Deep Regionlets for Object Detection

Hongyu Xu, Xutao Lv, Xiaoyu Wang, Zhou Ren,
Navaneeth Bodla and Rama Chellappa (ECCV 2018)

University at Buffalo The State University of New York

Introduction

- **Regionlets*** is an object detection approach for detecting objects of different scales, arbitrary viewpoints.
- The paper incorporates the concept of Regionlets to build an end-to-end trainable deep network.

* *Regionlets for Generic Object Detection*, Xiaoyu Wang Ming Yang Shenghuo Zhu Yuanqing Lin, Published in ICCV 2013 [link]

Previous Works

- Hand-crafted feature based models e.g. HoG* , SIFT**
- Deep Learning approaches for object detection, eg. Fast R-CNN, Faster R-CNN etc.
- DL approaches could be categorized into **single stage** and **two-stage** models.

* *Histograms of Oriented Gradients for Human Detection*, Navneet Dalal ,Bill Triggs, Published in CVPR '05 [Link]
** *Distinctive Image Features from Scale-Invariant Keypoints*, David G. Lowe , Published in IJCV'04 [Link]

Regionlets

- Proposed by Wang et al.
- These are a part of a hierarchical window design:
  - Candidate Bounding Box
  - Regions inside the Bounding Box
  - Sub-regions inside each Region

Regionlets

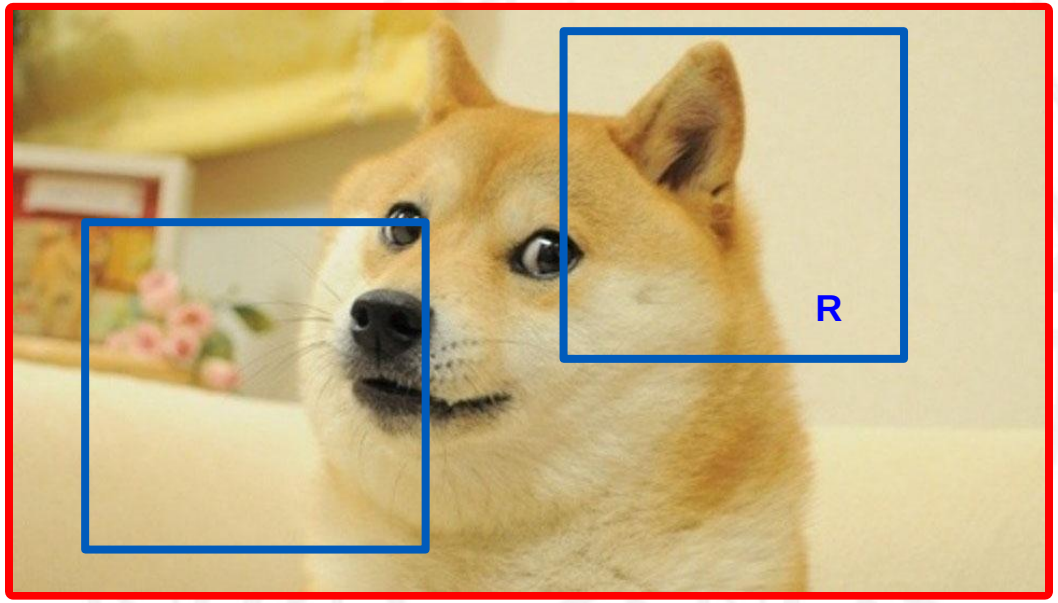
- Proposed by Wang et al.
- These are a part of a hierarchical window design:
  - Candidate Bounding Box
  - Regions inside the Bounding Box
  - Sub-regions inside each Region

Regionlets

- Proposed by Wang et al.
- These are a part of a hierarchical window design:
  - Candidate Bounding Box
  - Regions inside the Bounding Box
  - Sub-regions inside each Region

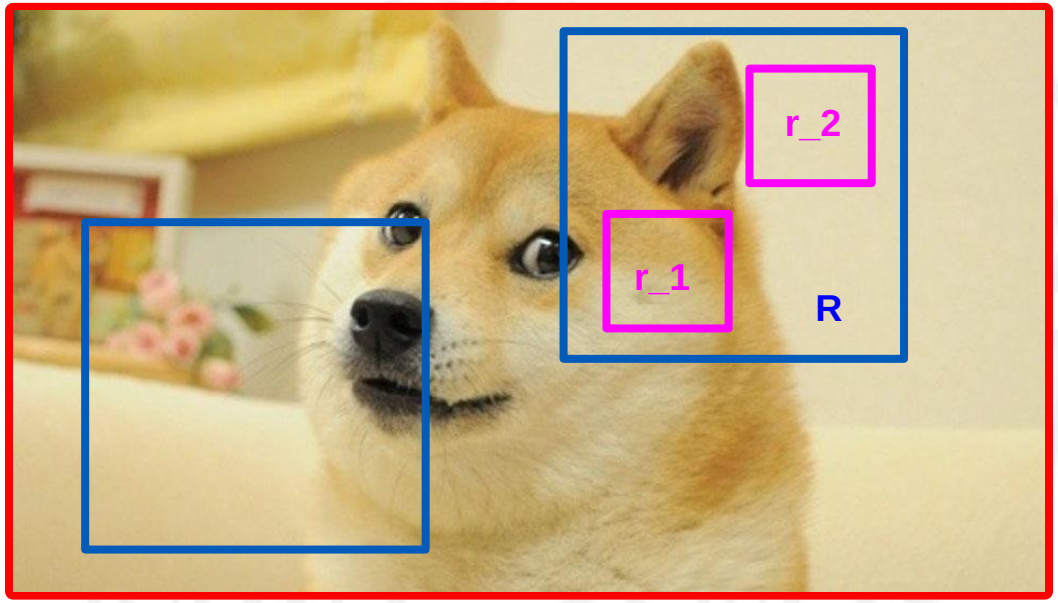
Regionlets

- Proposed by Wang et al.
- These are a part of a hierarchical window design:
  - Candidate Bounding Box
  - Regions inside the Bounding Box
  - Sub-regions inside each Region



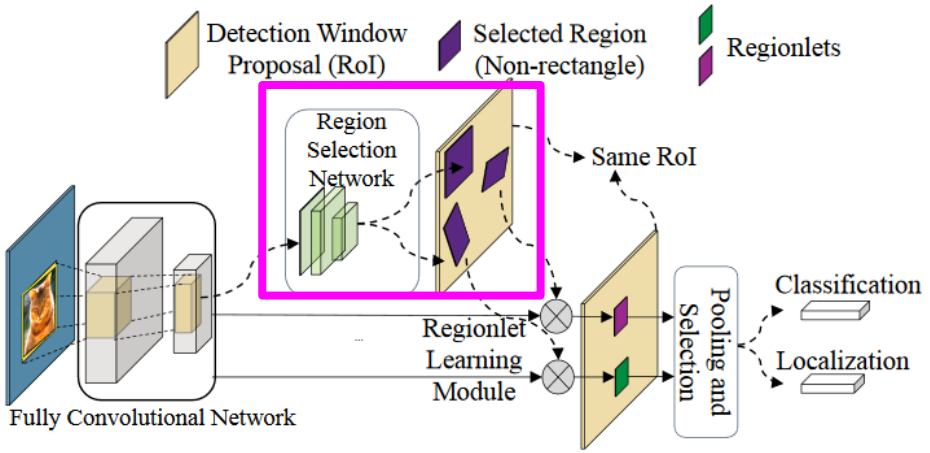
Regionlets

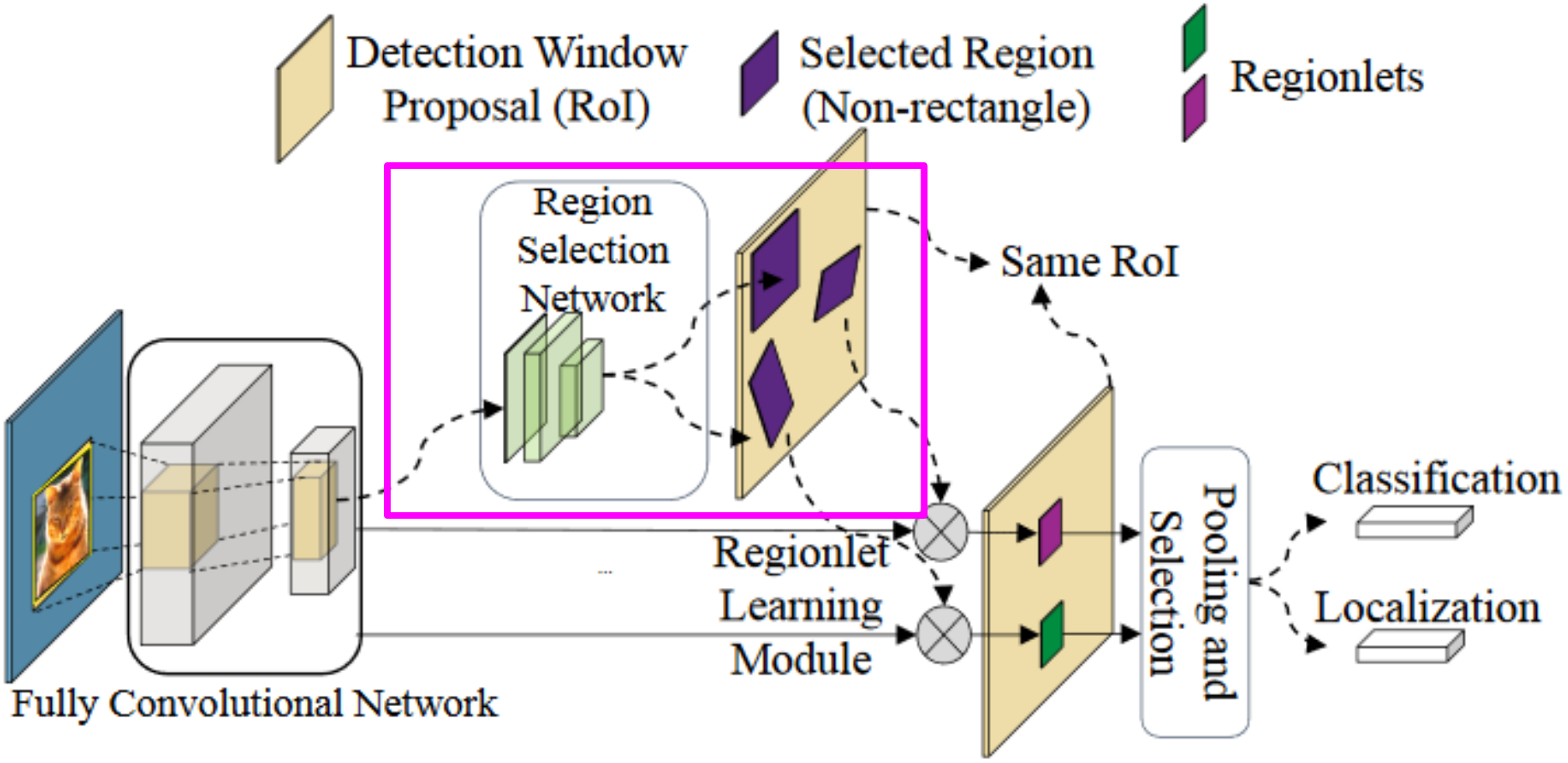
- Proposed by Wang et al.
- These are a part of a hierarchical window design:
  - Candidate Bounding Box
  - Regions inside the Bounding Box (R)
  - Sub-regions inside each Region (r_1, r_2, … r_n) (set of *regionlets*)



Architecture

- Two stage object detector
- It consists of a **Region Selection Network(RSN)** and a **Deep Regionlet learning module**.
- **Region Selection Network (RSN)**
  - predicts the transformation parameters to choose regions within a given candidate bounding box, generated from RPN.
  - Generates regions of arbitrary shapes which can be transformed using *affine* transformation.

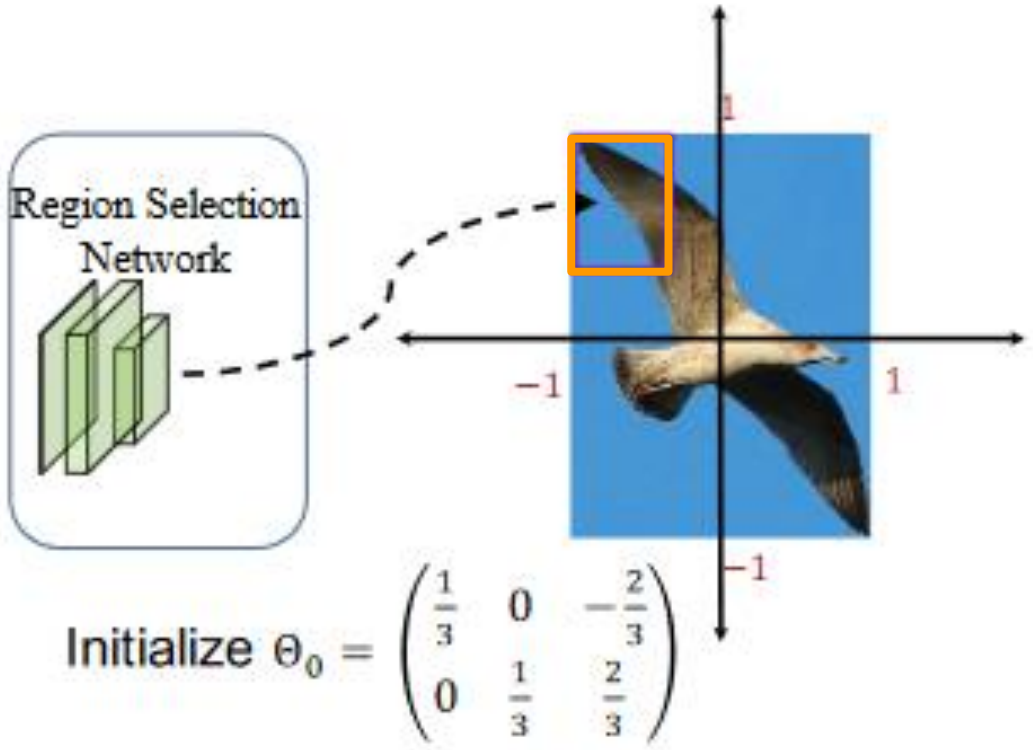



Detection Window Proposal (RoI)
Selected Region (Non-rectangle)
Regionlets
Region Selection Network
Same RoI
Regionlet Learning Module
Pooling and Selection
Classification
Localization
Fully Convolutional Network

- Goal : To predict a set of affine transformation parameters, rep. as :

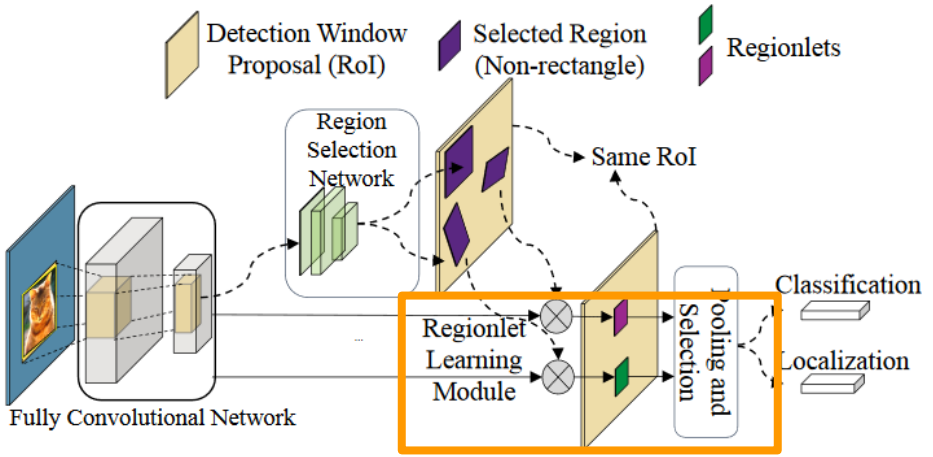
  θ = [θ_1, θ_2, θ_3 ;
  θ_4, θ_5, θ_6] where θ_i ϵ [-1, 1]

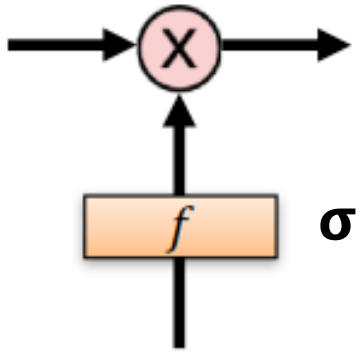
- Normalized affine transformation parameters
- RSN initialized by splitting the bounding box in sub-regions called *cells*

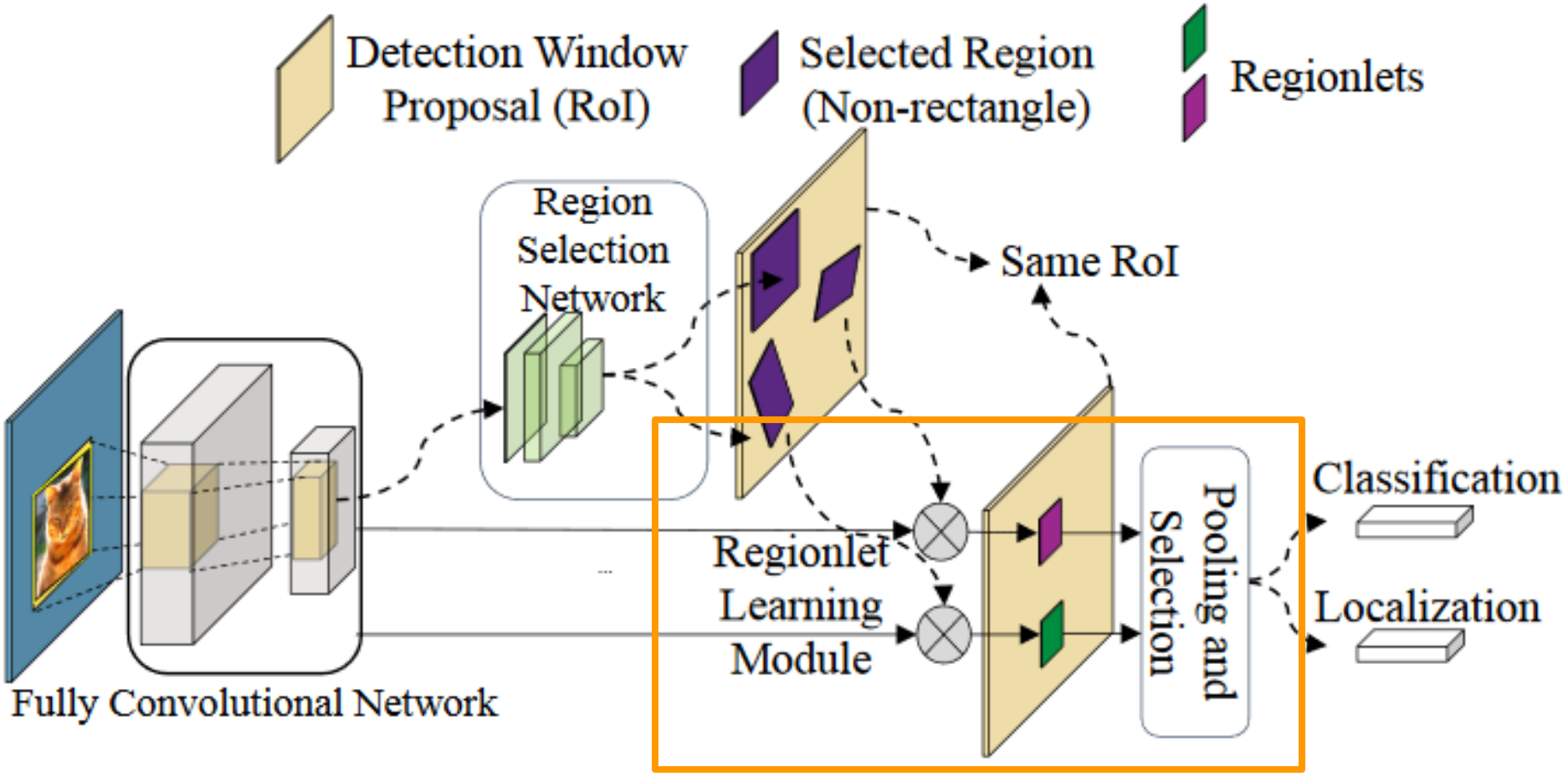


Architecture

- Two stage object detector
- It consists of a **Region Selection Network(RSN)** and a **Deep Regionlet learning module**.
- **Deep Regionlet Learning Module**
  - RSN selects the regions, now the regionlets need to be learned from the selected regions.
- **Gating Network**
  - assigns regionlets with different weights and generates regionlet feature representation.





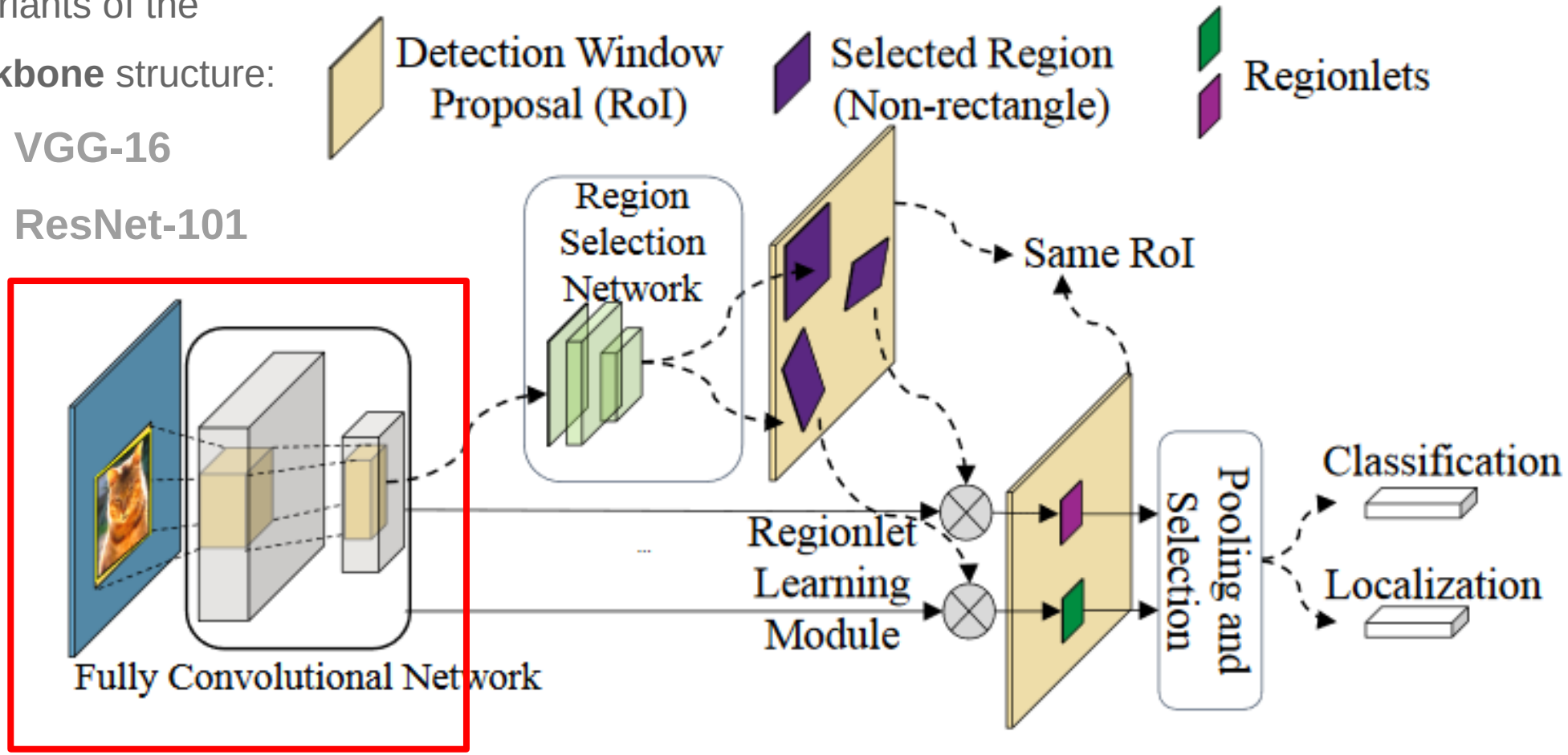

Detection Window Proposal (RoI)
Selected Region (Non-rectangle)
Regionlets
Region Selection Network
Same RoI
Regionlet Learning Module
Pooling and Selection
Classification
Localization
Fully Convolutional Network

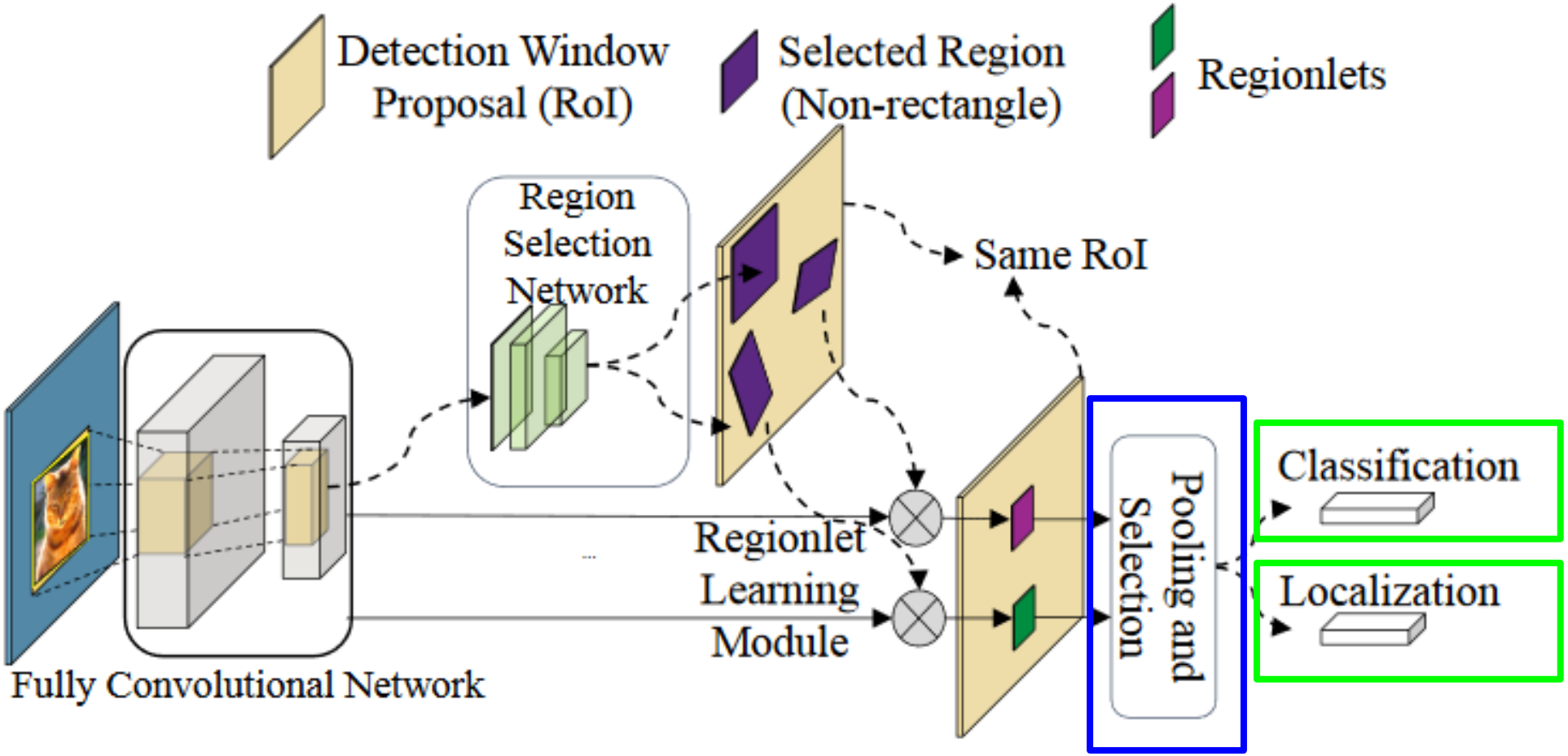
Erratum : $V(x_p^t, y_p^t, c|\Theta, R)$

$$V(x_p^s, y_p^s, c|\Theta, R) = \sum_n^H \sum_m^M U_{nm}^c \max(0, 1 - |x_p^s - m|) \max(0, 1 - |y_p^s - n|) \quad (1)$$

- Z: set of feature maps
- Spatial Location in feature map Z
- Value at location (n, m) in channel c of the input feature
- Total output feature map V

- 2 variants of the **backbone** structure:
  - VGG-16
  - ResNet-101




Detection Window Proposal (RoI)
Selected Region (Non-rectangle)
Regionlets
Region Selection Network
Same RoI
Regionlet Learning Module
Pooling and Selection
Classification
Localization
Fully Convolutional Network

Experiments and Results

- The model is evaluated using 2 datasets:
  - **MS COCO**
  - **PASCAL VOC**
- The model has 2 variants, one using VGG-16 and other using ResNet-101

Results: PASCAL VOC

Methods	training data	mAP@0.5(%)	training data	mAP@0.5(%)
Regionlet [47]	07	41.7	07 + 12	N/A
Faster R-CNN [40]	07	70.0	07 + 12	73.2
R-FCN [8]	07	69.6	07 + 12	76.6
SSD 512 [31]	07	71.6	07 + 12	76.8
Soft-NMS [4]	07	71.1	07 + 12	76.8
Ours	07	***73.0***	07 + 12	***79.2***
Ours[§]	07	**73.8**	07 + 12	**80.1**

Table 3: Detection results on PASCAL VOC using VGG16 as backbone architecture. Training data: "07": VOC2007 `trainval`, "07 + 12": VOC 2007 and 2012 `trainval`. Ours[§] denotes applying the soft-NMS [4] in the test stage.

Methods	mAP@0.5 / @0.7(%)	Methods	mAP@0.5 / @0.7(%)
Faster R-CNN [40]	78.1 / 62.1	SSD [31]	76.8 / N/A
DP-FCN [35]	78.1 / N/A	ION [3]	79.4 / N/A
LocNet [15]	78.4 / N/A	Deformable ConvNet [9]	78.6 / 63.3
Deformable ROI Pooling [9]	78.3 / 66.6	D-F-RCNN [9]	79.3 / 66.9
Ours	***82.0 / 67.0***	Ours[§]	**83.1 / 67.9**

Table 4: Detection results on PASCAL VOC using ResNet-101 [20] as backbone acchitecture. Training data: union set of VOC 2007 and 2012 `trainval`. Ours[§] denotes applying the soft-NMS [4] in the test stage.

Results: MS COCO

Methods	FRCN [40]	YOLO9000 [39]	FRCN OHEM	DSSD [14]	SSD* [31]
mAP@0.5(%)	73.8	73.4	76.3	76.3	78.5
Methods	ION [3]	R-FCN [8]	DP-FCN [35]	Ours	Ours§
mAP@0.5(%)	76.4	77.6	79.5	***80.4***	**81.2**

Table 5: Detection results on VOC2012 `test` set using training data "07++12": 2007 `trainvaltest` and 2012 `trainval`. SSD* denotes the new data augmentation. Ours§ denotes applying the soft-NMS [4] in the test stage.

Methods	Training Data	mmAP 0.5:0.95	mAP @0.5	mAP small	mAP medium	mAP large
Faster R-CNN [40]	trainval	24.4	45.7	7.9	26.6	37.2
SSD*[31]	trainval	31.2	50.4	10.2	34.5	49.8
DSSD [14]	trainval	33.2	53.5	13.0	35.4	51.1
R-FCN [8]	trainval	30.8	52.6	11.8	33.9	44.8
D-F-RCNN [9]	trainval	33.1	50.3	11.6	34.9	51.2
D-R-FCN [9]	trainval	34.5	55.0	14.0	37.7	50.3
Mask R-CNN [18]	trainval	38.2	60.3	20.1	41.1	50.2
RetinaNet500 [29]	trainval	34.4	53.1	14.7	38.5	49.1
Ours	trainval	**39.3**	59.8	**21.7**	**43.7**	50.9

Table 6: Object detection results on MS COCO 2017 `test-dev` using ResNet-101 backbone. Training data: 2017 `train` and `val` set. SSD* denotes the new data augmentation.

Ablation Study

- Study on **RSN**:
  - Global RSN: selects one global region and its initialized by \theta_0 = [1,0,0; 0,1,0]
  - Offset only RSN: force RSN to learn with fixed \theta_1, \theta_2, \theta_4, \theta_5. Only rectangular region are selected.
  - Non-Gating Selection: No gating network, hence each regionlet contributes equally to the final feature rep.

Ablation Study

- Study on **number of regions**:
  - mAP increase from 4(2x2) to 9(3x3) for fixed number of regionlets, but saturates at 16(4x4) selected regions
- Study on **number of regionlets**:
  - For regionlets, vary H & W.
  - Performance improves with increase from 4(2x2) to 25(5x5) with best performance at 16(4x4) or 25(5x5) (more regionlets, more spatial information , better performance)
  - However, performance degrades from 25(5x5) to 36(6x6) since dense regionlets may have redundant information.

Ablation Study

Methods	Global RSN	Offset-only RSN [9, 35]	Non-gating	Ours
mAP@0.5(%)	30.27	78.5	81.3 (+2.8)	82.0 (+3.5)

Table 1: Ablation study of each component in deep regionlet approach. Output size $H \times W$ is set to 4×4 for all the baselines

# of Regions \ Regionlets Density	2×2	3×3	4×4	5×5	6×6
$4(2 \times 2)$ regions	78.0	79.2	79.9	80.2	80.3
$9(3 \times 3)$ regions	79.6	80.3	80.9	81.5	81.3
$16(4 \times 4)$ regions	80.0	81.0	82.0	81.6	80.8

Table 2: Results of ablation studies when a region selection network (RSN) selects different number of regions and regionlets are learned at different level of density.

Conclusion

- The paper proposes a deep regionlet based approach for object detection.
- The approach uses arbitrarily shaped regions within a candidate bounding box, thus aids in building an end-to-end trainable model.