Yolo V4

Yolo V4

Backbone
)→ neck
Head

? Backbone의 역할: feature extraction

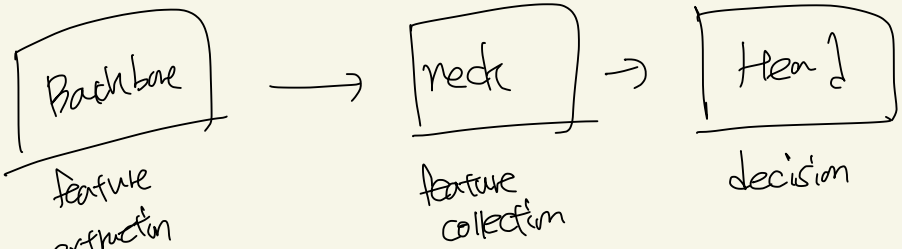
? Head의 역할: decision

Why neck? fluent extracted feature transfer.

Backbone: VGG / Resnet / ResneXT / Darknet / Densenet / Mobile
Head: Dense prediction (one stage) → RPN YOLO SSD Retina
Sparse prediction (two stage) → RCNN

Neck: FPN / PAN

Backbone: Increasing receptive field
Head: decision
Neck: used for feature collection (increasing activation size)



→ two stage detector

---

* RCNN (Region-CNN) — : bottle neck

RCNN: Selective Search + CNN for each RoI
+ FC for each RoI

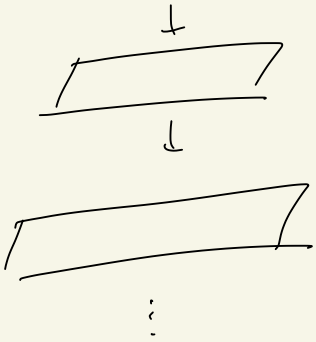
Fast RCNN: selective search + Conv + FC for each RoI

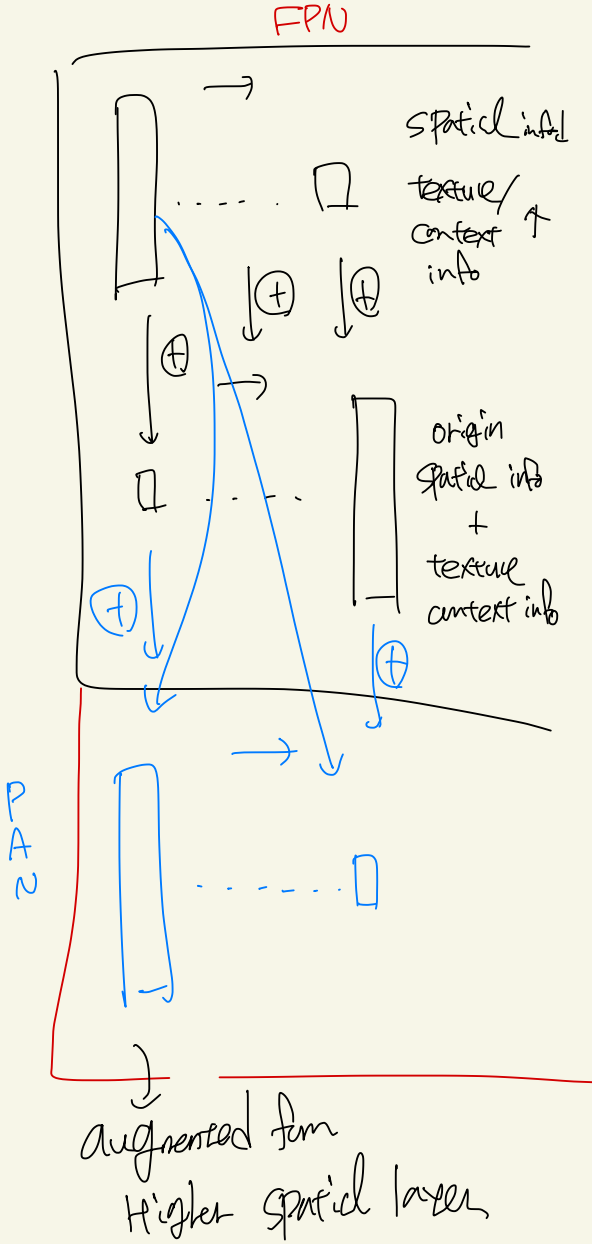
Faster RCNN: RPN (CNN) + RoI pooling for each RP
+ FC for each RoI

Anchor: Multiscale boxes that swipes out images
* slightly different from Yolo anchor

FPN: feature

How? deconv or upsample or d2s





Bag of freebies

⇒ offline cost↑

not online cost

Pixel wise

① photometric distortion

② geometric distortion

Block wise

① drop block

② Mix-up

③ Cut Mix

④ Mosaic

Semantic (class imbalance)

① Focal loss: $(1-p_t)^{\gamma} \log(p_t)$

② Knowledge distillation (Label smoothing)

: $y \cdot (1-\eta) + \frac{\eta}{n}$

BBox regression

→ coords → value base loss × (such as l_2)

→ IoU loss

Bag of specials

⇒ online cost↑

Accuracy ↑

Receptive field ↑

SPP ASPP RFB

Attention

softmax $\left(\frac{QK^T}{\sqrt{d_k}}\right)V$ (Self)

Activation

Swish/Mish

↳ differentiable or 0

Post processing

NMS.

Yolov4

① Select Backbone

② Select Neck

③ Select Head

---

① Backbone ⇒ CPSD Darknet +SPP

Note) Good classifier ≠ good at detector

1. Higher resolution
2. Higher receptive field
3. More params (for greater capacity)

→ Why? ① Big object size

② Context (deeper network size)

② Neck : PAN

③ Head : YoloV3

Additional methods

1. SAT (self adversarial) training

2. Genetic ⇒ Hyperparam Tuning

3. SAM, PAN, BN
   ↳ Modified
   ↳ easy letter paper!