Deep Learning

Mask R-CNN

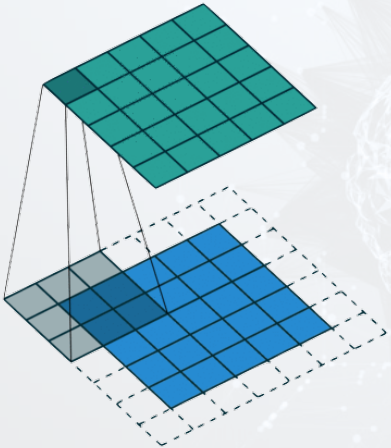
Lecturer: Duc Dung Nguyen, PhD.
Contact: nddung@hcmut.edu.vn

Faculty of Computer Science and Engineering
Hochiminh city University of Technology

Contents

Types of Convolution

Convolution

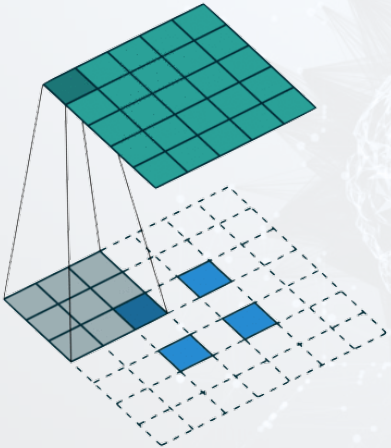


- **Kernel size** defines the field of view of the convolution.
- **Stride** defines the step size of the kernel when traversing the image.
- **Padding** defines how the border of a sample is handled.
- **Input & Output channels**: A convolutional layer takes a certain number of input channels (I) and calculates a specific number of output channels (O). The needed parameters for such a layer can be calculated by I*O*K.

Dilated Convolution

- **Dilation rate** defines a spacing between the values in a kernel
- Deliver a wider field of view at the same computational cost.
- It is popular in the field of real-time segmentation.

Transposed Convolution



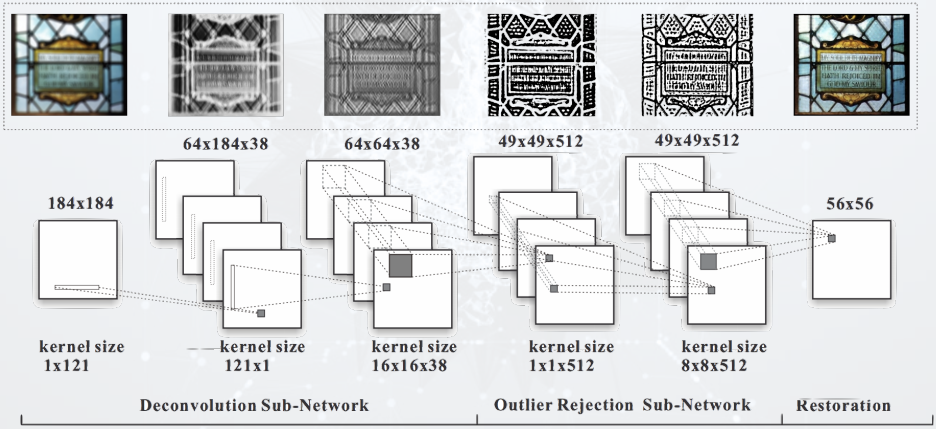
- **Not deconvolution**!
- A transposed convolutional layer carries out a regular convolution but reverts its spatial transformation.
- Combine the upscaling of an image with a convolution, instead of doing two separate processes.

Deconvolution

What is deconvolution?

Deconvolution

What is deconvolution?



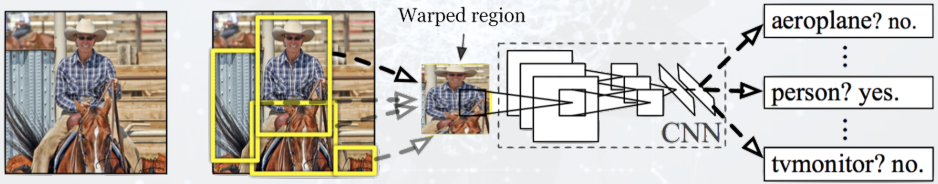
Deep Convolutional Neural Network for Image Deconvolution, NIPS2014

R-CNN

R-CNN

- R-CNN (Girshick et al., 2014): "Region-based Convolutional Neural Networks"
- Idea:
  - Using Selective search to identify bounding-box object region candidates
  - Extract CNN features from each region independently for classification.

- R-CNN (Girshick et al., 2014): "Region-based Convolutional Neural Networks"
- Idea:
  - Using Selective search to identify bounding-box object region candidates
  - Extract CNN features from each region independently for classification.



R-CNN

- Pre-train a CNN network on image classification tasks

R-CNN

- Pre-train a CNN network on image classification tasks
- Propose category-independent regions of interest by selective search (2k candidates per image). Those regions may contain target objects and they are of different sizes.

R-CNN

- Pre-train a CNN network on image classification tasks
- Propose category-independent regions of interest by selective search (2k candidates per image). Those regions may contain target objects and they are of different sizes.
- Region candidates are warped to have a fixed size.

R-CNN

- Pre-train a CNN network on image classification tasks
- Propose category-independent regions of interest by selective search (2k candidates per image). Those regions may contain target objects and they are of different sizes.
- Region candidates are warped to have a fixed size.
- Continue fine-tuning the CNN on warped proposal regions for $K + 1$ classes; The additional one class refers to the background (no object of interest).

R-CNN

- Pre-train a CNN network on image classification tasks
- Propose category-independent regions of interest by selective search (2k candidates per image). Those regions may contain target objects and they are of different sizes.
- Region candidates are warped to have a fixed size.
- Continue fine-tuning the CNN on warped proposal regions for K + 1 classes; The additional one class refers to the background (no object of interest).
- Given every image region, one forward propagation through the CNN generates a feature vector. This feature vector is then consumed by a binary SVM trained for each class independently.
  The positive samples are proposed regions with IoU (intersection over union) overlap threshold >= 0.3, and negative samples are irrelevant others.

R-CNN

- Pre-train a CNN network on image classification tasks
- Propose category-independent regions of interest by selective search (2k candidates per image). Those regions may contain target objects and they are of different sizes.
- Region candidates are warped to have a fixed size.
- Continue fine-tuning the CNN on warped proposal regions for K + 1 classes; The additional one class refers to the background (no object of interest).
- Given every image region, one forward propagation through the CNN generates a feature vector. This feature vector is then consumed by a binary SVM trained for each class independently.
  The positive samples are proposed regions with IoU (intersection over union) overlap threshold >= 0.3, and negative samples are irrelevant others.
- To reduce the localization errors, a regression model is trained to correct the predicted detection window on bounding box correction offset using CNN features.

Selective search:

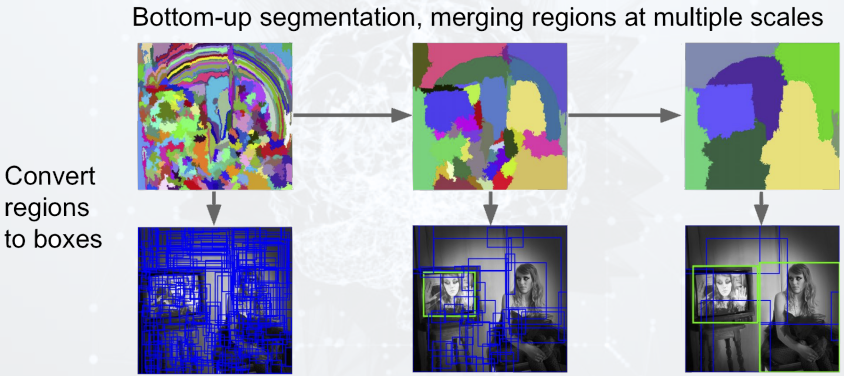
Algorithm 1: Hierarchical Grouping Algorithm

DontPrintSemicolon **Input**: (colour) image
Output: Set of object location hypotheses L

Obtain initial regions $R = \{r_1, \cdots, r_n\}$ using Felzenszwalb and Huttenlocher (2004) Initialise similarity set $S = \emptyset$;
foreach *Neighbouring region pair* (r_i, r_j) **do**
  Calculate similarity $s(r_i, r_j)$;
  $S = S \cup s(r_i, r_j)$;

while $S \neq \emptyset$ **do**
  Get highest similarity $s(r_i, r_j) = \max(S)$;
  Merge corresponding regions $r_t = r_i \cup r_j$;
  Remove similarities regarding $r_i : S = S \setminus s(r_i, r_*)$;
  Remove similarities regarding $r_j : S = S \setminus s(r_*, r_j)$;
  Calculate similarity set S_t between r_t and its neighbours;
  $S = S \cup S_t$;
  $R = R \cup r_t$;
Extract object location boxes L from all regions in R;

Selective search:



Uijlings et al, "Selective Search for Object Recognition", IJCV 2013

Better one? Check out EdgeBox.

Bottleneck

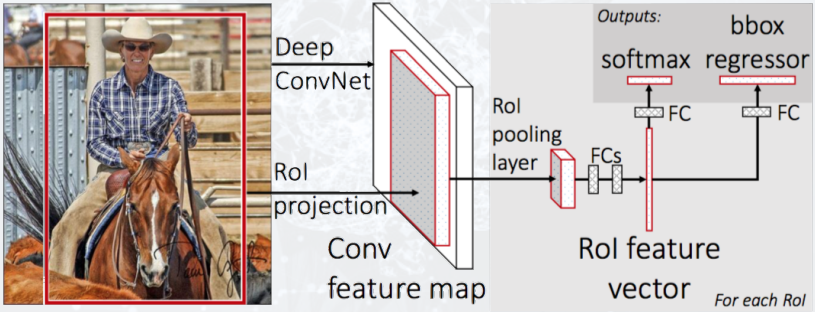
- Running selective search to propose 2000 region candidates
- Generating the CNN feature vector for every image region
- Involve three models separately without much shared computation

Fast R-CNN

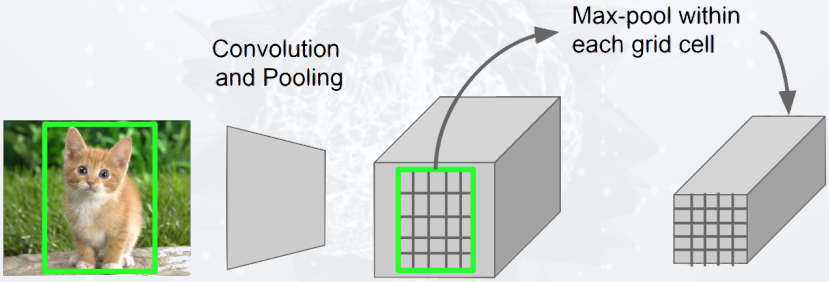
Fast R-CNN

- Proposed by Girshick in 2015
- Idea: speeding up by unifying 3 steps of R-CNN
- Aggregates features into one CNN forward pass over the entire image and the region proposals share this feature matrix.
- The same feature matrix is branched out to be used for learning the object classifier and the bounding-box regressor

Fast R-CNN


Deep
ConvNet
RoI
projection
Conv
feature map
RoI
pooling
layer
FCs
Outputs:
softmax
bbox
regressor
FC
FC
RoI feature
vector
For each RoI

RoI (Region of Interest)



Fast R-CNN

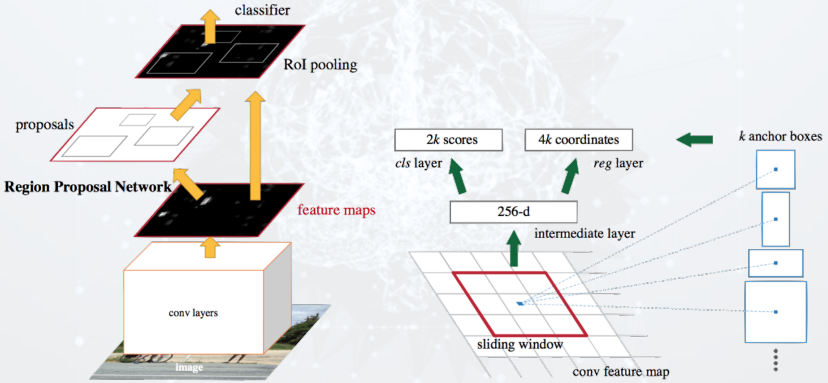
- Pre-train a CNN on image classification tasks.
- Propose regions by selective search (2k candidates per image).
- Replace the last max pooling layer of the pre-trained CNN with a **RoI pooling** layer. The RoI pooling layer outputs fixed-length feature vectors of region proposals.
- Replace the last fully connected layer and the last softmax layer (K classes) with a fully connected layer and softmax over $K + 1$ classes.
- The model branches into two output layers:
  - A softmax estimator of $K + 1$ classes, outputting a discrete probability distribution per RoI.
  - A bounding-box regression model which predicts offsets relative to the original RoI for each of K classes.

Fast R-CNN

Bottleneck

- The region proposals are generated separately by another model.

Idea: integrate the region proposal algorithm into the CNN model



Faster R-CNN model. (Ren et al., 2016)

Faster R-CNN

- Pre-train a CNN network
- Fine-tune the RPN (region proposal network) end-to-end for the region proposal task, which is initialized by the pre-train image classifier. Positive samples have IoU (intersection-over-union) > 0.7, while negative samples have IoU < 0.3.
  - Slide a small n x n spatial window over the conv feature map of the entire image.
  - At the center of each sliding window, we predict multiple regions of various scales and ratios simultaneously. An anchor is a combination of (sliding window center, scale, ratio).
- Train a Fast R-CNN object detection model using the proposals generated by the current RPN
- Use the Fast R-CNN network to initialize RPN training
- Fine-tune the unique layers of Fast R-CNN

Mask R-CNN

Mask R-CNN

- Detect not just object, but the exact pixels of the object
- Image segmentation problem

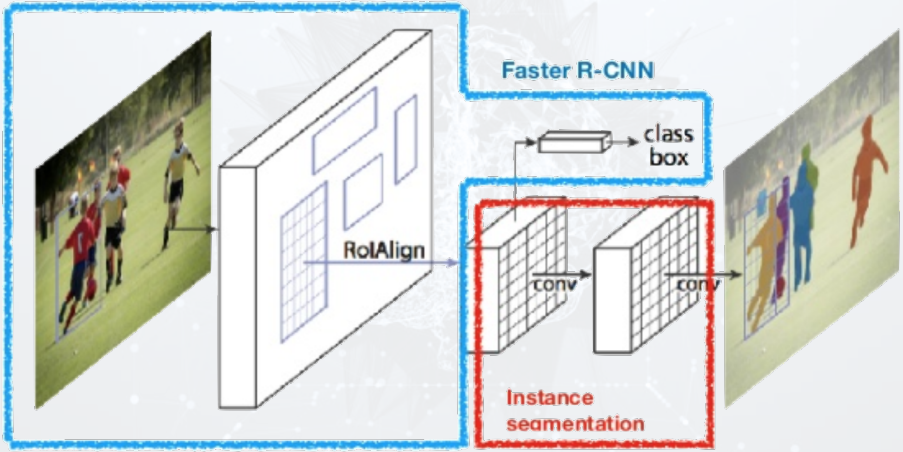
Mask R-CNN

Predictions by Mask R-CNN on COCO test set. (He et al., 2017)

Mask R-CNN

- **Mask R-CNN** (He et al., 2017) extends Faster R-CNN to pixel-level image segmentation.
- Idea: decouple the classification and the pixel-level mask prediction tasks
- Add a third branch for predicting an object mask in parallel with the existing branches for classification and localization.
- The mask branch is a small fully-connected network applied to each RoI.

Mask R-CNN



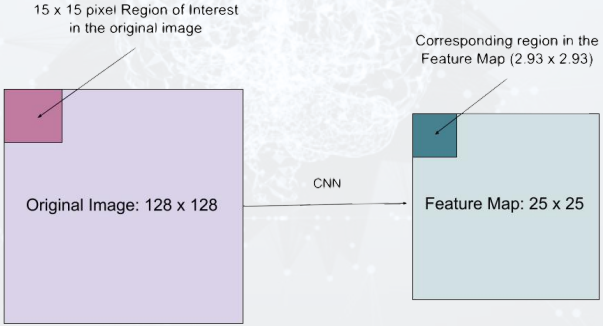
Mask R-CNN is Faster R-CNN model with image segmentation. (He et al., 2017)

Mask R-CNN

- Mask R-CNN improves the RoI pooling layer (**RoIAlign layer**) so that RoI can be better and more precisely mapped to the regions of the original image.

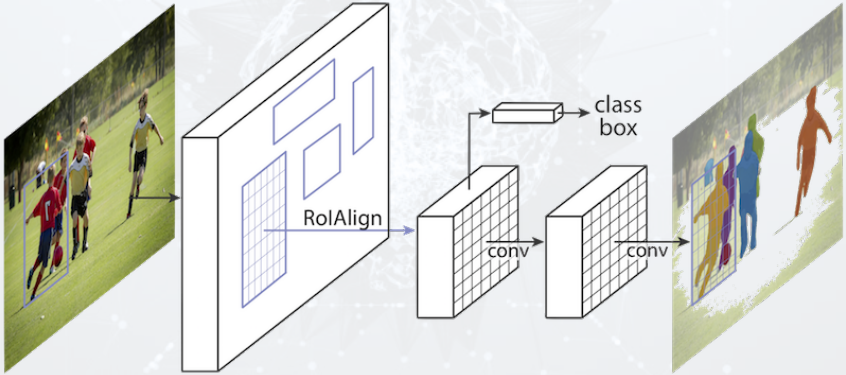
RoIAlign layer:

- Is designed to fix the location misalignment caused by quantization in the RoI pooling.
- RoIAlign removes the hash quantization
- Bilinear interpolation is used for computing the floating-point location values in the input.



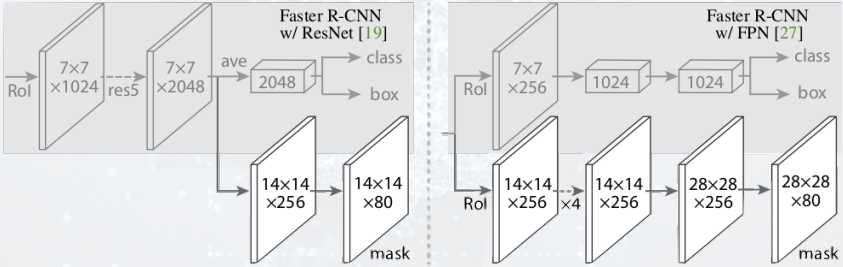
Mask R-CNN

- Model



Mask R-CNN

- Model



- Results on COCO test images