Lightweight Object Detection: A Brief Report

King-Siong Si, Lu Sun

Institute of Multimedia Knowledge Fusion and Engineering,
Xi'an JiaoTong University

sjsinx@stu.xjtu.edu.cn, *konautomaton@gmail.com*

January 22, 2024

1. Content and Achievements
   - NAS
   - Channel Pruning
   - SCConv
   - PConv
   - Re-Parameterization
   - NMS

2. Design and Results

3. References

NAS

- NAS(Neural Architecture Search)

NAS

- NAS(Neural Architecture Search)
- MAE-NAS[5]

NAS

- NAS(Neural Architecture Search)
- MAE-NAS[5]
- The topology of a network F can be abstracted as a graph $G(V, E)$. V represents the features, and E represents the operators.

NAS

- NAS(Neural Architecture Search)
- MAE-NAS[5]
- The topology of a network F can be abstracted as a graph $G(V, E)$. V represents the features, and E represents the operators.
- $S = \{h(v), h(e) : \forall e \in E, \forall v \in V\}$, here $h(\cdot)$ is the value function. S defines the continuous state space of F.

NAS

- NAS(Neural Architecture Search)
- MAE-NAS[5]
- The topology of a network F can be abstracted as a graph $G(V, E)$. V represents the features, and E represents the operators.
- $S = \{h(v), h(e) : \forall e \in E, \forall v \in V\}$, here $h(\cdot)$ is the value function. S defines the continuous state space of F.
- Zero-Cost NAS, proxy is defined as the entropy of S.

1 Content and Achievements

- NAS
- Channel Pruning
- SCConv
- PConv
- Re-Parameterization
- NMS

2 Design and Results

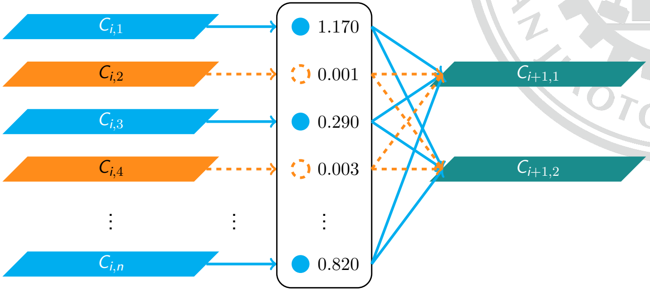
3 References

Channel Pruning

- Channel(Filter) Pruning[4]

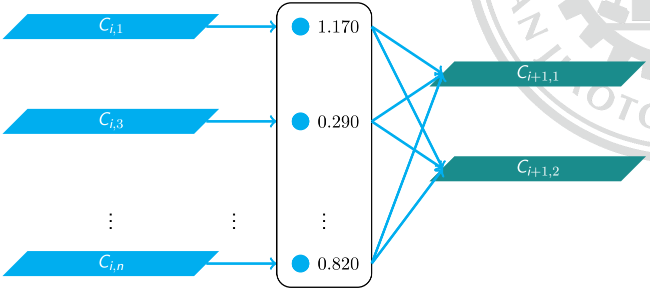
Channel Pruning

- Channel(Filter) Pruning[4]



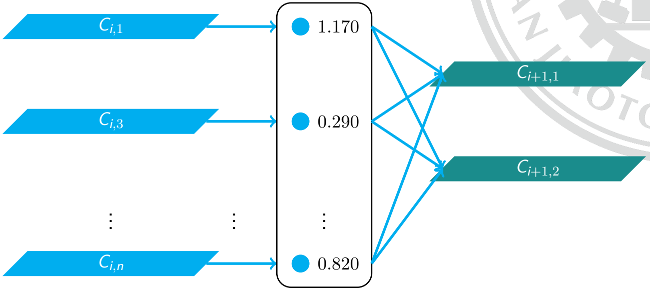
Channel Pruning

- Channel(Filter) Pruning[4]



Channel Pruning

- Channel(Filter) Pruning[4]



- mAP $= 10.3\%$ after sparsifying

1. Content and Achievements
   - NAS
   - Channel Pruning
   - SCConv
   - PConv
   - Re-Parameterization
   - NMS
2. Design and Results
3. References

SCConv

- *SCConv: Spatial and Channel Reconstruction Convolution for Feature Redundancy*[3]

SCConv

- *SCConv: Spatial and Channel Reconstruction Convolution for Feature Redundancy*[3]
- KingSiong's repository: SCConv-Implementation

SCConv

- *SCConv: Spatial and Channel Reconstruction Convolution for Feature Redundancy*[3]
- KingSiong's repository: SCConv-Implementation

parameter	**value**
dataset	CIFAR-10
epoch	200

SCConv

- *SCConv: Spatial and Channel Reconstruction Convolution for Feature Redundancy*[3]
- KingSiong's repository: SCConv-Implementation

parameter	**value**
dataset	CIFAR-10
epoch	200

Model	**Acc(%)**	**FLOPs(G)**	**Params(M)**
ResNet-50	94.93	2.62	23.52
ResNet-50 with SCConv	92.03	1.86	15.91

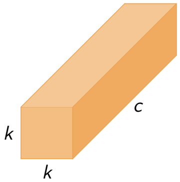
1. Content and Achievements
   - NAS
   - Channel Pruning
   - SCConv
   - PConv
   - Re-Parameterization
   - NMS
2. Design and Results
3. References

PConv

- *Run, Don't Walk: Chasing Higher FLOPS for Faster Neural Networks*[1]

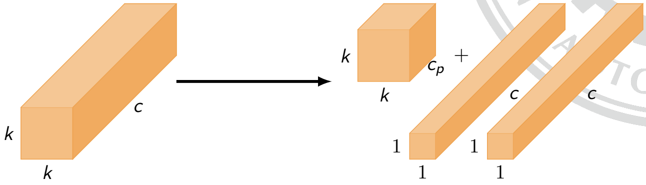
PConv

- *Run, Don't Walk: Chasing Higher FLOPS for Faster Neural Networks*[1]



PConv

- *Run, Don't Walk: Chasing Higher FLOPS for Faster Neural Networks*[1]



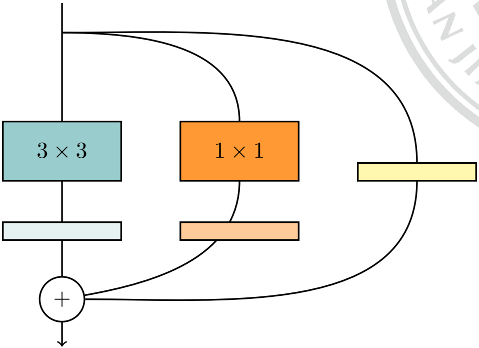
1. Content and Achievements
   - NAS
   - Channel Pruning
   - SCConv
   - PConv
   - Re-Parameterization
   - NMS
2. Design and Results
3. References

Re-Parameterization

- *RepVGG: Making VGG-style ConvNets Great Again*[2]

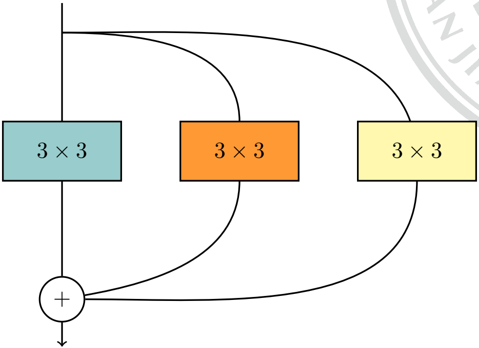
Re-Parameterization

- *RepVGG: Making VGG-style ConvNets Great Again*[2]



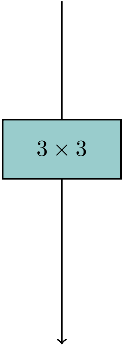
Re-Parameterization

- *RepVGG: Making VGG-style ConvNets Great Again*[2]



Re-Parameterization

- *RepVGG: Making VGG-style ConvNets Great Again*[2]



1. Content and Achievements
   - NAS
   - Channel Pruning
   - SCConv
   - PConv
   - Re-Parameterization
   - NMS
2. Design and Results
3. References

NMS

- NMS(Non Maximum Suppression) for Post-Processing

NMS

- NMS(Non Maximum Suppression) for Post-Processing
- Greedy-NMS(Original-NMS), taking $\mathcal{O}(N^2)$ time

NMS

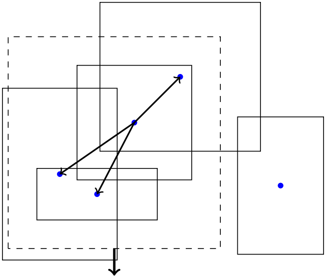
- NMS(Non Maximum Suppression) for Post-Processing
- Greedy-NMS(Original-NMS), taking $\mathcal{O}(N^2)$ time
- Dynamic density query and fast suppression detection using KNN structure

NMS

- NMS(Non Maximum Suppression) for Post-Processing
- Greedy-NMS(Original-NMS), taking $\mathcal{O}(N^2)$ time
- Dynamic density query and fast suppression detection using KNN structure
- Faster-NMS, taking $\mathcal{O}(NK \log N)$ time

NMS

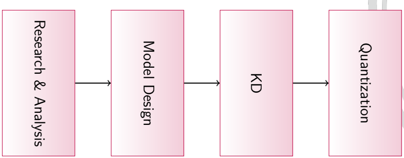
- NMS(Non Maximum Suppression) for Post-Processing
- Greedy-NMS(Original-NMS), taking $\mathcal{O}(N^2)$ time
- Dynamic density query and fast suppression detection using KNN structure
- Faster-NMS, taking $\mathcal{O}(NK\log N)$ time



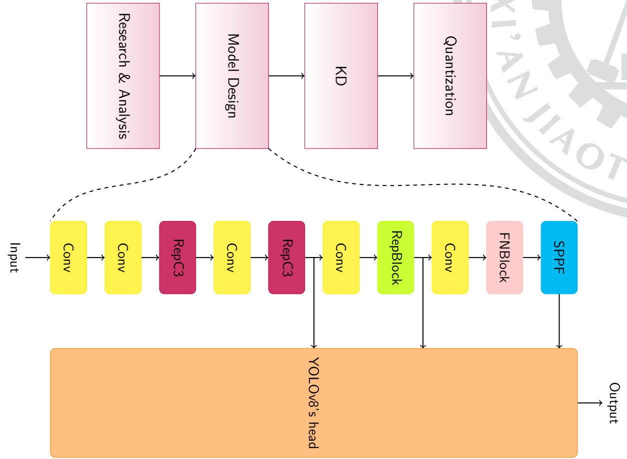
$K=3$, IoU/Density Query Box

1. Content and Achievements
2. Design and Results
3. References

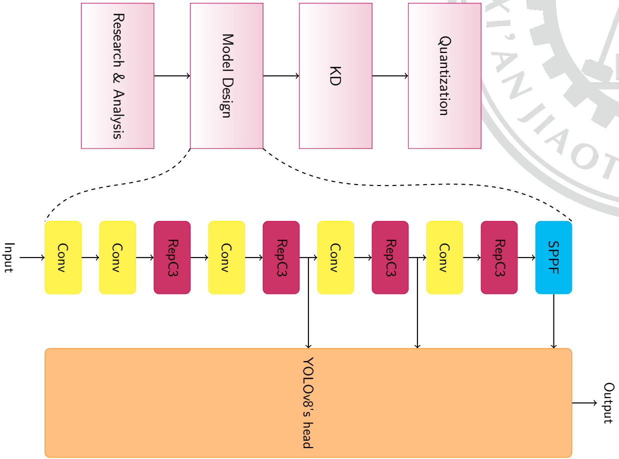
Design



Design


Research & Analysis
Model Design
KD
Quantization
Input
Conv
Conv
RepC3
Conv
RepC3
Conv
RepBlock
Conv
FNBlock
SPPF
YOLOv8's head
Output

Design


Research & Analysis
Model Design
KD
Quantization
Input
Conv
Conv
RepC3
Conv
RepC3
Conv
RepC3
Conv
RepC3
SPPF
YOLOv8's head
Output

Results

- Results,

Results

- Results,

	Model	$\mathbf{mAP}^{\mathbf{val}}_{\mathbf{50-95}}$	**latency(ms)**	**FPS(/s)**
env_1 [1]	yolov5s.pt	37.4	2.78	359.71
	yolov5su.pt	43.1	1.70	588.24($\uparrow 63.53\%$)
	yolov8n.pt	37.4	1.43	699.30($\uparrow 94.41\%$)
	yolov6n.pt	37.5	3.05	327.87($\downarrow 8.85\%$)
env_2 [2]	yolov8n.pt	37.4	2.18	458.72
	xjtun.pt(Ours)	36.7	1.91	523.56

[1] tested when server was idle.

[2] tested when server was not idle.

Results

- Results,

	Model	$\mathbf{mAP}^{\mathbf{val}}_{\mathbf{50-95}}$	latency(ms)	FPS(/s)
env_1[1]	yolov5s.pt	37.4	2.78	359.71
	yolov5su.pt	43.1	1.70	588.24($\uparrow$ 63.53%)
	yolov8n.pt	37.4	1.43	699.30($\uparrow$ 94.41%)
	yolov6n.pt	37.5	3.05	327.87($\downarrow$ 8.85%)
env_2[2]	yolov8n.pt	37.4	2.18	458.72
	xjtun.pt(Ours)	36.7	1.91	523.56

- Conclusion: The FPS improvement is at least 95% with 0.7% accuracy decrease.

[1] tested when server was idle.

[2] tested when server was not idle.

[1] Jierun Chen, Shiu-hong Kao, Hao He, Weipeng Zhuo, Song Wen, Chul-Ho Lee, and S-H Gary Chan.
Run, don't walk: Chasing higher flops for faster neural networks.
In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12021–12031, 2023.

[2] Xiaohan Ding, Xiangyu Zhang, Ningning Ma, Jungong Han, Guiguang Ding, and Jian Sun.
Repvgg: Making vgg-style convnets great again.
In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 13733–13742, 2021.

[3] Jiafeng Li, Ying Wen, and Lianghua He.
Scconv: Spatial and channel reconstruction convolution for feature redundancy.
In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6153–6162, 2023.

[4] Zhuang Liu, Jianguo Li, Zhiqiang Shen, Gao Huang, Shoumeng Yan, and Changshui Zhang.
Learning efficient convolutional networks through network slimming.
In *Proceedings of the IEEE international conference on computer vision*, pages 2736–2744, 2017.

[5] Zhenhong Sun, Ming Lin, Xiuyu Sun, Zhiyu Tan, Hao Li, and Rong Jin.
Mae-det: Revisiting maximum entropy principle in zero-shot nas for efficient object detection.
arXiv preprint arXiv:2111.13336, 2021.

Thanks!