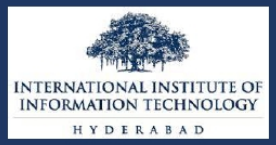


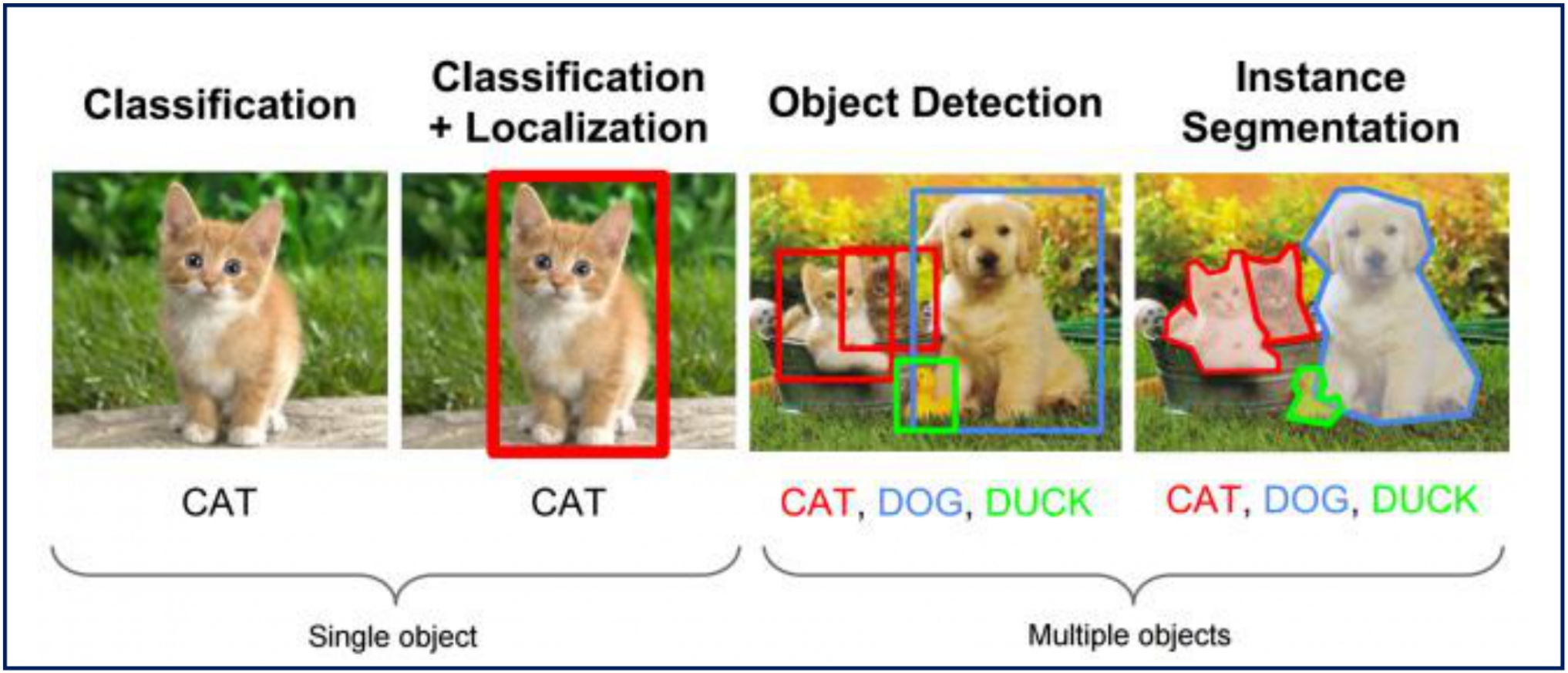
CVIT

Object Detection Concepts and Evolution in 20 Years

RRC Summer School – 2023
Date : 27th May, 2023
Time : 10:30 am – 12:00 pm

Seshadri Mazumder
PhD Research Scholar
IIIT Hyderabad
Telangana, India
(seshadri.mazumder@research.iiit.ac.in)

INTRODUCTION



Challenges in Object Detection

- Challenges in other computer vision tasks, such as
- Objects under different viewpoints,
- Illuminations,
- Intraclass variations
- Object rotation
- Scale changes
- Accurate object localization
- Dense object detection
- Occluded object detection
- Speedup of detection

Plan to proceed

INTERNATIONAL INSTITUTE OF INFORMATION TECHNOLOGY
HYDERABAD

- **A : Roadmap of Object Detection**

Plan to proceed

INTERNATIONAL INSTITUTE OF INFORMATION TECHNOLOGY
HYDERABAD

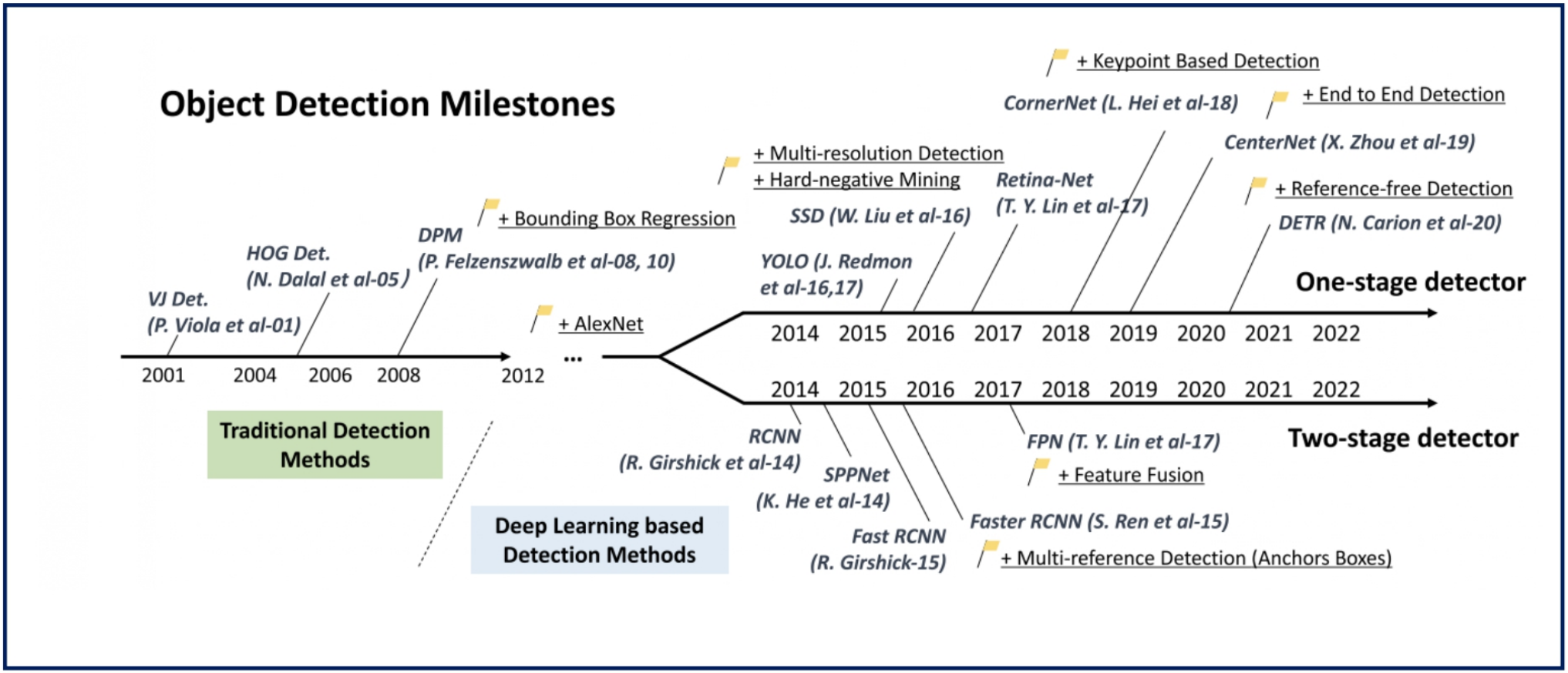
- **A : Roadmap of Object Detection**
- **B : Object Detection : Datasets & Metrics**

Plan to proceed

INTERNATIONAL INSTITUTE OF INFORMATION TECHNOLOGY
HYDERABAD

- A : Roadmap of Object Detection
- B : Object Detection : Datasets & Metrics
- C : Technical Evolution in Object Detection

Object Detection Milestones



Object Detection types

INTERNATIONAL INSTITUTE OF INFORMATION TECHNOLOGY
HYDERABAD

- **Milestone 1 : Traditional Detector**
  - Viola Jones Detector
  - Hogg Detector
  - DPMM

Object Detection types

- **Milestone 1 : Traditional Detector**
  - Viola Jones Detector
  - Hogg Detector
  - DPMM
- **Milestone 2 : CNN-Based Two-Stage Detectors**
  - RCNN
  - Fast RCNN
  - Faster RCNN
  - Feature Pyramid Network

Object Detection types

- **Milestone 1 : Traditional Detector**
  - Viola Jones Detector
  - Hogg Detector
  - DPMM
- **Milestone 2 : CNN-Based Two-Stage Detectors**
  - RCNN
  - Fast RCNN
  - Faster RCNN
  - Feature Pyramid Network
- **Milestone 3 : CNN-Based One-Stage Detectors**
  - YOLO
  - Single Shot Multi-box Detector
  - DETR

Object Detection types

- Milestone 1 : Traditional Detector
  - Viola Jones Detector
  - Hogg Detector
  - DPMM
- Milestone 2 : CNN-Based Two-Stage Detectors
  - RCNN
  - Fast RCNN
  - Faster RCNN
  - Feature Pyramid Network
- Milestone 3 : CNN-Based One-Stage Detectors
  - YOLO
  - Single Shot Multi-box Detector
  - DETR

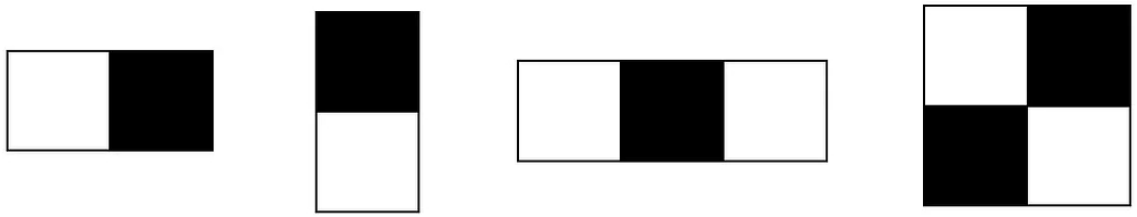
Object Detection types

- **Milestone 1 : Traditional Detector**
  - **Viola Jones Detector**
  - **Hogg Detector**
  - **DPMM**

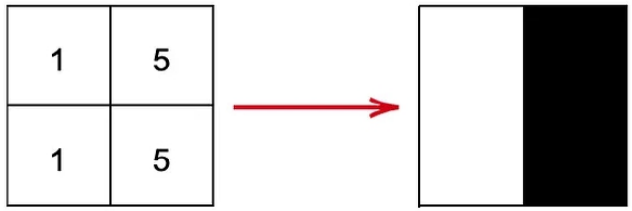
- **Milestone 2 : CNN-Based Two-Stage Detectors**
  - **RCNN**
  - **Fast RCNN**
  - **Faster RCNN**
  - **Feature Pyramid Network**

- **Milestone 3 : CNN-Based One-Stage Detectors**
  - **YOLO**
  - **Single Shot Multi-box Detector**
  - **DETR**

Haar Features



(A) (B) (C) (D)



$= 2 - 10 = -8$

Integral Features

	x_0	x_1	x_2	x_3
y_0	1	12	45	10
y_1	6	5	11	4
y_2	3	7	10	8
y_3	5	9	4	7

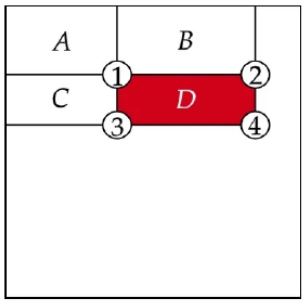
Original Image
(Grayscale)

$$ii(x, y) = \sum_{x' \leqslant x,\, y' \leqslant y} i(x', y')$$

i – original image
ii – integral image

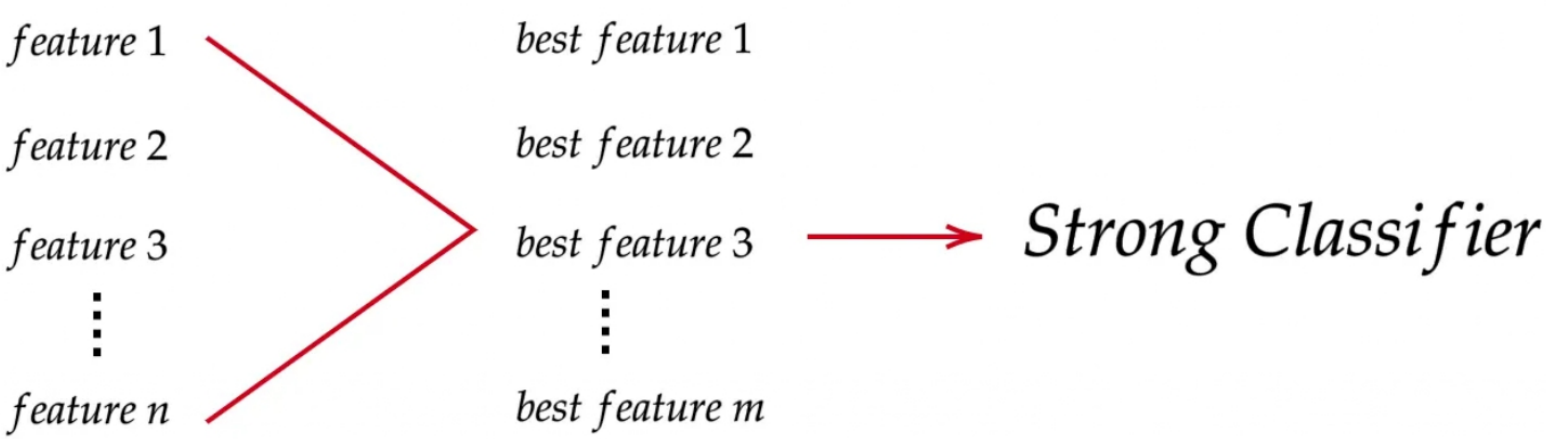
	x_0	x_1	x_2	x_3
y_0	1	13	58	68
y_1	7	24	80	94
y_2	10	34	100	122
y_3	15	48	118	147

Integral Image



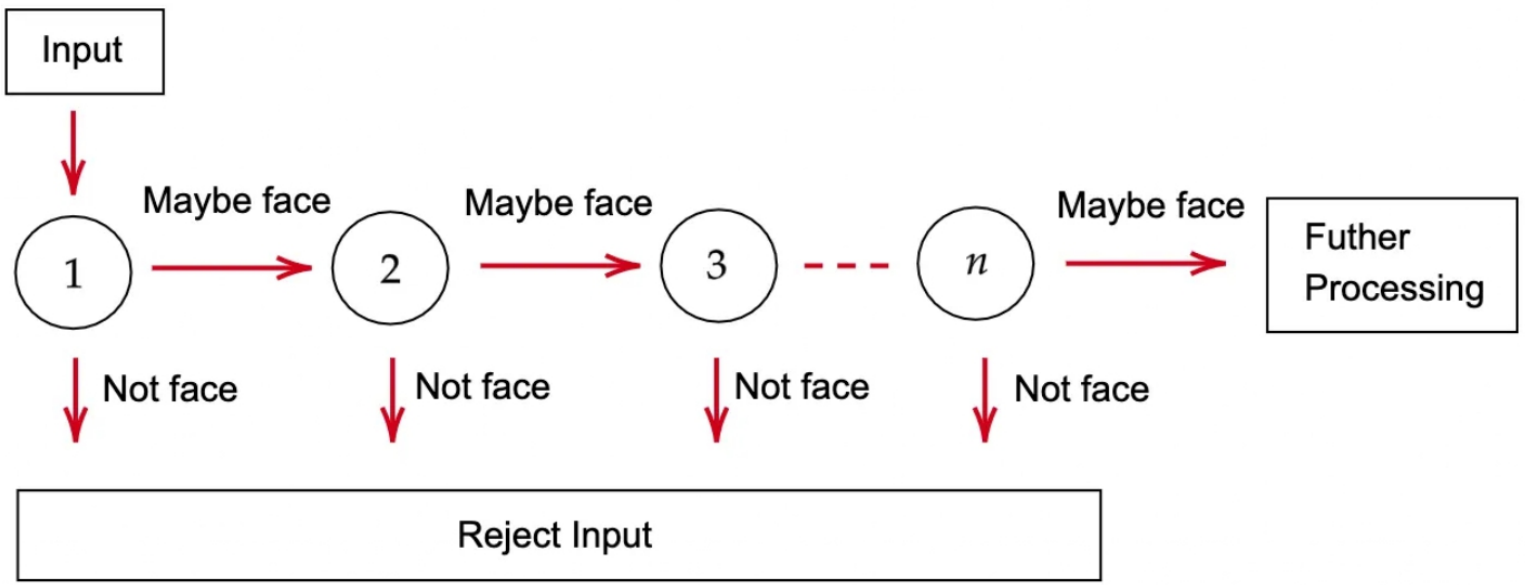
$$D = ④ - ② - ③ + ①$$
$$= ④ + ① - (② + ③)$$

Ada Boost Algorithm

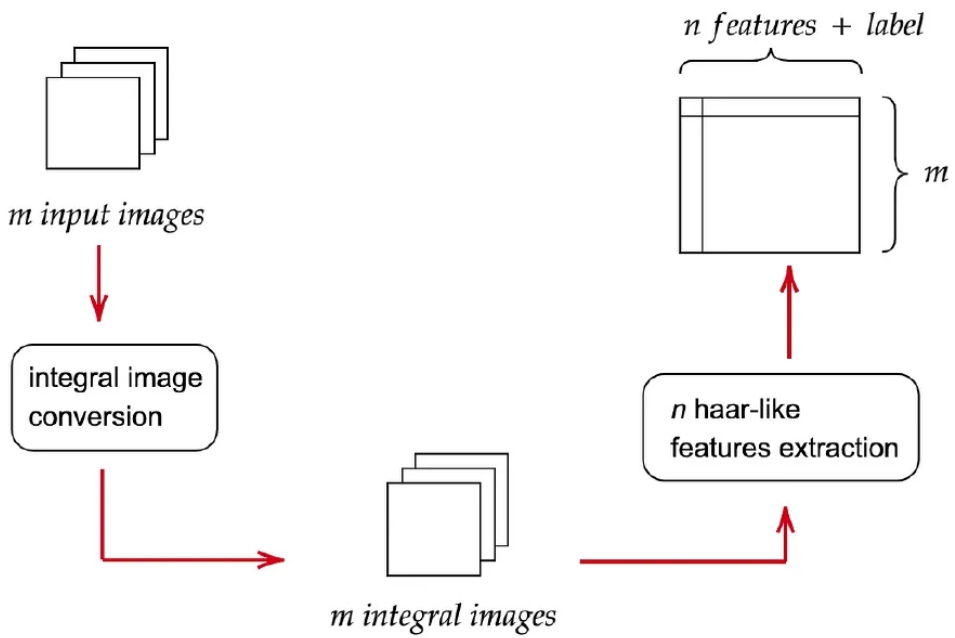


where $n > m$

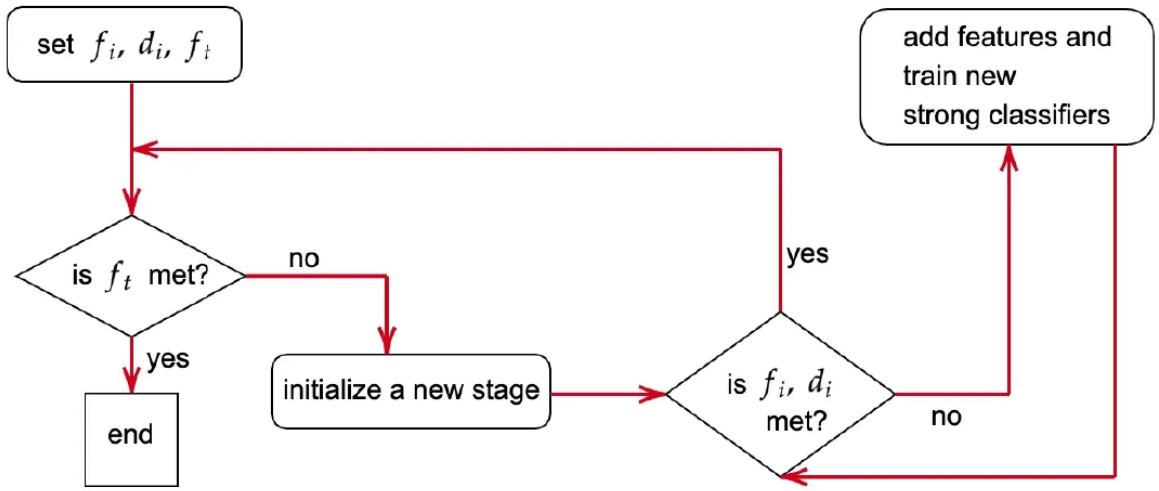
Cascade Classifier



Data Preparation for Viola Jones



Ada Boost + Cascade Classifier



f_i = maximum acceptable false positive rate per stage

d_i = minimum acceptable true positive rate per stage

f_t = target overall false positive rate

Object Detection types

- **Milestone 1 : Traditional Detector**
  - **Viola Jones Detector**
  - **Hogg Detector**
  - **DPMM**

- **Milestone 2 : CNN-Based Two-Stage Detectors**
  - **RCNN**
  - **Fast RCNN**
  - **Faster RCNN**
  - **Feature Pyramid Network**

- **Milestone 3 : CNN-Based One-Stage Detectors**
  - **YOLO**
  - **Single Shot Multi-box Detector**
  - **DETR**

Object Detection types

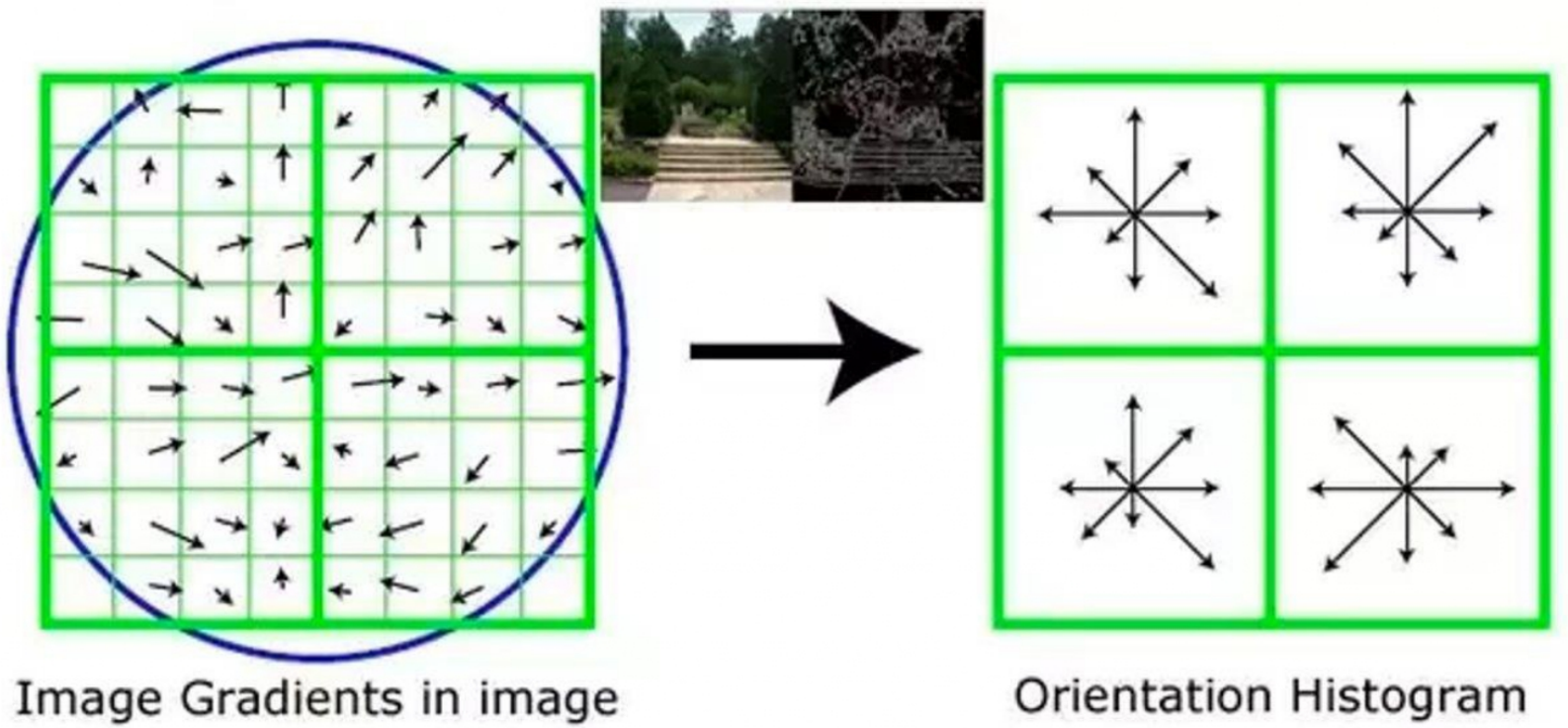
INTERNATIONAL INSTITUTE OF INFORMATION TECHNOLOGY
HYDERABAD

- **Milestone 1 : Traditional Detector**
  - **Viola Jones Detector**
  - **Hogg Detector**
  - **DPMM**

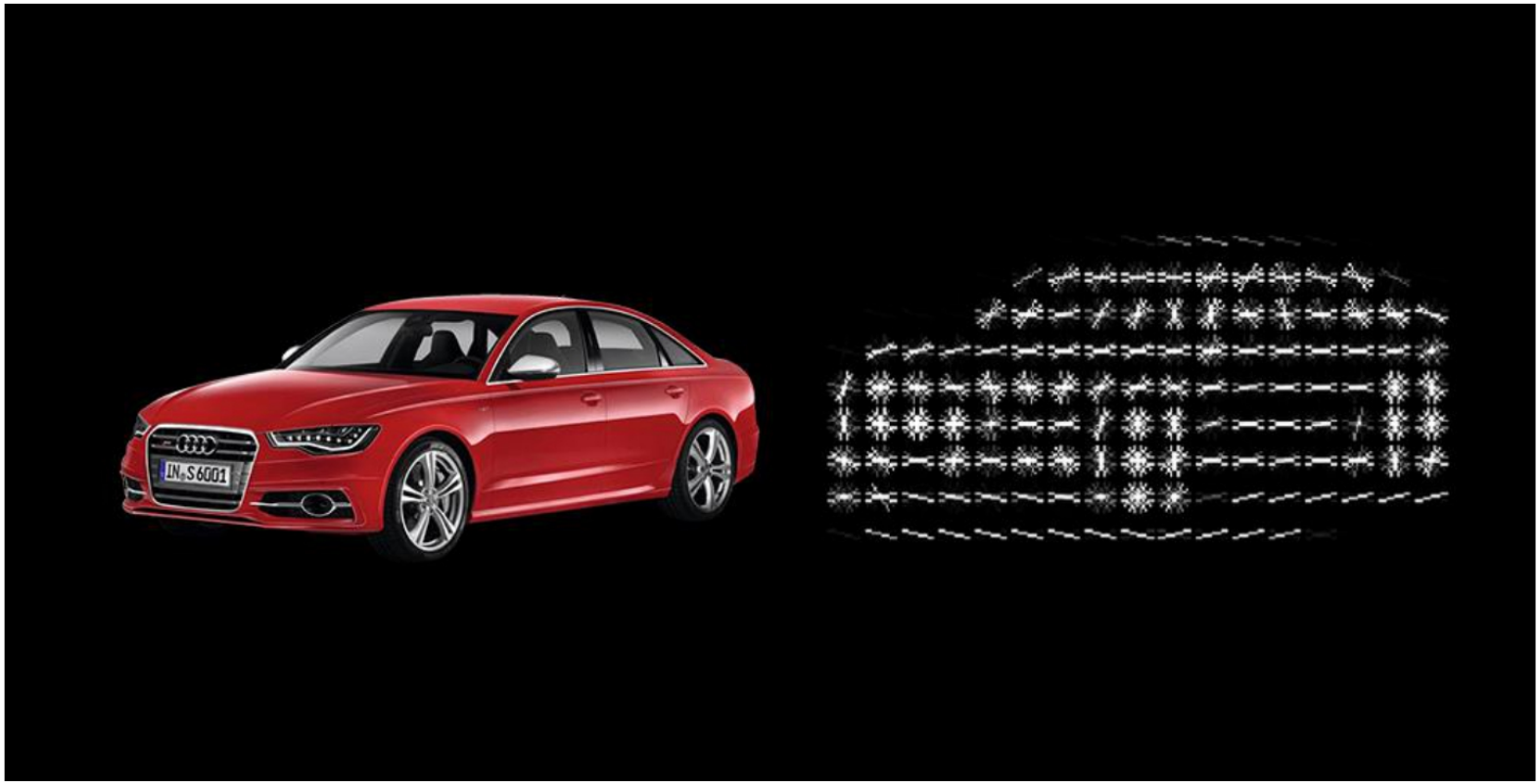
- **Milestone 2 : CNN-Based Two-Stage Detectors**
  - **RCNN**
  - **Fast RCNN**
  - **Faster RCNN**
  - **Feature Pyramid Network**

- **Milestone 3 : CNN-Based One-Stage Detectors**
  - **YOLO**
  - **Single Shot Multi-box Detector**
  - **DETR**

Hogg Detector

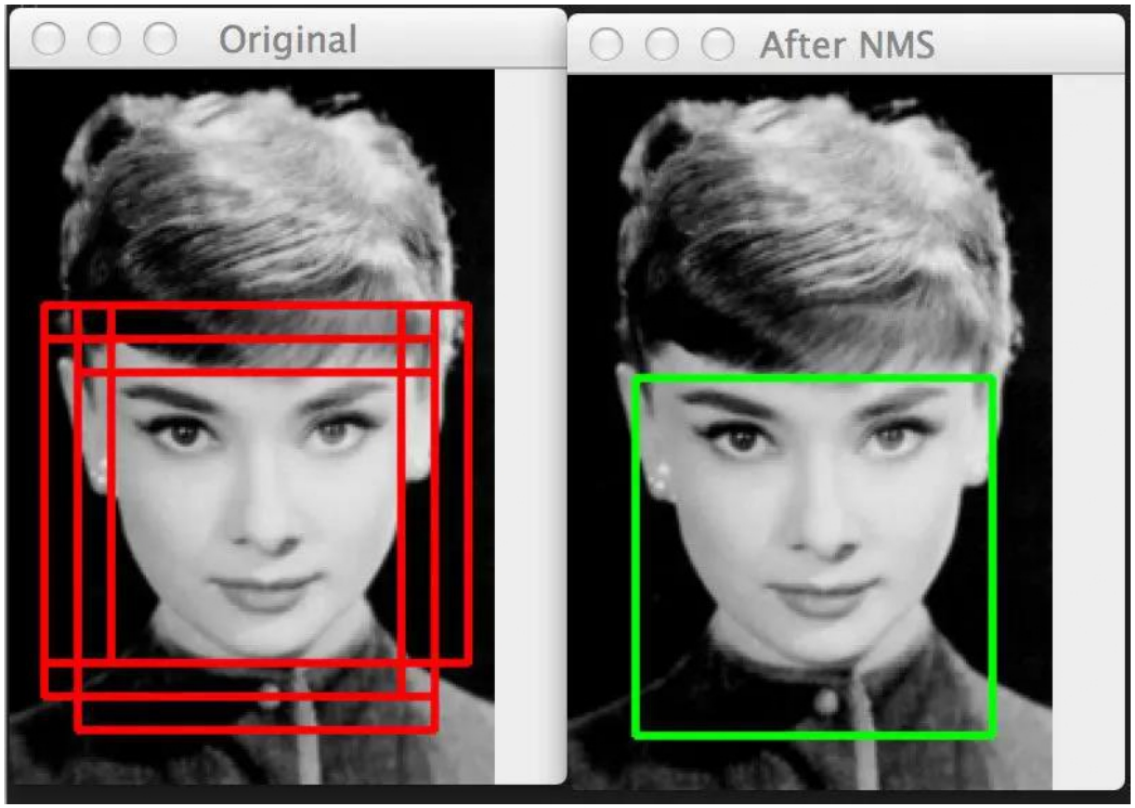


Hogg Detector



Hogg Detector

Non Maximum Suppression



Object Detection types

INTERNATIONAL INSTITUTE OF INFORMATION TECHNOLOGY
HYDERABAD

- **Milestone 1 : Traditional Detector**
  - **Viola Jones Detector**
  - **Hogg Detector**
  - **DPMM**

- **Milestone 2 : CNN-Based Two-Stage Detectors**
  - **RCNN**
  - **Fast RCNN**
  - **Faster RCNN**
  - **Feature Pyramid Network**

- **Milestone 3 : CNN-Based One-Stage Detectors**
  - **YOLO**
  - **Single Shot Multi-box Detector**
  - **DETR**

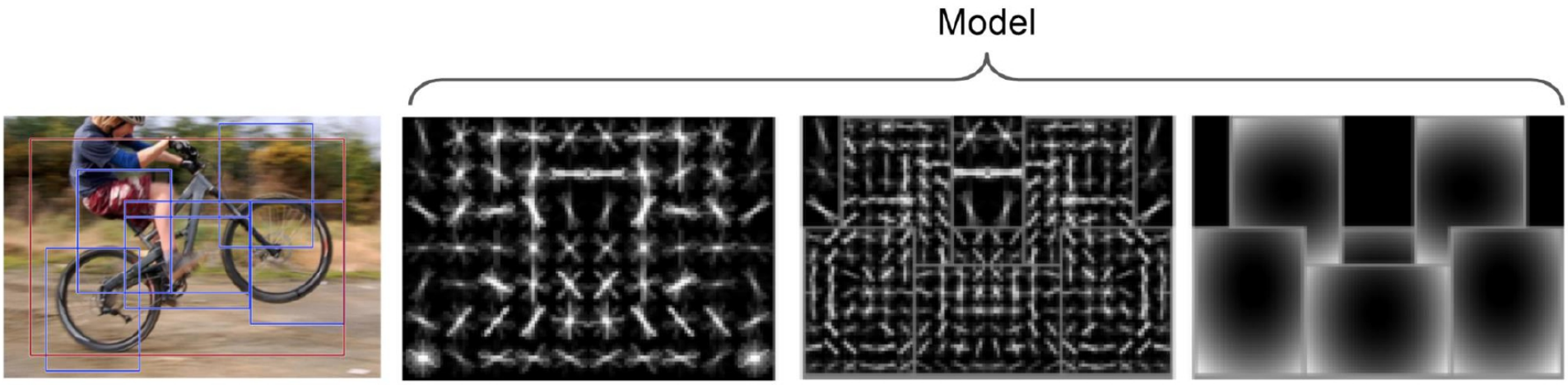
Object Detection types

- **Milestone 1 : Traditional Detector**
  - **Viola Jones Detector**
  - **Hogg Detector**
  - **DPMM**

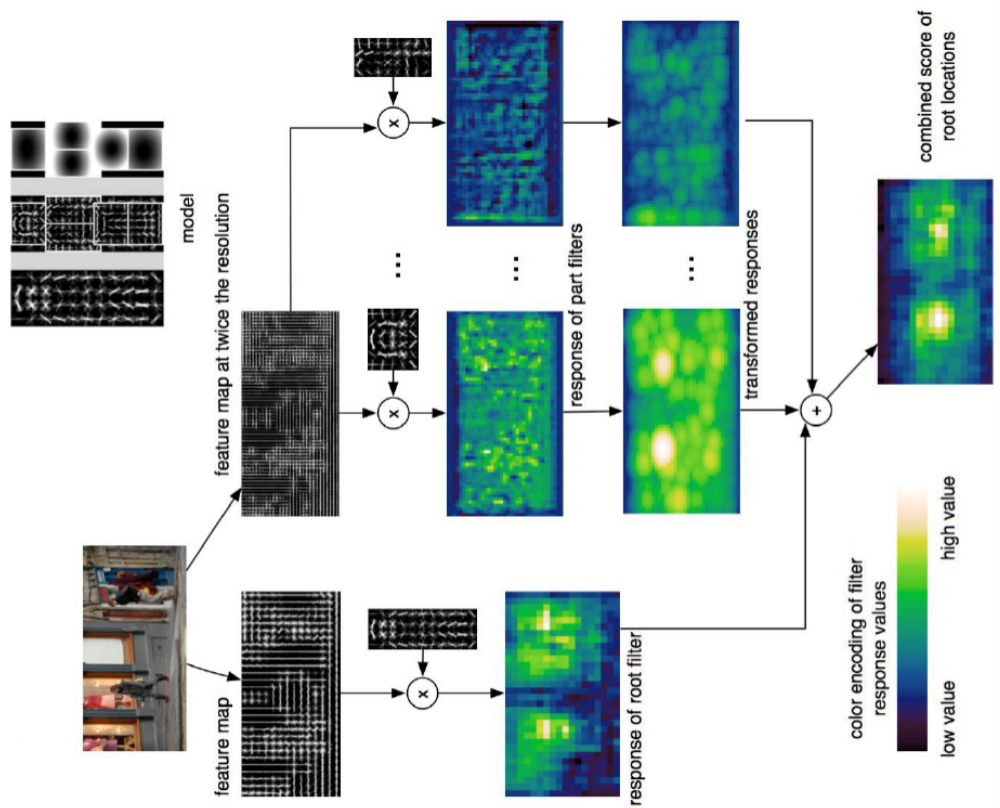
- **Milestone 2 : CNN-Based Two-Stage Detectors**
  - **RCNN**
  - **Fast RCNN**
  - **Faster RCNN**
  - **Feature Pyramid Network**

- **Milestone 3 : CNN-Based One-Stage Detectors**
  - **YOLO**
  - **Single Shot Multi-box Detector**
  - **DETR**

DPMM


Model
(a) Root filter
(b) Part filters in higher resolution
(c) A spatial model for part locations

DPMM


model
feature map at twice the resolution
feature map
response of part filters
transformed responses
response of root filter
combined score of root locations
color encoding of filter response values
high value
low value

Object Detection types

- **Milestone 1 : Traditional Detector**
  - **Viola Jones Detector**
  - **Hogg Detector**
  - **DPMM**

- **Milestone 2 : CNN-Based Two-Stage Detectors**
  - **RCNN**
  - **Fast RCNN**
  - **Faster RCNN**
  - **Feature Pyramid Network**

- **Milestone 3 : CNN-Based One-Stage Detectors**
  - **YOLO**
  - **Single Shot Multi-box Detector**
  - **DETR**

Object Detection types

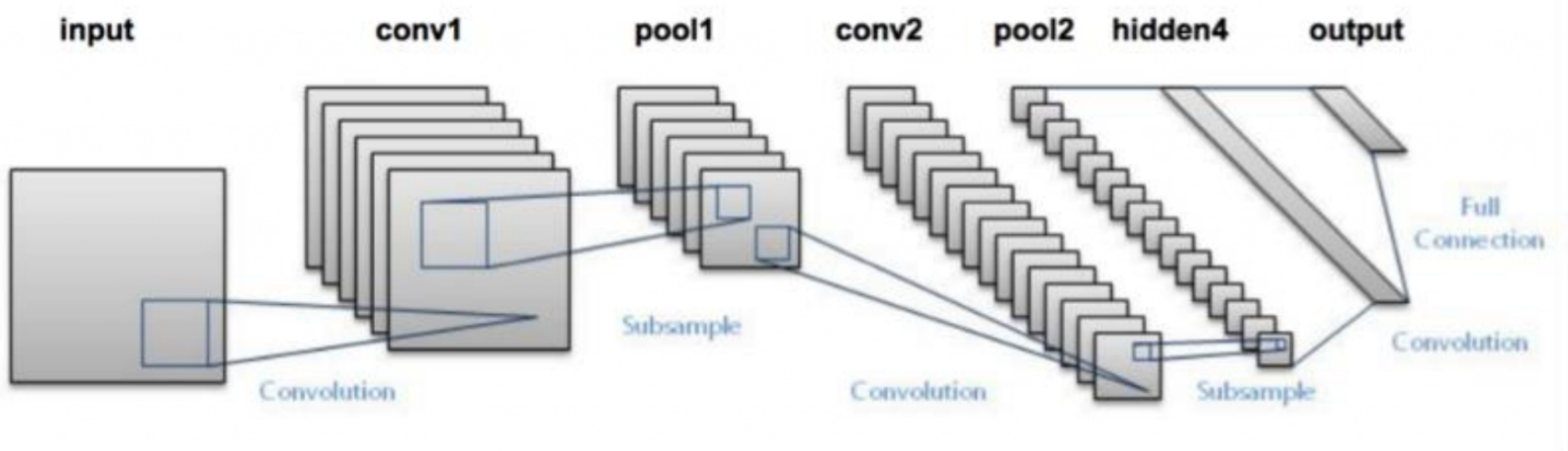
INTERNATIONAL INSTITUTE OF INFORMATION TECHNOLOGY
HYDERABAD

- **Milestone 1 : Traditional Detector**
  - **Viola Jones Detector**
  - **Hogg Detector**
  - **DPMM**

- **Milestone 2 : CNN-Based Two-Stage Detectors**
  - **RCNN**
  - **Fast RCNN**
  - **Faster RCNN**
  - **Feature Pyramid Network**

- **Milestone 3 : CNN-Based One-Stage Detectors**
  - **YOLO**
  - **Single Shot Multi-box Detector**
  - **DETR**

Basic CNN Architecture



Let's look at how we can solve a general object detection problem using a CNN...!!

CNN for Object Detection

INTERNATIONAL INSTITUTE OF INFORMATION TECHNOLOGY
HYDERABAD

Step 1 : Taking the Image as an input

CNN for Object Detection

INTERNATIONAL INSTITUTE OF INFORMATION TECHNOLOGY
HYDERABAD

Step 2 : Divide the image into various regions

CNN for Object Detection

Step 3 : Consider each region as a separate image

CNN for Object Detection

Step 3 : Consider each region as a separate image

Step 4 : Pass all these regions(images) into CNN and classify them into various classes.

CNN for Object Detection

INTERNATIONAL INSTITUTE OF INFORMATION TECHNOLOGY
HYDERABAD

Step 3 : Consider each region as a separate image
Step 4 : Pass all these regions(images) into CNN and classify them into various classes.
Step 5 : Once we have divided each regions into multiple classes, we can aggregate the result, to get the detected image with objects.

CNN for Object Detection

CNN for Object Detection

Problems?

Identifying Problems

- **Object in the image can have different Aspect Ratio.**
- **Object in the image can have different Spatial Locations.**
- **In some cases :**
  - **the object might be covering most of the image**
  - **the object might only be covering a small percentage of the image.**
- The shapes of the objects might also be different (happens a lot in real-life use cases).

Identifying Problems

- **Object in the image can have different Aspect Ratio.**
- **Object in the image can have different Spatial Locations.**
- **In some cases :**
  - **the object might be covering most of the image**
  - **the object might only be covering a small percentage of the image.**
- The shapes of the objects might also be different (happens a lot in real-life use cases).

- **Solution?**

Identifying Problems

- **Object in the image can have different Aspect Ratio.**
- **Object in the image can have different Spatial Locations.**
- **In some cases :**
  - **the object might be covering most of the image**
  - **the object might only be covering a small percentage of the image.**
- The shapes of the objects might also be different (happens a lot in real-life use cases).

- **Solution?**
  - A very large number of regions

Identifying Problems

- **Object in the image can have different Aspect Ratio.**
- **Object in the image can have different Spatial Locations.**
- **In some cases :**
  - **the object might be covering most of the image**
  - **the object might only be covering a small percentage of the image.**
- The shapes of the objects might also be different (happens a lot in real-life use cases).

- **Solution?**
  - A very large number of regions

- **Problem?**
  - Computational Time

Identifying Problems

INTERNATIONAL INSTITUTE OF INFORMATION TECHNOLOGY
HYDERABAD

- **Object in the image can have different Aspect Ratio.**
- **Object in the image can have different Spatial Locations.**
- **In some cases :**
  - **the object might be covering most of the image**
  - **the object might only be covering a small percentage of the image.**
- The shapes of the objects might also be different (happens a lot in real-life use cases).

- **Solution?**
  - A very large number of regions

- **Problem?**
  - Computational Time

- **Solution?**
  - Region-based CNN

Intuition of RCNN

INTERNATIONAL INSTITUTE OF INFORMATION TECHNOLOGY
HYDERABAD

- Instead of working on a massive number of regions, the RCNN algorithm proposes a bunch of boxes in the image and checks if any of these boxes contain any object.
- RCNN **uses selective search to extract these boxes from an image (these boxes are called regions).**
- Understand what selective search is and how it identifies the different regions.

Understanding Selective Search

- There are basically four regions that form an object:
  - **varying scales,**
  - **colors,**
  - **textures, and**
  - **enclosure**

- Selective search identifies these patterns in the image and based on that, proposes various regions.

Brief Overview of Selective Search

INTERNATIONAL INSTITUTE OF INFORMATION TECHNOLOGY
HYDERABAD

- **Step 1 :** Take Image as an Input

Brief Overview of Selective Search

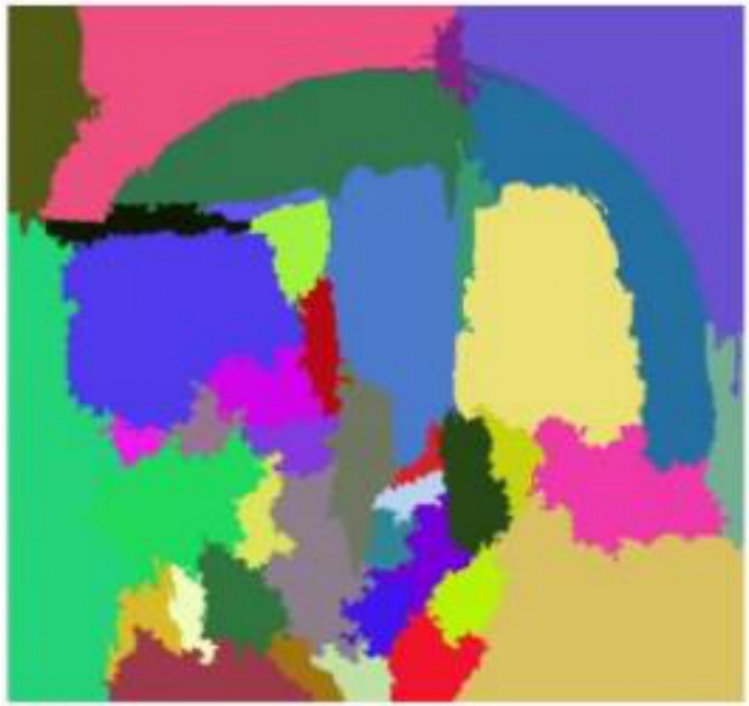
INTERNATIONAL INSTITUTE OF INFORMATION TECHNOLOGY
HYDERABAD

- **Step 2 :** Generates initial sub-segmentations so that we have multiple regions from this image.

Brief Overview of Selective Search

INTERNATIONAL INSTITUTE OF INFORMATION TECHNOLOGY
HYDERABAD

- **Step 3 :** The technique then combines the similar regions to form a larger region (based on color similarity, texture similarity, size similarity, and shape compatibility)



Summary of the steps for RCNN

- We first take a pre-trained convolutional neural network.
- Then, this model is retrained.
- We train the last layer of the network based on the number of classes that need to be detected.
- The third step is to get the Region of Interest for each image.
- We then reshape all these regions so that they can match the CNN input size.
- After getting the regions, we train SVM to classify objects and background.
- For each class, we train one binary SVM.
- Finally, we train a linear regression model to generate tighter bounding boxes for each identified object in the image.

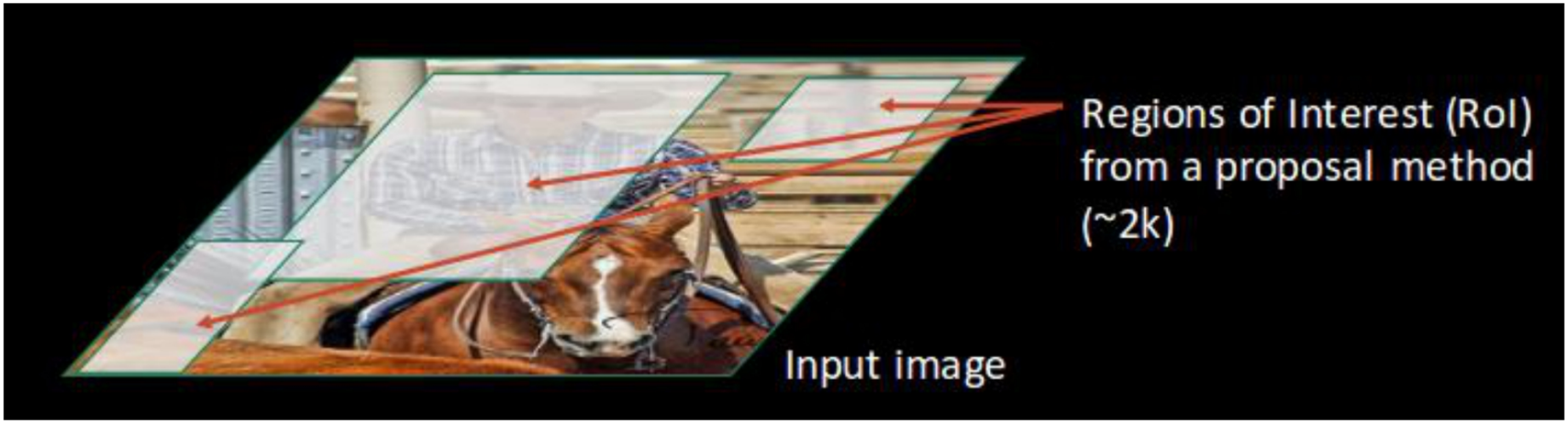
Simple Step by Step Visualization

INTERNATIONAL INSTITUTE OF INFORMATION TECHNOLOGY
HYDERABAD

- **Step 1 : Input Image**

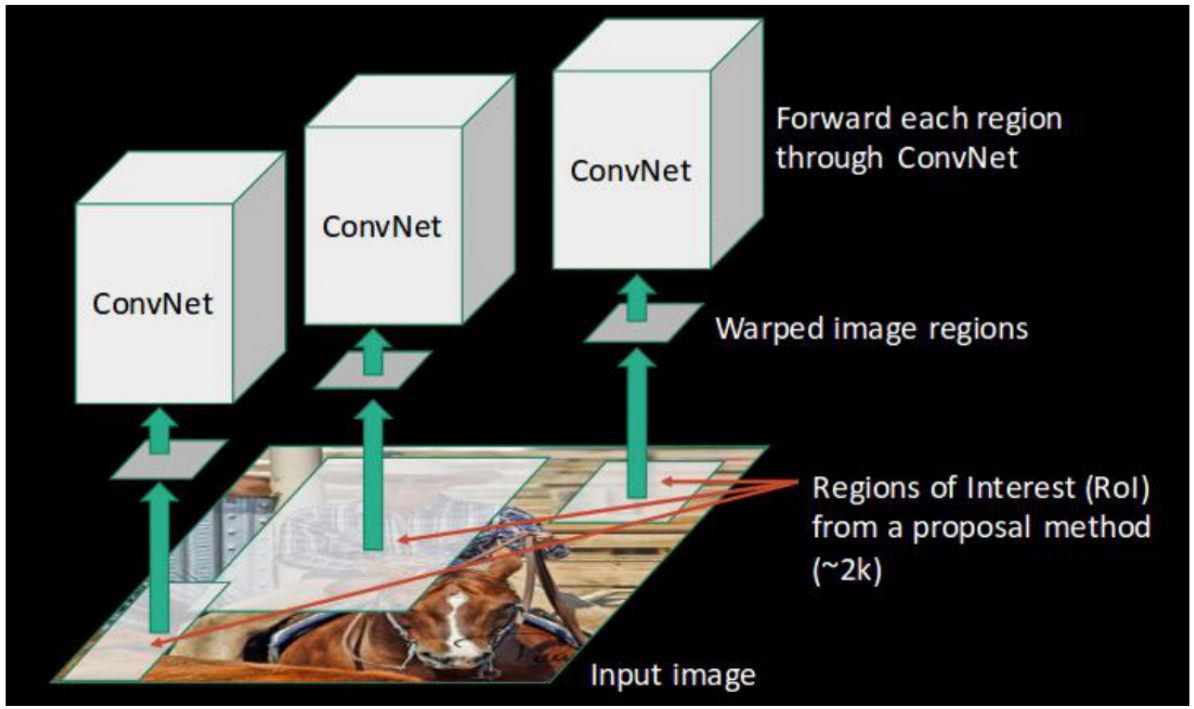
Simple Step by Step Visualization

- **Step 2 : Then, we get the Regions of Interest (ROI) using some proposal method.**



Simple Step by Step Visualization

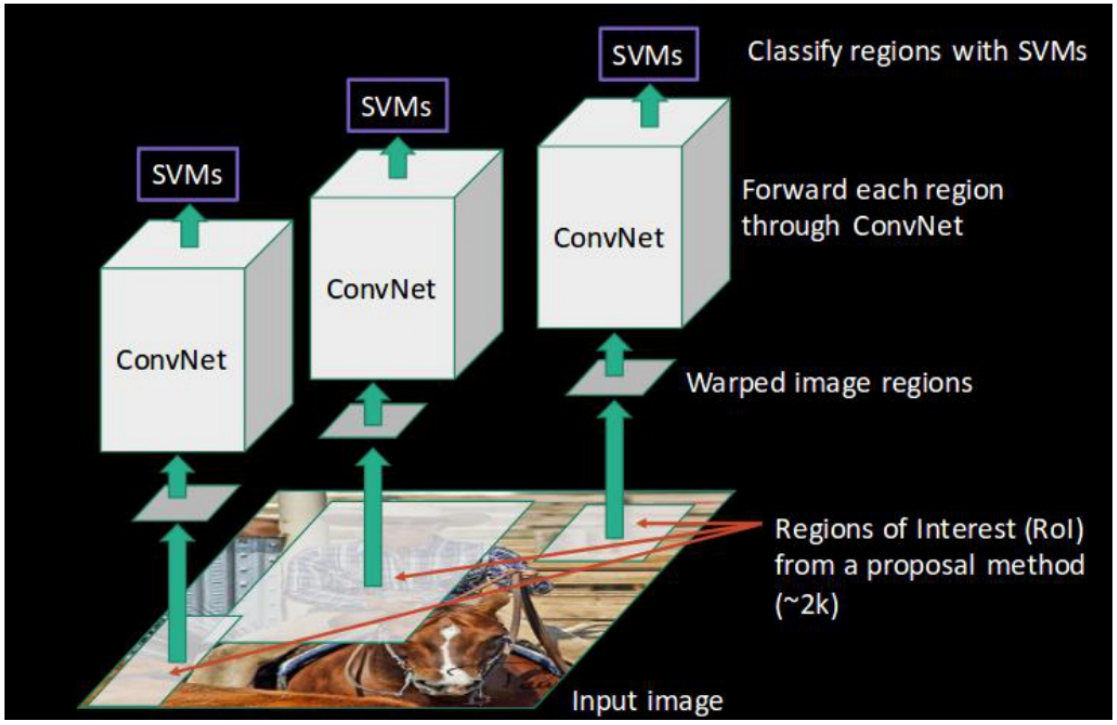
- **Step 3 : All these regions are then reshaped as per the input of the CNN, and each region is passed to the ConvNet.**



Simple Step by Step Visualization

INTERNATIONAL INSTITUTE OF INFORMATION TECHNOLOGY
HYDERABAD

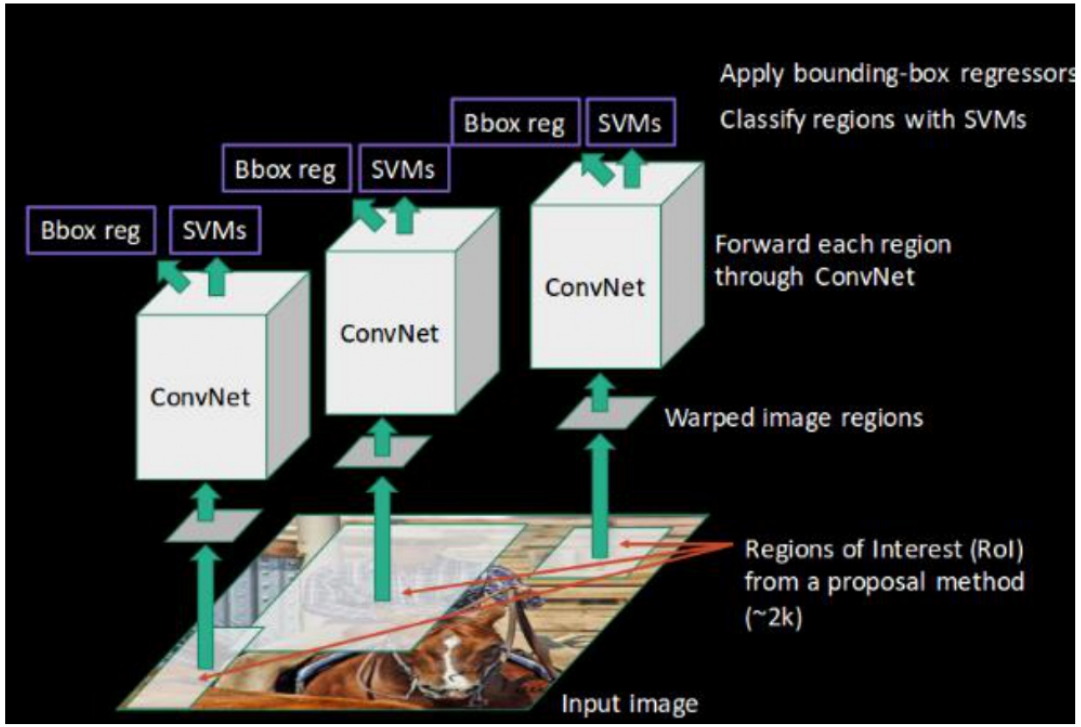
- **Step 4 : CNN then extracts features for each region and SVMs are used to divide these regions into different classes:**



Simple Step by Step Visualization

INTERNATIONAL INSTITUTE OF INFORMATION TECHNOLOGY
HYDERABAD

- **Step 5 : Finally, a bounding box regression (*Bbox reg*) is used to predict the bounding boxes for each identified region**



Problems with RCNN

INTERNATIONAL INSTITUTE OF INFORMATION TECHNOLOGY
HYDERABAD

- Training an RCNN model is expensive and slow thanks to the below steps:
- Extracting 2,000 regions for each image based on selective search
- Extracting features using CNN for every image region. Suppose we have N images, then the number of CNN features will be N*2,000

- The entire process of object detection using RCNN has three models:

    - CNN for feature extraction
    - Linear SVM classifier for identifying objects
    - Regression model for tightening the bounding boxes.
    - All these processes combine to make RCNN very slow. It takes around 40-50 seconds to make predictions for each new image, which essentially makes the model cumbersome and practically impossible to build when faced with a gigantic dataset.

Object Detection types

INTERNATIONAL INSTITUTE OF INFORMATION TECHNOLOGY
HYDERABAD

- **Milestone 1 : Traditional Detector**
  - **Viola Jones Detector**
  - **Hogg Detector**
  - **DPMM**

- **Milestone 2 : CNN-Based Two-Stage Detectors**
  - **RCNN**
  - **Fast RCNN**
  - **Faster RCNN**
  - **Feature Pyramid Network**

- **Milestone 3 : CNN-Based One-Stage Detectors**
  - **YOLO**
  - **Single Shot Multi-box Detector**
  - **DETR**

Object Detection types

INTERNATIONAL INSTITUTE OF INFORMATION TECHNOLOGY
HYDERABAD

- **Milestone 1 : Traditional Detector**
  - **Viola Jones Detector**
  - **Hogg Detector**
  - **DPMM**

- **Milestone 2 : CNN-Based Two-Stage Detectors**
  - **RCNN**
  - **Fast RCNN**
  - **Faster RCNN**
  - **Feature Pyramid Network**

- **Milestone 3 : CNN-Based One-Stage Detectors**
  - **YOLO**
  - **Single Shot Multi-box Detector**
  - **DETR**

Intuition of Fast RCNN

- What else can we do to reduce the computation time a RCNN algorithm typically takes?
- Instead of running a CNN 2,000 times per image, we can run it just once per image and get all the regions of interest (regions containing some object).
- Ross Girshick, the author of RCNN, came up with this idea of running the CNN just once per image and then finding a way to share that computation across the 2,000 regions.
- In Fast RCNN, we feed the input image to the CNN, which in turn generates the convolutional feature maps.
- Using these maps, the regions of proposals are extracted.
- We then use a RoI pooling layer to reshape all the proposed regions into a fixed size, so that it can be fed into a fully connected network.

Summary of steps

- As with the earlier two techniques, we take an image as an input.
- This image is passed to a ConvNet which in turns generates the Regions of Interest.
- A RoI pooling layer is applied on all of these regions to reshape them as per the input of the ConvNet.
- Then, each region is passed on to a fully connected network.
- A softmax layer is used on top of the fully connected network to output classes.
- Along with the softmax layer, a linear regression layer is also used parallely to output bounding box coordinates for predicted classes.
- So, instead of using three different models (like in RCNN), Fast RCNN uses a single model which extracts features from the regions, divides them into different classes, and returns the boundary boxes for the identified classes simultaneously.

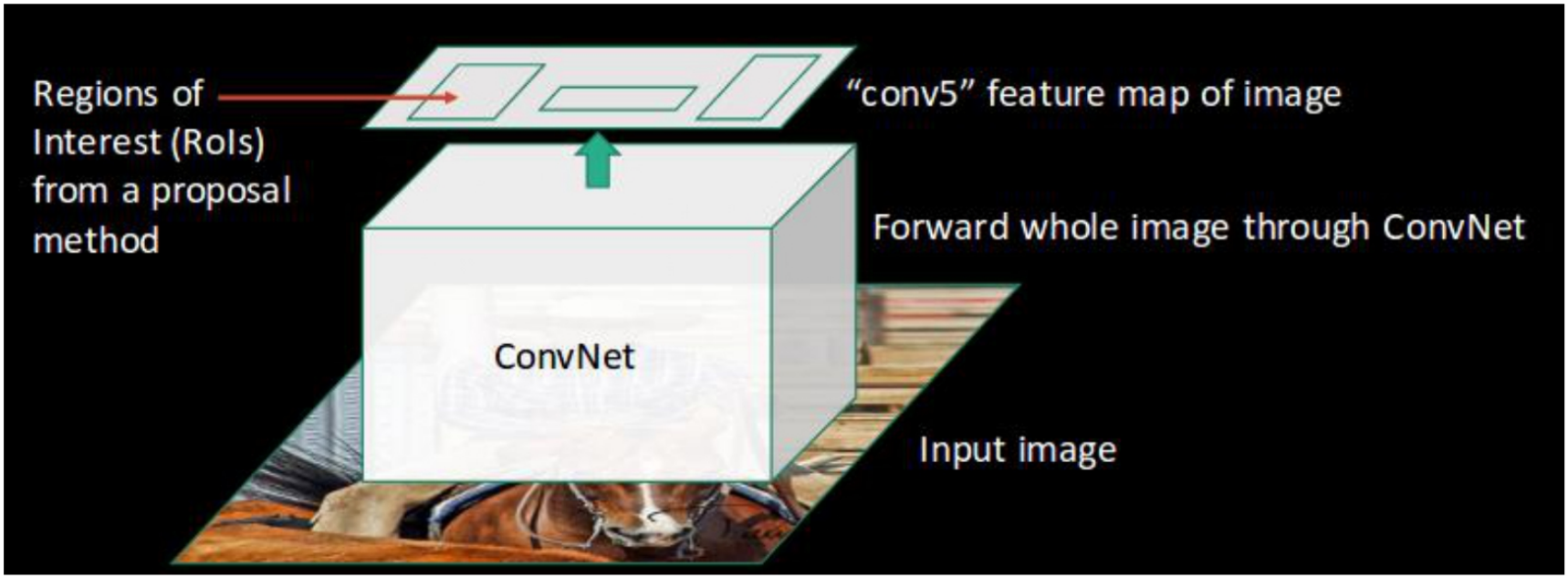
Step by Step Visualization

INTERNATIONAL INSTITUTE OF INFORMATION TECHNOLOGY
HYDERABAD

- **Step 1 :**

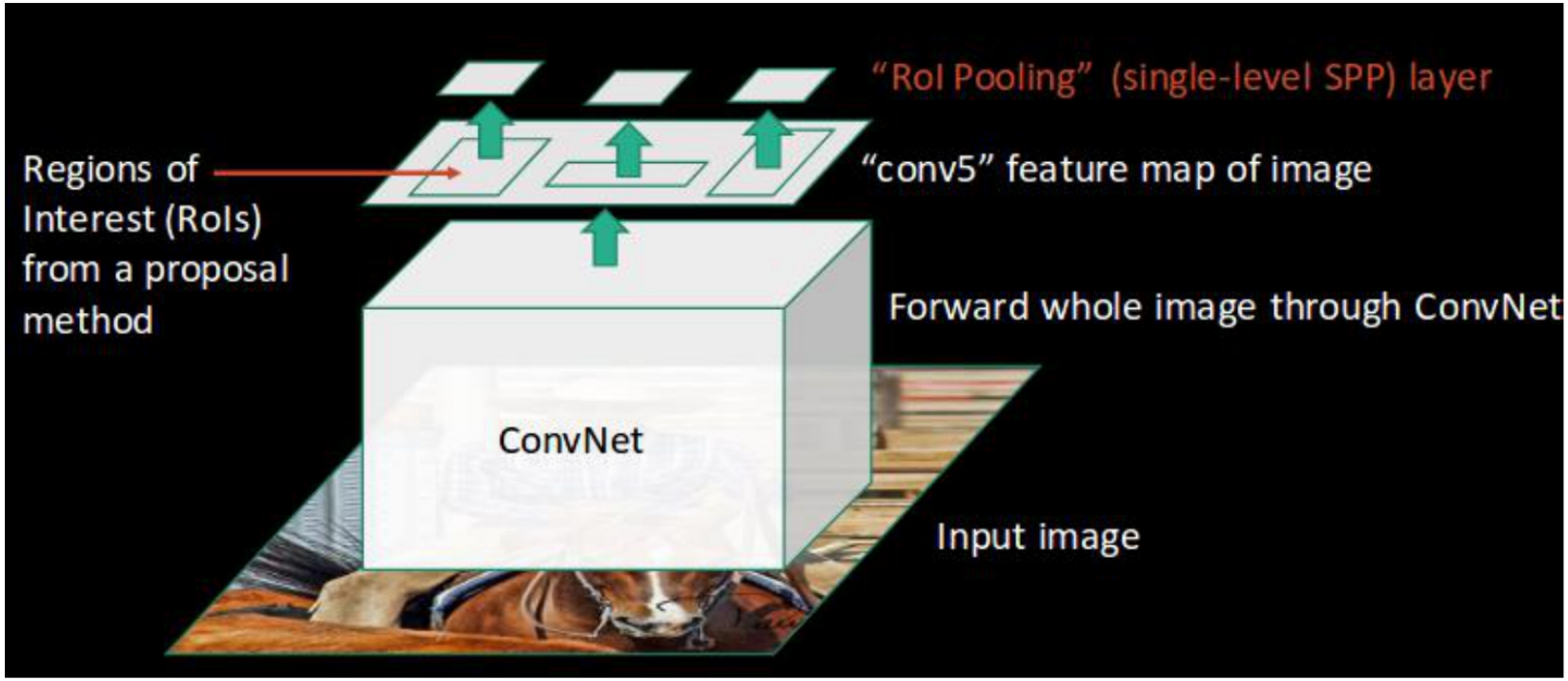
Step by Step Visualization

- **Step 2 :**



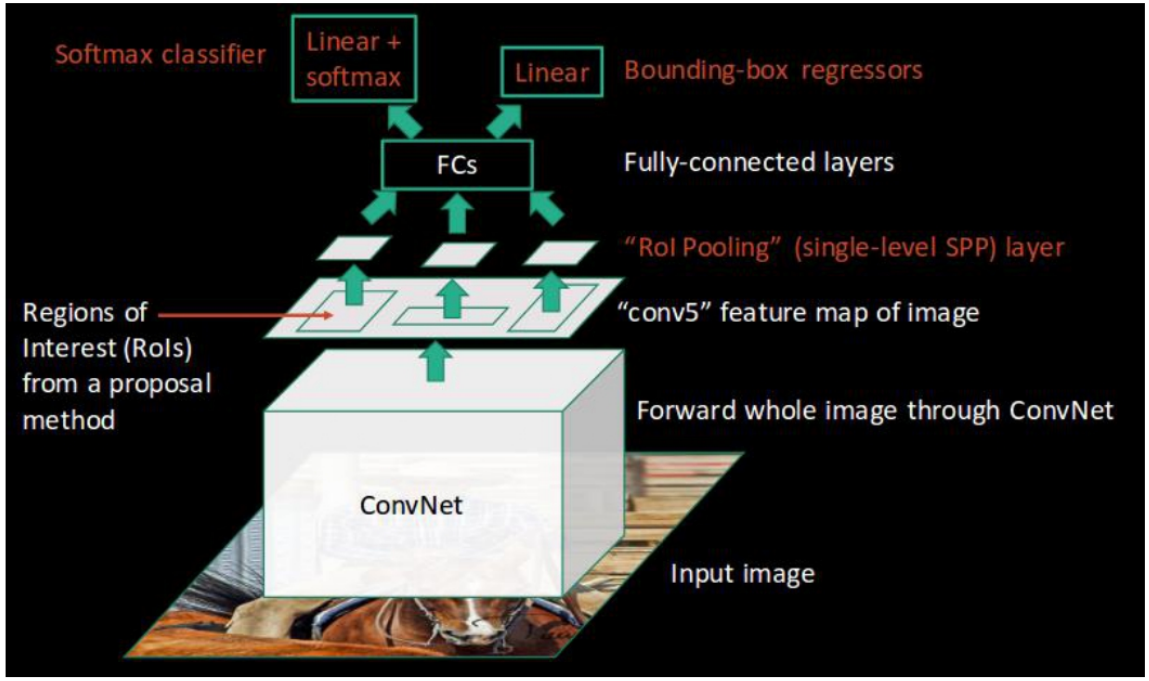
Step by Step Visualization

- **Step 3 :**



Step by Step Visualization

- **Step 4 :**



Summary of steps

- As with the earlier two techniques, we take an image as an input.
- This image is passed to a ConvNet which in turns generates the Regions of Interest.
- A RoI pooling layer is applied on all of these regions to reshape them as per the input of the ConvNet.
- Then, each region is passed on to a fully connected network.
- A softmax layer is used on top of the fully connected network to output classes.
- Along with the softmax layer, a linear regression layer is also used parallely to output bounding box coordinates for predicted classes.
- So, instead of using three different models (like in RCNN), Fast RCNN uses a single model which extracts features from the regions, divides them into different classes, and returns the boundary boxes for the identified classes simultaneously.

Object Detection types

INTERNATIONAL INSTITUTE OF INFORMATION TECHNOLOGY
HYDERABAD

- **Milestone 1 : Traditional Detector**
  - **Viola Jones Detector**
  - **Hogg Detector**
  - **DPMM**

- **Milestone 2 : CNN-Based Two-Stage Detectors**
  - **RCNN**
  - **Fast RCNN**
  - **Faster RCNN**
  - **Feature Pyramid Network**

- **Milestone 3 : CNN-Based One-Stage Detectors**
  - **YOLO**
  - **Single Shot Multi-box Detector**
  - **DETR**

Object Detection types

INTERNATIONAL INSTITUTE OF INFORMATION TECHNOLOGY
HYDERABAD

- **Milestone 1 : Traditional Detector**
  - **Viola Jones Detector**
  - **Hogg Detector**
  - **DPMM**

- **Milestone 2 : CNN-Based Two-Stage Detectors**
  - **RCNN**
  - **Fast RCNN**
  - **Faster RCNN**
  - **Feature Pyramid Network**

- **Milestone 3 : CNN-Based One-Stage Detectors**
  - **YOLO**
  - **Single Shot Multi-box Detector**
  - **DETR**

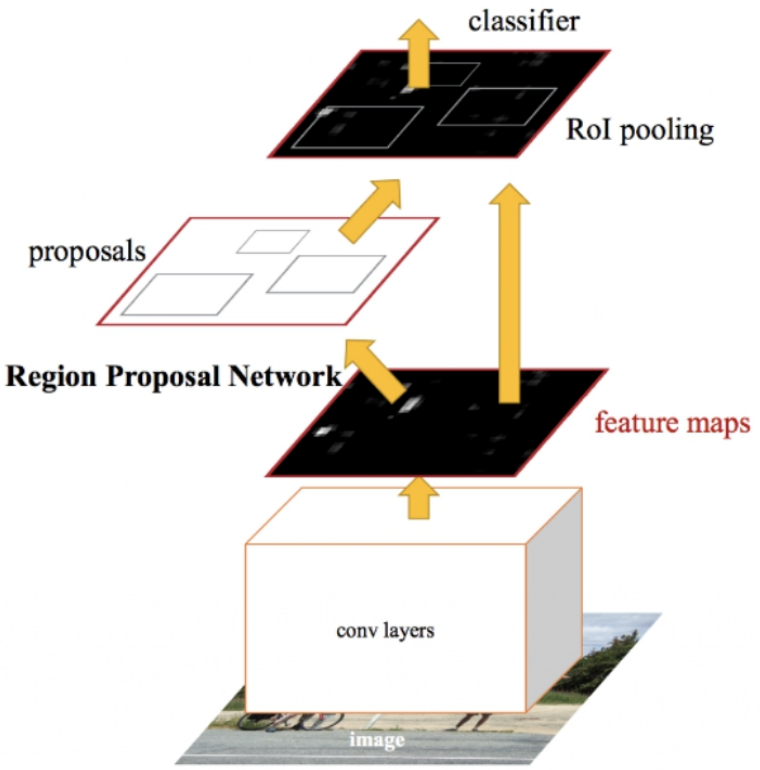
Intuition of faster RCNN

INTERNATIONAL INSTITUTE OF INFORMATION TECHNOLOGY
HYDERABAD

- Faster RCNN is the modified version of Fast RCNN.
- The major difference between them is that Fast RCNN uses selective search for generating Regions of Interest, while Faster RCNN uses “Region Proposal Network”, aka RPN.
- RPN takes image feature maps as an input and generates a set of object proposals, each with an objectness score as output.

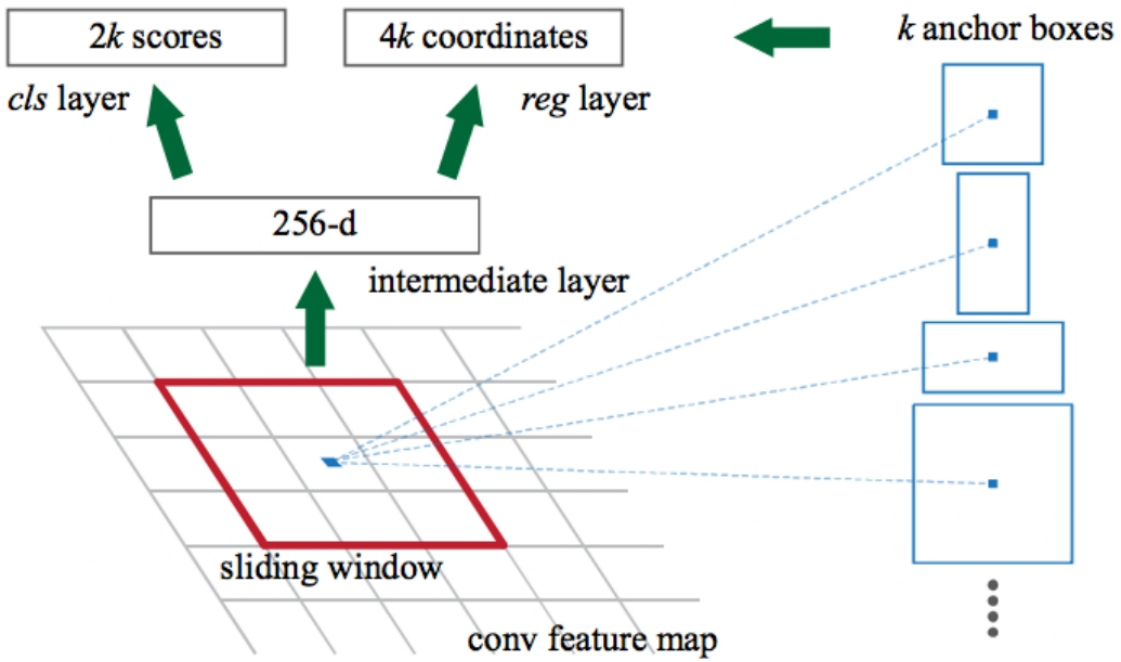
Steps in Faster RCNN

- We take an image as input and pass it to the ConvNet which returns the feature map for that image.
- Region proposal network is applied on these feature maps. This returns the object proposals along with their objectness score.
- A RoI pooling layer is applied on these proposals to bring down all the proposals to the same size.
- Finally, the proposals are passed to a fully connected layer which has a softmax layer and a linear regression layer at its top, to classify and output the bounding boxes for objects.



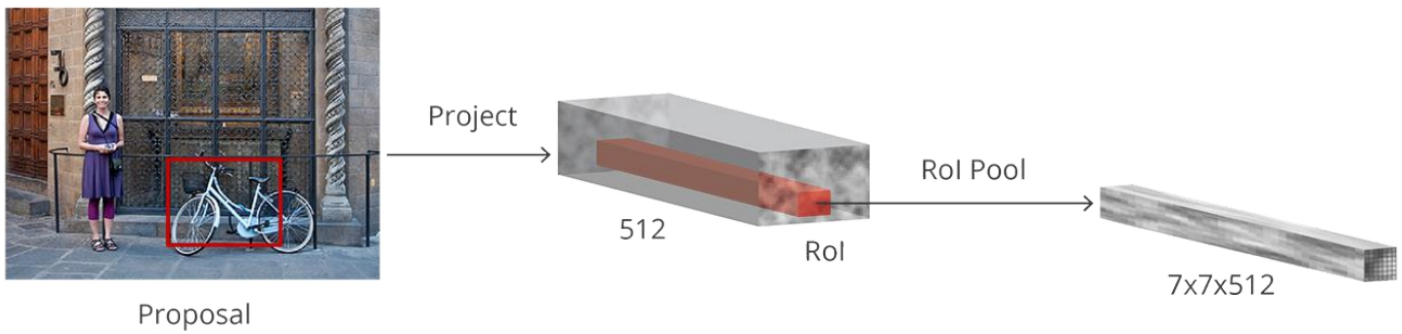
Region Proposal Network

- To begin with, Faster RCNN takes the feature maps from CNN and passes them on to the Region Proposal Network.
- RPN uses a sliding window over these feature maps, and at each window, it generates *k* Anchor boxes of different shapes and sizes:
- Anchor boxes are fixed sized boundary boxes that are placed throughout the image and have different shapes and sizes.
- For each anchor, RPN predicts two things:
  - The first is the probability that an anchor is an object (it does not consider which class the object belongs to)
  - Second is the bounding box regressor for adjusting the anchors to better fit the object



Region Proposal Network

- We now have bounding boxes of different shapes and sizes which are passed on to the RoI pooling layer.
- Now it might be possible that after the RPN step, there are proposals with no classes assigned to them.
- We can take each proposal and crop it so that each proposal contains an object.
- This is what the RoI pooling layer does. It extracts fixed sized feature maps for each anchor:



- Then these feature maps are passed to a fully connected layer which has a softmax and a linear regression layer.
- It finally classifies the object and predicts the bounding boxes for the identified objects.

Problems of faster RCNN

- All of the object detection algorithms we have discussed so far use regions to identify the objects.
- The network does not look at the complete image in one go, but focuses on parts of the image sequentially. This creates two complications:
- The algorithm requires many passes through a single image to extract all the objects
- As there are different systems working one after the other, the performance of the systems further ahead depends on how the previous systems performed

Object Detection types

INTERNATIONAL INSTITUTE OF INFORMATION TECHNOLOGY
HYDERABAD

- **Milestone 1 : Traditional Detector**
  - **Viola Jones Detector**
  - **Hogg Detector**
  - **DPMM**

- **Milestone 2 : CNN-Based Two-Stage Detectors**
  - **RCNN**
  - **Fast RCNN**
  - **Faster RCNN**
  - **Feature Pyramid Network**

- **Milestone 3 : CNN-Based One-Stage Detectors**
  - **YOLO**
  - **Single Shot Multi-box Detector**
  - **DETR**

Object Detection types

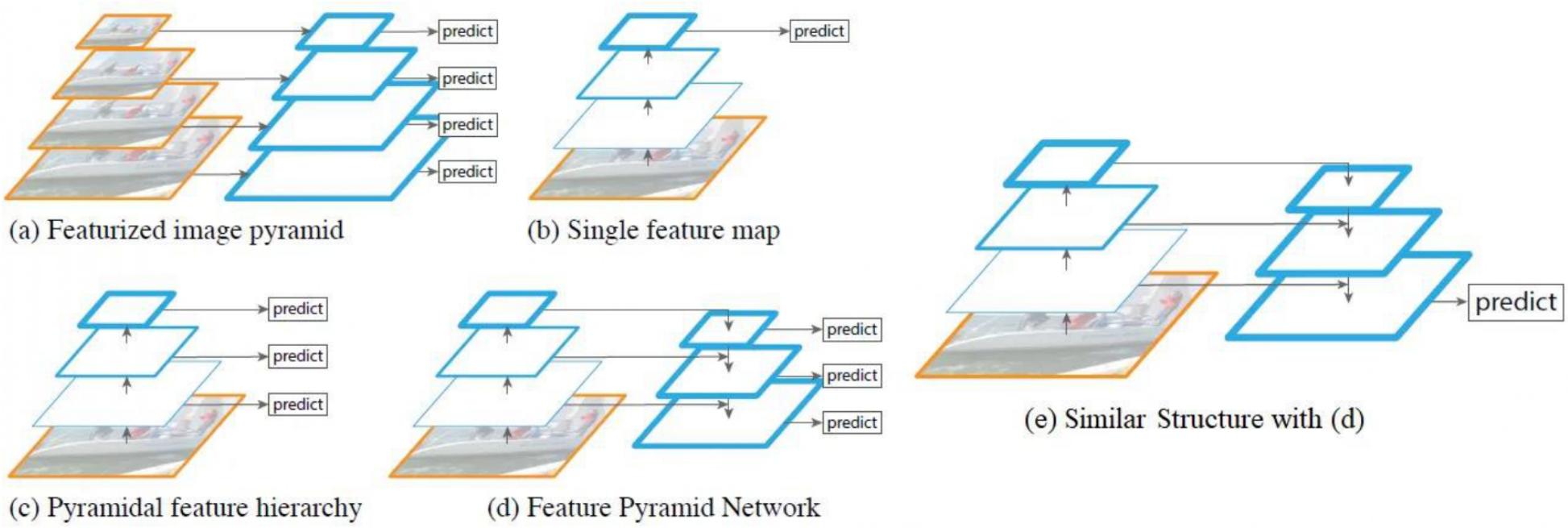
INTERNATIONAL INSTITUTE OF INFORMATION TECHNOLOGY
HYDERABAD

- **Milestone 1 : Traditional Detector**
  - **Viola Jones Detector**
  - **Hogg Detector**
  - **DPMM**

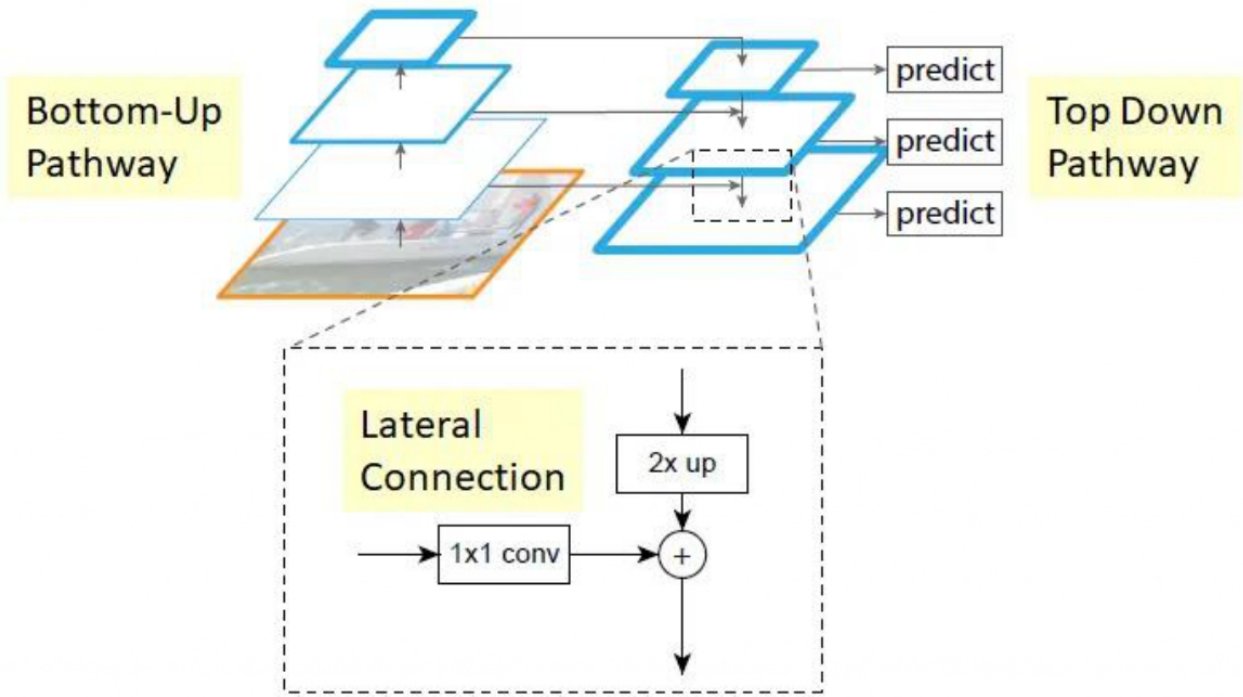
- **Milestone 2 : CNN-Based Two-Stage Detectors**
  - **RCNN**
  - **Fast RCNN**
  - **Faster RCNN**
  - **Feature Pyramid Network**

- **Milestone 3 : CNN-Based One-Stage Detectors**
  - **YOLO**
  - **Single Shot Multi-box Detector**
  - **DETR**

Feature Pyramid Networks



Feature Pyramid Networks



Feature Pyramid Networks

- Multi-scale Feature Representation: FPNs address the challenge of detecting objects at different scales by constructing a feature pyramid that captures information at multiple resolutions. This enables the network to effectively handle objects of varying sizes.

Feature Pyramid Networks

INTERNATIONAL INSTITUTE OF INFORMATION TECHNOLOGY
HYDERABAD

- Multi-scale Feature Representation: FPNs address the challenge of detecting objects at different scales by constructing a feature pyramid that captures information at multiple resolutions. This enables the network to effectively handle objects of varying sizes.

- Bottom-up and Top-down Pathways: FPNs utilize a bottom-up pathway, where features are extracted from lower layers of the network with high spatial resolution but low semantic information. These features are then passed through a top-down pathway, where they are up sampled and combined with higher-level features to create a feature pyramid.

Feature Pyramid Networks

INTERNATIONAL INSTITUTE OF INFORMATION TECHNOLOGY
HYDERABAD

- Multi-scale Feature Representation: FPNs address the challenge of detecting objects at different scales by constructing a feature pyramid that captures information at multiple resolutions. This enables the network to effectively handle objects of varying sizes.

- Bottom-up and Top-down Pathways: FPNs utilize a bottom-up pathway, where features are extracted from lower layers of the network with high spatial resolution but low semantic information. These features are then passed through a top-down pathway, where they are up sampled and combined with higher-level features to create a feature pyramid.

- Feature Fusion: FPNs employ lateral connections to fuse features from different levels of the pyramid. This fusion process allows for the integration of high-level semantic information with detailed spatial information, enhancing the representation power of the network.

Feature Pyramid Networks

INTERNATIONAL INSTITUTE OF INFORMATION TECHNOLOGY
HYDERABAD

- Multi-scale Feature Representation: FPNs address the challenge of detecting objects at different scales by constructing a feature pyramid that captures information at multiple resolutions. This enables the network to effectively handle objects of varying sizes.
- Bottom-up and Top-down Pathways: FPNs utilize a bottom-up pathway, where features are extracted from lower layers of the network with high spatial resolution but low semantic information. These features are then passed through a top-down pathway, where they are up sampled and combined with higher-level features to create a feature pyramid.
- Feature Fusion: FPNs employ lateral connections to fuse features from different levels of the pyramid. This fusion process allows for the integration of high-level semantic information with detailed spatial information, enhancing the representation power of the network.
- Object Detection across Scales: FPNs enable object detection at different scales by performing detection on features from multiple levels of the feature pyramid. This helps in detecting both small and large objects effectively.

Feature Pyramid Networks

INTERNATIONAL INSTITUTE OF INFORMATION TECHNOLOGY
HYDERABAD

- Multi-scale Feature Representation: FPNs address the challenge of detecting objects at different scales by constructing a feature pyramid that captures information at multiple resolutions. This enables the network to effectively handle objects of varying sizes.
- Bottom-up and Top-down Pathways: FPNs utilize a bottom-up pathway, where features are extracted from lower layers of the network with high spatial resolution but low semantic information. These features are then passed through a top-down pathway, where they are up sampled and combined with higher-level features to create a feature pyramid.
- Feature Fusion: FPNs employ lateral connections to fuse features from different levels of the pyramid. This fusion process allows for the integration of high-level semantic information with detailed spatial information, enhancing the representation power of the network.
- Object Detection across Scales: FPNs enable object detection at different scales by performing detection on features from multiple levels of the feature pyramid. This helps in detecting both small and large objects effectively.
- Improved Localization Accuracy: FPNs enhance the localization accuracy of object detection by providing high-resolution feature maps that preserve fine-grained spatial details, especially for small objects

Limitations of Feature Pyramid Networks

- Loss of Localization Precision: While FPNs improve localization accuracy compared to earlier models, there can still be a loss of localization precision for small or densely packed objects. This is because the pooling and upsampling operations involved in creating the feature pyramid can lead to some loss of spatial information.

Limitations of Feature Pyramid Networks

- Loss of Localization Precision: While FPNs improve localization accuracy compared to earlier models, there can still be a loss of localization precision for small or densely packed objects. This is because the pooling and upsampling operations involved in creating the feature pyramid can lead to some loss of spatial information.

- Increased Computational Complexity: FPNs involve additional computations for feature fusion and pyramid construction, leading to increased computational complexity compared to single-scale detection methods. This can impact real-time performance, especially on resource-constrained devices.

Limitations of Feature Pyramid Networks

- Loss of Localization Precision: While FPNs improve localization accuracy compared to earlier models, there can still be a loss of localization precision for small or densely packed objects. This is because the pooling and upsampling operations involved in creating the feature pyramid can lead to some loss of spatial information.

- Increased Computational Complexity: FPNs involve additional computations for feature fusion and pyramid construction, leading to increased computational complexity compared to single-scale detection methods. This can impact real-time performance, especially on resource-constrained devices.

- Complexity of Architecture: The design and implementation of FPNs require careful consideration of the network architecture and parameter settings. Selecting appropriate scales, levels, and fusion strategies can be challenging and may require manual tuning or extensive experimentation.

Limitations of Feature Pyramid Networks

- Loss of Localization Precision: While FPNs improve localization accuracy compared to earlier models, there can still be a loss of localization precision for small or densely packed objects. This is because the pooling and upsampling operations involved in creating the feature pyramid can lead to some loss of spatial information.
- Increased Computational Complexity: FPNs involve additional computations for feature fusion and pyramid construction, leading to increased computational complexity compared to single-scale detection methods. This can impact real-time performance, especially on resource-constrained devices.
- Complexity of Architecture: The design and implementation of FPNs require careful consideration of the network architecture and parameter settings. Selecting appropriate scales, levels, and fusion strategies can be challenging and may require manual tuning or extensive experimentation.
- Potential Semantic Gap: FPNs rely on features from both low-level and high-level layers to construct the feature pyramid. However, there can still be a semantic gap between these features, potentially affecting the network's ability to capture contextual information accurately.

Object Detection types

INTERNATIONAL INSTITUTE OF INFORMATION TECHNOLOGY
HYDERABAD

- **Milestone 1 : Traditional Detector**
  - **Viola Jones Detector**
  - **Hogg Detector**
  - **DPMM**

- **Milestone 2 : CNN-Based Two-Stage Detectors**
  - **RCNN**
  - **Fast RCNN**
  - **Faster RCNN**
  - **Feature Pyramid Network**

- **Milestone 3 : CNN-Based One-Stage Detectors**
  - **YOLO**
  - **Single Shot Multi-box Detector**
  - **DETR**

Object Detection types

- Milestone 1 : Traditional Detector
  - Viola Jones Detector
  - Hogg Detector
  - DPMM
- Milestone 2 : CNN-Based Two-Stage Detectors
  - RCNN
  - Fast RCNN
  - Faster RCNN
  - Feature Pyramid Network
- Milestone 3 : CNN-Based One-Stage Detectors
  - YOLO
  - Single Shot Multi-box Detector
  - DETR

Understanding : YOLO

INTERNATIONAL INSTITUTE OF INFORMATION TECHNOLOGY
HYDERABAD

Take an Input Image

Understanding : YOLO

INTERNATIONAL INSTITUTE OF INFORMATION TECHNOLOGY
HYDERABAD

Divide that into 3x3 grid

Understanding : YOLO

INTERNATIONAL INSTITUTE OF INFORMATION TECHNOLOGY
HYDERABAD

Divide that into 3x3 grid

Image Localization and Classification applied to each grid.

Understanding : YOLO

If we have 3 classes to detect :

The target vector would look like :

y =	pc
	bx
	by
	bh
	bw
	c1
	c2
	c3

Understanding : YOLO

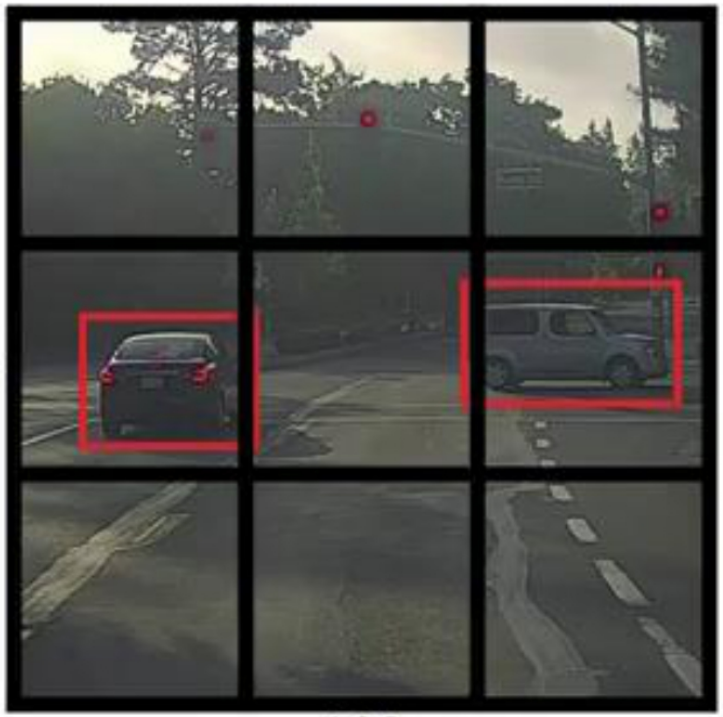
If there is no object, like the first grid: 

Understanding : YOLO

If there is no object, like the first grid:

Then the target vector looks like:

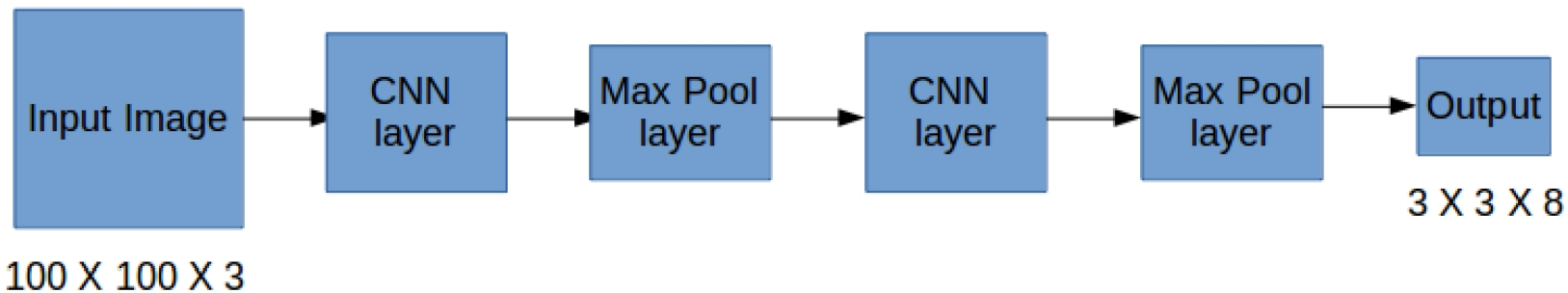
y =	0
	?
	?
	?
	?
	?
	?
	?



Understanding : YOLO

So now we have an input image and its corresponding target image.

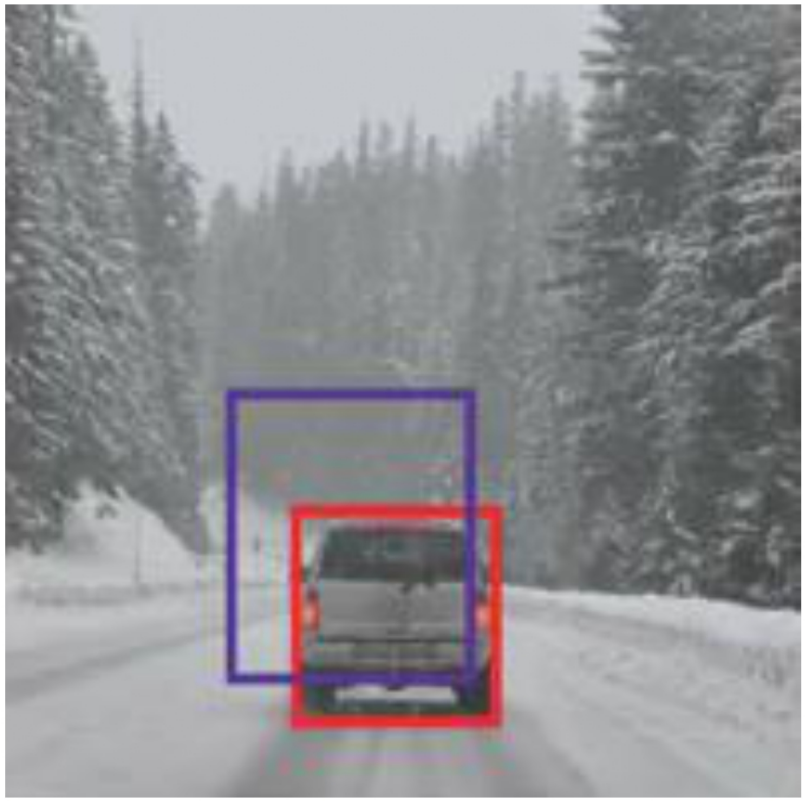
So, using the previous example we can train it using a normal CNN network like this :



Intersection over Union

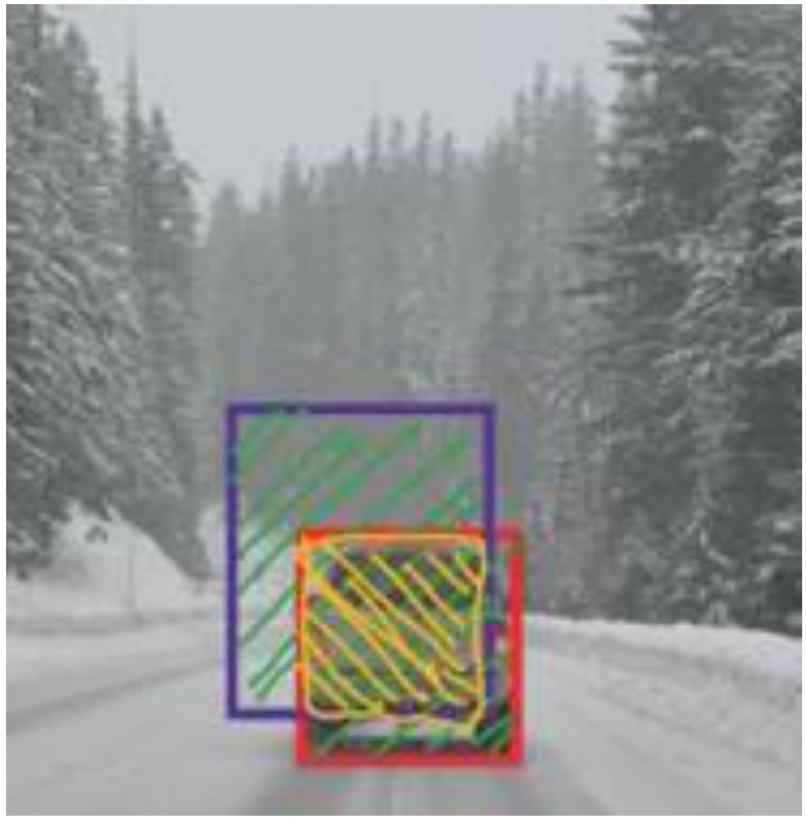
INTERNATIONAL INSTITUTE OF INFORMATION TECHNOLOGY
HYDERABAD

How can we decide whether the predicted bounding box is giving us a good outcome (or a bad one)?

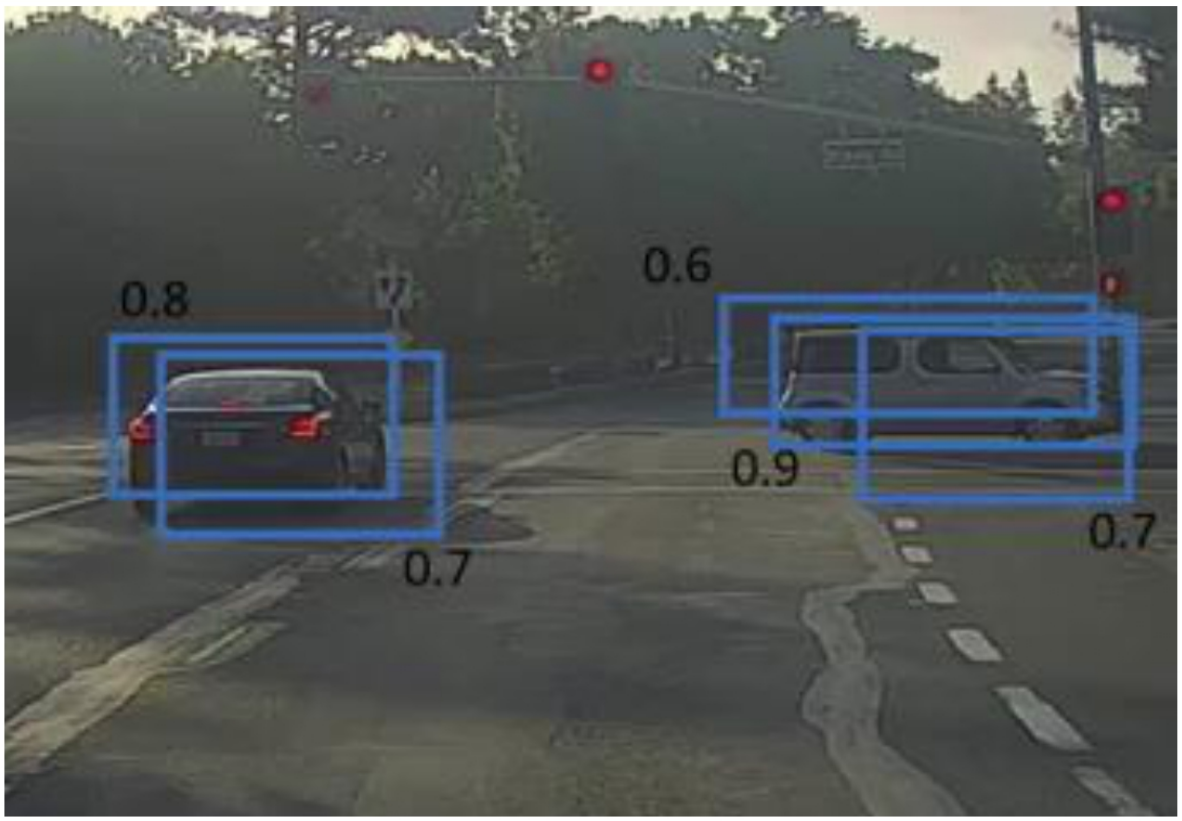


Intersection over Union

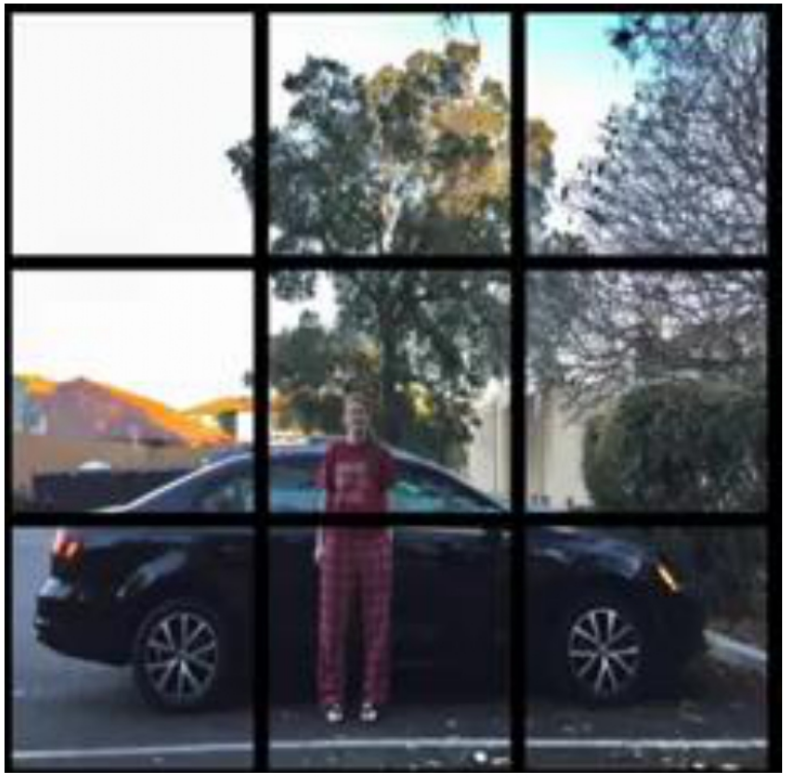
How can we decide whether the predicted bounding box is giving us a good outcome (or a bad one)?

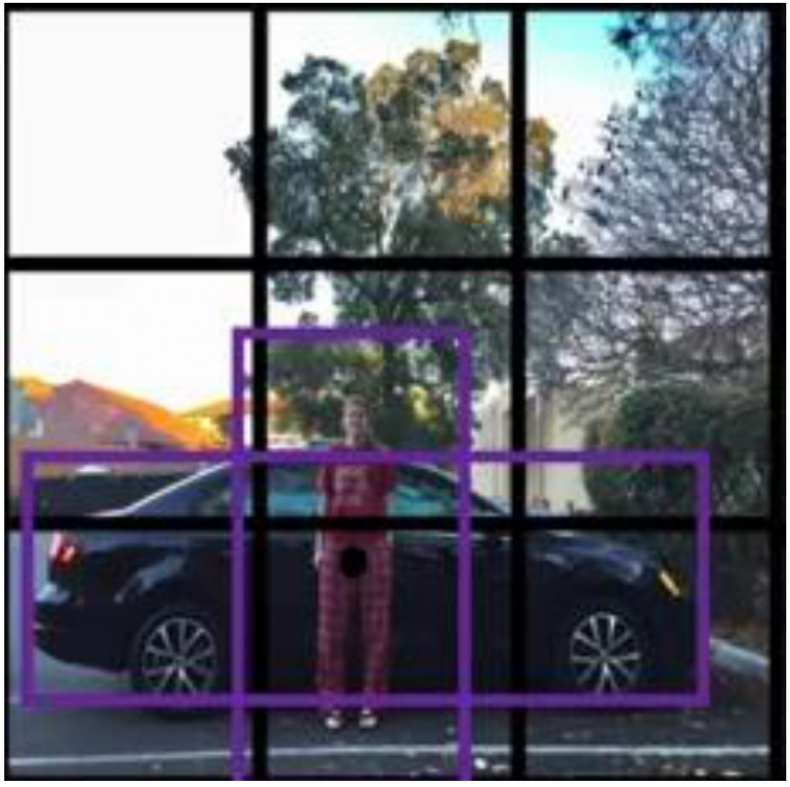


Non-Maximum Suppression

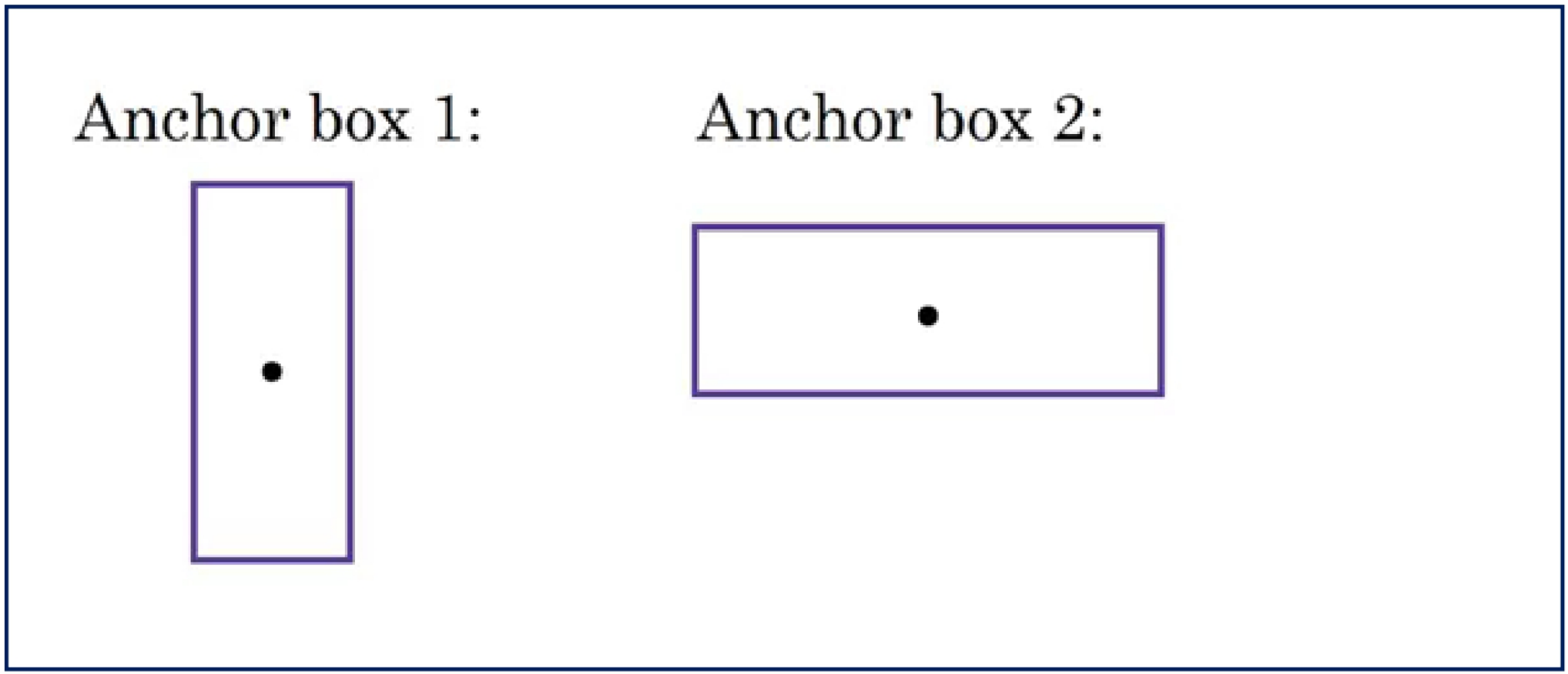


Anchor boxes



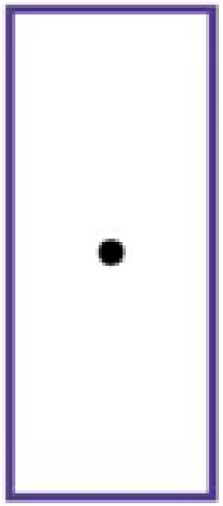


Anchor boxes

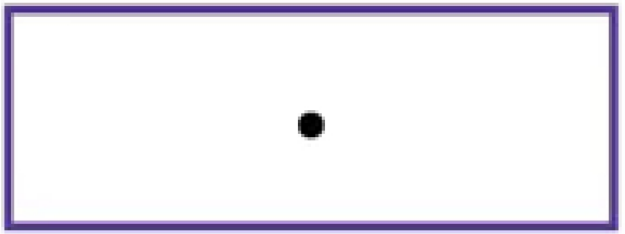


Anchor boxes

Anchor box 1:



Anchor box 2:



y =	pc
	bx
	by
	bh
	bw
	c1
	c2
	c3
	pc
	bx
	by
	bh
	bw
	c1
	c2
	c3

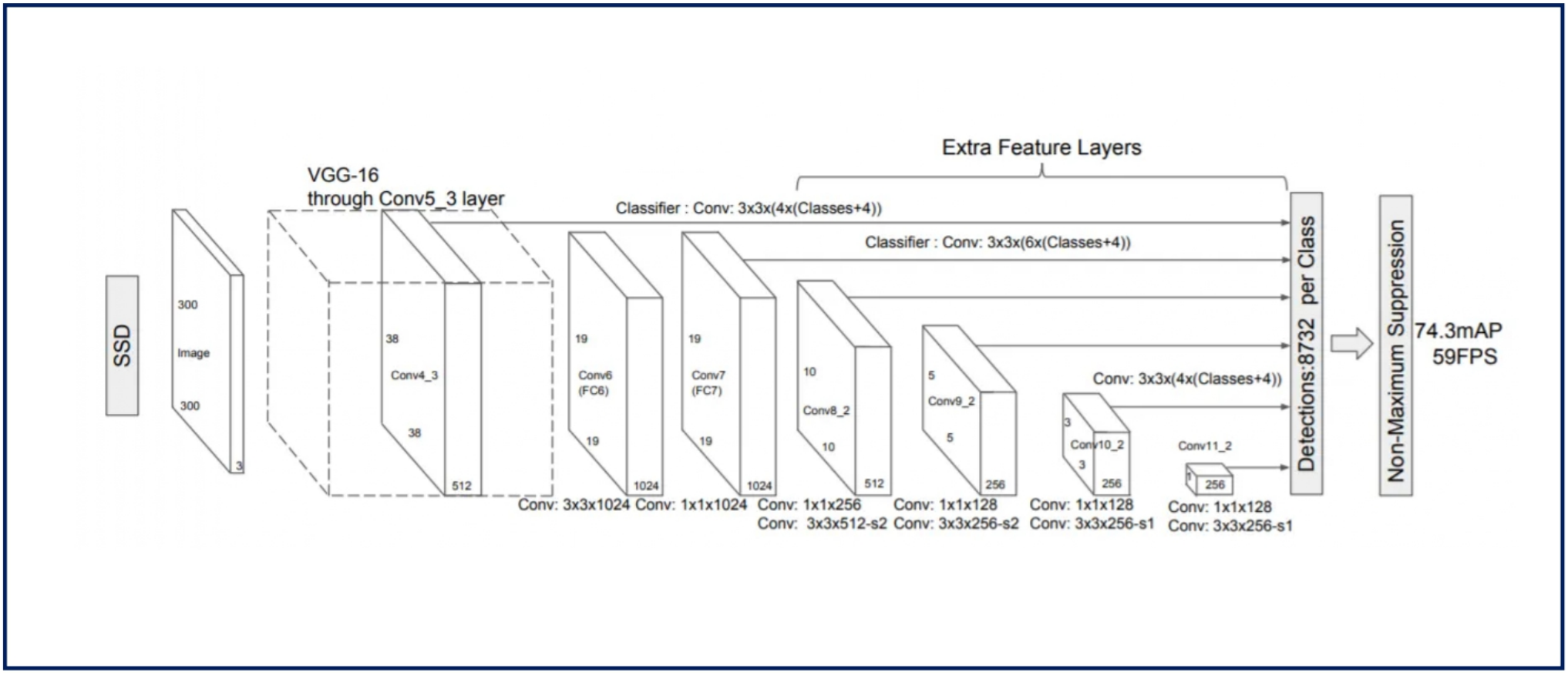
Object Detection types

- Milestone 1 : Traditional Detector
  - Viola Jones Detector
  - Hogg Detector
  - DPMM
- Milestone 2 : CNN-Based Two-Stage Detectors
  - RCNN
  - Fast RCNN
  - Faster RCNN
  - Feature Pyramid Network
- Milestone 3 : CNN-Based One-Stage Detectors
  - YOLO
  - Single Shot Multi-box Detector
  - DETR

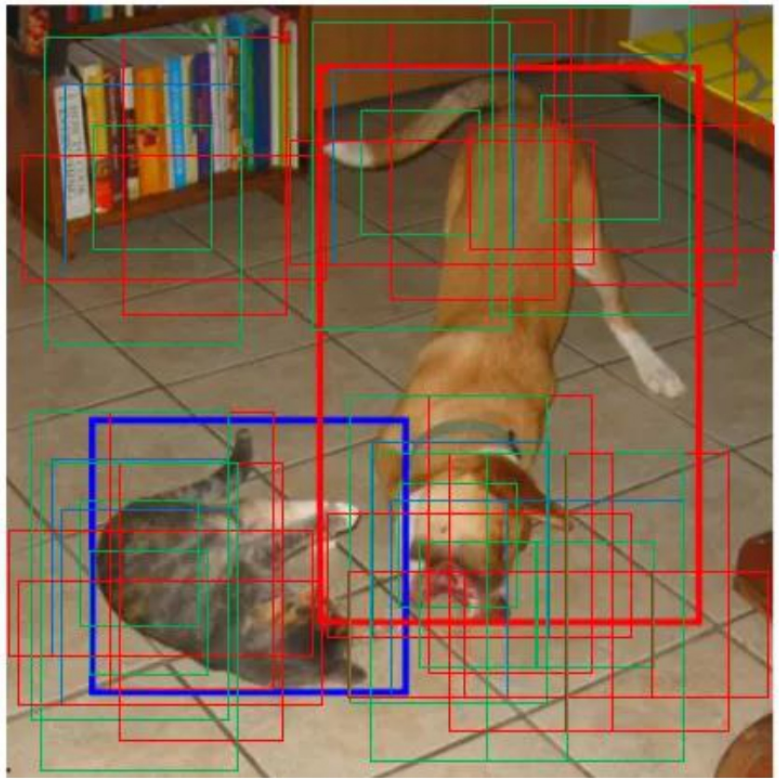
Object Detection types

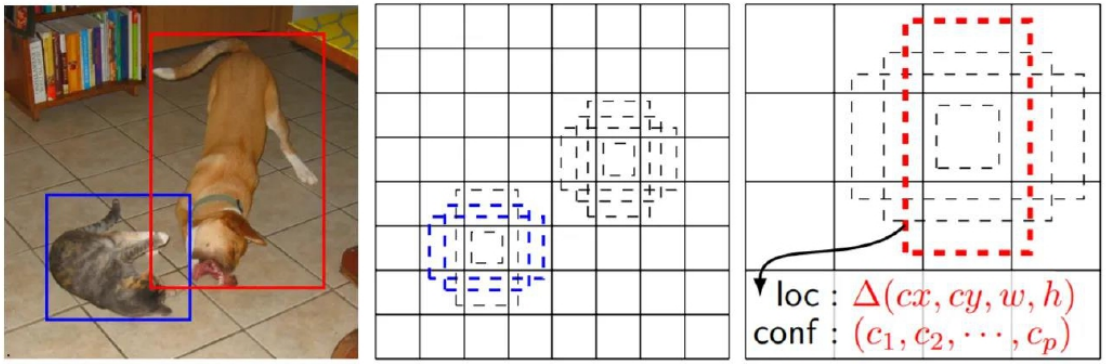
- Milestone 1 : Traditional Detector
  - Viola Jones Detector
  - Hogg Detector
  - DPMM
- Milestone 2 : CNN-Based Two-Stage Detectors
  - RCNN
  - Fast RCNN
  - Faster RCNN
  - Feature Pyramid Network
- Milestone 3 : CNN-Based One-Stage Detectors
  - YOLO
  - Single Shot Multi-box Detector
  - DETR

Single Shot Multibox Detector



Single Shot Multibox Detector

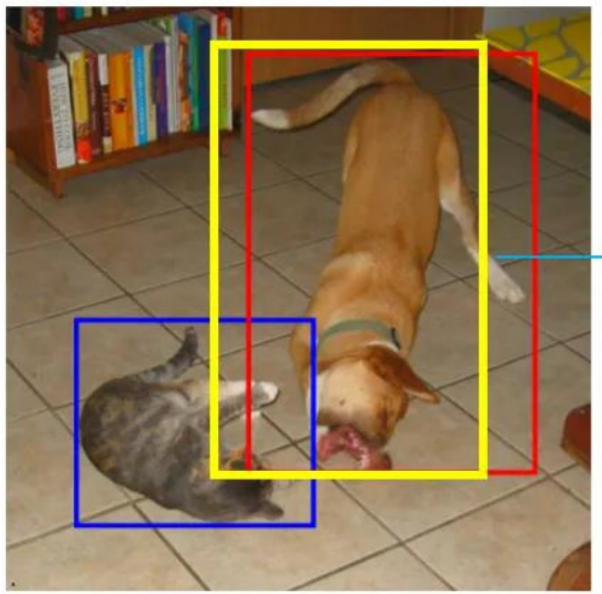




(a) Image with GT boxes (b) 8×8 feature map (c) 4×4 feature map

Single Shot Multibox Detector

- **Single-stage Detector:** SSD is a single-stage object detector that predicts object bounding boxes and class probabilities directly from feature maps. It eliminates the need for a separate region proposal stage, making it faster compared to two-stage detectors.

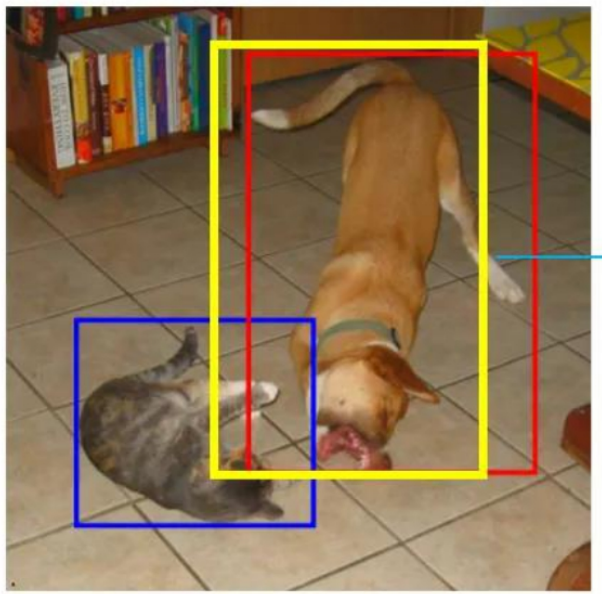


$loc \rightarrow \Delta(Cx, C_y, h, w)$

Conf-> c1, c2,, cn

Single Shot Multibox Detector

- **Single-stage Detector:** SSD is a single-stage object detector that predicts object bounding boxes and class probabilities directly from feature maps. It eliminates the need for a separate region proposal stage, making it faster compared to two-stage detectors.
- **Multi-scale Feature Maps:** SSD utilizes feature maps at multiple scales to detect objects of various sizes. This is achieved by applying convolutional layers with different kernel sizes to extract features at different receptive fields.

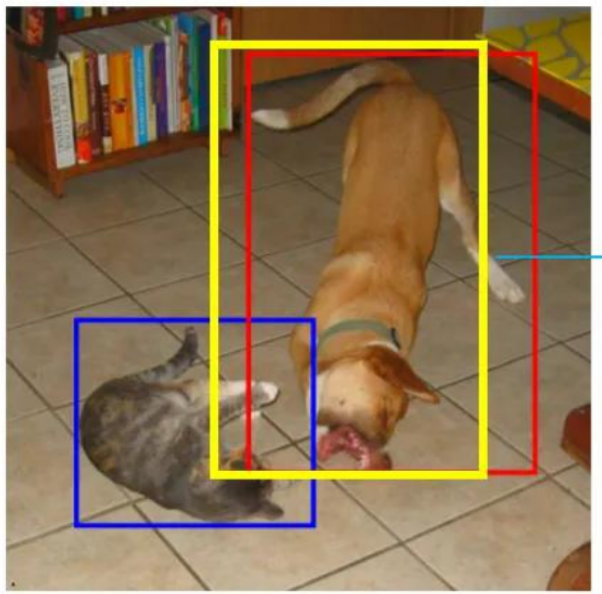


$loc \rightarrow \Delta(Cx, C_y, h, w)$

Conf-> c1, c2,, cn

Single Shot Multibox Detector

- **Single-stage Detector:** SSD is a single-stage object detector that predicts object bounding boxes and class probabilities directly from feature maps. It eliminates the need for a separate region proposal stage, making it faster compared to two-stage detectors.
- **Multi-scale Feature Maps:** SSD utilizes feature maps at multiple scales to detect objects of various sizes. This is achieved by applying convolutional layers with different kernel sizes to extract features at different receptive fields.
- **Default Boxes (Priors):** SSD employs a set of predefined default boxes (or priors) at each feature map location, with different aspect ratios and scales. These default boxes serve as anchors for predicting object locations and sizes.

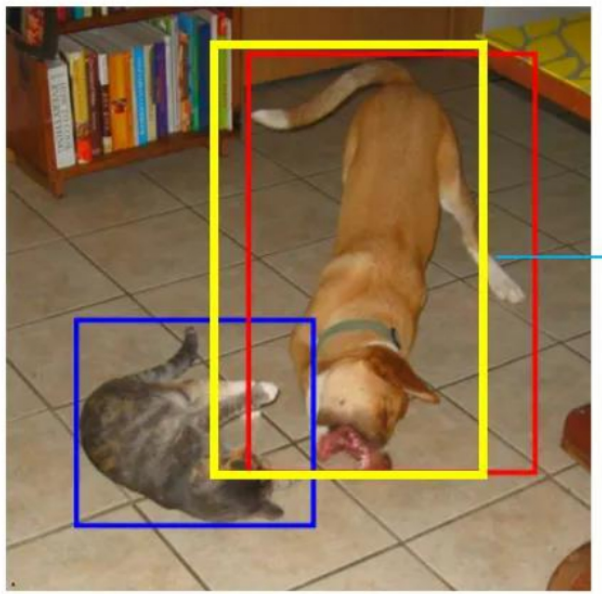


$loc \rightarrow \Delta(Cx, C_y, h, w)$

Conf-> c1, c2,, cn

Single Shot Multibox Detector

- **Multi-level Predictions:** SSD performs multi-level predictions by applying convolutional filters of different sizes to each feature map. This enables the network to capture object information at various scales and resolutions.

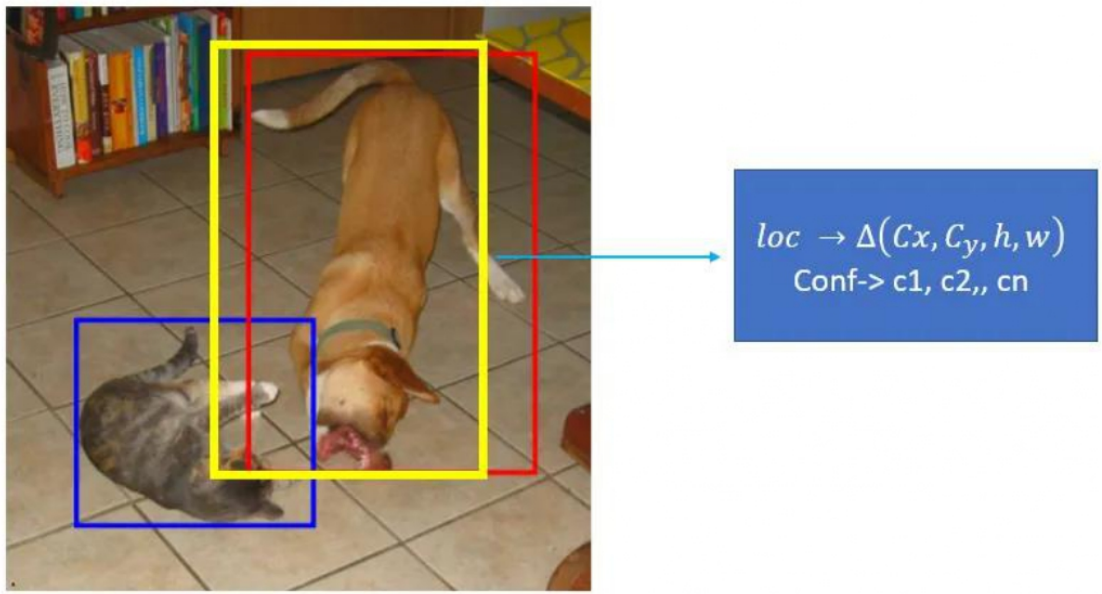


$loc \rightarrow \Delta(Cx, C_y, h, w)$

Conf-> c1, c2,, cn

Single Shot Multibox Detector

- **Multi-level Predictions:** SSD performs multi-level predictions by applying convolutional filters of different sizes to each feature map. This enables the network to capture object information at various scales and resolutions.
- **Feature Fusion:** SSD incorporates feature fusion techniques, such as concatenating feature maps from different layers, to improve the representation power of the network and enhance detection accuracy.



Limitations : Single Shot Multibox Detector

- **Localization Accuracy:** SSDs can sometimes struggle with precise localization of objects, especially small or densely packed ones. This is because the fixed aspect ratios and scales of the default boxes may not adequately cover all object variations, leading to localization errors.

Limitations : Single Shot Multibox Detector

- **Localization Accuracy:** SSDs can sometimes struggle with precise localization of objects, especially small or densely packed ones. This is because the fixed aspect ratios and scales of the default boxes may not adequately cover all object variations, leading to localization errors.

- **Difficulty in Handling Aspect Ratio Variations:** While SSDs provide default boxes with different aspect ratios, they may not capture objects with extreme aspect ratios accurately. This limitation can affect the detection performance on objects with non-standard shapes.

Limitations : Single Shot Multibox Detector

INTERNATIONAL INSTITUTE OF INFORMATION TECHNOLOGY
HYDERABAD

- **Localization Accuracy:** SSDs can sometimes struggle with precise localization of objects, especially small or densely packed ones. This is because the fixed aspect ratios and scales of the default boxes may not adequately cover all object variations, leading to localization errors.

- **Difficulty in Handling Aspect Ratio Variations:** While SSDs provide default boxes with different aspect ratios, they may not capture objects with extreme aspect ratios accurately. This limitation can affect the detection performance on objects with non-standard shapes.

- **Object Class Imbalance:** In scenarios where the dataset has imbalanced class distributions, SSDs might face challenges in accurately detecting objects from underrepresented classes. The network tends to prioritize classes with higher occurrence, potentially leading to biased results.

Limitations : Single Shot Multibox Detector

INTERNATIONAL INSTITUTE OF INFORMATION TECHNOLOGY
HYDERABAD

- **Localization Accuracy:** SSDs can sometimes struggle with precise localization of objects, especially small or densely packed ones. This is because the fixed aspect ratios and scales of the default boxes may not adequately cover all object variations, leading to localization errors.

- **Difficulty in Handling Aspect Ratio Variations:** While SSDs provide default boxes with different aspect ratios, they may not capture objects with extreme aspect ratios accurately. This limitation can affect the detection performance on objects with non-standard shapes.

- **Object Class Imbalance:** In scenarios where the dataset has imbalanced class distributions, SSDs might face challenges in accurately detecting objects from underrepresented classes. The network tends to prioritize classes with higher occurrence, potentially leading to biased results.

- **Parameter Sensitivity:** The performance of SSDs can be sensitive to the selection of parameters, such as the number and aspect ratios of default boxes, and the design of the feature pyramid. Careful tuning and hyperparameter optimization are crucial for achieving optimal results.

Limitations : Single Shot Multibox Detector

INTERNATIONAL INSTITUTE OF INFORMATION TECHNOLOGY
HYDERABAD

- **Localization Accuracy:** SSDs can sometimes struggle with precise localization of objects, especially small or densely packed ones. This is because the fixed aspect ratios and scales of the default boxes may not adequately cover all object variations, leading to localization errors.
- **Difficulty in Handling Aspect Ratio Variations:** While SSDs provide default boxes with different aspect ratios, they may not capture objects with extreme aspect ratios accurately. This limitation can affect the detection performance on objects with non-standard shapes.
- **Object Class Imbalance:** In scenarios where the dataset has imbalanced class distributions, SSDs might face challenges in accurately detecting objects from underrepresented classes. The network tends to prioritize classes with higher occurrence, potentially leading to biased results.
- **Parameter Sensitivity:** The performance of SSDs can be sensitive to the selection of parameters, such as the number and aspect ratios of default boxes, and the design of the feature pyramid. Careful tuning and hyperparameter optimization are crucial for achieving optimal results.
- **Trade-off Between Speed and Accuracy:** While SSDs are faster compared to two-stage detectors, there is often a trade-off between speed and accuracy. In certain scenarios, SSDs may sacrifice some detection accuracy to achieve real-time or faster inference speed.

Object Detection types

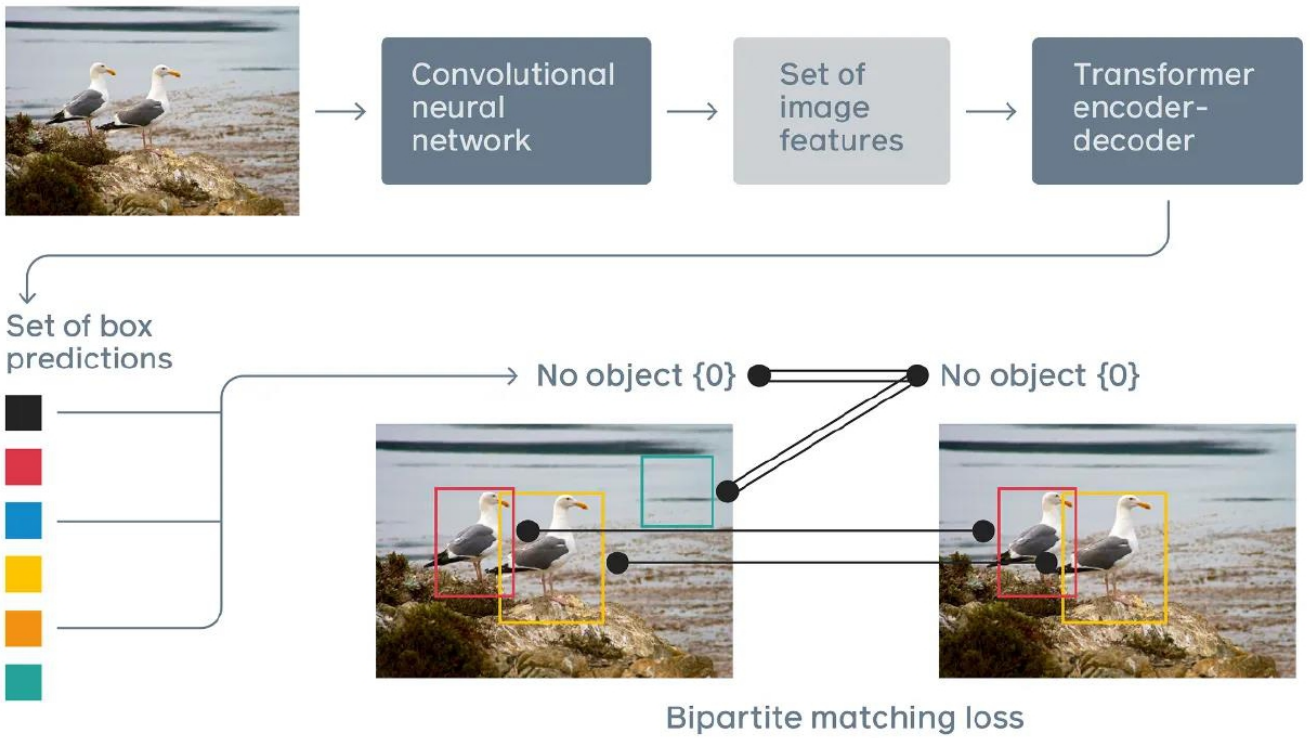
- Milestone 1 : Traditional Detector
  - Viola Jones Detector
  - Hogg Detector
  - DPMM
- Milestone 2 : CNN-Based Two-Stage Detectors
  - RCNN
  - Fast RCNN
  - Faster RCNN
  - Feature Pyramid Network
- Milestone 3 : CNN-Based One-Stage Detectors
  - YOLO
  - Single Shot Multi-box Detector
  - DETR

Object Detection types

INTERNATIONAL INSTITUTE OF INFORMATION TECHNOLOGY
HYDERABAD

- Milestone 1 : Traditional Detector
  - Viola Jones Detector
  - Hogg Detector
  - DPMM
- Milestone 2 : CNN-Based Two-Stage Detectors
  - RCNN
  - Fast RCNN
  - Faster RCNN
  - Feature Pyramid Network
- Milestone 3 : CNN-Based One-Stage Detectors
  - YOLO
  - Single Shot Multi-box Detector
  - DETR

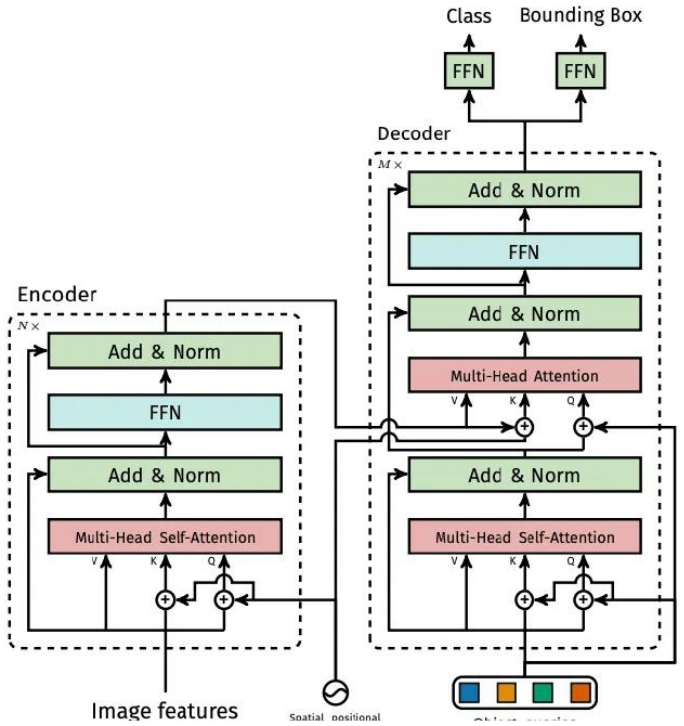
DETR : *End-to-End Object Detection with Transformers*



DETR : *End-to-End Object Detection with Transformers*

INTERNATIONAL INSTITUTE OF INFORMATION TECHNOLOGY HYDERABAD

- **Transformer-based Architecture:** DETR leverages the Transformer architecture, originally introduced for natural language processing tasks, for object detection. This architecture allows for capturing global context and modeling long-range dependencies, making it suitable for object detection tasks.

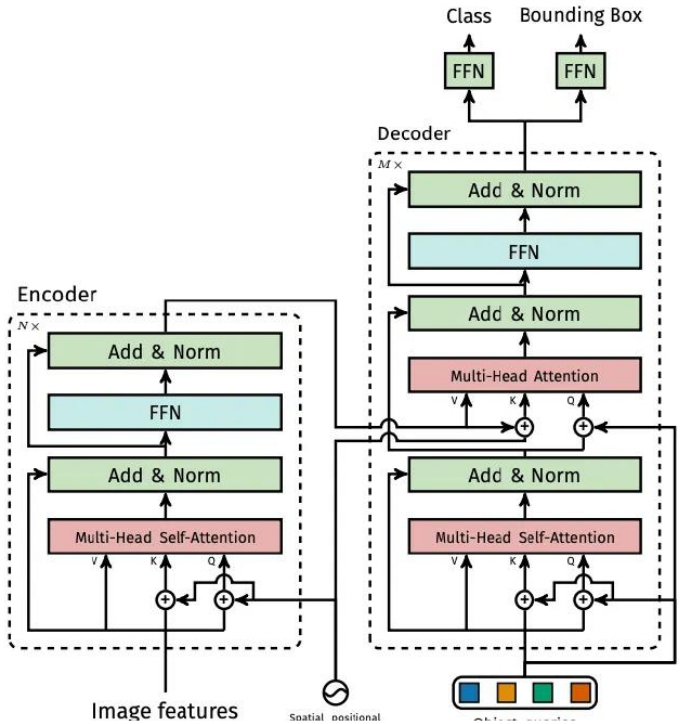


DETR : *End-to-End Object Detection with Transformers*

INTERNATIONAL INSTITUTE OF INFORMATION TECHNOLOGY HYDERABAD

- **Transformer-based Architecture:** DETR leverages the Transformer architecture, originally introduced for natural language processing tasks, for object detection. This architecture allows for capturing global context and modeling long-range dependencies, making it suitable for object detection tasks.

- **End-to-End Training:** DETR enables end-to-end training by formulating object detection as a set prediction problem. It predicts object bounding boxes and their corresponding class labels simultaneously, without the need for anchor boxes or region proposal stages.



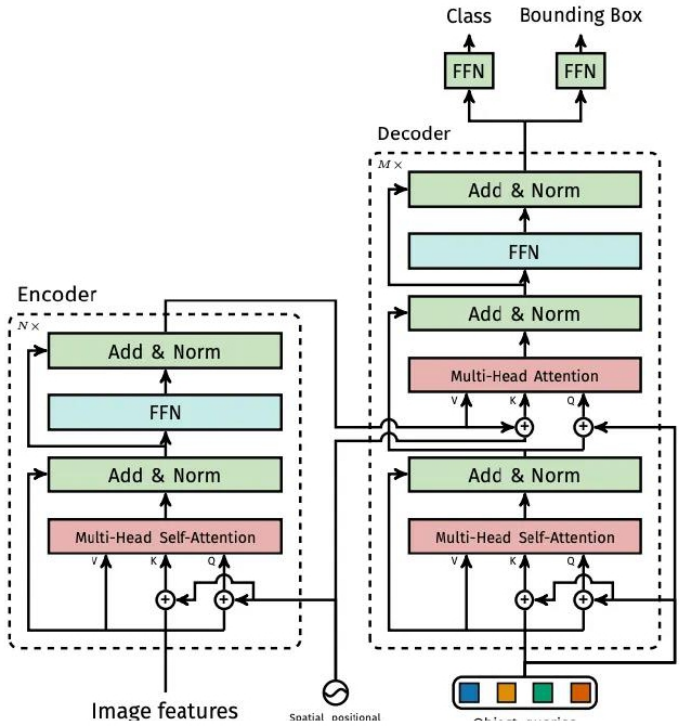
DETR : *End-to-End Object Detection with Transformers*

INTERNATIONAL INSTITUTE OF INFORMATION TECHNOLOGY HYDERABAD

- **Transformer-based Architecture:** DETR leverages the Transformer architecture, originally introduced for natural language processing tasks, for object detection. This architecture allows for capturing global context and modeling long-range dependencies, making it suitable for object detection tasks.

- **End-to-End Training:** DETR enables end-to-end training by formulating object detection as a set prediction problem. It predicts object bounding boxes and their corresponding class labels simultaneously, without the need for anchor boxes or region proposal stages.

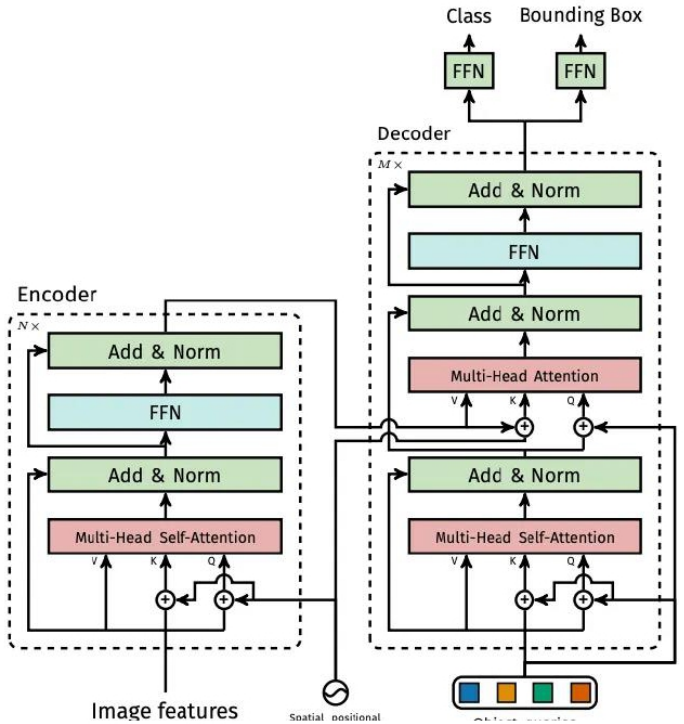
- **Set Prediction and Bipartite Matching:** DETR uses a set prediction approach, where it predicts a fixed number of object detections as a permutation of the input objects. It employs a bipartite matching algorithm to assign predictions to ground truth objects, ensuring one-to-one correspondence.



DETR : *End-to-End Object Detection with Transformers*

INTERNATIONAL INSTITUTE OF INFORMATION TECHNOLOGY HYDERABAD

- **Positional Encoding:** DETR incorporates positional encoding to provide spatial information to the Transformer. This encoding helps in preserving the relative positions of objects within the image, which is crucial for accurate localization.

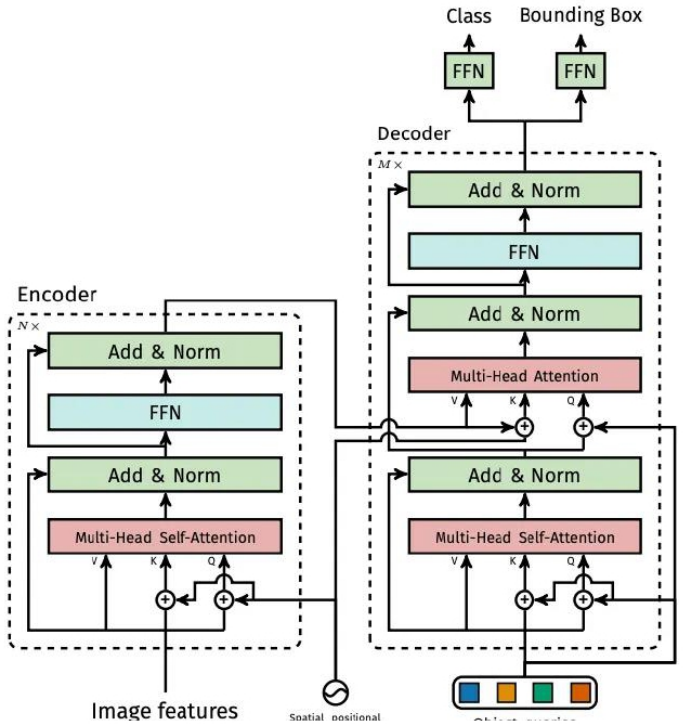


DETR : *End-to-End Object Detection with Transformers*

INTERNATIONAL INSTITUTE OF INFORMATION TECHNOLOGY HYDERABAD

- **Positional Encoding:** DETR incorporates positional encoding to provide spatial information to the Transformer. This encoding helps in preserving the relative positions of objects within the image, which is crucial for accurate localization.

- **Object Queries and Decoder:** DETR utilizes object queries, which act as learnable representations of objects, to attend to relevant features in the image. The decoder part of the Transformer processes these object queries and generates the final object predictions.



Limitations of DETR

INTERNATIONAL INSTITUTE OF INFORMATION TECHNOLOGY
HYDERABAD

- **Lack of Spatial Invariance:** DETR's attention mechanism operates on a fixed-size grid of features, which can lead to a loss of spatial invariance. This limitation may affect the model's ability to accurately handle object deformations, rotations, or translations.

Limitations of DETR

INTERNATIONAL INSTITUTE OF INFORMATION TECHNOLOGY
HYDERABAD

- **Lack of Spatial Invariance:** DETR's attention mechanism operates on a fixed-size grid of features, which can lead to a loss of spatial invariance. This limitation may affect the model's ability to accurately handle object deformations, rotations, or translations.

- **Difficulty in Handling Large Object Classes:** DETR's set prediction approach can be challenging for datasets with a large number of object classes. The model may struggle to assign correct labels to objects from underrepresented classes, leading to lower detection accuracy for those classes.

Limitations of DETR

INTERNATIONAL INSTITUTE OF INFORMATION TECHNOLOGY
HYDERABAD

- **Lack of Spatial Invariance:** DETR's attention mechanism operates on a fixed-size grid of features, which can lead to a loss of spatial invariance. This limitation may affect the model's ability to accurately handle object deformations, rotations, or translations.

- **Difficulty in Handling Large Object Classes:** DETR's set prediction approach can be challenging for datasets with a large number of object classes. The model may struggle to assign correct labels to objects from underrepresented classes, leading to lower detection accuracy for those classes.

- **Localization Accuracy for Small Objects:** DETR can face challenges in accurately localizing small objects due to the fixed-size grid representation. The limited spatial resolution can result in imprecise bounding box predictions for small or densely packed objects.

Limitations of DETR

INTERNATIONAL INSTITUTE OF INFORMATION TECHNOLOGY
HYDERABAD

- **Lack of Spatial Invariance:** DETR's attention mechanism operates on a fixed-size grid of features, which can lead to a loss of spatial invariance. This limitation may affect the model's ability to accurately handle object deformations, rotations, or translations.
- **Difficulty in Handling Large Object Classes:** DETR's set prediction approach can be challenging for datasets with a large number of object classes. The model may struggle to assign correct labels to objects from underrepresented classes, leading to lower detection accuracy for those classes.
- **Localization Accuracy for Small Objects:** DETR can face challenges in accurately localizing small objects due to the fixed-size grid representation. The limited spatial resolution can result in imprecise bounding box predictions for small or densely packed objects.
- **Computational Complexity:** The Transformer architecture used in DETR can be computationally expensive, especially for high-resolution images or large-scale datasets. The extensive self-attention operations and the need to process the entire image in parallel can increase the model's computational requirements.

Limitations of DETR

INTERNATIONAL INSTITUTE OF INFORMATION TECHNOLOGY
HYDERABAD

- **Lack of Spatial Invariance:** DETR's attention mechanism operates on a fixed-size grid of features, which can lead to a loss of spatial invariance. This limitation may affect the model's ability to accurately handle object deformations, rotations, or translations.
- **Difficulty in Handling Large Object Classes:** DETR's set prediction approach can be challenging for datasets with a large number of object classes. The model may struggle to assign correct labels to objects from underrepresented classes, leading to lower detection accuracy for those classes.
- **Localization Accuracy for Small Objects:** DETR can face challenges in accurately localizing small objects due to the fixed-size grid representation. The limited spatial resolution can result in imprecise bounding box predictions for small or densely packed objects.
- **Computational Complexity:** The Transformer architecture used in DETR can be computationally expensive, especially for high-resolution images or large-scale datasets. The extensive self-attention operations and the need to process the entire image in parallel can increase the model's computational requirements.
- **Training Data Dependencies:** DETR's performance heavily relies on the availability of large-scale, diverse, and annotated training data. Insufficient or biased training data can impact the model's generalization ability and lead to suboptimal performance on unseen data.

Object Detection Datasets

1. **PASCAL-VOC (The PASCAL Visual Object Classes):**
- Dataset Size: The PASCAL-VOC dataset contains around 11,530 images.
- Train/Val/Test Split:
  - Training Set: Approximately 9,000 images.
  - Validation Set: Around 1,000 images.
  - Testing Set: Approximately 1,530 images.

Object Detection Datasets

INTERNATIONAL INSTITUTE OF INFORMATION TECHNOLOGY
HYDERABAD

1. **PASCAL-VOC (The PASCAL Visual Object Classes):**
- Dataset Size: The PASCAL-VOC dataset contains around 11,530 images.
- Train/Val/Test Split:
  - Training Set: Approximately 9,000 images.
  - Validation Set: Around 1,000 images.
  - Testing Set: Approximately 1,530 images.

2. **ISLVRC (ImageNet Large Scale Visual Recognition Challenge):**
- Dataset Size: The ImageNet dataset used for ISLVRC is extensive, containing millions of images.
- Train/Val/Test Split:
  - Training Set: Approximately 1.2 million images.
  - Validation Set: Around 50,000 images.
  - Testing Set: Approximately 100,000 images.

Object Detection Datasets

3. **MS COCO (Microsoft Common Objects in Context):**
- Dataset Size: The MS COCO dataset comprises approximately 330,000 images.
- Train/Val/Test Split:
  - Training Set: Around 118,000 images.
  - Validation Set: Approximately 5,000 images.
  - Testing Set: Around 40,000 images.

Object Detection Datasets

INTERNATIONAL INSTITUTE OF INFORMATION TECHNOLOGY
HYDERABAD

3. **MS COCO (Microsoft Common Objects in Context):**
- Dataset Size: The MS COCO dataset comprises approximately 330,000 images.
- Train/Val/Test Split:
  - Training Set: Around 118,000 images.
  - Validation Set: Approximately 5,000 images.
  - Testing Set: Around 40,000 images.

4. **Open Images :**
- Dataset Size: The Open Images dataset is a large-scale dataset containing millions of images.
- Train/Val/Test Split:
  - The dataset provides different splits for training, validation, and testing. The specific numbers for each split may vary depending on the version or subset of the dataset used.

Object Detection Metrics

INTERNATIONAL INSTITUTE OF INFORMATION TECHNOLOGY
HYDERABAD

- **Intersection over Union (IoU):**
  - IoU measures the overlap between the predicted bounding box and the ground truth bounding box.
  - Formula: IoU = Intersection Area / Union Area.

- **Precision and Recall:**
  - Precision measures the accuracy of positive detections, while recall measures the ability to detect positive instances.
  - Formulas: Precision = TP / (TP + FP), Recall = TP / (TP + FN).

- **Average Precision (AP):**
  - AP summarizes the precision-recall curve and provides a single scalar value for detection performance.
  - Formula: AP = ∫(Recall) d(Precision).

- **Mean Average Precision (mAP):**
  - mAP is the mean value of AP across multiple object categories.
  - Formula: mAP = (AP_1 + AP_2 + ... + AP_N) / N.

- **F1 Score:**
  - The F1 score is the harmonic mean of precision and recall, providing a balanced measure.
  - Formula: F1 Score = 2 * (Precision * Recall) / (Precision + Recall).

Object Detection Metrics

INTERNATIONAL INSTITUTE OF INFORMATION TECHNOLOGY
HYDERABAD

- **True Positive Rate (TPR) and False Positive Rate (FPR):**
  - TPR (also recall) measures the proportion of true positives out of all positive instances, while FPR measures the proportion of false positives out of all negative instances.
  - Formulas: TPR = TP / (TP + FN), FPR = FP / (FP + TN).

- **Log Average Miss Rate (logMAR):**
  - logMAR quantifies detection error at different levels of false positives per image.
  - Formula: logMAR = log(mean(1 - Miss Rate)).

Technical Evolution of Multi-Scale Detection

INTERNATIONAL INSTITUTE OF INFORMATION TECHNOLOGY
HYDERABAD

1. **Single-Scale Object Detection:**
   - Initially, object detection models focused on detecting objects at a fixed scale within an image.
   - These models typically employed a fixed receptive field size or used a single-scale image pyramid for object detection.
   - The limitations of single-scale detection became evident when objects appeared at different scales or underwent scale variations.

Technical Evolution of Multi-Scale Detection

INTERNATIONAL INSTITUTE OF INFORMATION TECHNOLOGY
HYDERABAD

1. **Single-Scale Object Detection:**
   - Initially, object detection models focused on detecting objects at a fixed scale within an image.
   - These models typically employed a fixed receptive field size or used a single-scale image pyramid for object detection.
   - The limitations of single-scale detection became evident when objects appeared at different scales or underwent scale variations.

2. **Scale-Invariant Object Detection:**
   - To address the limitations of single-scale detection, researchers introduced scale-invariant approaches.
   - These methods aimed to detect objects at multiple scales by employing image pyramids or multi-scale feature extraction.
   - Image pyramids involved generating scaled versions of the input image, allowing detection at different resolutions.
   - Multi-scale feature extraction involved incorporating features from multiple layers of a convolutional neural network (CNN) to capture objects at different scales.

Technical Evolution of Multi-Scale Detection

INTERNATIONAL INSTITUTE OF INFORMATION TECHNOLOGY
HYDERABAD

3. **Image Pyramids and Sliding Windows:**
   - Image pyramids, combined with sliding window techniques, became popular for multi-scale object detection.
   - Sliding windows involved moving a fixed-sized window across different positions and scales within an image.
   - The use of image pyramids and sliding windows enabled detection of objects at various scales, but it incurred high computational costs due to the exhaustive search process.

Technical Evolution of Multi-Scale Detection

INTERNATIONAL INSTITUTE OF INFORMATION TECHNOLOGY
HYDERABAD

3. **Image Pyramids and Sliding Windows:**
   - Image pyramids, combined with sliding window techniques, became popular for multi-scale object detection.
   - Sliding windows involved moving a fixed-sized window across different positions and scales within an image.
   - The use of image pyramids and sliding windows enabled detection of objects at various scales, but it incurred high computational costs due to the exhaustive search process.

4. **Feature Pyramid Networks (FPN):**
   - Feature Pyramid Networks (FPN) introduced a more efficient approach to multi-scale detection.
   - FPN utilized a top-down pathway to fuse features from different layers of a CNN, creating a feature pyramid.
   - The feature pyramid allowed for capturing objects at multiple scales while maintaining high-level semantic information.
   - FPN improved the accuracy of object detection by enabling better handling of objects at different scales and achieving more robust feature representation.

Technical Evolution of Multi-Scale Detection

INTERNATIONAL INSTITUTE OF INFORMATION TECHNOLOGY
HYDERABAD

5. **Anchor-based Approaches:**
   - Alongside the development of multi-scale detection techniques, anchor-based approaches gained popularity.
   - These methods introduced anchor boxes, which are predefined bounding boxes of various scales and aspect ratios.
   - Anchor boxes were matched with objects at different scales during training and used for localization and classification.
   - By incorporating anchor boxes, models could efficiently handle objects at multiple scales and aspect ratios, leading to improved object detection accuracy.

Technical Evolution of Multi-Scale Detection

INTERNATIONAL INSTITUTE OF INFORMATION TECHNOLOGY
HYDERABAD

5. **Anchor-based Approaches:**
   - Alongside the development of multi-scale detection techniques, anchor-based approaches gained popularity.
   - These methods introduced anchor boxes, which are predefined bounding boxes of various scales and aspect ratios.
   - Anchor boxes were matched with objects at different scales during training and used for localization and classification.
   - By incorporating anchor boxes, models could efficiently handle objects at multiple scales and aspect ratios, leading to improved object detection accuracy.

6. **Efficient Backbone Networks:**
   - The evolution of multi-scale detection also involved the development of more efficient backbone networks.
   - Lightweight architectures, such as MobileNet and EfficientNet, were designed to provide high-level feature representations while minimizing computational costs.
   - Efficient backbone networks enabled real-time multi-scale object detection on resource-constrained devices, expanding the practicality and accessibility of the technology.

Technical Evolution of Multi-Scale Detection

INTERNATIONAL INSTITUTE OF INFORMATION TECHNOLOGY
HYDERABAD

7. **Attention Mechanisms and Context Integration:**
   - Recent advancements in multi-scale detection have explored the integration of attention mechanisms and contextual information.
   - Attention mechanisms allow models to dynamically focus on informative regions and scales within an image.
   - Context integration techniques leverage global context or scene information to enhance object detection performance across different scales and complex scenarios.

Technical Evolution of Loss Functions

INTERNATIONAL INSTITUTE OF INFORMATION TECHNOLOGY
HYDERABAD

1.**Early Loss Functions:**

- In the early stages of object detection, common loss functions such as mean squared error (MSE) or cross-entropy loss were used.
- For example, in the pioneering work of Viola and Jones, a variant of AdaBoost algorithm with exponential loss was used as the loss function for face detection [1].

Technical Evolution of Loss Functions

INTERNATIONAL INSTITUTE OF INFORMATION TECHNOLOGY
HYDERABAD

1.**Early Loss Functions:**
- In the early stages of object detection, common loss functions such as mean squared error (MSE) or cross-entropy loss were used.
- For example, in the pioneering work of Viola and Jones, a variant of AdaBoost algorithm with exponential loss was used as the loss function for face detection [1].

2. **Localization Loss:**
- As object detection moved towards localizing objects with bounding boxes, specific loss functions were introduced to address the localization task.
- The localization loss function typically consists of a combination of terms, such as the smooth L1 loss, which penalizes the difference between predicted and ground-truth bounding box coordinates.
- Research papers like "Fast R-CNN" by Girshick [2] and "Faster R-CNN" by Ren et al. [3] introduced these loss functions to improve object localization accuracy.

Technical Evolution of Loss Functions

3. **Classification Loss:**
   - Object detection involves both localization and classification tasks. Therefore, appropriate loss functions for classification were incorporated into object detection models.
   - Cross-entropy loss, softmax loss, or sigmoid loss functions have been widely used for the classification component of object detection models.
   - Papers like "Fast R-CNN" by Girshick [2] and "YOLOv3" by Redmon and Farhadi [4] employed these loss functions for classification in object detection models.

Technical Evolution of Loss Functions

INTERNATIONAL INSTITUTE OF INFORMATION TECHNOLOGY
HYDERABAD

3. **Classification Loss:**
   - Object detection involves both localization and classification tasks. Therefore, appropriate loss functions for classification were incorporated into object detection models.
   - Cross-entropy loss, softmax loss, or sigmoid loss functions have been widely used for the classification component of object detection models.
   - Papers like "Fast R-CNN" by Girshick [2] and "YOLOv3" by Redmon and Farhadi [4] employed these loss functions for classification in object detection models.

4. **IoU-based Loss:**
   - Intersection over Union (IoU) is a widely used metric for evaluating the accuracy of bounding box predictions.
   - IoU-based loss functions, such as the IoU loss or the GIoU (Generalized IoU) loss, directly optimize the IoU between predicted and ground-truth bounding boxes.
   - Research papers like "RetinaNet" by Lin et al. [5] and "EfficientDet" by Tan et al. [6] introduced IoU-based loss functions to improve the alignment between predicted and ground-truth bounding boxes.

Technical Evolution of Loss Functions

INTERNATIONAL INSTITUTE OF INFORMATION TECHNOLOGY
HYDERABAD

5. **Focal Loss:**
   - Focal loss was introduced to address the problem of class imbalance in object detection, where background samples significantly outnumber object samples.
   - The focal loss function assigns higher weights to hard examples (misclassified or difficult samples) and reduces the emphasis on easy examples during training.
   - The paper "Focal Loss for Dense Object Detection" by Lin et al. [7] introduced the focal loss, which significantly improved object detection performance, especially for challenging samples.

Technical Evolution of Loss Functions

INTERNATIONAL INSTITUTE OF INFORMATION TECHNOLOGY
HYDERABAD

5. **Focal Loss:**
   - Focal loss was introduced to address the problem of class imbalance in object detection, where background samples significantly outnumber object samples.
   - The focal loss function assigns higher weights to hard examples (misclassified or difficult samples) and reduces the emphasis on easy examples during training.
   - The paper "Focal Loss for Dense Object Detection" by Lin et al. [7] introduced the focal loss, which significantly improved object detection performance, especially for challenging samples.

6. **Center-based Loss:**
   - Center-based loss functions were developed to improve the localization accuracy of small or densely packed objects.
   - These loss functions aim to predict the center coordinates of objects instead of directly regressing bounding box coordinates.
   - Papers like "CenterNet" by Zhou et al. [8] and "Objects as Points" by Zhou et al. [9] introduced center-based loss functions, which achieved high performance in detecting small objects.

Thank You,

Seshadri Mazumder,
PhD Computer Science & Engineering [3rd Year],
Center for Visual Information Technology [CVIT],
International Institute of Information Technology-Hyderabad [IIITH]