南京航空航天大學

Nanjing University of Aeronautics and Astronautics

Large Selective Kernel Network for Remote Sensing Object Detection

Yuxuan Li, Qibin Hou, Zhaohui Zheng, Ming-Ming Cheng, Jian Yang and Xiang Li*
IMPlus@PCALab & TMCC, CS, Nankai University
yuxuan.li.17@ucl.ac.uk, andrewhoux@gmail.com,
{zh_zheng, cmm, csjyang, xiang.li.implus}@nankai.edu.cn

Background

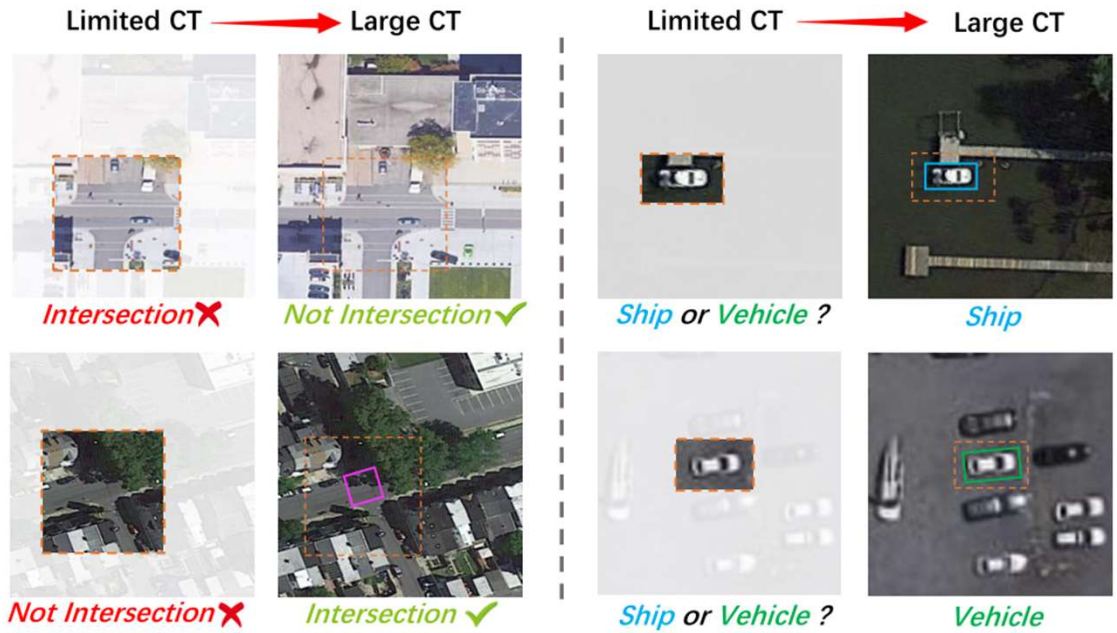


Figure 1: Successfully detecting remote sensing objects requires the use of a wide range of contextual information. Detectors with a limited receptive field may easily lead to incorrect detection results. "CT" stands for Context.

Background

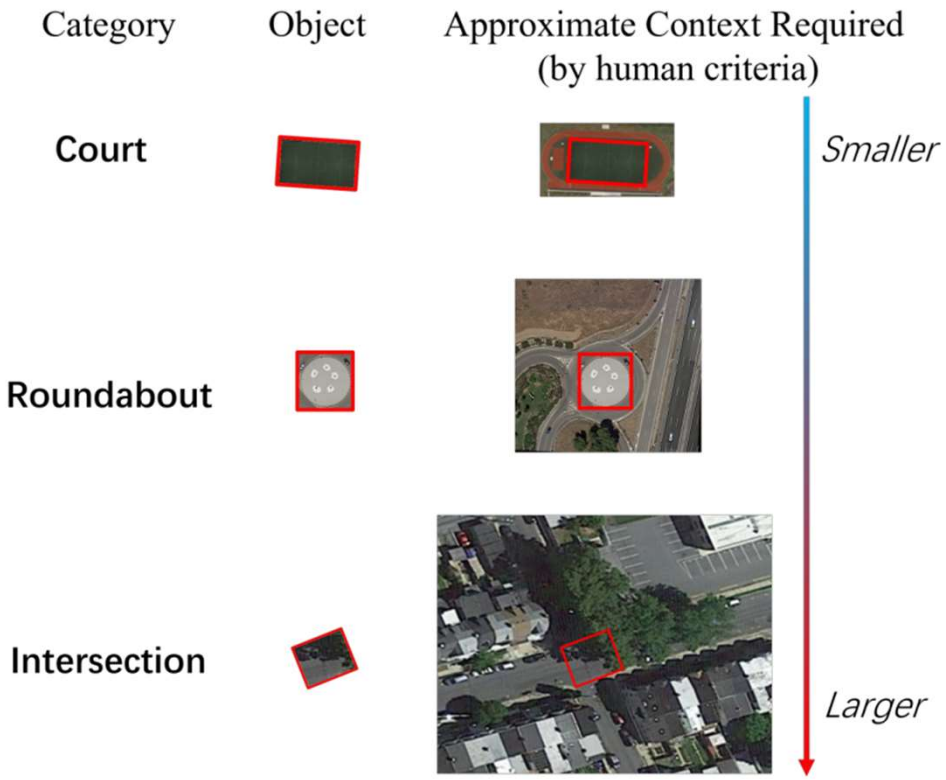


Figure 2: The wide range of contextual information required for different object types is very different by human criteria. The objects with red boxes are the exact ground-truth annotations.

Method

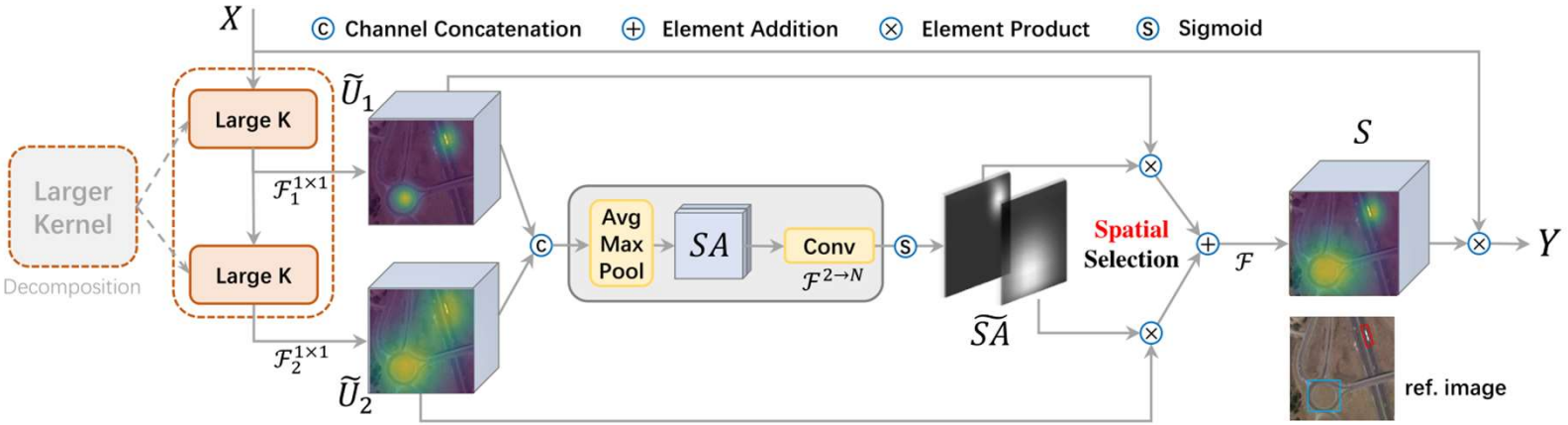


Figure 4: A conceptual illustration of LSK module.

RF	(k, d) sequence	#P	FLOPs
23	(23, 1)	40.4K	42.4G
	(5, 1) ⟶ (7, 3)	11.3K	11.9G
29	(29, 1)	60.4K	63.3G
	(3, 1) ⟶ (5, 2) ⟶ (7, 3)	11.3K	13.6G

Table 2: **Theoretical efficiency comparisons of two representative examples** by expanding single large depth-wise kernel into a sequence, given channels being 64. k: kernel size; d: dilation.

Method

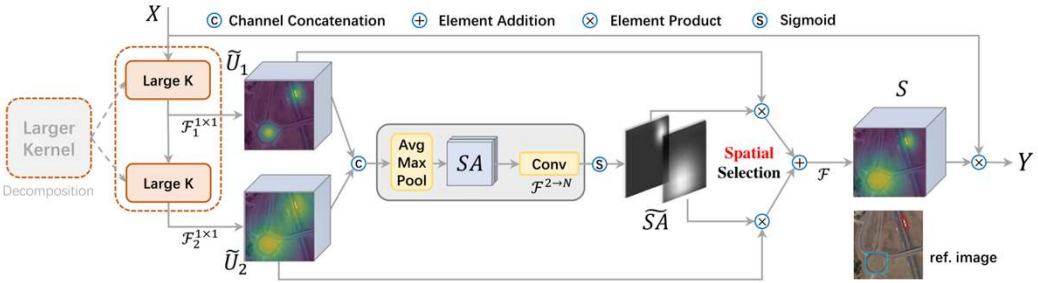


Figure 4: A conceptual illustration of LSK module.

```
classLSKmodule(nn.Module):
def__init__(self, dim):
        super().__init__()
        self.conv0= nn.Conv2d(dim, dim, 5, padding=2, groups=dim)
        self.convl= nn.Conv2d(dim, dim, 7, stride=1, padding=9, groups=dim, dilation=3)
        self.conv0_s= nn.Conv2d(dim, dim//2, 1)
        self.conv1_s= nn.Conv2d(dim, dim//2, 1)
        self.conv_squeeze= nn.Conv2d(2, 2, 7, padding=3)
        self.conv_m= nn.Conv2d(dim//2, dim, 1)

defforward(self, x):
        attn1= self.conv0(x)
        attn2= self.convl(attn1)

        attn1= self.conv0_s(attn1)
        attn2= self.conv1_s(attn2)

        attn= torch.cat([attn1, attn2], dim=1)
        avg_attn= torch.mean(attn, dim=1, keepdim=True)
        max_attn, _= torch.max(attn, dim=1, keepdim=True)
        agg= torch.cat([avg_attn, max_attn], dim=1)
        sig= self.conv_squeeze(agg).sigmoid()
        attn= attn1* sig[:,0,:,:].unsqueeze(1)+ attn2* sig[:,1,:,:].unsqueeze(1)
        attn= self.conv_m (attn)
return x* attn
```

Experiments

(k, d) sequence	RF	Num.	FPS	mAP (%)
(29, 1)	29	1	18.6	80.66
(5, 1) $\longrightarrow$ (7, 4)	29	2	**20.5**	**80.91**
(3, 1) $\longrightarrow$ (5, 2) $\longrightarrow$ (7, 3)	29	3	19.2	80.77

Table 3: **The effects of the number of decomposed large kernels** on the inference FPS and mAP, given theoretical receptive field being 29. We adopt LSKNet-T backbones pretrained on ImageNet for 100 epochs. Decomposing the large kernel into two depth-wise kernels achieves the best performance of speed and accuracy.

(k_1, d_1)	(k_2, d_2)	CS	SS	RF	FPS	mAP (%)
(3, 1)	(5, 2)	-	-	11	22.1	80.80
(5, 1)	(7, 3)	-	-	23	21.7	80.94
(7, 1)	(9, 4)	-	-	39	21.3	80.84
(5, 1)	(7, 3)	✓	-	23	19.6	80.57
(5, 1)	(7, 3)	-	✓	23	20.7	**81.31**

Table 4: **The effectiveness of the key design components** of the LSKNet when the large kernel is decomposed into a sequence of two depth-wise kernels. CS: channel selection (likewise in SKNet [30]); SS: spatial selection **(ours)**. We adopt LSKNet-T backbones pretrained on ImageNet for 100 epochs. The LSKNet achieves best performance when using a reasonably large receptive field with spatial selection.

Experiments

Pooling		FPS	mAP (%)
Max.	Avg.		
✓		20.7	81.23
	✓	20.7	81.12
✓	✓	20.7	**81.31**

Table 5: Ablation study on the effectiveness of the **maximum and average pooling in spatial selection** of our proposed LSK module. We adopt LSKNet-T backbones pretrained on ImageNet for 100 epochs. Best result is obtained when using both.

Framework \ mAP (%)	ResNet-18	⋆ LSKNet-T
Oriented RCNN [62]	79.27	81.31 **(+2.04)**
RoI Transformer [12]	78.32	80.89 **(+2.57)**
S^2A-Net [20]	76.82	80.15 **(+3.33)**
R3Det [68]	74.16	78.39 **(+4.23)**
#P (backbone only)	11.2M	4.3M **(-62%)**
FLOPs (backbone only)	38.1G	19.1G **(-50%)**

Table 6: **Comparison of LSKNet-T and ResNet-18** as backbones with different detection frameworks on DOTA-v1.0. The LSKNet-T backbone is pretrained on ImageNet for 100 epochs. The lightweight LSKNet-T achieves significant higher mAP in various frameworks than ResNet-18.

Experiments

Group	Model (backbone only)	#P	FLOPs	mAP (%)
Baseline	ResNet-18	11.2M	38.1G	79.27
Large Kernel	VAN-B1 [17]	13.4M	52.7G	81.15
	ConvNeXt V2-N [59]	15.0M	51.2G	80.81
	MSCAN-S [18]	13.1M	45.0G	81.12
Selective Attention	SKNet-26 [30]	14.5M	58.5G	80.67
	ResNeSt-14 [77]	8.6M	57.9G	79.51
	SCNet-18 [34]	14.0M	50.7G	79.69
Ours	⋆ LSKNet-S	14.4M	54.4G	**81.48**
Prev Best	CSPNeXt [43]	26.1M	87.6G	81.33

Table 7: **Comparison on LSKNet-S and other (large kernel/selective attention) backbones** under O-RCNN [62] framework on DOTA-v1.0, except that the Prev Best is under RTMDet [43] framework. All backbones are pretrained on ImageNet for 100 epochs. Our LSKNet achieves the best mAP under similar complexity budgets, whilst surpassing the previous best public records [43].

Method	Pre.	mAP (07) ↑	mAP (12) ↑	#P ↓	FLOPs ↓
DRN [46]	IN	-	92.70	-	-
CenterMap [56]	IN	-	92.80	41.1M	198G
Rol Trans. [12]	IN	86.20	-	55.1M	200G
G. V. [64]	IN	88.20	-	41.1M	198G
R3Det [68]	IN	89.26	96.01	41.9M	336G
DAL [44]	IN	89.77	-	36.4M	216G
GWD [70]	IN	89.85	97.37	47.4M	456G
S^2ANet [20]	IN	90.17	95.01	38.6M	198G
AOPG [6]	IN	90.34	96.22	-	-
ReDet [21]	IN	90.46	97.63	31.6M	-
O-RCNN [62]	IN	90.50	97.60	41.1M	199G
RTMDet [43]	CO	90.60	97.10	52.3M	205G
⋆ LSKNet-S **(ours)**	IN	**90.65**	**98.46**	**31.0M**	**161G**

Table 8: Comparison with state-of-the-art methods on the **HRSC2016** dataset. The LSKNet-S backbone is pretrained on ImageNet for 300 epochs, the same with most compared methods [68, 20, 62]. mAP (07/12): VOC 2007 [15]/2012 [16] metrics.

Experiments

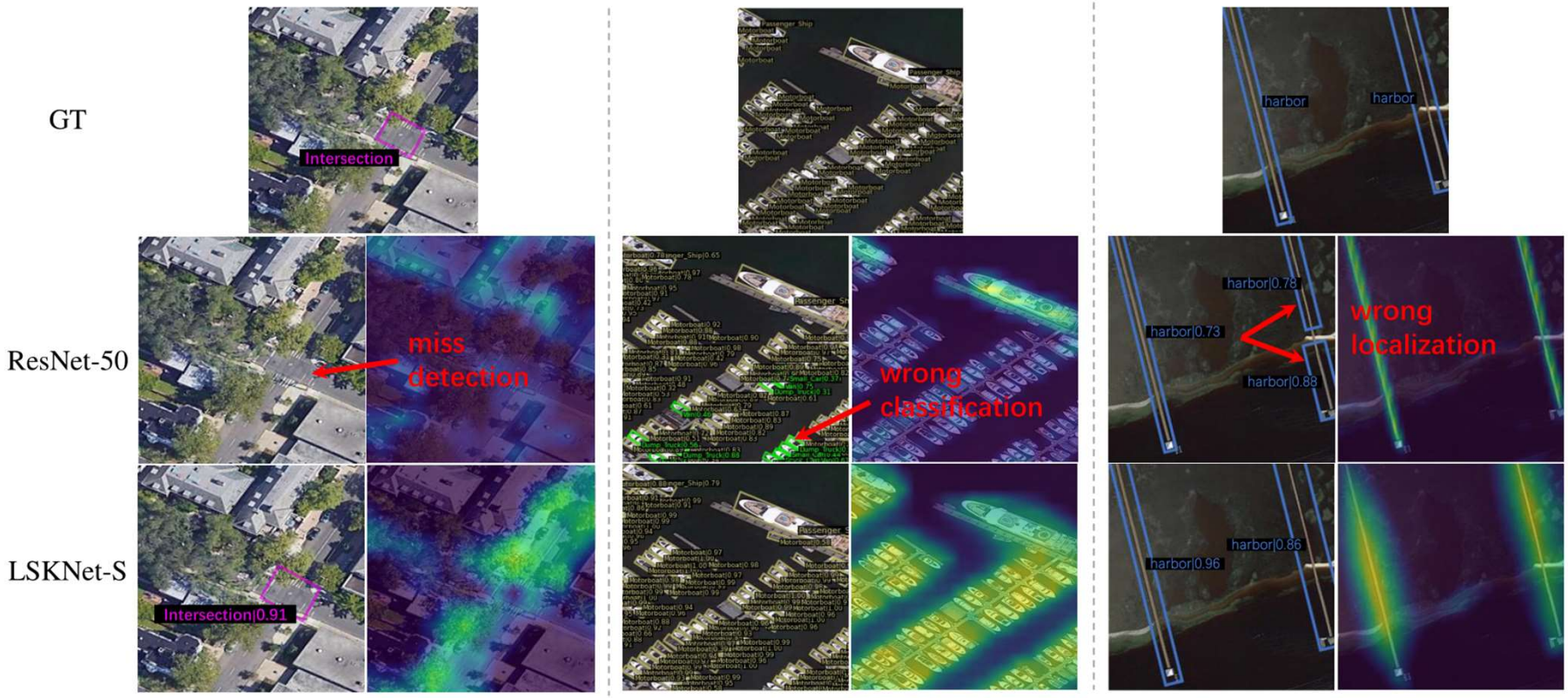
Method	Pre.	mAP ↑	#P ↓	FLOPs ↓	PL	BD	BR	GTF	SV	LV	SH	TC	BC	ST	SBF	RA	HA	SP	HC
One-stage																			
R3Det [68]	IN	76.47	41.9M	336G	89.80	83.77	48.11	66.77	78.76	83.27	87.84	90.82	85.38	85.51	65.57	62.68	67.53	78.56	72.62
CFA [19]	IN	76.67	36.6M	194G	89.08	83.20	54.37	66.87	81.23	80.96	87.17	90.21	84.32	86.09	52.34	69.94	75.52	80.76	67.96
DAFNe [28]	IN	76.95	-	-	89.40	86.27	53.70	60.51	82.04	81.17	88.66	90.37	83.81	87.27	53.93	69.38	75.61	81.26	70.86
SASM [24]	IN	79.17	36.6M	194G	89.54	85.94	57.73	78.41	79.78	84.19	89.25	90.87	58.80	87.27	63.82	67.81	78.67	79.35	69.37
AO2-DETR [9]	IN	79.22	74.3M	304G	89.95	84.52	56.90	74.83	80.86	83.47	88.47	90.87	86.12	**88.55**	63.21	65.09	79.09	**82.88**	73.46
S^2ANet [20]	IN	79.42	38.6M	198G	88.89	83.60	57.74	81.95	79.94	83.19	**89.11**	90.78	84.87	87.81	70.30	68.25	78.30	77.01	69.58
R3Det-GWD [70]	IN	80.23	41.9M	336G	89.66	84.99	59.26	82.19	78.97	84.83	87.70	90.21	86.54	86.85	**73.47**	67.77	76.92	79.22	74.92
RTMDet-R [43]	IN	80.54	52.3M	205G	88.36	84.96	57.33	80.46	80.58	84.88	88.08	**90.90**	86.32	87.57	69.29	70.61	78.63	80.97	79.24
R3Det-KLD [72]	IN	80.63	41.9M	336G	89.92	85.13	59.19	81.33	78.82	84.38	87.50	89.80	87.33	87.00	72.57	71.35	77.12	79.34	78.68
RTMDet-R [43]	CO	81.33	52.3M	205G	88.01	86.17	58.54	82.44	81.30	84.82	88.71	90.89	**88.77**	87.37	71.96	71.18	81.23	81.40	77.13
Two-stage																			
SCRDet [71]	IN	72.61	-	-	89.98	80.65	52.09	68.36	68.36	60.32	72.41	90.85	87.94	86.86	65.02	66.68	66.25	68.24	65.21
RoI Trans. [12]	IN	74.61	55.1M	200G	88.65	82.60	52.53	70.87	77.93	76.67	86.87	90.71	83.83	82.51	53.95	67.61	74.67	68.75	61.03
G.V. [64]	IN	75.02	41.1M	198G	89.64	85.00	52.26	77.34	73.01	73.14	86.82	90.74	79.02	86.81	59.55	70.91	72.94	70.86	57.32
CenterMap [56]	IN	76.03	41.1M	198G	89.83	84.41	54.60	70.25	77.66	78.32	87.19	90.66	84.89	85.27	56.46	69.23	74.13	71.56	66.06
CSL [69]	IN	76.17	37.4M	236G	**90.25**	85.53	54.64	75.31	70.44	73.51	77.62	90.84	86.15	86.69	69.60	68.04	73.83	71.10	68.93
ReDet [21]	IN	80.10	31.6M	-	88.81	82.48	60.83	80.82	78.34	86.06	88.31	90.87	**88.77**	87.03	68.65	66.90	79.26	79.71	74.67
DODet [7]	IN	80.62	-	-	89.96	85.52	58.01	81.22	78.71	85.46	88.59	90.89	87.12	87.80	70.50	71.54	82.06	77.43	74.47
AOPG [6]	IN	80.66	-	-	89.88	85.57	60.90	81.51	78.70	85.29	88.85	90.89	87.60	87.65	71.66	68.69	82.31	77.32	73.10
O-RCNN [62]	IN	80.87	41.1M	199G	89.84	85.43	61.09	79.82	79.71	85.35	88.82	90.88	86.68	87.73	72.21	70.80	82.42	78.18	74.11
KFIoU [73]	IN	80.93	58.8M	206G	89.44	84.41	62.22	82.51	80.10	86.07	88.68	**90.90**	87.32	88.38	72.80	71.95	78.96	74.95	75.27
RVSA [55]	MA	81.24	114.4M	414G	88.97	85.76	61.46	81.27	79.98	85.31	88.30	90.84	85.06	87.50	66.77	**73.11**	**84.75**	81.88	77.58
⋆ LSKNet-T (**ours**)	IN	81.37	**21.0M**	**124G**	89.14	84.90	61.78	83.50	81.54	85.87	88.64	90.89	88.02	87.31	71.55	70.74	78.66	79.81	78.16
⋆ LSKNet-S (**ours**)	IN	81.64	31.0M	161G	89.57	**86.34**	**63.13**	**83.67**	**82.20**	**86.10**	88.66	90.89	88.41	87.42	71.72	69.58	78.88	81.77	76.52
⋆ LSKNet-S* (**ours**)	IN	**81.85**	31.0M	161G	89.69	85.70	61.47	83.23	81.37	86.05	88.64	90.88	88.49	87.40	71.67	71.35	79.19	81.77	**80.86**

Table 9: Comparison with state-of-the-art methods on the **DOTA-v1.0** dataset with multi-scale training and testing. The LSKNet backbones are pretrained on ImageNet for 300 epochs, similarly to the compared methods [68, 20, 62]. *: With EMA finetune similarly to the compared methods [43].

Model	G. V.* [64]	RetinaNet* [32]	C-RCNN* [2]	F-RCNN* [52]	RoI Trans.* [12]	O-RCNN [62]	⋆ LSKNet-T	⋆ LSKNet-S
mAP(%)	29.92	30.67	31.18	32.12	35.29	45.60	46.93	**47.87**

Table 10: Comparison with state-of-the-art methods on the **FAIR1M-v1.0** dataset. The LSKNet backbones are pretrained on ImageNet for 300 epochs, similarly to [68, 20, 62]. *: Results are referenced from FAIR1M paper [53].

Experiments



(a) Robustness to obstacles (b) Accurate Object Classification (c) Accurate Object Localization

Figure 5: **Eigen-CAM visualization** of Oriented RCNN detection framework with ResNet-50 and LSKNet-S. Our proposed LSKNet can model a much long range of context information, leading to better performance in various hard cases.

The end

Thanks for listening!!!