# How detectron works?

## Architecture:

1. Detectron2’s main working mechanism is through a CNN like ResNet, ResNeXt and FPN (Depending upon the situation).

-> ResNet 50 and 101 are most commonly used while FPN is added to implement multi-scaled features. ResNet or residual network is used to solve vanishing gradient problems in a deep learning environment. ResNet allows deep layer trainings (upwards of 50 layers) because of its ability to have gradient flow in all layers and it skips if the gradient is not changing coming from one layer to the other. This way it sometimes skips the FPN.

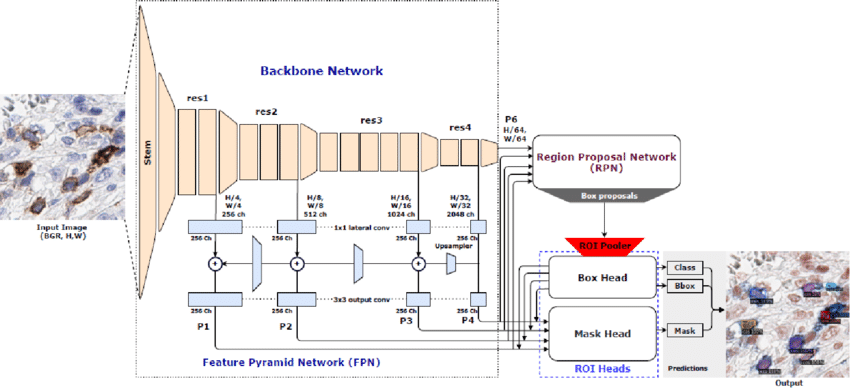
1. Detectron uses Region Proposal Network (RPN) to narrow down the areas of interest for better analysis.
2. The output from the RPN is provided to the Region of Interest (ROI) for further classification and refinement. ROI has mask for segmentation as well.
3. It also has task specific heads to do different tasks such as box regression for object localization, mask prediction and keypoint prediction to estimate the pose of the object (specially humans).

## Training

1. Detectron prepares data in a COCO format (Images, Annotations and Metadata) for easy integration.
2. Data augmentation is done through random flipping, scaling and cropping.
3. Loss function and cross-entropy loss are calculated during the training process.

## Prediction

1. The images pass through the architecture that generates feature maps.
2. The RPN passes the images to ROI which selects and narrows down the bounding boxes and predicts the output.
3. The main CNN architecture is selected according to the required tasks.



# References

<https://www.analyticsvidhya.com/blog/2021/08/your-guide-to-object-detection-with-detectron2-in-pytorch/>

<https://viso.ai/deep-learning/detectron2/>

https://datascientistassoc.org/detectron2/

Deep Residual Learning for Image Recognition: https://arxiv.org/pdf/1512.03385