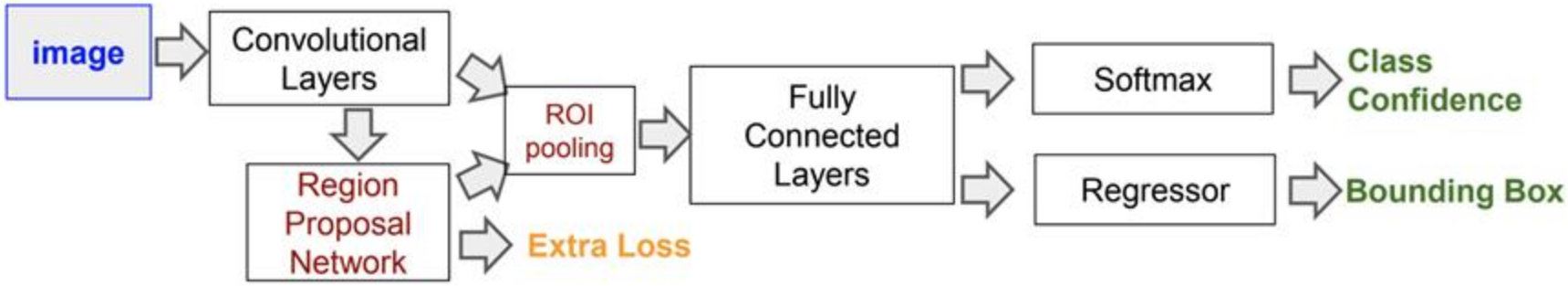
Name	lr sched	train time (s/iter)	inference time (s/im)	train mem (GB)	box AP	model id	download
R50-C4	1x	0.551	0.102	4.8	35.7	137257644	model | metrics
R50-DC5	1x	0.380	0.068	5.0	37.3	137847829	model | metrics
R50-FPN	1x	0.210	0.038	3.0	37.9	137257794	model | metrics
R50-C4	3x	0.543	0.104	4.8	38.4	137849393	model | metrics
R50-DC5	3x	0.378	0.070	5.0	39.0	137849425	model | metrics
R50-FPN	3x	0.209	0.038	3.0	40.2	137849458	model | metrics
R101-C4	3x	0.619	0.139	5.9	41.1	138204752	model | metrics
R101-DC5	3x	0.452	0.086	6.1	40.6	138204841	model | metrics
R101-FPN	3x	0.286	0.051	4.1	42.0	137851257	model | metrics
X101-FPN	3x	0.638	0.098	6.7	43.0	139173657	model | metrics

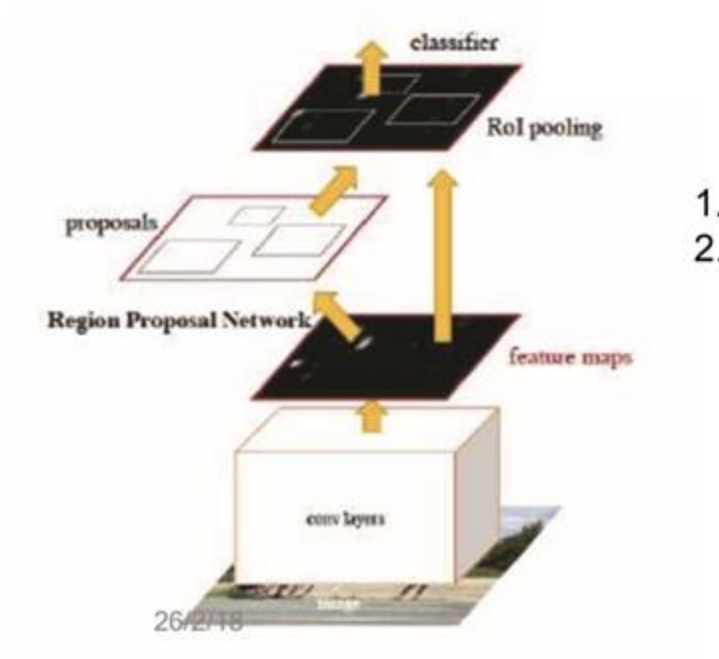
COCO Instance Segmentation Baselines with Mask R-CNN

Name	lr sched	train time (s/iter)	inference time (s/im)	train mem (GB)	box AP	mask AP	model id	download
R50-C4	1x	0.584	0.110	5.2	36.8	32.2	137259246	model | metrics
R50-DC5	1x	0.471	0.076	6.5	38.3	34.2	137260150	model | metrics
R50-FPN	1x	0.261	0.043	3.4	38.6	35.2	137260431	model | metrics
R50-C4	3x	0.575	0.111	5.2	39.8	34.4	137849525	model | metrics
R50-DC5	3x	0.470	0.076	6.5	40.0	35.9	137849551	model | metrics
R50-FPN	3x	0.261	0.043	3.4	41.0	37.2	137849600	model | metrics
R101-C4	3x	0.652	0.145	6.3	42.6	36.7	138363239	model | metrics
R101-DC5	3x	0.545	0.092	7.6	41.9	37.3	138363294	model | metrics
R101-FPN	3x	0.340	0.056	4.6	42.9	38.6	138205316	model | metrics
X101-FPN	3x	0.690	0.103	7.2	44.3	39.5	139653917	model | metrics

M5 – Introduction to Object Detection and Segmentation

Faster R-CNN [paper]



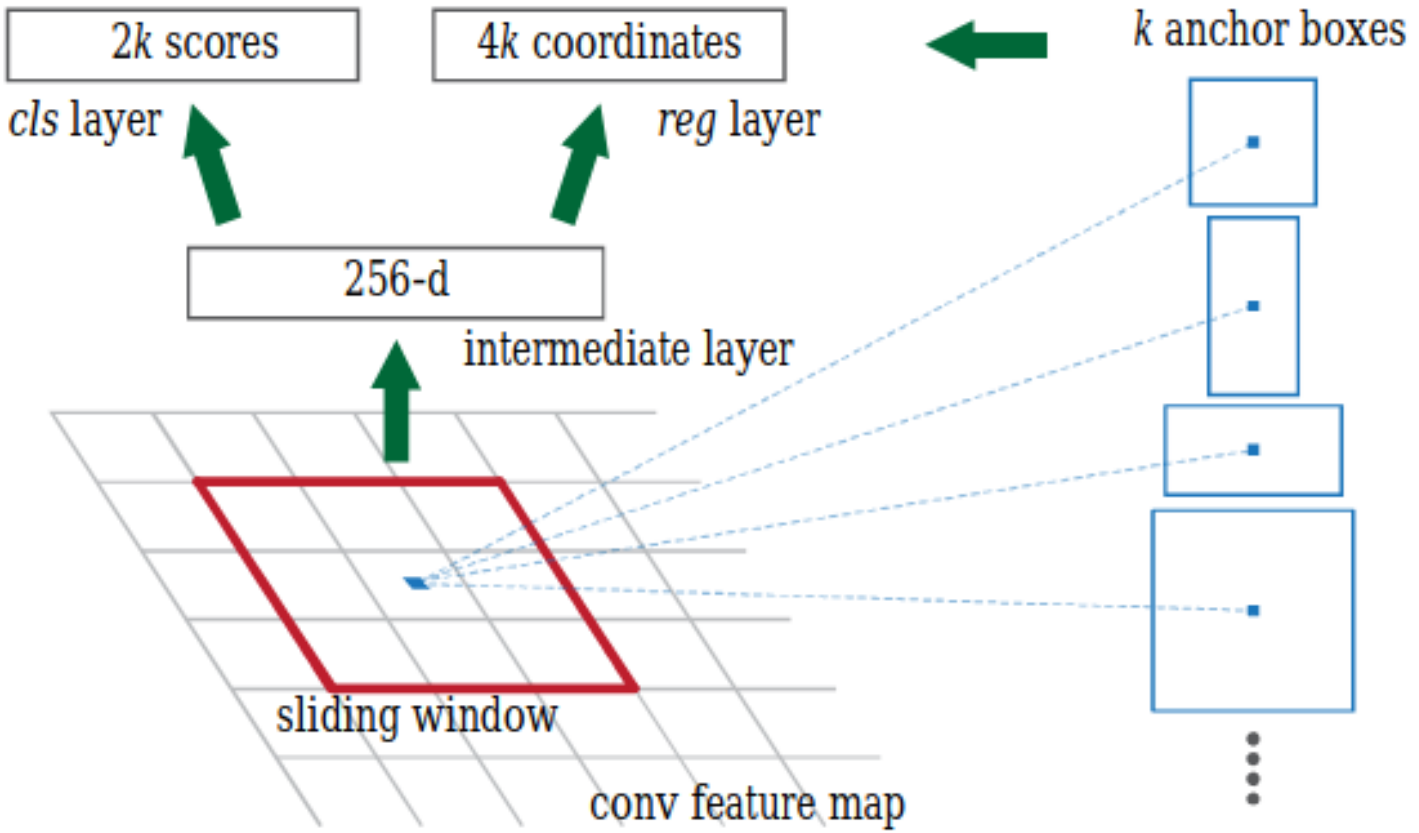


1. Extract feature map and region proposals
2. Infer class, confidence and bounding box for each proposal

M5 – Introduction to Object Detection and Segmentation

Faster R-CNN [paper]

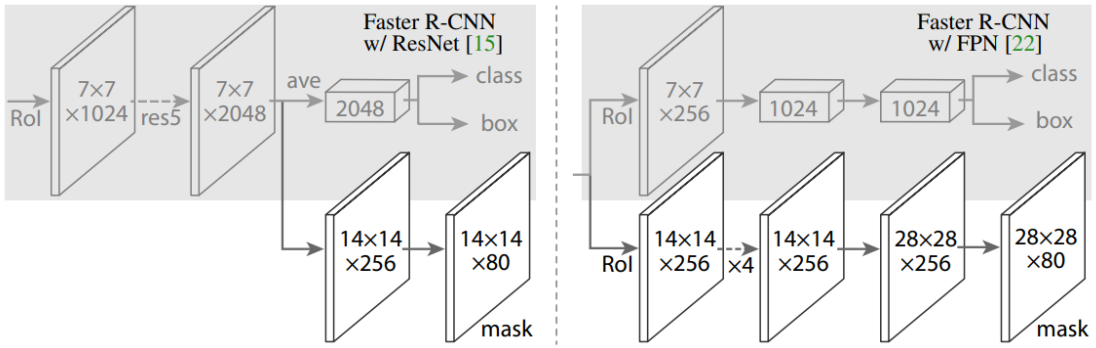
- Novel RPN



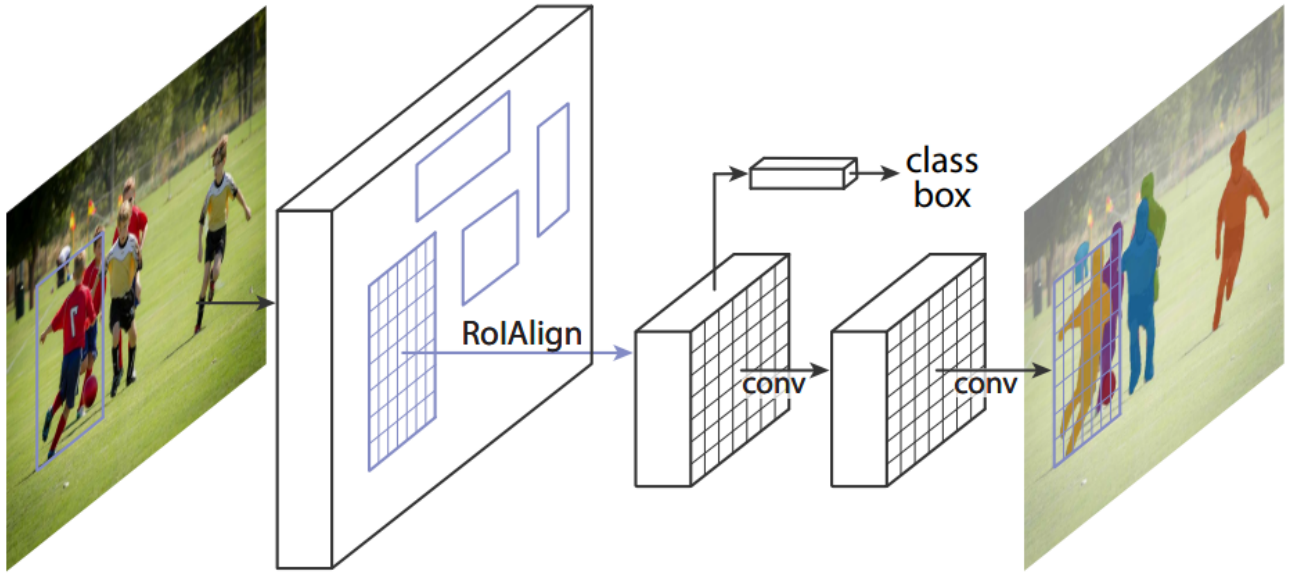
M5 – Introduction to Object Detection and Segmentation

Mask R-CNN [paper]

- Extension of Faster R-CNN
- Predicts a binary mask for each RoI head.



Head architecture

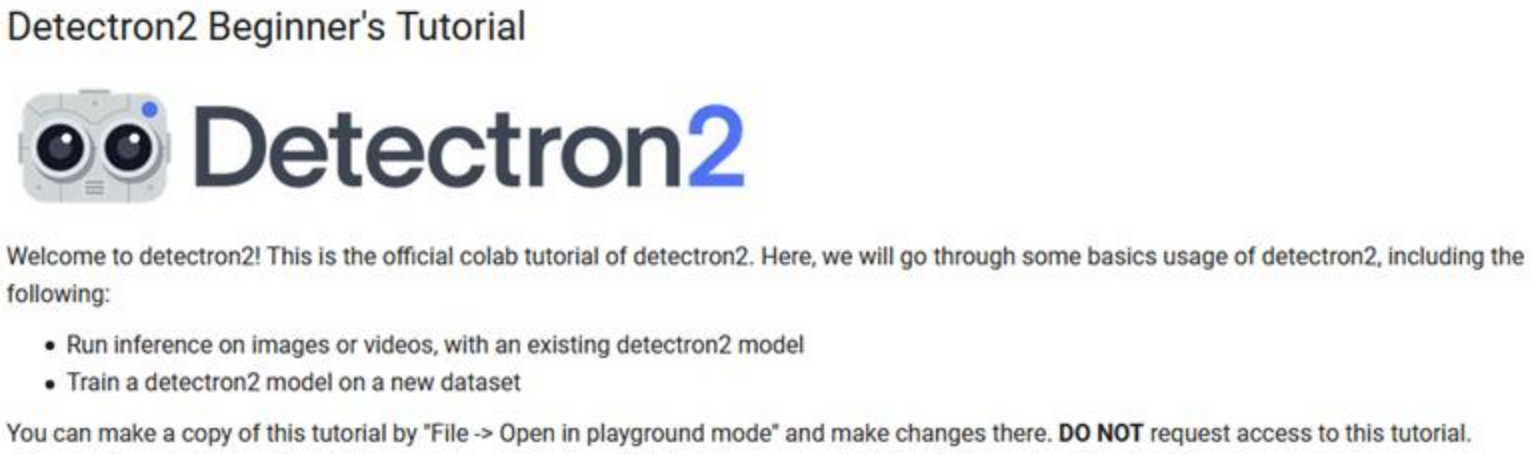


Mask R-CNN

M5 – Introduction to Object Detection and Segmentation

How to start using Detectron2?

- Detectron2 beginner's tutorial



Detectron2 Beginner's Tutorial

Detectron2

Welcome to detectron2! This is the official colab tutorial of detectron2. Here, we will go through some basics usage of detectron2, including the following:

- Run inference on images or videos, with an existing detectron2 model
- Train a detectron2 model on a new dataset

You can make a copy of this tutorial by "File -> Open in playground mode" and make changes there. **DO NOT** request access to this tutorial.

M5 – Introduction to Object Detection and Segmentation

How to start using Detectron2?

- Official installation instructions
- M5 installation instructions

Note: Detectron2 beginner's tutorial is thought to be run in Google Colab, not on the server. Therefore:

- You need to install opencv-python (included in instructions)
- Check PIL version => If problems:

```
!pip install Pillow==5.3.0
```

- You can't use cv2_imshow from google.colab.patches but

```
cv2.imwrite(img_filename,img) #otherwise
```

M5 – Introduction to Object Detection and Segmentation

Dataset: KITTI-MOTS

- Tracking and segmentation of CARS and PEDESTRIANS
  - Training data
    - 12 sequences
    - 8,073 pedestrian masks + 18,831 car masks
  - Validation data
    - 9 sequences
    - 3,347 pedestrian masks + 8,068 car masks
  - Testing data: 29 sequences

(*) More details on training and validation split in the original paper: link

M5 – Introduction to Object Detection and Segmentation

Dataset: KITTI-MOTS

Annotation format

- Class ids:
  - 1 -> car
  - 2 -> pedestrian
- Class id is obtained by floor division by 1000
- Instance id is obtained by modulo 1000
- Car instances: 1000, 1001, 1002, etc.
- Pedestrian instances: 2000, 2001, 2002, etc.

Week 2. Introduction to Object Detection and Instance segmentation

Details on tasks, deliverables, and marks for this week

Week 1	**Introduction to Pytorch - Image Classification**
Week 2	**Object Detection, Recognition and Segmentation I**
Week 3	**Object Detection, Recognition and Segmentation II**
Week 4	**Image Retrieval**
Week 5	**Cross-modal Retrieval**
Week 6	**Presentation**

M5 Project: Goals per week

Week 2: Introduction to Object Detection and Instance Segmentation

Goals

a. Get familiar with Detectron2 framework.
b. Set up project
c. Run inference with pre-trained Faster R-CNN (detection) and Mask R-CNN (detection and segmentation) on KITTI-MOTS dataset.
d. Evaluate pre-trained Faster R-CNN (detection) and Mask R-CNN (detection and segmentation) on KITTI-MOTS dataset.
e. Fine-tune Faster R-CNN and Mask R-CNN on KITTI-MOTS
f. Start writing paper

Marks

(C) Achieve (a)-(d)
(B) Achieve (a)-(e)
(A) Achieve (a)-(f) goals

Deliverable (for next week)

- **Github** repository (code explanation & instructions)
- Presentation with information about models and results.
  - Include 1 final summary slide
- Report on overleaf about object detection and segmentation.

M5 – Introduction to Object Detection and Segmentation

Task (a): **Get familiar with Detectron2 framework**

- Installation of the framework
- Follow Detectron2 beginner's tutorial

Task (b): **Set up project**

- Review descriptions of the official challenge (KITTI-MOTS).
- Get familiar with how to read images and annotations.
- You will find KITTI-MOTS dataset on the server.
  - /home/mcv/datasets/KITTI-MOTS/

M5 – Introduction to Object Detection and Segmentation

Task (c): **Run inference with pre-trained Faster R-CNN (detection) and Mask R-CNN (detection and segmentation) on KITTI-MOTS dataset.**

- Apply Faster R-CNN using Detectron2 framework with pretrained weights of COCO on KITTI-MOTS dataset.
- Apply Mask R-CNN using Detectron2 framework with pretrained weights of COCO on KITTI-MOTS dataset.
- Provide dataset description as well as qualitative results on your **presentation**.

M5 – Introduction to Object Detection and Segmentation

Task (d): **Evaluate pre-trained Faster R-CNN (detection) and Mask R-CNN (detection and segmentation) on KITTI-MOTS dataset**

- Apply Faster R-CNN using Detectron2 framework with pretrained weights of COCO on KITTI-MOTS dataset.
- Apply Mask R-CNN using Detectron2 framework with pretrained weights of COCO on KITTI-MOTS dataset.
- Use official validation partition of KITTI-MOTS as your test set.
- Don't use KITTI-MOTS evaluation metrics. Instead use official COCO metrics provided by Detectron2.
  - You will have to map class labels of KITTI-MOTS to class labels of COCO. Modify MetadataCatalog: Official documentation, detectron2 thread
- Provide metric description as well as quantitative results on your **presentation**.

M5 – Introduction to Object Detection and Segmentation

Task (e): **Fine-tune Faster R-CNN and Mask R-CNN on KITTI-MOTS.**

- Train Faster R-CNN and Mask R-CNN using Detectron2 framework on KITTI-MOTS dataset.
  - Split original training set into training and validation sets.
- Evaluate fine-tuned models on your test set using COCO metrics.
- Compare results with pre-trained models without finetuning.
- Include quantitative results and comparison in your **presentation**.

M5 – Introduction to Object Detection and Segmentation

Task (f): **Start writing paper**.

- You can use CVPR paper format.
- Include a brief introduction to the task (Max ½ page)
- Related work of object detection and instance segmentation (Max 1 page).
- Description of Datasets and metrics
- Write your experiments with your quantitative and qualitative results.
  - Experiments in presentation should be more extensive than in the paper.

Recommended

- Use appropriate references wherever required.
- Use the feedback from the presentations to improve your paper.

M5 – Introduction to Object Detection and Segmentation

General tips

- The **weekly presentation** objective is to follow the **students' progress**. Therefore, extensive experiments are welcome. You should also include problems you faced and examples you find interesting.
- The final **report/paper** objective is to **summarize** your work and teach you how to write a paper. Only the most relevant experiments and qualitative results are expected. Those, from where you can get **relevant** conclusions.
- Don't limit the results section to show the results. You must compare and get some insights or conclusions of the results of your experiments.

M5 – Introduction to Object Detection and Segmentation

- **Code** on Github project

- Report your results in your **presentation**.
  - Remember **1 minute slide** to present summary of conclusions and difficulties.

- **Overleaf** link on your Github

Due date: Monday 21st March before 10:00 AM