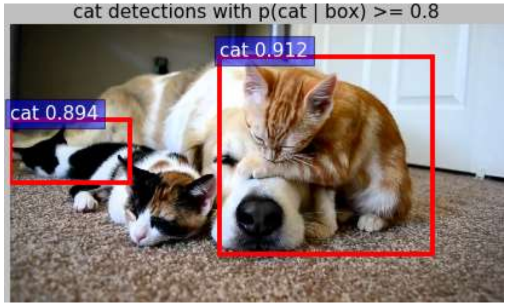
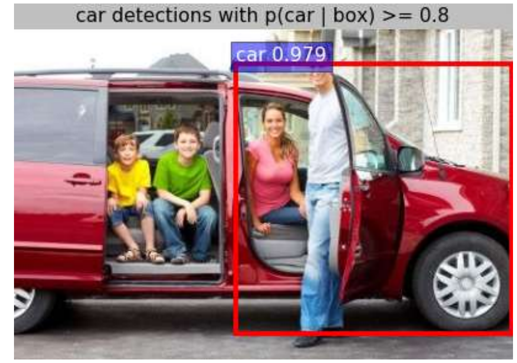
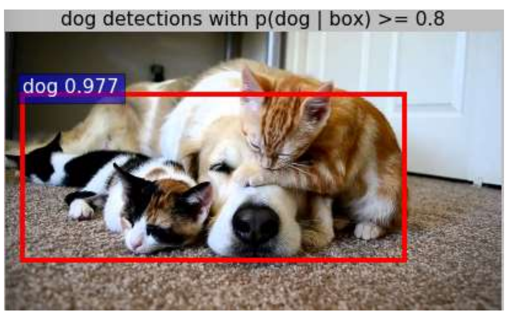




(Towards) Real-Time Object Detection with DeepNets

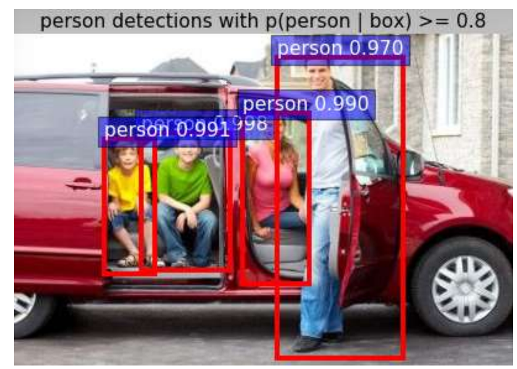
Implementer's Perspective of R-CNN, Fast R-CNN, and **Faster R-CNN**



McGill Deep Learning Reading Group

Anqi Xu

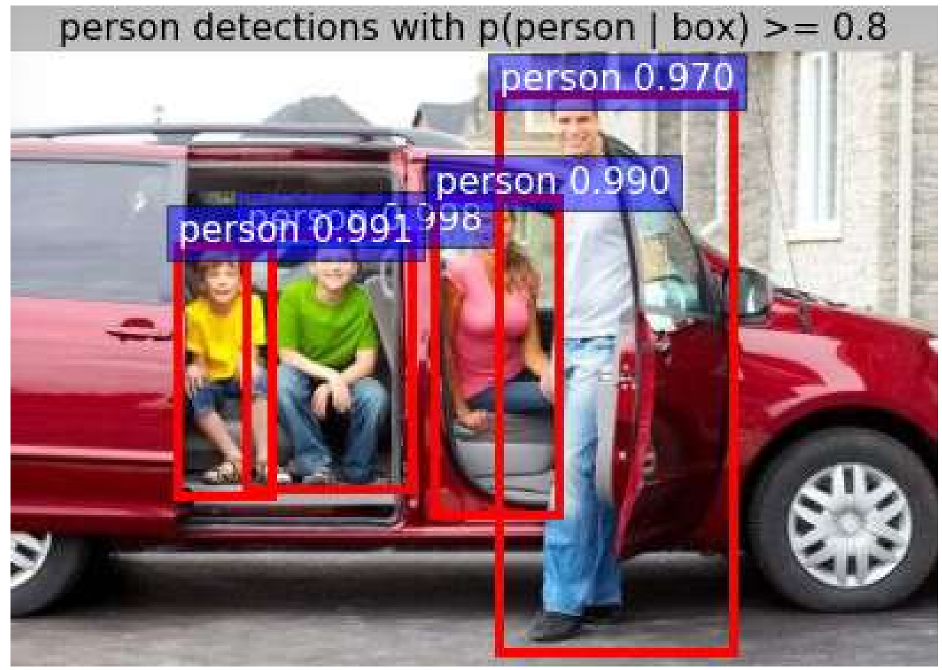
Nov. 30th, 2016



Object Detection Problem

- What: *Locate* and *detect* **object (classes)** in images

- Why?
  - Automated Scene Understanding
  - Vision-based Robotics Control
  - Visual Human-Automation Interaction
  - CV is cool!
  - Etc.

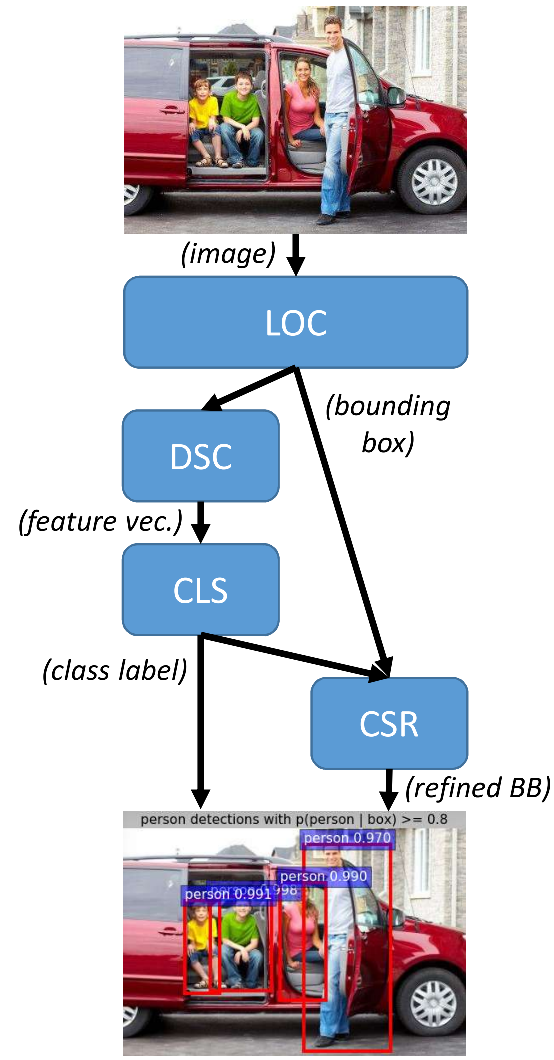


Object Detection Problem: How

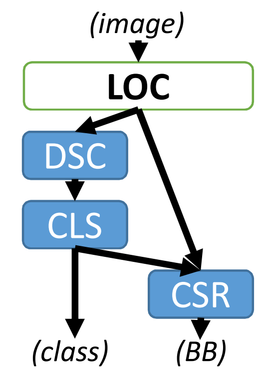
- Object Localization (LOC)

- Object Classification (via Features)
  - Feature Descriptor (DSC)
  - Feature-Based Classifier (CLS)

- Class-Specific Localization Refinement (CSR)



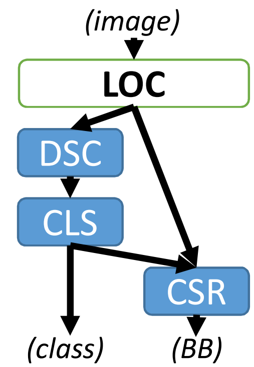
LOC - Object Localization Overview



- Region Proposal: region (e.g. BB) *possibly* containing object

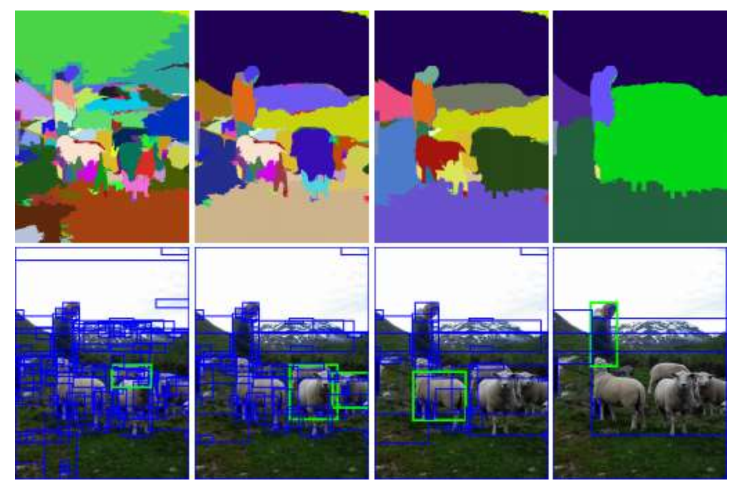
- Approaches
  - Greedy Search (a.k.a. exhaustive convolution of window-based object detection)
  - Objectness: object likelihood of image windows
  - Selective Search
  - Region Proposal Network

LOC: Selective Search

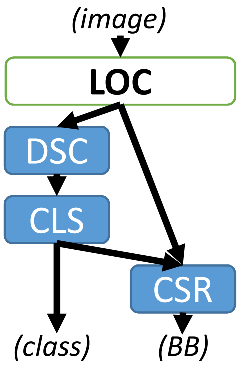


- Uijlings *et al.*; Selective Search for Object Recognition; IJCV '12.

- Algorithm: Hierarchical Grouping
  - Graph-based super-pixel segmentation into regions $R = \{r_i\}$
  - Initialize similarity set with pairwise similarity $S = \{s(r_i, r_j)\}$
  - While $S \neq \emptyset$:
    - Get $s^{max}(r_i, r_j)$
    - Merge $r_t = r_i \cup r_j$
    - Remove (r_i, r_*), (r_*, r_j) and add (r_t, r_*) to S
    - Update $r_t \rightarrow R$
  - Return BB of each region in R



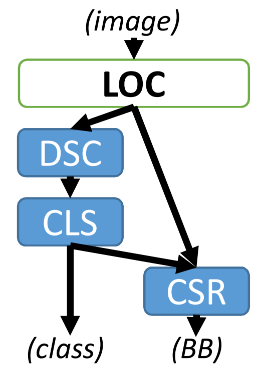
LOC: Selective Search (cont.)



- Diversification Strategy A: multiple colour spaces
  - RGB, intensity, Lab, normalized rg, HSV, normalized rgb, C, Hue

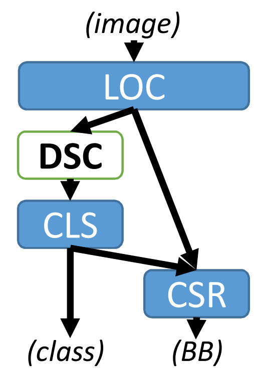
- Diversification Strategy B: multiple region similarity metrics
  - Color: histogram similarity
  - Texture: HOG histogram
  - Size: pixel count
  - Fill: joint BB size - pixel count i – pixel count j

LOC: Selective Search (cont.)



- Full Algorithm
  - Compute groupings using combinations of colour spaces x similarity metrics
  - Rank all object hypotheses based on grouping order * rand[0,1]
  - “Filter out lower ranked duplicates” (NMS based on IoU overlap?)

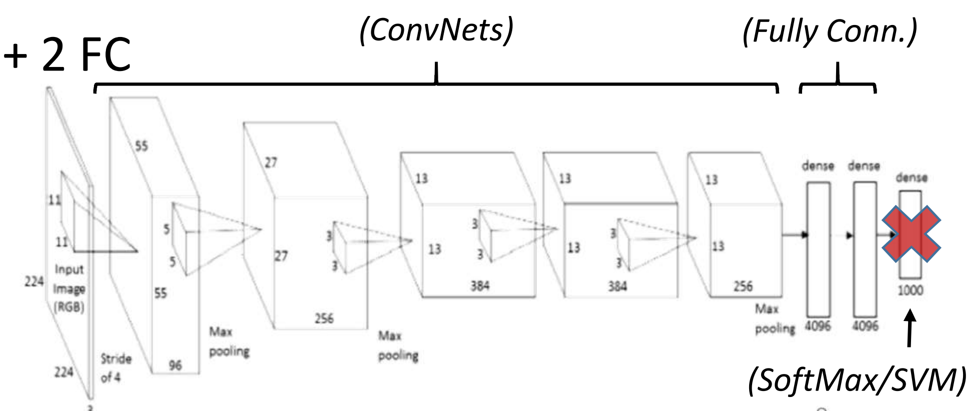
- Selective Search “Fast Mode”
  - {HSV, Lab, C+T+S+F, T+S+F} x {k=50,100}
  - 8 strategies, ~2k windows, 0.799 MABO, 3.79s



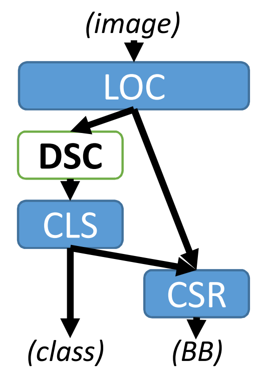
DSC: Feature Descriptor

- (Semantic) vectorized data compression of pixel data
  - Engineered edge-based descriptors (e.g. HOG, wavelet, etc.)
  - Truncate classifier Deep Nets: Conv Layers + FC Layers
  - Fixed-size RoI descriptor via RoI Pooling

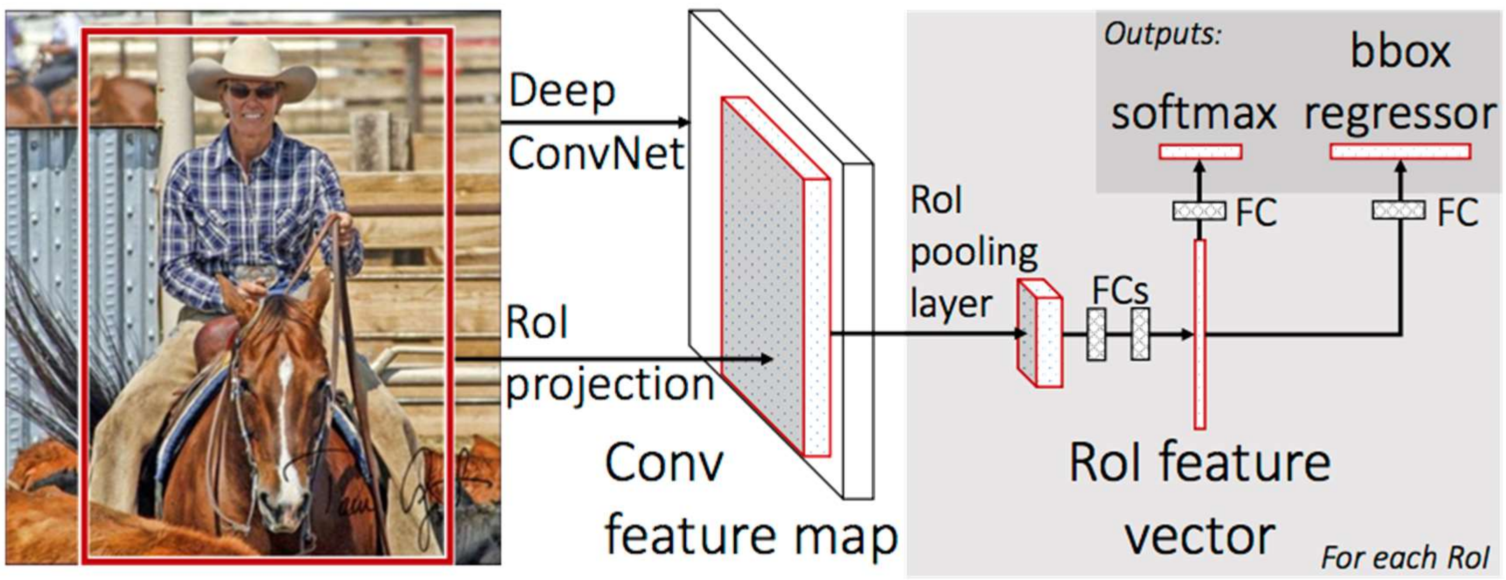
- Deep Nets, yo!
  - AlexNet/T-Net/CaffeNet (Hinton): 5 CN + 2 FC
  - Zeiler+Fergus: 5 CN + ? FC
  - VGG16 (Zisserman): 16 CN + 3(?) FC



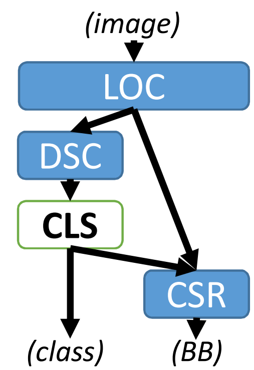
DSC: RoI Pooling (Fast R-CNN)



- Downsize (h,w) RoI BB into fixed H x W (e.g. 7 x 7) descriptor
  - Per-channel max-pooling

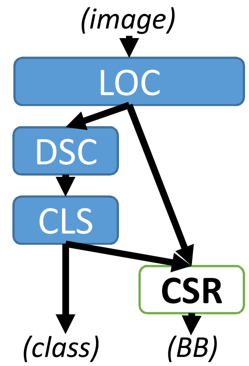


CLS: Feature-Based Classifier



- R-CNN Approach:
  - (Source image) -> warped RoI image -> Conv Layers -> (feature) **-> FC Layers -> class-specific SVMs**

- Fast(er) R-CNN Approach:
  - (Source image) -> RoI from region proposal step -> RoI Pooling Layer-> (feature) **-> FC Layers -> N+1 softmax Layer**



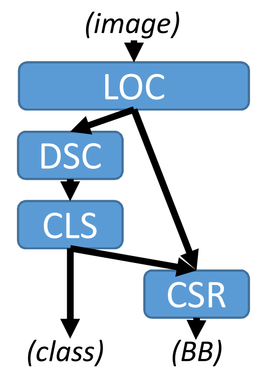
CSR: Class-Specific BB Refinement

- R-CNN Approach: Bounding Box ridge regression

$$
\begin{aligned}
t_x &= (G_x - P_x)/P_w \\
t_y &= (G_y - P_y)/P_h \\
t_w &= \log(G_w/P_w) \\
t_h &= \log(G_h/P_h).
\end{aligned}
$$

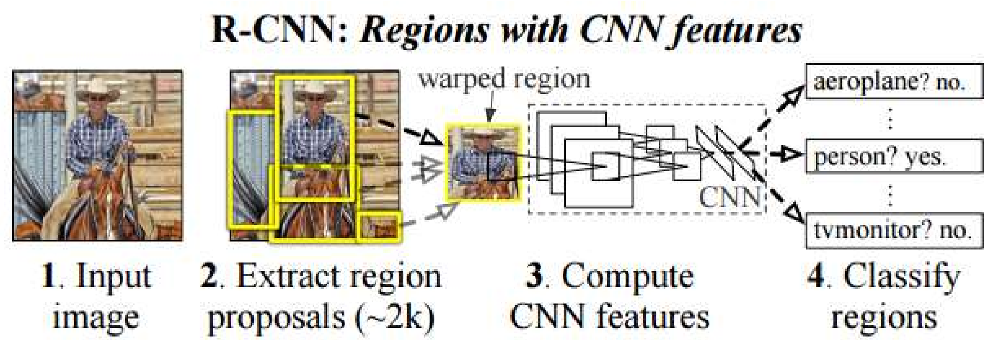
- Fast(er) R-CNN Approach: FC(s) + bounding box regressor (layer?)

$$
\begin{aligned}
t_{\mathrm{x}} &= (x - x_{\mathrm{a}})/w_{\mathrm{a}}, \quad t_{\mathrm{y}} = (y - y_{\mathrm{a}})/h_{\mathrm{a}}, \\
t_{\mathrm{w}} &= \log(w/w_{\mathrm{a}}), \quad t_{\mathrm{h}} = \log(h/h_{\mathrm{a}}), \\
t^*_{\mathrm{x}} &= (x^* - x_{\mathrm{a}})/w_{\mathrm{a}}, \quad t^*_{\mathrm{y}} = (y^* - y_{\mathrm{a}})/h_{\mathrm{a}}, \\
t^*_{\mathrm{w}} &= \log(w^*/w_{\mathrm{a}}), \quad t^*_{\mathrm{h}} = \log(h^*/h_{\mathrm{a}}),
\end{aligned}
$$



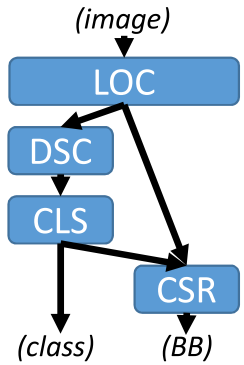
Big Picture: R-CNN (2012)

- Components
  - LOC: Selective Search “fast mode”
  - DSC: ConvNet (AlexNet/VGG16) + FCs
  - CLS: class-specific SVMs
  - CSR: class-specific BB ridge regression



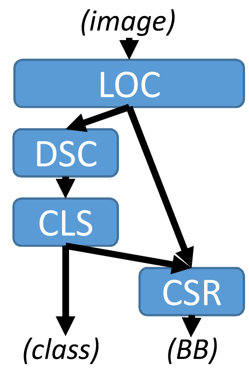
- Other Contributions
  - Supervised Pre-Training: ILSVRC12 classification (image-level annotations only, w/o BB labels)
  - Domain-specific fine-tuning: SGD on warped proposal windows (N=20 for VOC, N=200 for ILSVRC13)
  - Empirical analysis: warped RoI better than “tightest square with context” & “tightest square without context” (a.k.a. *who needs aspect ratio?*)

Big Picture: R-CNN (2012) (cont.)

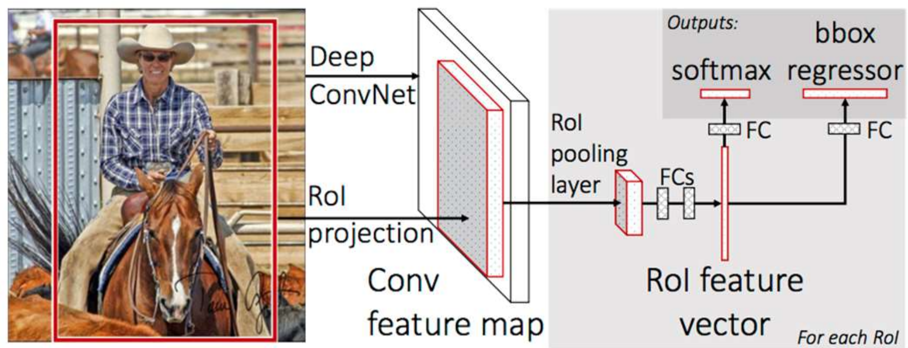


- Training Time: a few days on GTX560
- Run Time: ~10 secs on 2012-era GPU
- VOC07 test mAP: 58.5% (R-CNN BB) vs 34.3% (DPM HSC)
- ILSVRC13 mAP: 31.4% (R-CNN BB) vs 24.3% (Overfeat posthoc)

Big Picture: Fast R-CNN (2014)

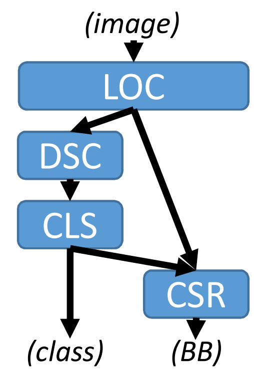


- Updated Components
  - LOC: *Selective Search "fast mode" (unchanged)*
  - DSC: ConvNet (CaffeNet/VGG_CNN_M_1024/VGG16) **on whole image** + RoI pooling layer + FCs
  - CLS: FC + softmax layer
  - CSR: FC + regressor layer



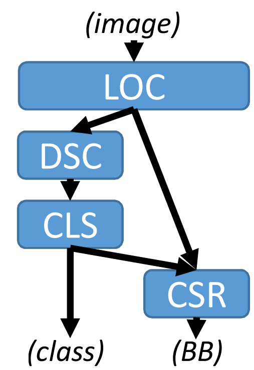
- Other Contributions
  - Efficient backprop via Mini-Batch SGD: N=2 images, R=128 total regions
  - Multi-task loss: log class likelihood + L-1 (x,y,w,h) BB regression
  - Approximate scale normalization by matching image pyramid w/ RoI size
  - Truncated SVD approximation of FC layers

Big Picture: Fast R-CNN (2014) (cont.)

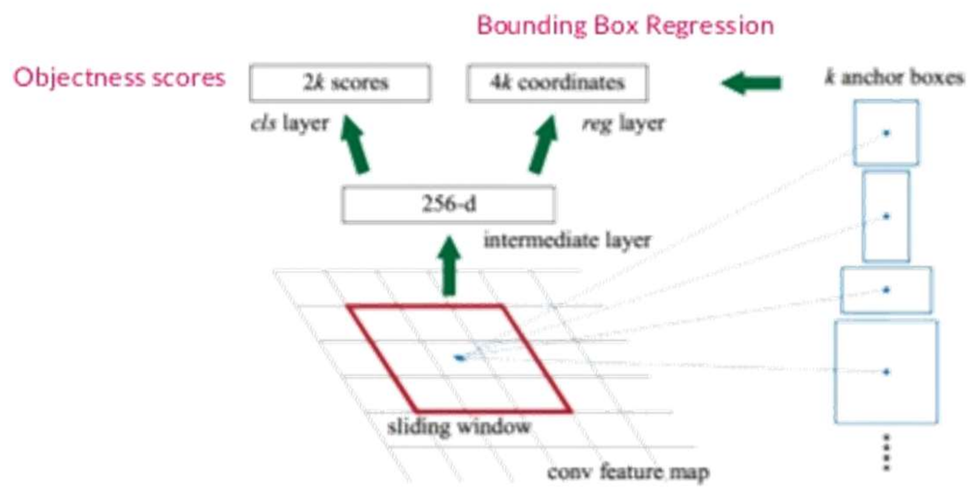


- Training Time: 1.2-9.5h (8-18x speedup)
  - On GTX 770?
- Run Time: 0.10-0.32s w/o SVD, 0.06-0.22s w/ SVD
- VOC07: 70.0% (Fast R-CNN) vs 66.0% (R-CNN BB) vs 63.1% (SPPNet BB)
- VOC12: 68.4% (Fast R-CNN) vs 62.4% (R-CNN BB) vs 63.2% (BabyLearning)

Big Picture: Faster R-CNN (2015)



- Region Proposal Network



In practice, k = 9 (3 different scales and 3 aspect ratios)

- Updated Components
  - LOC: RPN (into k anchor boxes)
  - DSC: ConvNet (ZG/VGG16) on whole image + RoI pooling layer + FCs
  - CLS / CSR: *FC + softmax layer / FC + regressor layer (unchanged)*

Faster R-CNN: Results, Demo, Discussion

- Results: see paper
- Live demo
- Discussion Seed Points
  - Failure cases
  - Improvements to individual steps ("SqueezeNet, anyone?")
  - YOLO!