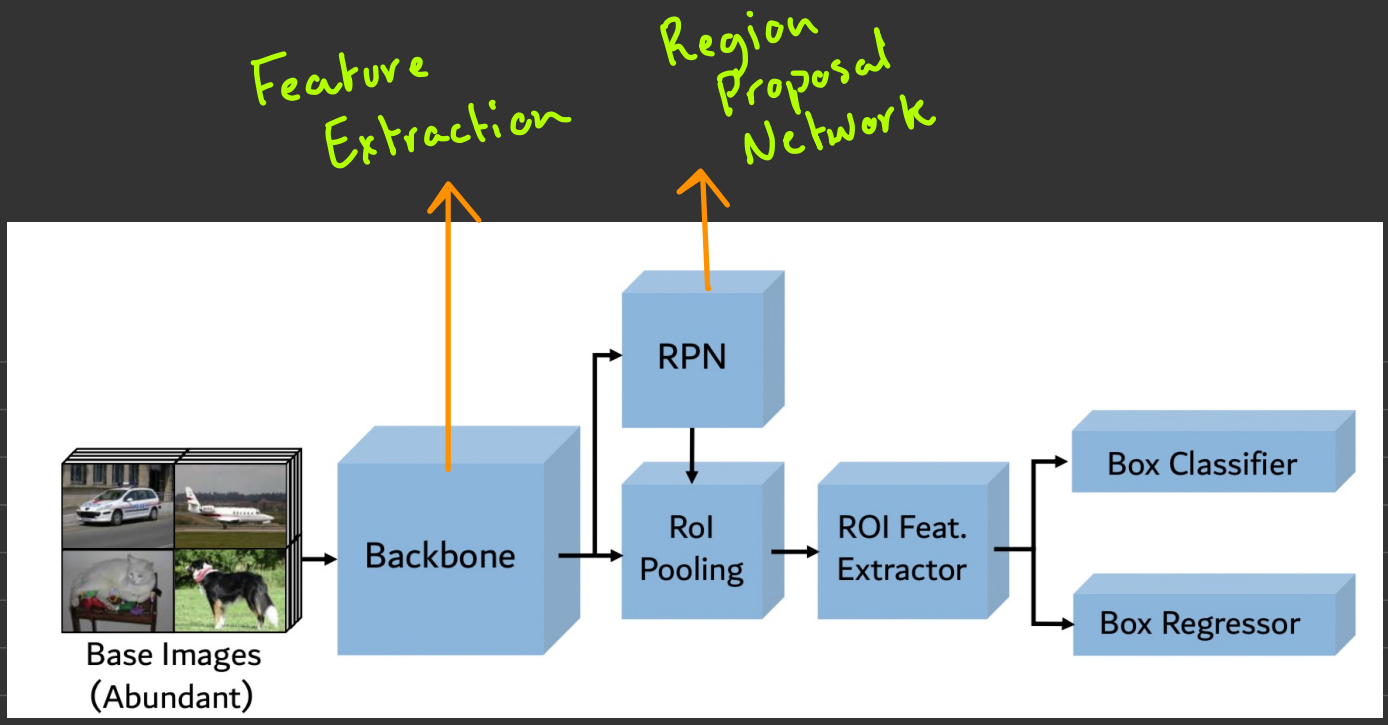
Abstract: Simple and interesting but usually the obvious
Intro: What the problem is

Detectron2 has pretrained weights, on PyTorch

COCO - Common Objects in Context Dataset

Faster-RCNN - Efficient model



Look at articles

Admin part at supervisor

Few shot: Learn to class

Meta-Learning? → Classify but don't localise

Meta feature + light weight reweighting.

Proposal based

RCNN
pretrained

↔

Proposal free

single CNN

YOLOv2 simpler and faster

↓

If it is video?

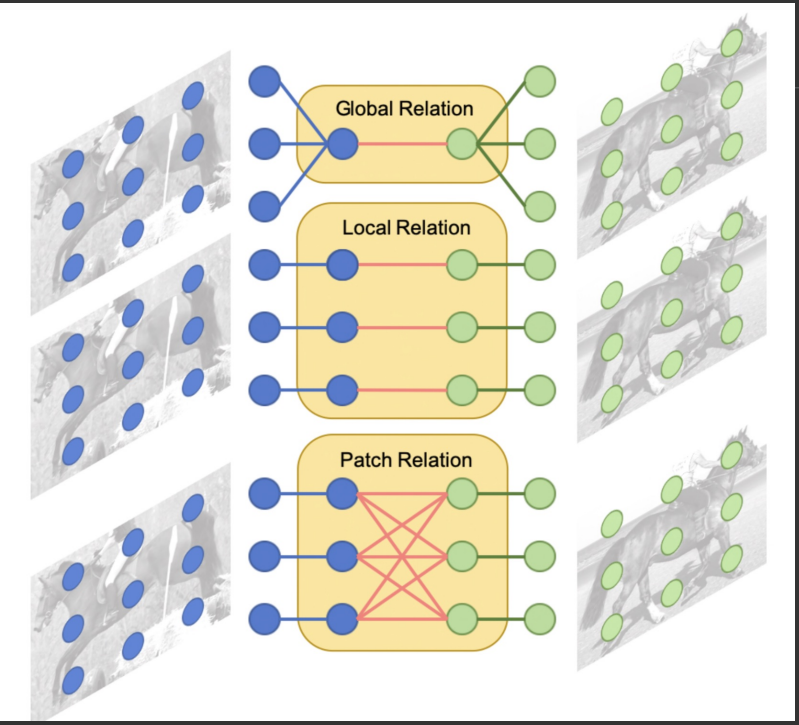
FSOD: Few Shot detection dataset

Interesting: Less images more categories

Attention network, slightly better performance

No fine-tuning required

Multi relation ->



Mask RCNN!

Simpler

<-> recognition precedes segm.

Our models can run at about 200ms per frame on a GPU, and training on COCO takes one to two days on a single 8-GPU machine. We believe the fast train and test speeds, together with the framework's flexibility and accuracy, will benefit and ease future research on instance segmentation.

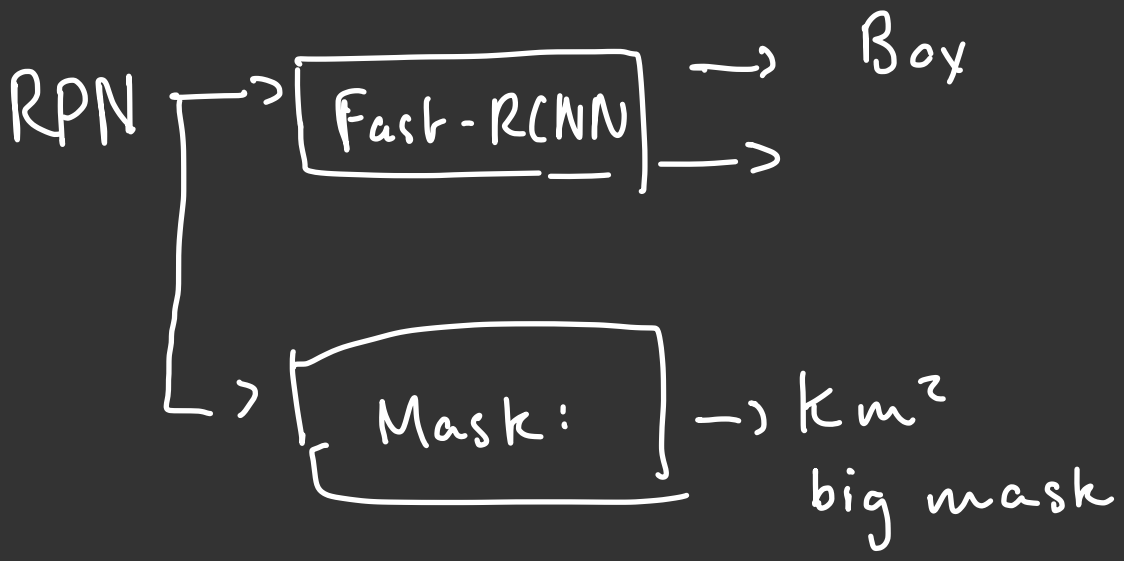
<- Siames Mask RCNN?!

Faster R-CNN:

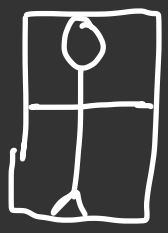
RPN -> Fast-RCNN -> Box
Fast-RCNN -> Class

Features using RoI Pool

Mask-R-CNN: Detectron?

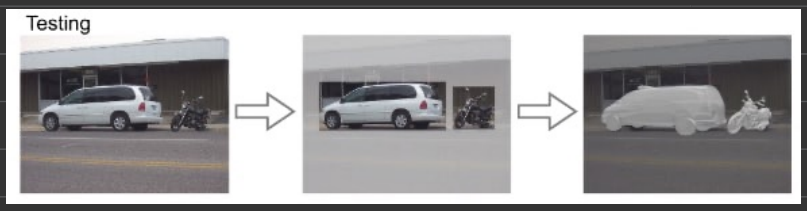


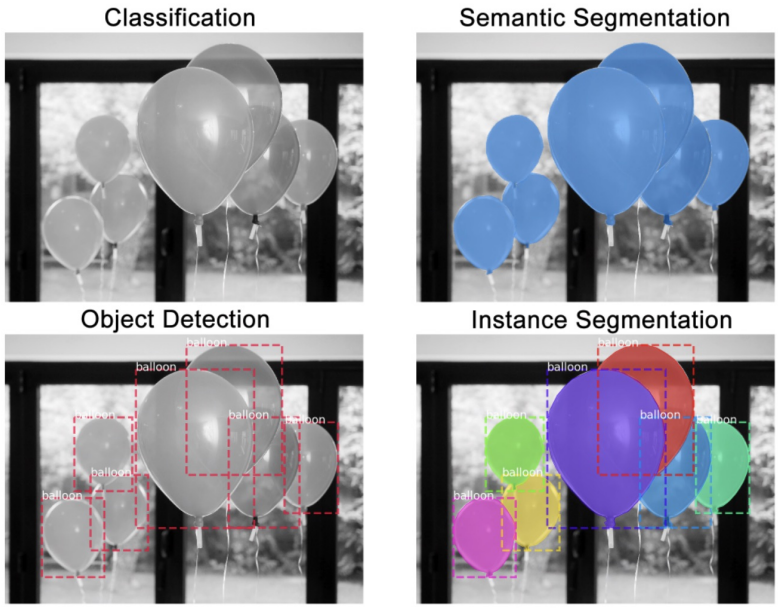
?



RoI align is used

View point estimation! 3D view

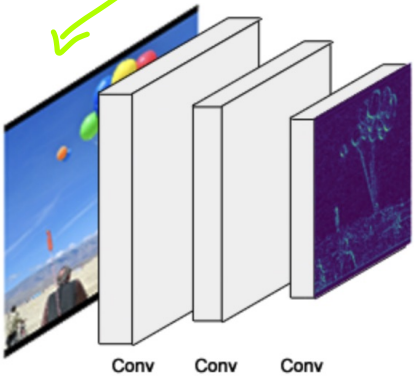




- **Classification:** There is a balloon in this image.
- **Semantic Segmentation:** These are all the balloon pixels.
- **Object Detection:** There are 7 balloons in this image at these locations. We're starting to account for objects that overlap.
- **Instance Segmentation**: There are 7 balloons at these locations, and these are the pixels that belong to each one.

What were looking for!

1. Backbone



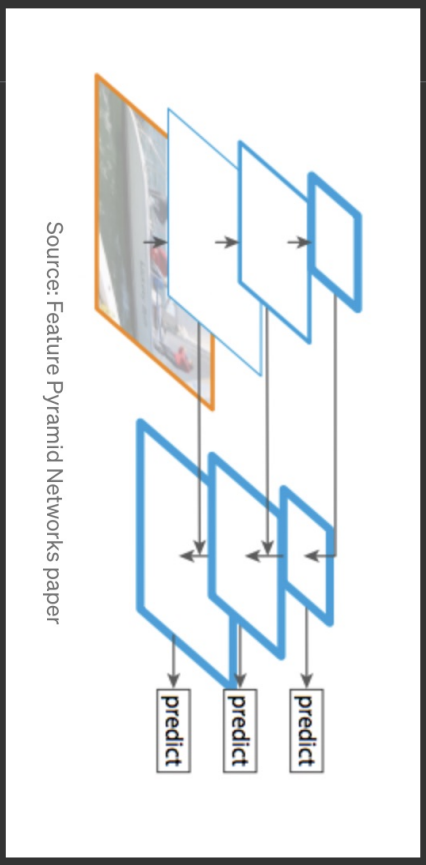
Simplified illustration of the backbone nework

1024x1024 x RGB

↓

32x32x2048

Feature map



Source: Feature Pyramid Networks paper

FPN → Get features of all scales

https://arxiv.org/abs/1612.03144

RPN → 10ms to parallel scan the whole image (Feature extraction map)

1. **Anchor Class:** One of two classes: foreground or background. The FG class implies that there is likely an object in that box.
2. **Bounding Box Refinement:** A foreground anchor (also called positive anchor) might not be centered perfectly over the object. So the RPN estimates a delta (% change in x, y, width, height) to refine the anchor box to fit the object better.

↳ then takes this and produces two outputs:
Class and BBox

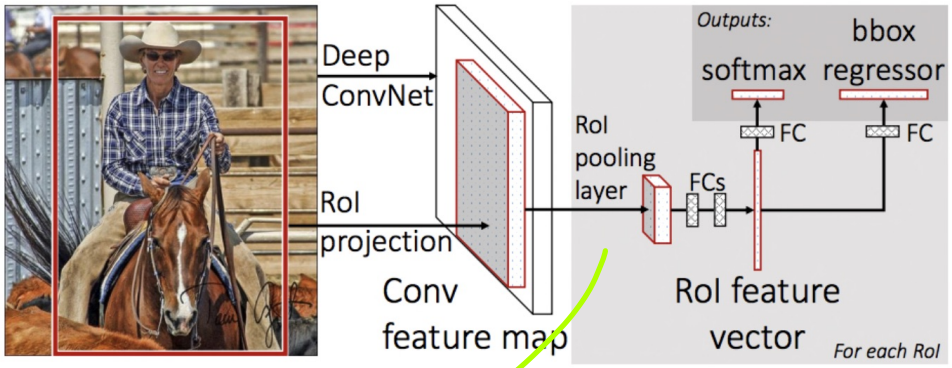
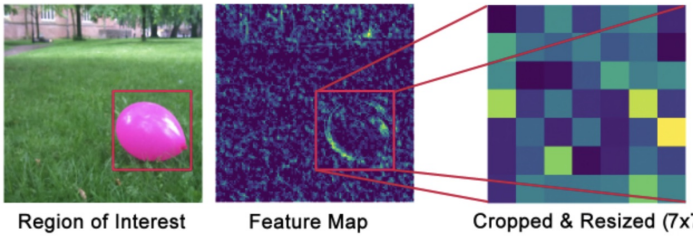


Illustration of stage 2. Source: Fast R-CNN (https://arxiv.org/abs/1504.08083)

Classifier needs fixed size,
thus crop and resize to 7x7



This is Faster-RCNN.

Mask RCNN add parallel masker
which does 28x28 pixel mask
with FP values