AIGS 538 Homework Assignment 3 20222421 심재준 (JAEYOON SIM)

We have to summerize and compare 4 object detection papers and our report should include architecture details, training details, and performance evaluation (mAP, speed, memory etc). First, we would like to summarize the 4 object detection papers and briefly compare them at the end.

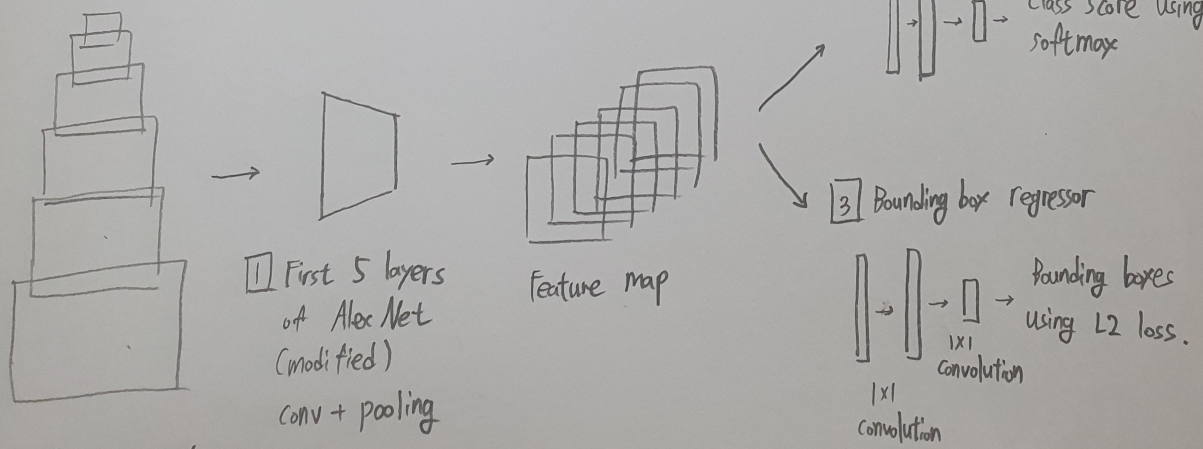
#1. OverFeat : Integrated Recognition, Localization and Detection using Convolutional Networks. Pierre Sermanet, David Eigen, Xiang Zhang, Michael Mathieu, Rob Fergus, Yann LeCun, CVPR 2014

This paper presents an integrated framework for using Convolutional Networks for classification, localization and detection. They show how a multiscale and sliding window approach can be efficiently implemented within a ConvNet. They also introduce a novel deep learning approach to localization by learning to predict object boundaries. Bounding boxes are then accumulated rather than suppressed in order to increase detection confidence. There are 3 vision tasks in increasing order of difficulty and they are classification, localization, and detection. Each task is a sub-task of the next. All tasks are adressed using a single framework and a shared feature learning base.

In the classification task of this challenge, each image is assigned a single label corresponding to the main object in the image. 5 guesses are allowed to find the correct answer. The localization task is similar in that 5 guesses are allowed per image, but in addition, a bounding box for the predicted object must be turned with each guess. To be considered correct, the predicted box must match the groundtruth by at least 50%, as well as be labeled with the correct class. The detection task differes from localization in that there can be any number of objects in each image, and false positives are penalized by the mean average precision (mAP) measure.

Their classification architecture is similar to the best ILSVRC12 architecture. But, they improve on the network design and the inference step. They train the network on the ImageNet 2012 training set. Their model uses the same fixed input size approach during training but turns to multi-scale for classification. Each image is downsampled so that the smallest dimension is 256 pixels. Then, they extract 5 random crops of size 221 x 221 pixels and present these to the network in mini-batches of size 128. The weights in the network are initialized randomly and they are updated by stochastic gradient descent. The learning rate is successively decreased and dropout is employed on the fully connected layers in the classifier. During the training, they treat their architecture

as non-spatial, as opposed to the inference step, which produces spatial outputs. They release a feature extractor named "OverFeat" in order to provide powerful features. Two models are provided, a fast and accurate one. They explore the entire image by densely running the network at each location and at multiple scales. While the sliding window approach may be computationally prohibitive for certain types of model, it is inherently efficient in the case of ConvNets. In contrast to many sliding-window approaches that compute an entire pipeline for each window of the input one at a time, ConvNets are inherently efficient when applied in a sliding fashion because they naturally share computations common to overlapping regions. When applying their network to larger images at test time, they simply apply each convolution over the extent of the full image. This extends the output of each layer to cover the new image size, eventually producing a map of output class predictions, with one spatial location for each "window" (field of view) of input. Starting from their classification-trained network, they replace the classifier layers by a regression network and train it to predict object bounding boxes at each spatial location and scale. They then combine the regression predictions together, along with the classification results at each location. Detection training is similar to classification training but in spatial manner. Multiple location of an image may be trained simultaneously. Since the model is conventional, all weights are shared among all locations. The main difference with the localization task, is the necessity to predict a background class when no object is present. From now on, we would like to summarize more. Now about the architecture of overfeat, it has 6-scale images as a input and get feature map with feature extractor which is trained for classification. Classifier and bounding box regressor use this feature map as a input and return spatial map. In bounding box regressor, it use greedy merge strategy algorithm to predict bounding box. Following illustration shows overall architecture of overfeat detection.



20222421 심재윤(JAEYOON SIM)

For training of classifier, the overfeat model trained in advance for the classification task up to layer 5 and use it as a feature as a feature extractor, and add layer 6 (5x5 conv), layer 7 (1x1 conv) and layer 8 (1x1 conv). The image is input to the feature extractor (1~5 layers) and the output is 256 5x5 size feature maps. The input is 5x5 size feature map to layer 6, 7, and 8 and the output is 1x1x # of classes. Lastly, they learn through a softmax loss function.

For training of bounding box regressor, they load the overfeat model trained in advance for the classification task up to layer 5 and use it as a feature extractor and add layer 6, 7, and 8 like in training of classifier. The image is input to feature extractor (1~5 layers) and the output is 256 5x5 size feature maps. Next, the input is the 5x5 size feature map to layer 6, 7, 8 and the output is the 1x1x4x # of classes size feature map. The reason why they use 4 is that there are X center, Y center, W(width), and H(height). And they use greedy merge strategy to merge unnecessary bounding boxes.

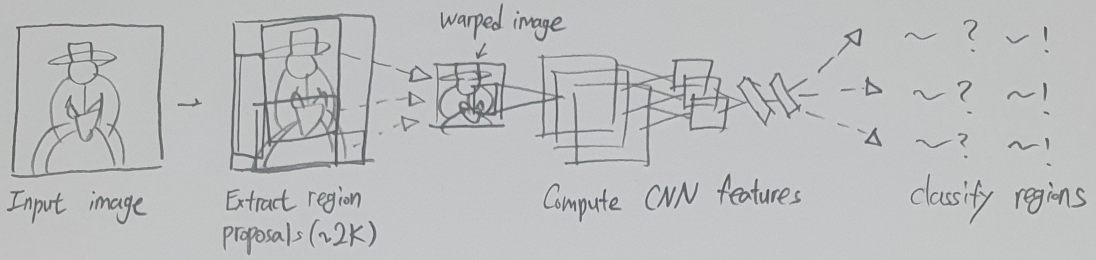
For performance evaluation in object detection, we have to know precision, recall, average precision, and IoU before studying overfeat, R-CNN, fast R-CNN, and faster R-CNN. The precision is TP/(TP+FP), and the recall is TP/(TP+FN). These two are usually in inverse proportion relation. So, average precision (AP) is used which is the area of recall-precision graph. The IoU is intersection over union and it is ratio of two bounding box overlap. It is area of overlap / area of union. The overfeat is the winner of ILSVRC13 localization competition with 29.9% error. The detection results in ILSVRC13 test set, overfeat got 24.3% mAP.

#2. Rich feature hierarchies for accurate object detection and semantic segmentation.
Girshick et al, CVPR 2014.

In this paper, they propose a simple and scalable detection algorithm that improves mean average precision (mAP) by more than 30% relative to the previous best result on VOC 2012 - achieving a mAP of 53.3%. Their approach combines two key insights where one can apply high-capacity convolutional neural networks (CNNs) to bottom-up region proposals in order to localize and sement objects and when labeled training data is scarce, supervised pre-training for an auxilary task, follwed by domain-specific fine-tuning, yields a significant performance boost. They combine region proposals with CNNs, so they called their method "R-CNN" and they also compare R-CNN to OverFeat,

20222421 심재윤(JAEYOON SIM)

a proposed sliding-window detector based on a similar CNN architectures. object detection models find the location of the objects (localization) and find what classes the objects belong to (classification). R-CNN is two-stage detector that sequentially proceeds localization and classification. The following is the overall R-CNN architecture.



Their system takes an input image, extracts around 2000 bottom-up region proposals, computes features for each proposal using a large convolutional neural network (CNN), then classifies each region using class-specific linear SVMs. For more details, first, 2000 region proposals which are likely location of objects are extracted thrugh the selective search algorithm and each is then warped to size of 227 x 227. Second, all the warped region proposals are input of the fine-tuned AlexNet to extract feature vectors of size 2000 x 4096. Next, the extracted feature vectors are input of a linear SVM model and a bounding box regressor model to obtain a confidence score and bounding box coordinates, respectively. Finally, the non maximum suppression algorithm is applied to output minimum and optimal bounding box.

For region proposal by selective search, the R-CNN model uses a selective search algorithm to extract a region proposal before estimating the location of specific object; and region proposal is the likely location of an object. Selective search algorithm groups pixels based on various criteria such as color, pattern, contrast and others. After extracting 2000 candidate regions from a single image, they warp them to size of 227 x 227 for input into the next CNN model. But, this step is done in CPU, not GPU. So, it takes so much time.

For extraction by fine-tuned AlexNet, 2000 (regions) x 4096 (dimension of feature vector) size feature vectors are extracted by putting 2000 candidate regions into fine-tuned AlexNet. They propose a method of fine-tuning a pre-trained CNN model according to the domain through the existing ImageNet dataset to predict a class suitable for specific domain during object detection. Using fine-tuned model, makes it possible to extract a more suitable feature vector domain.

2022421 심재윤 (JAEYOON SIM)

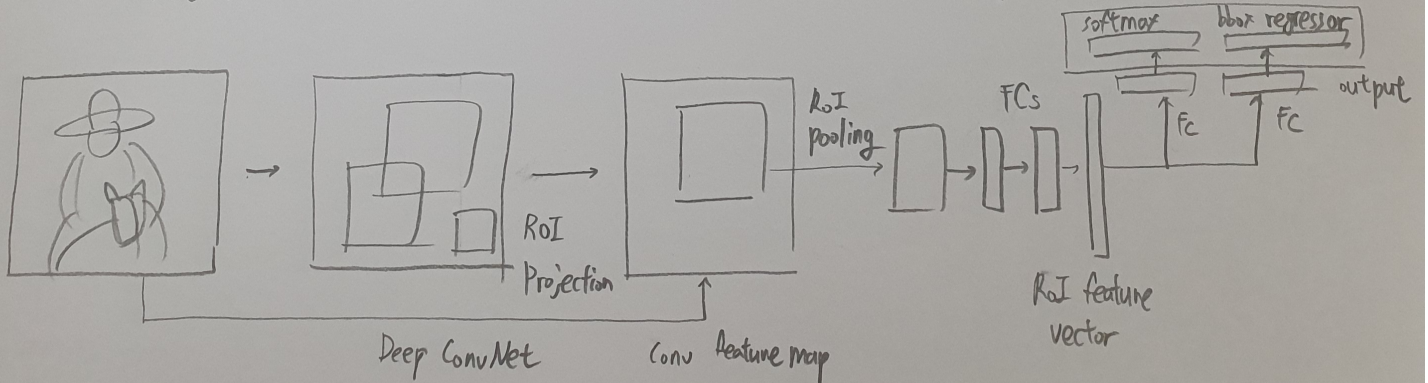
For classification of SVM, linear support vector machine (SVM) model recieves a 2000x4096 feature vector as input, predicts the class, and returns a confidence score. In this case, the linear SVM model is a binary classifier that only judges whether or not and it corresponds to specific class. Therefore, when predicting N classes, they need to train N+1 independent linear SVM models including the background.

For localization by bounding box regressor, there is bounding box for fine-tuning. Because the location of objects obtained through the selective search algorithm may be inaccurate. R-CNN achieves a mAP of 31.4% in ILSVRC 2013 competition, which is significantly ahead of second best result 24.3% from overfeat. Also, for detection average precision (%) on VOC 2010 test, R-CNN is most directly comparable to UVA and Regionlets since all methods use selective search region proposals. The mAP of R-CNN is 50.2% and that of R-CNN BB is 53.7%. And at this time, the performance of R-CNN was good on most datasets.

#3. Fast R-CNN. Ross Girshick. ICCV 2015.

This paper propses a Fast Region-based Convolutional Network method (Fast R-CNN) for object detection. Fast R-CNN builds on previous work to efficiently classify object proposals using deep convolutional networks. Fast R-CNN employs several innovations to improve training and testing speed while also increasing detection accuracy. Fast R-CNN trains the very deep VGG16 network 9x faster than R-CNN, is 213x faster at test-time, and achieves a higher mAP on PASCAL VOC 2012.

R-CNN model takes very long time to train, and the detection speed also took 47 seconds per image. Also, there is a problem that is impossible to share computations or update weight value by independently training three model (AlexNet, linear SVM, and bounding box regressor). Therefore, it is hard to find global optimal solution. Fast R-CNN is a model that shows a significant improvement in speed compared to R-CNN. Fast R-CNN can be trained in end-to-end and following is overall architecture.



20222421 심재윤 (JAEYOON SIM)

An input image and multiple regions of interest (RoIs) are input into a fully convolutional network. Each RoI is pooled into a fixed-size feature map and then mapped to a feature vector by fully connected layers (FCs). The network has two output vectors per RoI: softmax probabilities and per-class bounding-box regression offsets. The architecture is trained end-to-end with a multi-task loss. The R-CNN model takes a lot of time by putting 2000 region proposals into the CNN model and learning each one independently. Fast R-CNN improves on this problem and receives only one image, and without the need to warp the size of region proposals, a feature vector of a fixed size through RoI pooling is input into fully connected layer. It also uses multi-task loss to train the model at once without the need to train the model individually. This significantly reduced training and detection time. But, selection search is still done in CPU. so it is not that fast.

Their experiments use three pre-trained ImageNet models that are available online. The first is the CaffeNet (essentially AlexNet) from R-CNN. The second network is VGG_CNN_M_1024, which has the same depth as first model, but is wider. The final network is the very deep VGG16 model and all experiments use single-scale training and testing. For training fast R-CNN, they initialize pre-trained network and propose region by selective search. Next, feature extraction (1~13 layers pre-pooling) is done by VGG16. They then do a max pooling on the feature map by RoI pooling. Feature vectors are extracted by FC layers. Prediction of class by classifier is done and localization is done by bounding box regressor. Classifier and bounding box regressor returns multi task loss. Finally, two model classifier and regressor are trained at the same time through back-propagation.

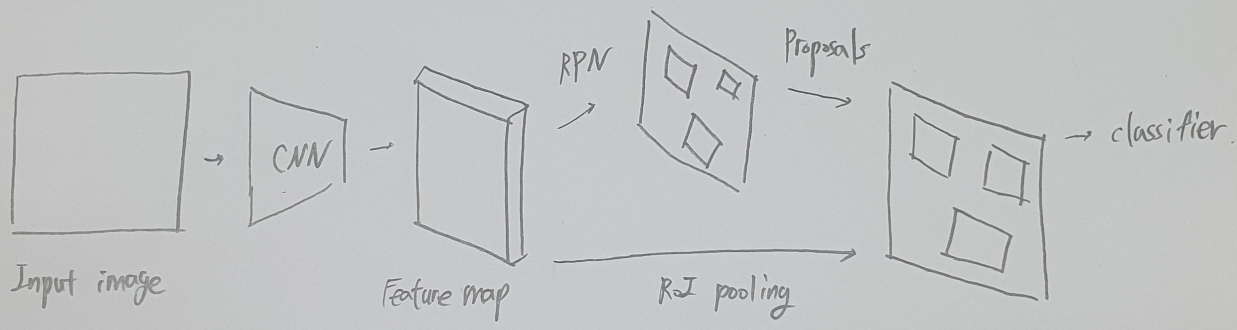
Fast R-CNN was applied to the MS COCO dataset to establish a preliminary baseline. The PASCAL-style mAP is 35.9%, the new COCO-style AP, which also averages over IoU thresholds is 19.7%.

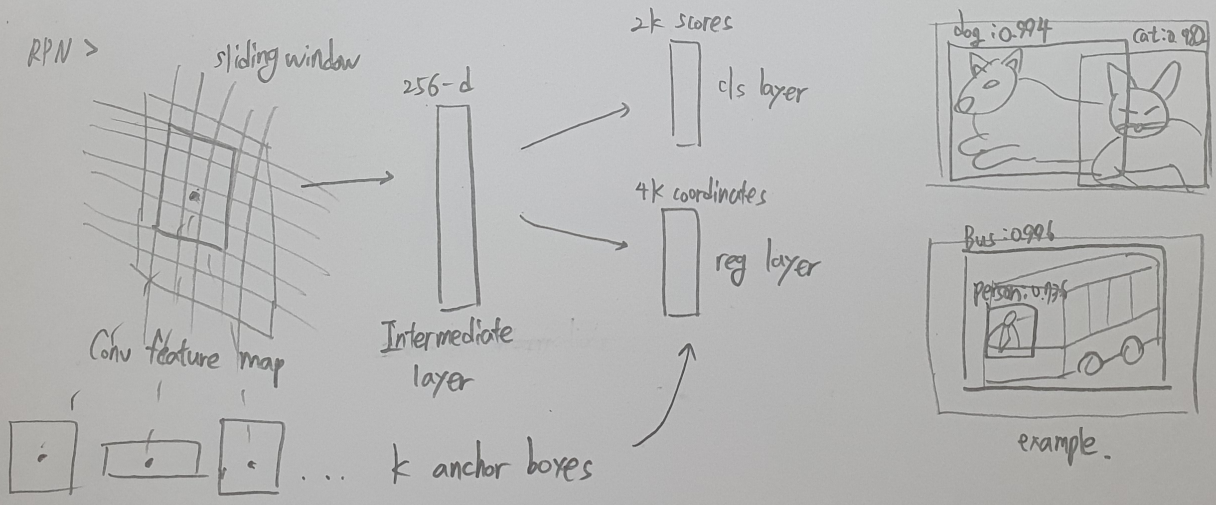
#4. Faster R-CNN: Towards Real-Time object Detection with Region Proposal Networks.

Ren et al, NIPS 2015.

Most of object detection networks depend on region proposal algorithm to hypothesize object locations. Advances like SPPnet and Fast R-CNN have reduced the running time of these detection networks, exposing region proposal computation as a bottleneck. In this paper, they introduce a Region Proposal Network (RPN) that shares full-image convolutional features with the detection network,

thus enabling nearly cost-free region proposals. Fast R-CNN model still have limitations in improving training and detection speed because they extract region proposals through a selective search algorithm. An RPN performs candidate region extraction to solve this problem and it is a fully convolutional network that simultaneously predicts object bounds and objectness scores at each position. The RPN is trained end-to-end to generate high-quality region proposals, which are used by Fast R-CNN for detection. They further merge RPN and Fast R-CNN into a single network by sharing their convolutional features. They perform good accuracy on lots of datasets like PASCAL VOC 2007, 2012, and MS COCO datasets with only 300 proposals per image. In ILSVRC and COCO 2015 competition, Faster R-CNN and RAN are the foundations of the 1st-place winning entries in several tracks. Now, the following is the overall architecture of faster R-CNN.





RPN introduces anchor boxes, which are bounding boxes of various sizes and aspect ratios to more precisely extract region proposals. Faster R-CNN can be seen as a combination of RPN and Fast R-CNN. It extracts region proposals from the RPN and forwards them to a fast R-CNN network to predict the object's class and location. This makes it possible to train the network end-to-end without any bottlenecks as the entire process of the model runs on the GPU. The overall operation

sequence is as follows. First, the original image is input to the pre-trained CNN models to obtain the feature map. Next, the feature map is passed to the RPN to generate appropriate region proposals. Then, they perform RoI pooling through region proposals and feature map obtained to obtain fixed size feature map. Finally, classification and bounding box regression are performed by putting a fixed size feature map as input to the model. Mainly, the RPN can be trained end-to-end by back-propagation and stochastic gradient descent (SGD). They also randomly initialize all new layers by drawing weights from zero-mean Gaussian distribution with standard deviation 0.01. Using COCO training dataset to learn, Faster R-CNN has 42.1% mAP and 21.5% mAP on COCO test-dev set. This is 2.8% higher for mAP and 2.2% higher for mAP than fast R-CNN counter part. This indicates that RPN performs excellent for improving the localization accuracy at higher IoU thresholds. Using the COCO train val set to train, faster R-CNN has 42.7% mAP and 21.9% mAP on the COCO test-dev set.

#5. Comparison between four models (OverFeat, R-CNN, Fast R-CNN, Faster R-CNN)

OverFeat is one stage detector and R-CNN series models are two stage detector. There are some differences. One stage detector can do regional proposal and classification at the same time, but two stage detector can do the proposal first and then do the classification later. In R-CNN series, R-CNN and fast R-CNN use regional proposal on CPU by selective search, but faster R-CNN uses regional proposal network on GPU which is much more faster. Fast R-CNN uses CNN once to get feature map but RCNN uses CNN 2000 times once in each proposal. R-CNN and fast R-CNN use selective search, but faster R-CNN uses sliding window method. There are some pros and cons about R-CNN series. R-CNN uses CNN to classify each region, but it cannot be learned in end-to-end method. Therefore, it is hard to find global optimal solution. Fast R-CNN can group feature extraction, RoI pooling, region classification, bounding box regression steps together and use end-to-end method to train; but it is still slow because selective search is done on CPU. Finally, faster R-CNN proposes RPN network and it can use end-to-end method. However, it still consist of many components and in region classification, each feature vector is forwarded to FC layer, respectively. Although object detection research has been actively conducted recently, these 4 methodologies have brought breakthrough results.