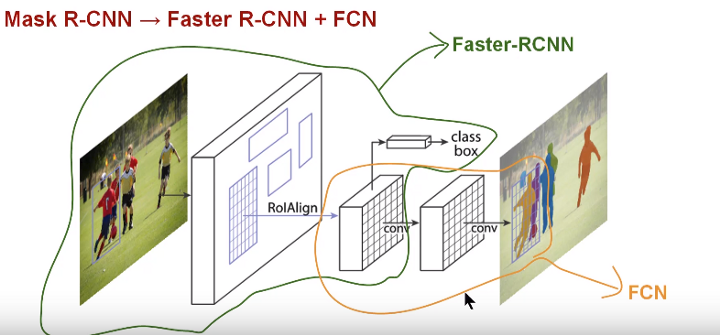
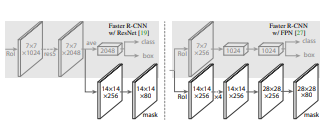
**MASK RCNN OBJECT DETECTION MODEL**



Model Introduction:

* Mask R-CNN, extends Faster R-CNN by adding a branch for predicting segmentation masks on each Region of Interest (RoI), in parallel with the existing branch for classification and bounding box regression .
* The mask branch is a small FCN applied to each RoI, predicting a segmentation mask in a pixel-to pixel manner. Mask R-CNN is simple to implement and train given the Faster R-CNN framework, which facilitates a wide range of flexible architecture designs.
* Additionally, the mask branch only adds a small computational overhead, enabling a fast system and rapid experimentation.
* Image segmentation technique called Mask R-CNN is used to solve an instance segmentation problem.

Model Architecture:



* Head Architecture: We extend two existing Faster RCNN heads . Left/Right panels show the heads for the ResNet C4 and FPN backbones respectively, to which a mask branch is added.
* Numbers denote spatial resolution and channels. Arrows denote either conv, deconv, or fc layers as can be inferred from context (conv preserves spatial dimension while deconv increases it).
* All convs are 3×3, except the output conv which is 1×1, deconvs are 2×2 with stride 2, and we use ReLU in hidden layers.
* Left: ‘res5’ denotes ResNet’s fifth stage, which for simplicity we altered so that the first conv operates on a 7×7 RoI with stride 1 (instead of 14×14 / stride 2 ). Right: ‘×4’ denotes a stack of four consecutive convs.

Mask RCNN Working:

* Mask R-CNN is conceptually simple, Faster R-CNN has two outputs for each candidate object, a class label and a bounding-box offset; to this we add a third branch that outputs the object mask.
* Mask R-CNN is thus a natural and intuitive idea but the additional mask output is distinct from the class and box outputs, requiring extraction of much finer spatial layout of an object.
* Next, we introduce the key elements of Mask R-CNN, including pixel-to-pixel alignment, which is the main missing piece of Fast/Faster R-CNN.
* Mask R-CNN adopts the same two-stage procedure, with an identical first stage (which is RPN). In the second stage, in parallel to predicting the class and box offset, Mask R-CNN also outputs a binary mask for each RoI.
* This approach follows the spirit of Fast R-CNN, that applies bounding-box classification and regression in parallel which turned out to largely simplify the multi-stage pipeline of original R-CNN.
* Our definition of( L mask) allows the network to generate masks for every class without competition among classes; we rely on the dedicated classification branch to predict the class label used to select the output mask.
* This decouples mask and class prediction. This is different from common practice when applying FCNs to semantic segmentation, which typically uses a per-pixel softmax and a multinomial cross-entropy loss. In that case, masks across classes compete; in our case, with a per-pixel sigmoid and a binary loss, they do not. This formulation is key for good instance segmentation results.

Mask Representation:

**Mask Representation**: A mask encodes an input object’s spatial layout. Thus, unlike class labels or box offsets that are inevitably collapsed into short output vectors by fully-connected (fc) layers, extracting the spatial structure of masks can be addressed naturally by the pixel-to-pixel correspondence provided by convolutions.

Specifically, we predict an m × m mask from each RoI using an FCN. This allows each layer in the mask branch to maintain the explicit m × m object spatial layout without collapsing it into a vector representation that lacks spatial dimensions. Unlike previous methods that resort to fc layers for mask prediction , our fully convolutional representation requires fewer parameters, and is more accurate.

**RoIAlign:** RoIPool is a standard operation for extracting a small feature map (e.g., 7×7) from each RoI. RoIPool first quantizes a floating-number RoI to the discrete granularity of the feature map, this quantized RoI is then subdivided into spatial bins which are themselves quantized, and finally feature values covered by each bin are aggregated (usually by max pooling).

Quantization is performed. These quantization introduce misalignments between the RoI and the extracted features. While this may not impact classification, which is robust to small translations, it has a large negative effect on predicting pixel-accurate masks.

To address this, we propose an **RoIAlign** layer that removes the harsh quantization of RoI Pool, properly aligning the extracted features with the input. Our proposed change is simple: we avoid any quantization of the RoI boundaries 3 or bins.