Mask-Net

A Hardware-efficient Object Detection Network with Masked Region Proposals

Hanqiu Chen[1*], Cong (Callie) Hao[2]

1. Nanjing University
2. Georgia Institute of Technology

* Work done during internship at Georgia Tech

Georgia Tech

Overview

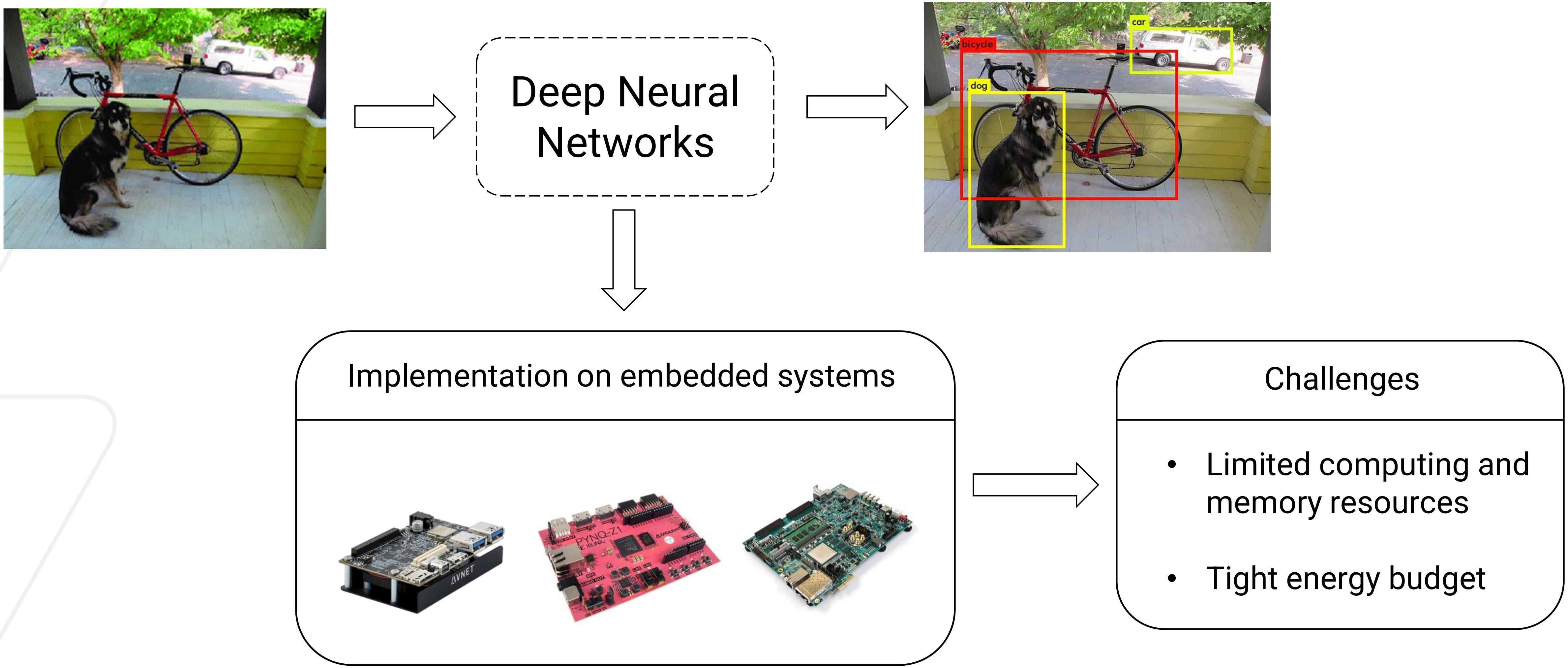
- Background & Motivation
- Related work
  - Region proposal
  - Cascade network
- Mask-Net
  - The architecture
  - Promising features
  - Algorithm and hardware innovations
- Experiment results
- Design space exploration
- Future research directions

Overview

- Background & Motivation
- Related work
  - Region proposal
  - Cascade network
- Mask-Net
  - The architecture
  - Promising features
  - Algorithm and hardware innovations
- Experiment results
- Design space exploration
- Future research directions

Background & Motivation

Challenges for object detection on embedded systems with DNNs



Deep Neural Networks

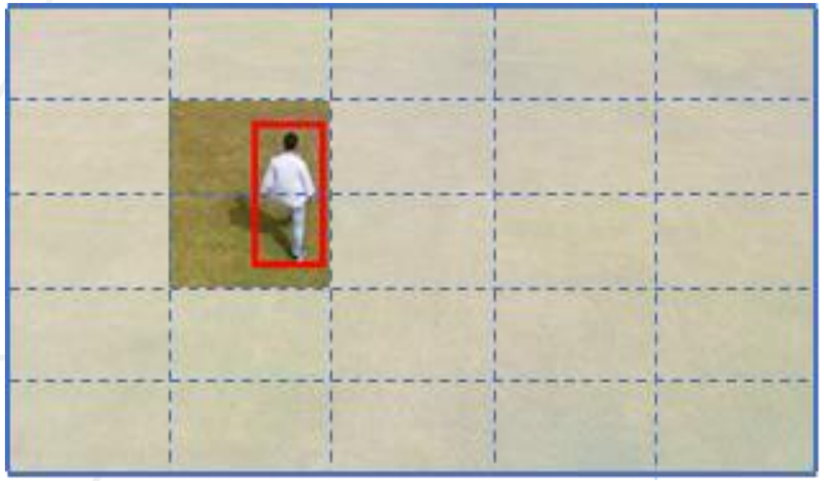
Implementation on embedded systems

Challenges

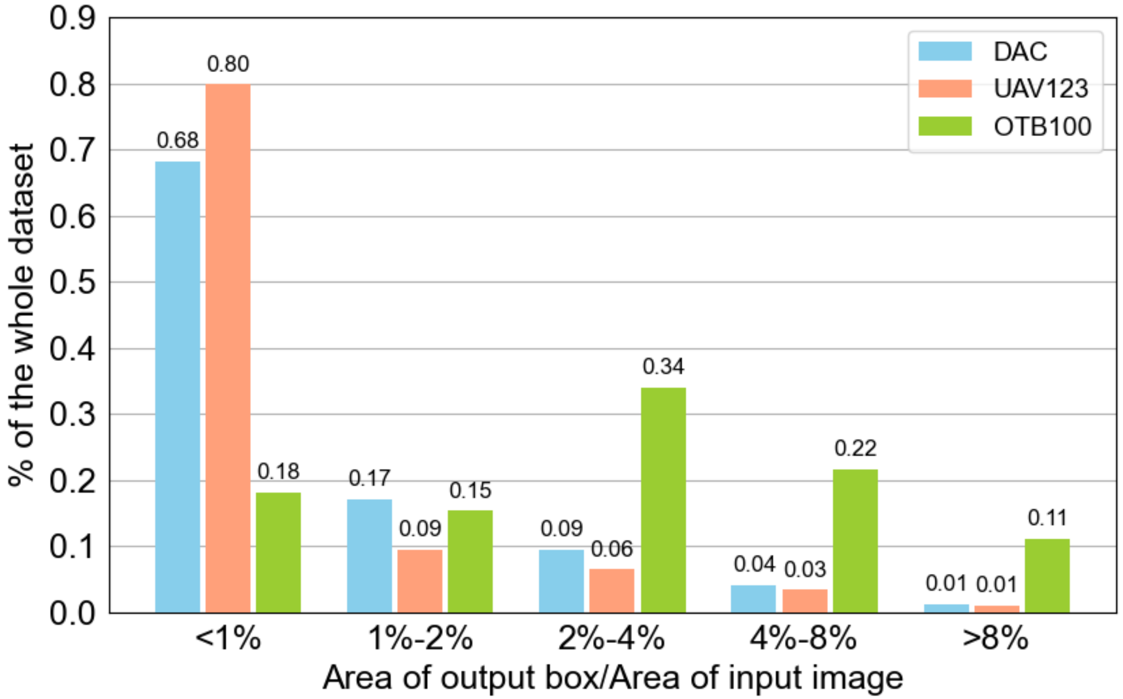
- Limited computing and memory resources
- Tight energy budget

Background & Motivation

Redundant computation: a large part of an image is background and it is unnecessary to focus on these regions.



Sample image from DAC-SDC [1] dataset



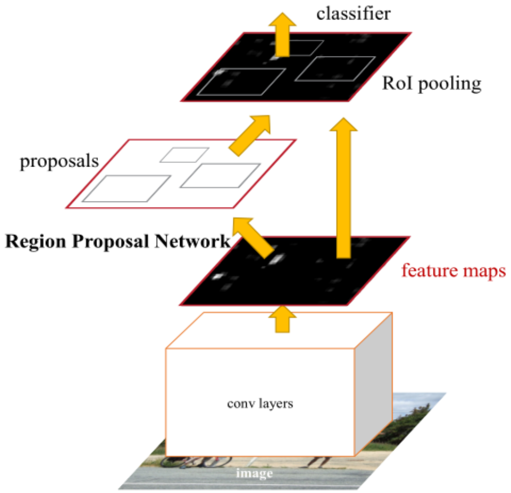
The distribution of bounding box relative size in three different datasets

[1] Xiaowei Xu, Xinyi Zhang, Bei Yu, X Sharon Hu, Christopher Rowen, Jingtong Hu, and Yiyu Shi. Dac-sdc low power object detection challenge for uav applications. IEEE transactions on pattern analysis and machine intelligence, 2019.

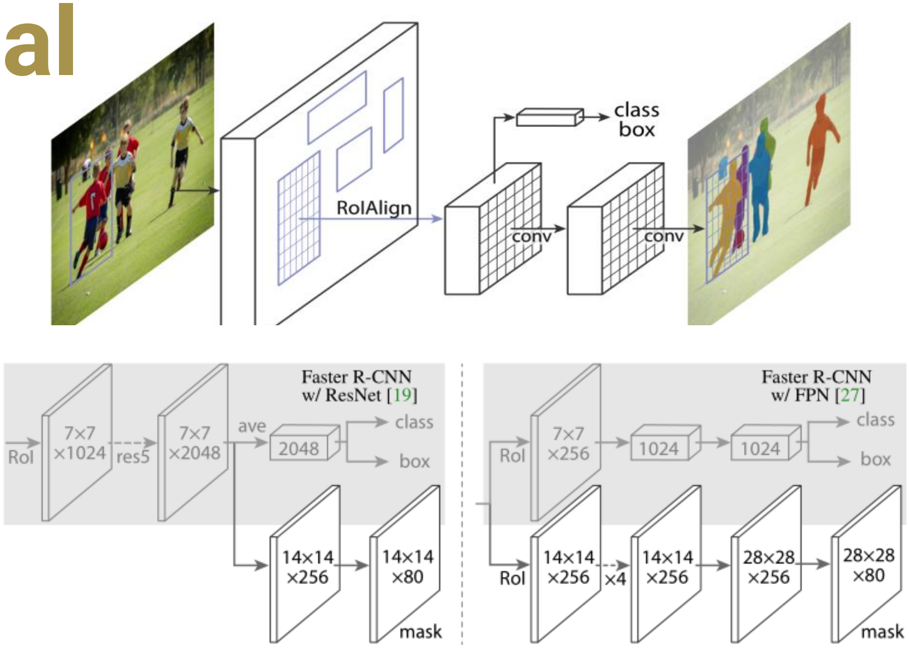
Overview

- Background & Motivation
- Related work
  - Region proposal
  - Cascade network
- Mask-Net
  - The architecture
  - Promising features
  - Algorithm and hardware innovations
- Experiment results
- Design space exploration
- Future research directions

Related Work: Region Proposal



Faster R-CNN[1] is a single, unified network for object detection. The RPN module serves as the 'attention' of this unified network.



The Mask-RCNN[2] framework for instance segmentation and its extension

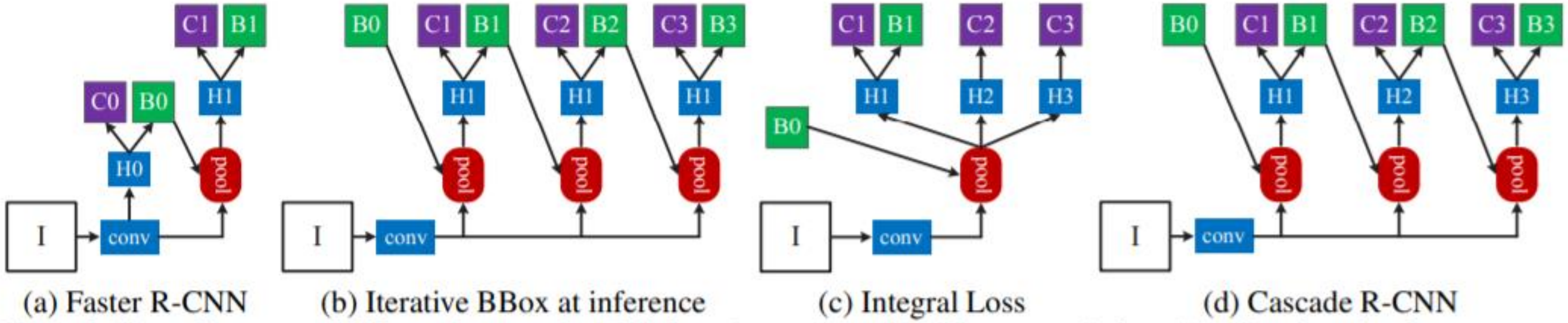
Faster-RCNN	Mask-RCNN
Computationally expensive: needs deep convolution layers to extract enough features	**No rectangular regions:** not beneficial for hardware acceleration

[1] Ren, Shaoqing, et al. "Faster r-cnn: Towards real-time object detection with region proposal networks." Advances in neural information processing systems 28 (2015): 91-99.
[2] He, Kaiming, et al. "Mask r-cnn." Proceedings of the IEEE international conference on computer vision. 2017.

Related Work : Cascade

- A famous example: Cascade R-CNN

Pros	Cons
➢ Higher quality detectors are only required to operate on higher quality hypotheses.	➢ **Non hardware efficient:** has many branches. ➢ **Unsuitable for edge devices:** fits better in dense DNNs instead of lightweight DNNs.



(a) Faster R-CNN (b) Iterative BBox at inference (c) Integral Loss (d) Cascade R-CNN

Figure 3. The architectures of different frameworks. "I" is input image, "conv" backbone convolutions, "pool" region-wise feature extraction, "H" network head, "B" bounding box, and "C" classification. "B0" is proposals in all architectures.

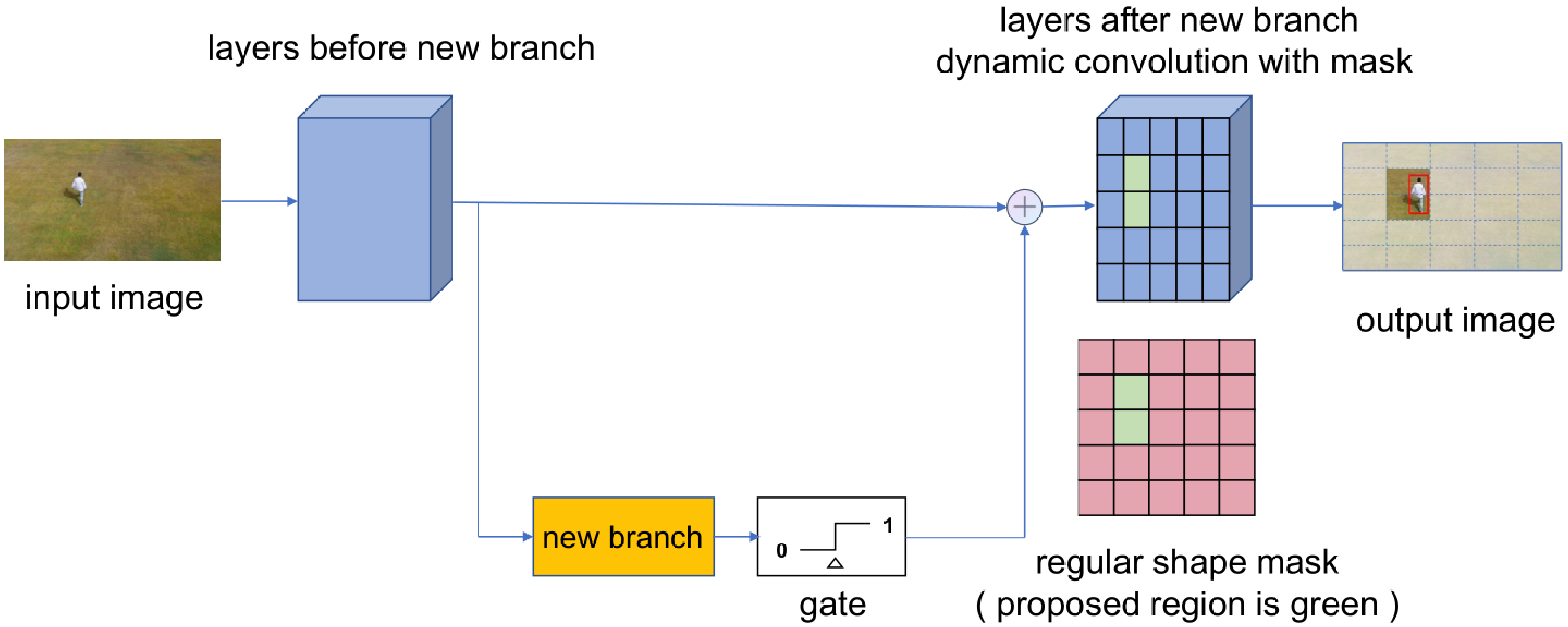
Four common cascade network in object detection

Image credit: Cai, Zhaowei, and Nuno Vasconcelos. "Cascade r-cnn: Delving into high quality object detection." *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2018.

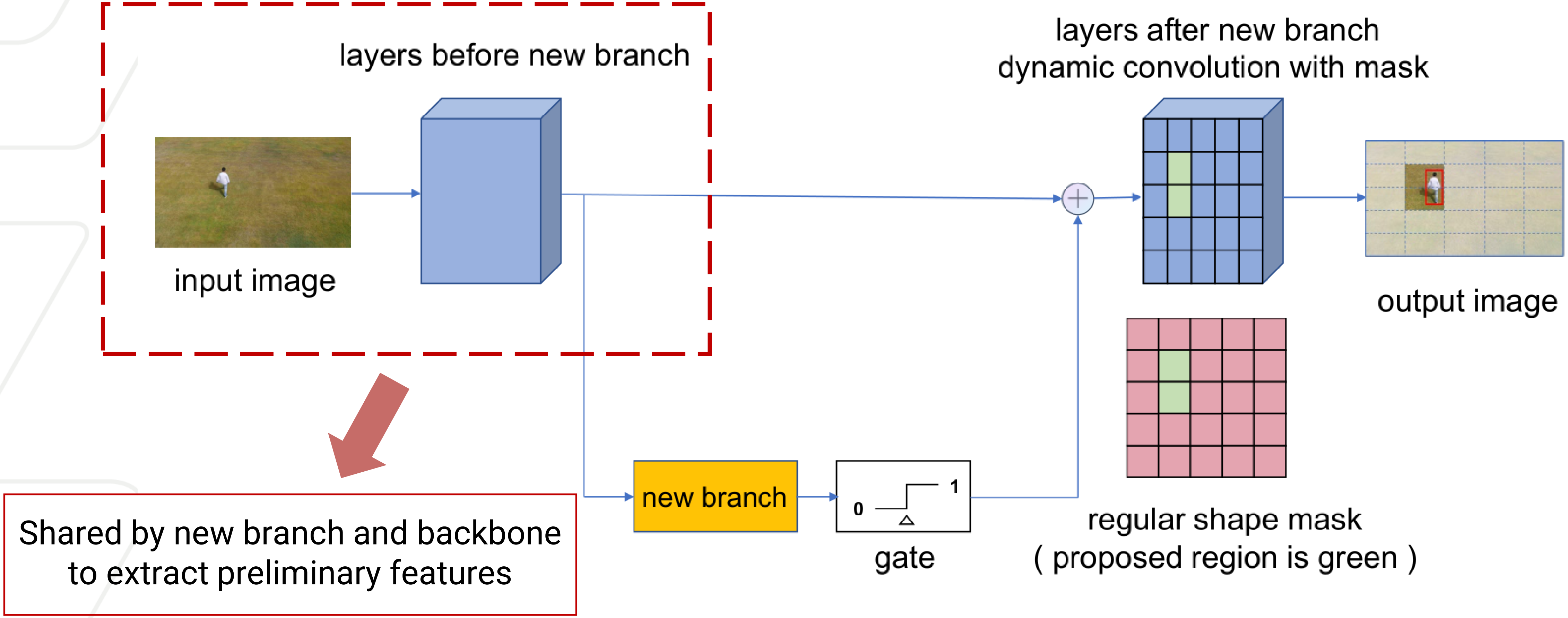
Overview

- Background & Motivation
- Related work
  - Region proposal
  - Cascade network
- Mask-Net
  - The architecture
  - Promising features
  - Algorithm and hardware innovations
- Experiment results
- Design space exploration
- Future research directions

The architecture of Mask-Net



The architecture of Mask-Net



layers before new branch

input image

layers after new branch
dynamic convolution with mask

output image

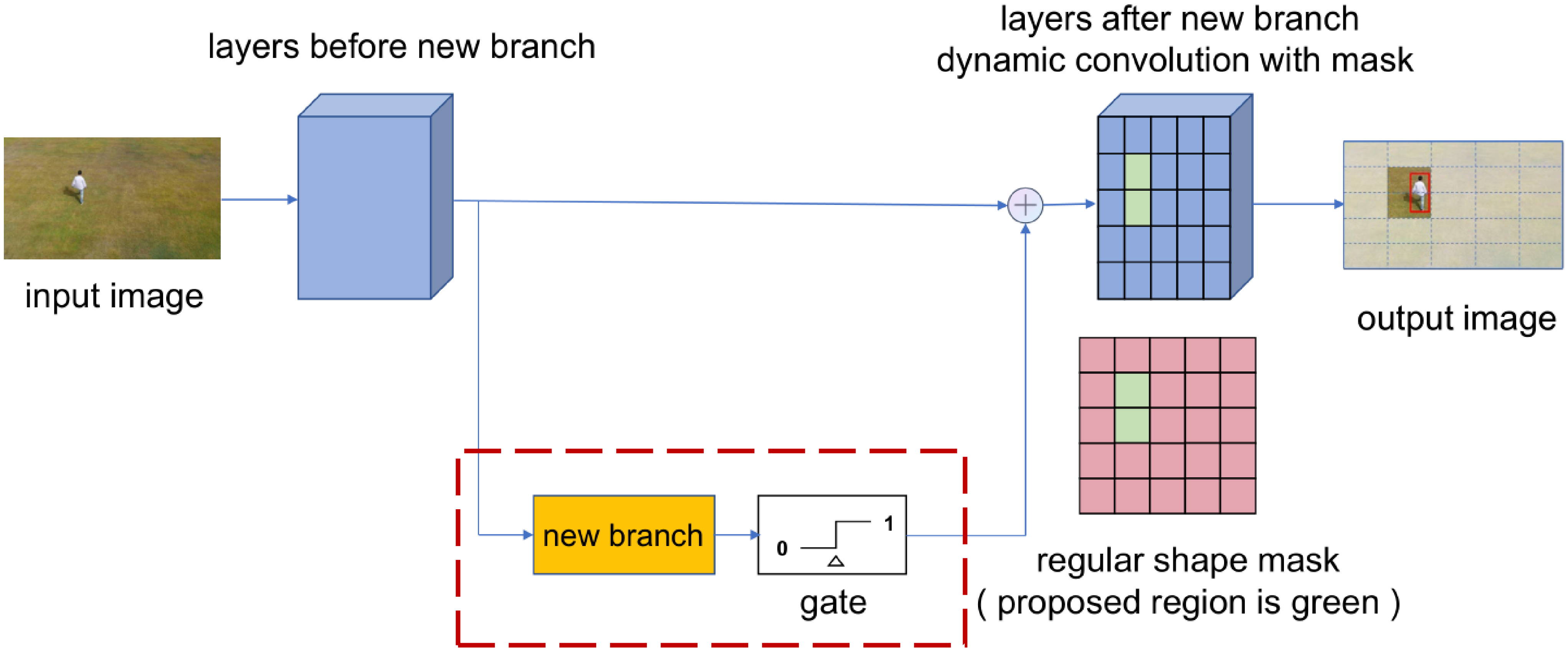
new branch

gate

regular shape mask
(proposed region is green)

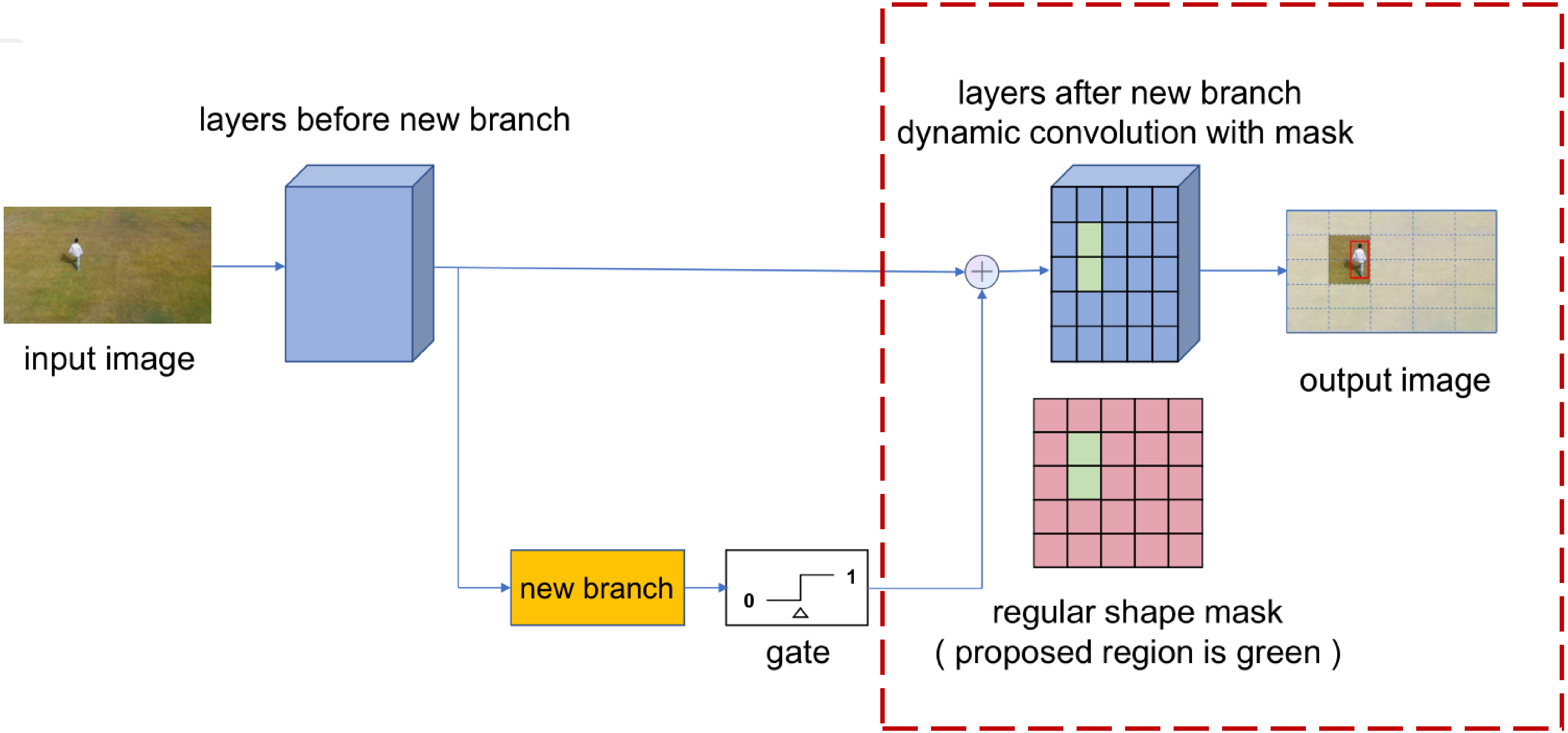
Shared by new branch and backbone
to extract preliminary features

The architecture of Mask-Net



Generate a mask with proposed regions

The architecture of Mask-Net

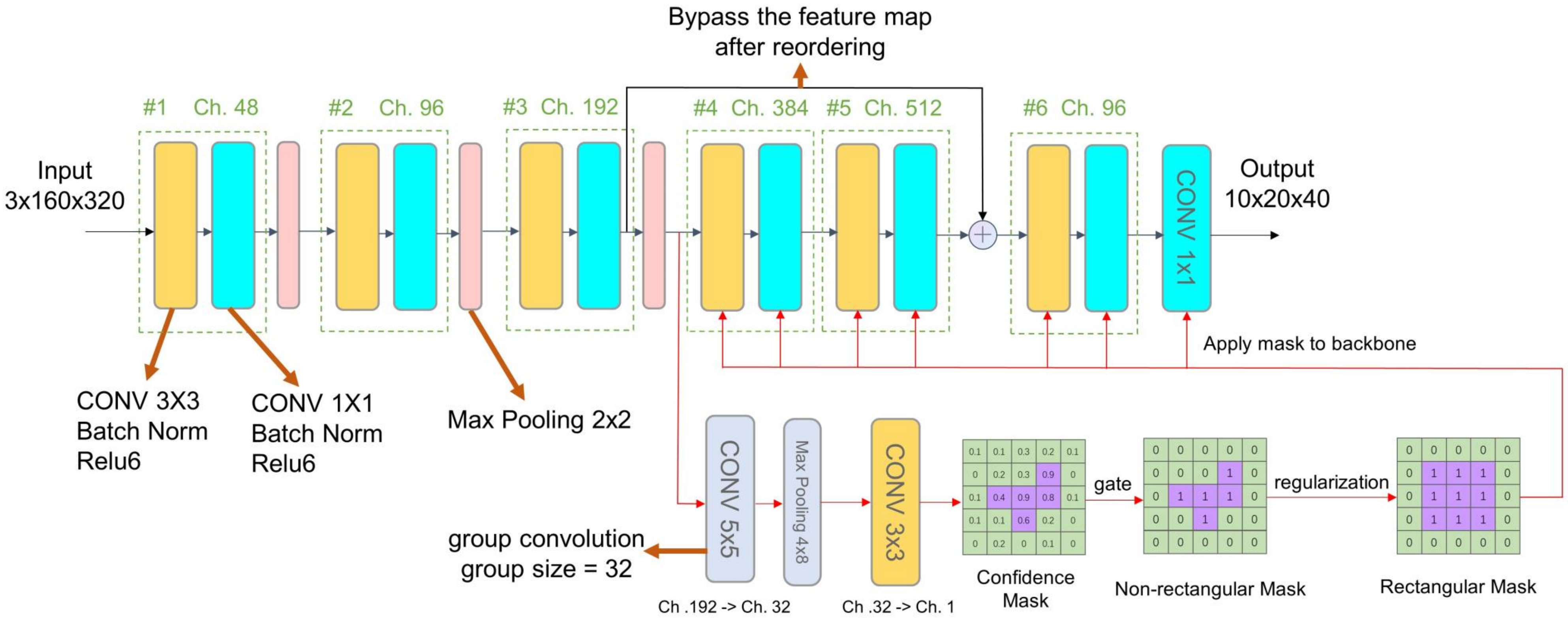


Only compute the proposed regions to generate bounding box

A case study: Mask-SkyNet

SkyNet[1] is a hardware-efficient object detection and tracking backbone.

We choose SkyNet as base model to design a FPGA accelerator.



Confidence Mask

0.1	0.1	0.3	0.2	0.1
0	0.2	0.3	0.9	0
0.1	0.4	0.9	0.8	0.1
0.1	0.1	0.6	0.2	0
0	0.2	0	0.1	0

Non-rectangular Mask

0	0	0	0	0
0	0	0	1	0
0	1	1	1	0
0	0	1	0	0
0	0	0	0	0

Rectangular Mask

0	0	0	0	0
0	1	1	1	0
0	1	1	1	0
0	1	1	1	0
0	0	0	0	0

Mask-SkyNet architecture

[1] Xiaofan Zhang, Haoming Lu, Cong Hao, Jiachen Li, Bowen Cheng, Yuhong Li, Kyle Rupnow, Jinjun Xiong, Thomas Huang, Honghui Shi, et al. Skynet: a hardware-efficient method for object detection and tracking on embedded systems. Proceedings of Machine Learning and Systems, 2:216–229, 2020.

Promising features of Mask-Net

- Small Overhead
  - The mask generation branch’s computation cost is about 6% of the whole network.
- Hardware friendly
  - The mask generation branch can reuse convolution modules in the backbone.
  - Mask shape regularization can help avoid complex control logic.
  - Channel shuffle can help reduce data movement between DRAM and on-chip memory.
- Generalizable
  - Can be applied to different object detection or tracking backbones, including SkyNet, UltraNet and ResNet-18.
  - Works well in different scenarios, including DAC-SDC, UAV123 and OTB100 dataset.

Promising features of Mask-Net

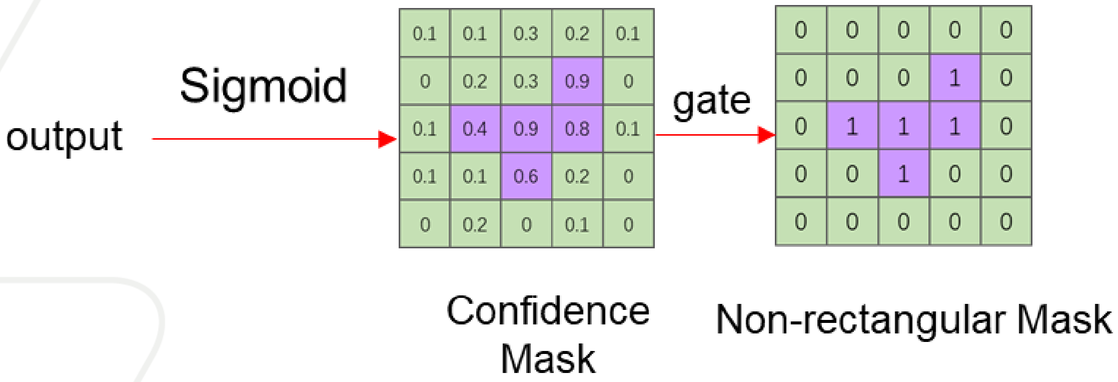
- Small Overhead
  - The mask generation branch's computation cost is about 6% of the whole network.
- Hardware friendly
  - The mask generation branch can reuse convolution modules in the backbone.
  - Mask shape regularization can help avoid complex control logic.
  - Channel shuffle can help reduce data movement between DRAM and on-chip memory.
- Generalizable
  - Can be applied to different object detection or tracking backbones, including SkyNet, UltraNet and ResNet-18.
  - Works well in different scenarios, including DAC-SDC, UAV123 and OTB100 dataset.

Promising features of Mask-Net

- Small Overhead
  - The mask generation branch’s computation cost is about 6% of the whole network.
- Hardware friendly
  - The mask generation branch can reuse convolution modules in the backbone.
  - Mask shape regularization can help avoid complex control logic.
  - Channel shuffle can help reduce data movement between DRAM and on-chip memory.
- Generalizable
  - Can be applied to different object detection or tracking backbones, including SkyNet, UltraNet and ResNet-18.
  - Works well in different scenarios, including DAC-SDC, UAV123 and OTB100 dataset.

Algorithm Innovations

- Confidence mask and regions of interest generation
  - The confidence mask is generated by Sigmoid.
  - The patch’s score in the mask is proportional to the probability of objects’ existence.



0.1	0.1	0.3	0.2	0.1
0	0.2	0.3	0.9	0
0.1	0.4	0.9	0.8	0.1
0.1	0.1	0.6	0.2	0
0	0.2	0	0.1	0

0	0	0	0	0
0	0	0	1	0
0	1	1	1	0
0	0	1	0	0
0	0	0	0	0

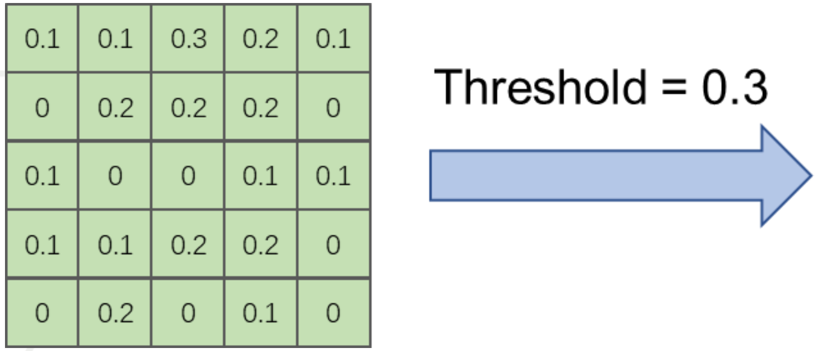
Mask generation process

$$\theta(x)_{ij} = \begin{cases} 1, & x_{ij} > threshold \\ 0, & x_{ij} \leq threshold \end{cases}$$

The gate function

Algorithm Innovations

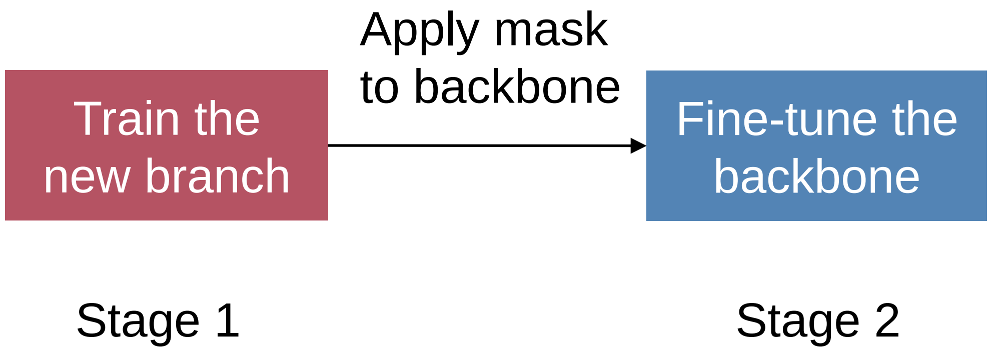
- All pass mechanism
  - If the score of all patches does not exceed the threshold, then we need to calculate the whole image to improve the robustness of mask generation.
- Two-stage training process
  - The first step is to fix the weights in the backbone and only train the new branch.
  - The second step is to fix the weights of the new branch and fine-tune the backbone.



0.1	0.1	0.3	0.2	0.1
0	0.2	0.2	0.2	0
0.1	0	0	0.1	0.1
0.1	0.1	0.2	0.2	0
0	0.2	0	0.1	0

1	1	1	1	1
1	1	1	1	1
1	1	1	1	1
1	1	1	1	1
1	1	1	1	1

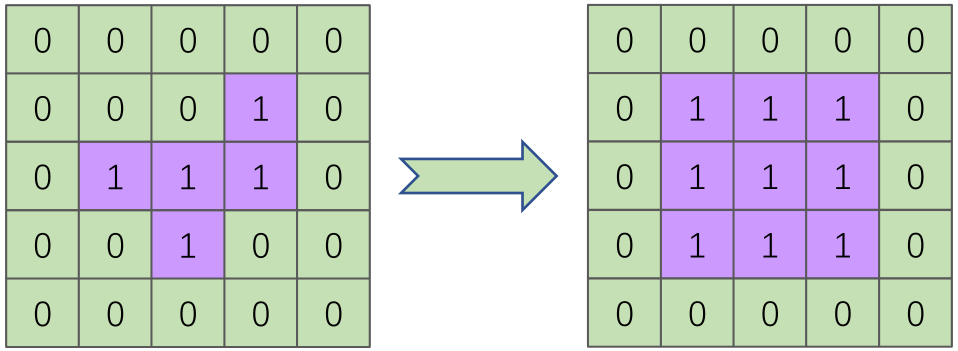
All pass mechanism



Two stage training process

Hardware Innovations

- Region of Interest Shape Regularization
  - Our FPGA accelerator is tile-based. If the shape of regions of interest is not rectangular, additional judgement is needed before loading and calculating the tile.
  - Regularize the shape of regions of interest in the mask into rectangular will avoid the complex control logic introduced by the judgement of patches' score.



0	0	0	0	0
0	0	0	1	0
0	1	1	1	0
0	0	1	0	0
0	0	0	0	0

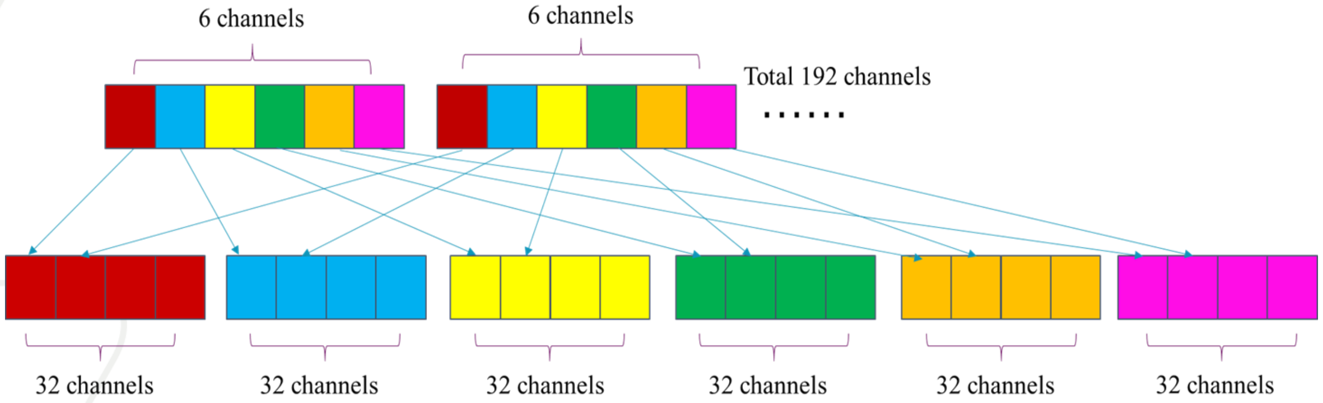
Non-rectangular Shape Mask

0	0	0	0	0
0	1	1	1	0
0	1	1	1	0
0	1	1	1	0
0	0	0	0	0

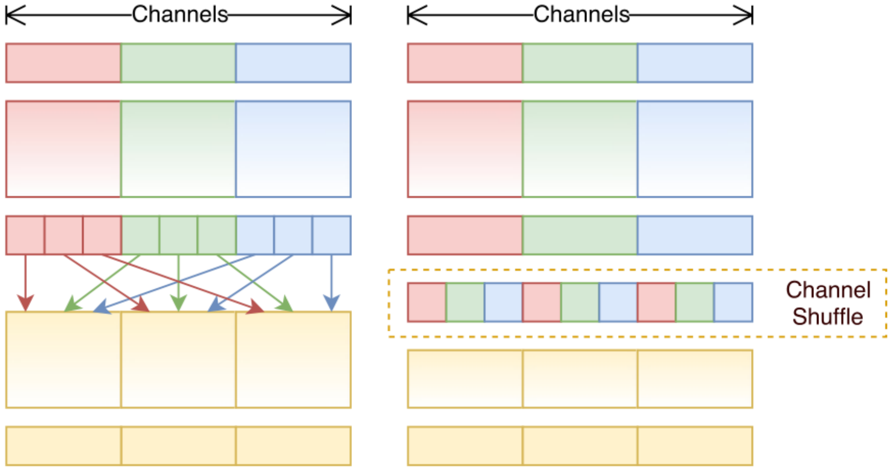
Rectangular Shape Mask

Hardware Innovations

- Channel shuffle
  - Reduce data movement between DRAM and on-chip memory to reduce computation cost.
  - Our channel shuffle method is the inverse process of that in ShuffleNet.



Channel shuffle in Mask-Net



Channel shuffle in ShuffleNet[1]

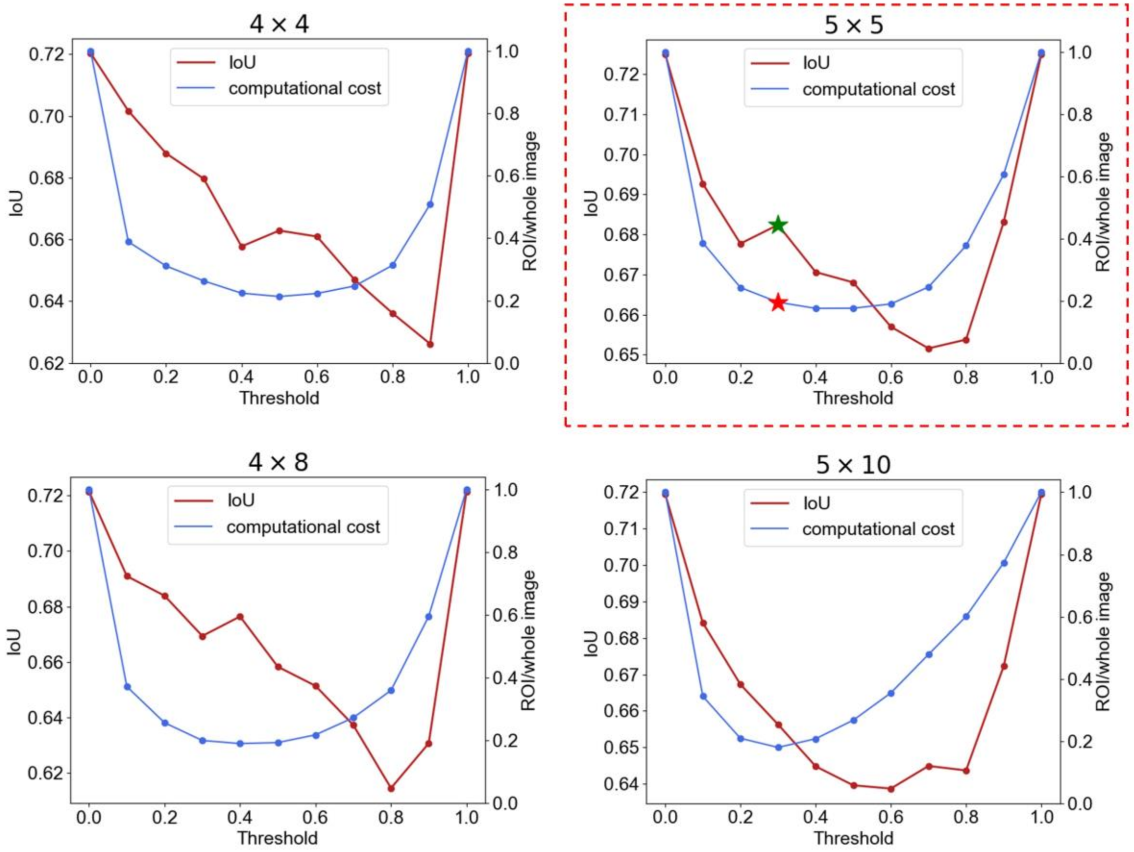
[1] Xiangyu Zhang, Xinyu Zhou, Mengxiao Lin, and Jian Sun. Shufflenet: An extremely efficient convolutional neural network for mobile devices. In Proceedings of the IEEE conference on computer vision and pattern recognition, pages 6848–6856, 2018.

Overview

- Background & Motivation
- Related work
    - Region proposal
    - Cascade network
- Mask-Net
    - The architecture
    - Promising features
    - Algorithm and hardware innovations
- Experiment results
- Design space exploration
- Future research directions

Experiment Results

- Mask shape and threshold study
  - We study the impact of different mask shapes on Mask-SkyNet performance. We choose threshold from 0-1 and different mask shapes including 4 × 4, 5 × 5, 4 × 8 and 5 × 10. The experiment is done on DAC-SDC dataset.

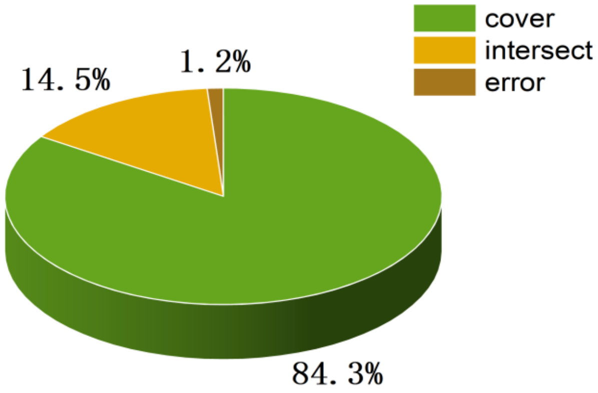


The shape we select is **5 × 5**

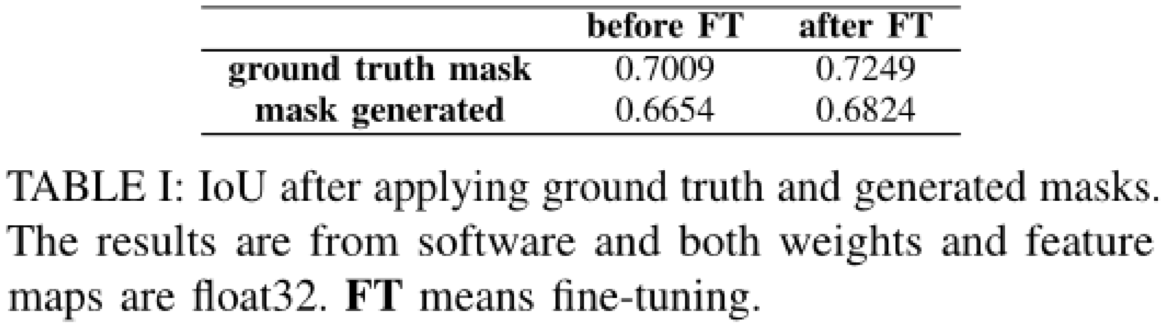
The threshold we select is **0.3**

Experiment Results

- Mask quality
  - The experiment is done using Mask-SkyNet on DAC-SDC dataset.
  - 84.3% of proposed regions can completely cover the object while only 1.2% are completely wrong.
  - Masked region proposals will only cause 5%~6% IoU loss.



Mask quality analysis



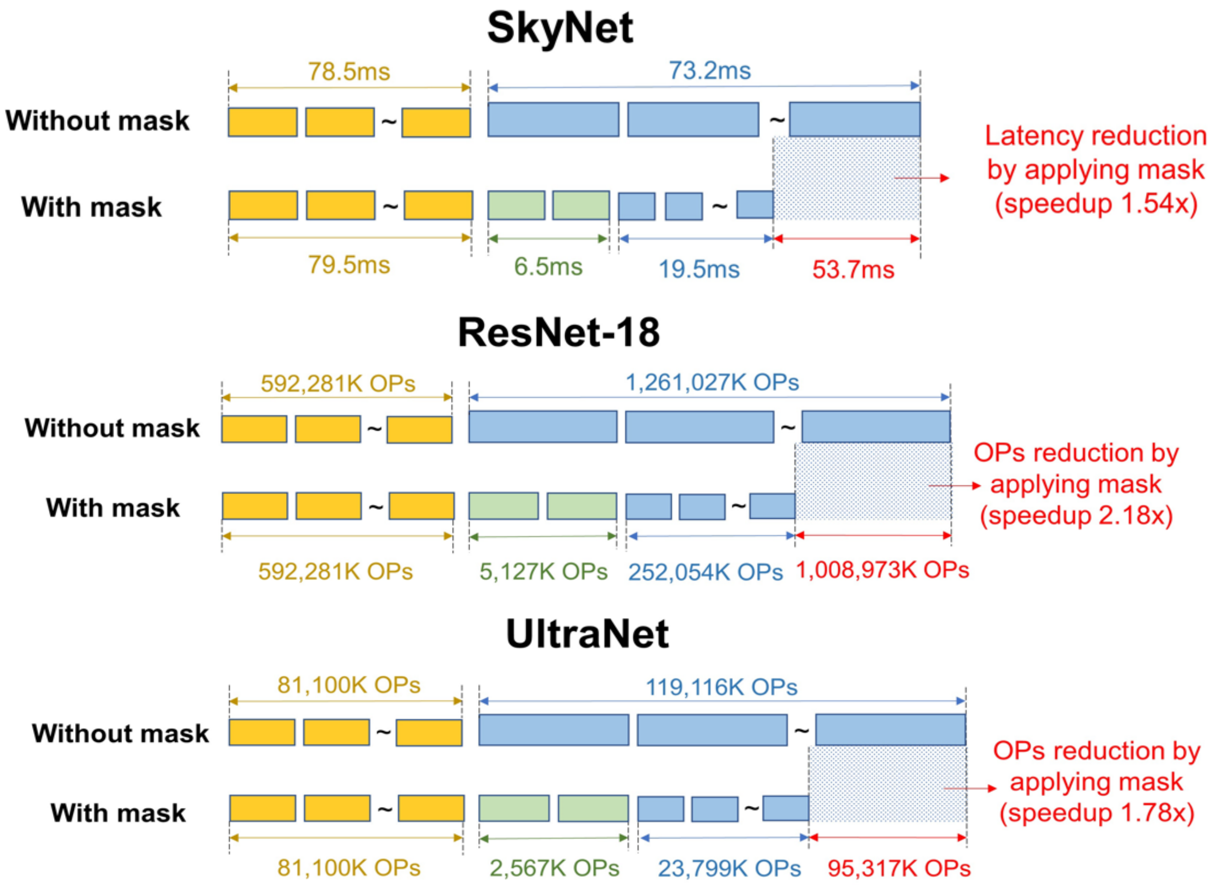
	before FT	after FT
ground truth mask	0.7009	0.7249
mask generated	0.6654	0.6824

TABLE I: IoU after applying ground truth and generated masks. The results are from software and both weights and feature maps are float32. **FT** means fine-tuning.

IoU loss comparison

Experiment Results

- C/RTL co-simulation results
  - Only 6% of total inference time allows the network to correctly distinguish between objects and background.



C/RTL co-simulation results from Vitis

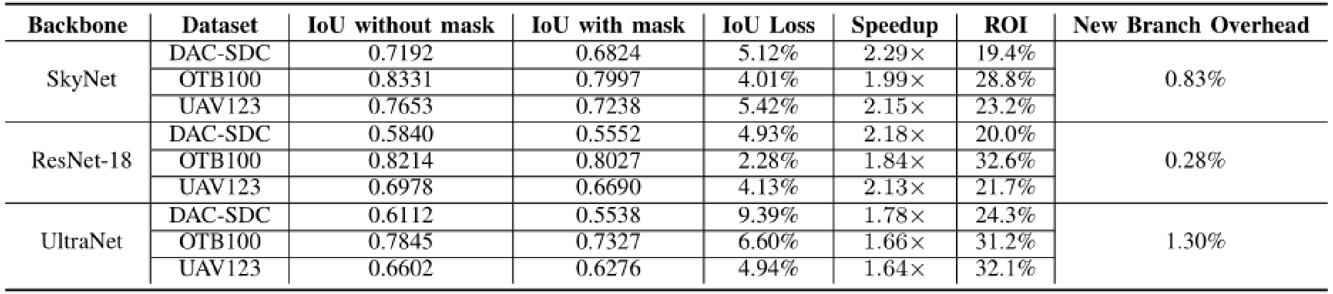
🟨 Layers before new branch

🟦 Layers after new branch

🟩 New branch layers

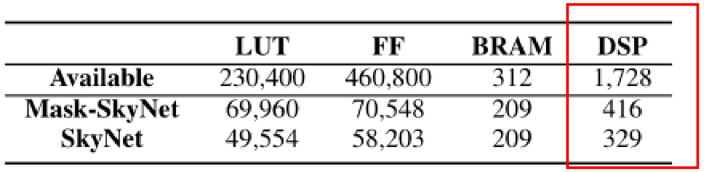
Experiment Results

- Software and hardware evaluation results
  - At software level, we evaluate Mask-Net using three detection backbones: SkyNet, UltraNet and ResNet-18 and three datasets: DAC-SDC, OTB100 and UAV123. The weights and feature maps are 32-bit.
  - At hardware level, we use the same datasets to test our Mask-SkyNet accelerator on ZCU106 FPGA. The weights are 11 bits and feature maps are 9 bits.



Backbone	Dataset	IoU without mask	IoU with mask	IoU Loss	Speedup	ROI	New Branch Overhead
SkyNet	DAC-SDC	0.7192	0.6824	5.12%	2.29×	19.4%	0.83%
	OTB100	0.8331	0.7997	4.01%	1.99×	28.8%	
	UAV123	0.7653	0.7238	5.42%	2.15×	23.2%	
ResNet-18	DAC-SDC	0.5840	0.5552	4.93%	2.18×	20.0%	0.28%
	OTB100	0.8214	0.8027	2.28%	1.84×	32.6%	
	UAV123	0.6978	0.6690	4.13%	2.13×	21.7%	
UltraNet	DAC-SDC	0.6112	0.5538	9.39%	1.78×	24.3%	1.30%
	OTB100	0.7845	0.7327	6.60%	1.66×	31.2%	
	UAV123	0.6602	0.6276	4.94%	1.64×	32.1%	

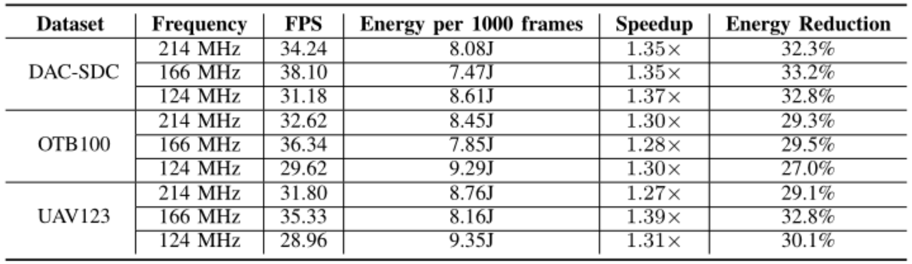
Software evaluation results



	LUT	FF	BRAM	DSP
Available	230,400	460,800	312	1,728
Mask-SkyNet	69,960	70,548	209	416
SkyNet	49,554	58,203	209	329

Resource utilization report

73 of the 87 DSPs added come from the new branch



Dataset	Frequency	FPS	Energy per 1000 frames	Speedup	Energy Reduction
DAC-SDC	214 MHz	34.24	8.08J	1.35×	32.3%
	166 MHz	38.10	7.47J	1.35×	33.2%
	124 MHz	31.18	8.61J	1.37×	32.8%
OTB100	214 MHz	32.62	8.45J	1.30×	29.3%
	166 MHz	36.34	7.85J	1.28×	29.5%
	124 MHz	29.62	9.29J	1.30×	27.0%
UAV123	214 MHz	31.80	8.76J	1.27×	29.1%
	166 MHz	35.33	8.16J	1.39×	32.8%
	124 MHz	28.96	9.35J	1.31×	30.1%

Hardware evaluation results

Overview

- Background & Motivation
- Related work
  - Region proposal
  - Cascade network
- Mask-Net
  - The architecture
  - Promising features
  - Algorithm and hardware innovations
- Experiment results
- Design space exploration
- Future research directions

Design Space Exploration

- Reasons for DSE
  - The time imbalance between different parts of the accelerator will affect the overall performance.
  - We want to optimally allocating DSPs to different parts of the accelerator to balance the computations under a fixed number of DSPs.

- DSE model
  - The model is used to calculate the theoretical speedup under fixed number of DSPs after DSP redistribution.

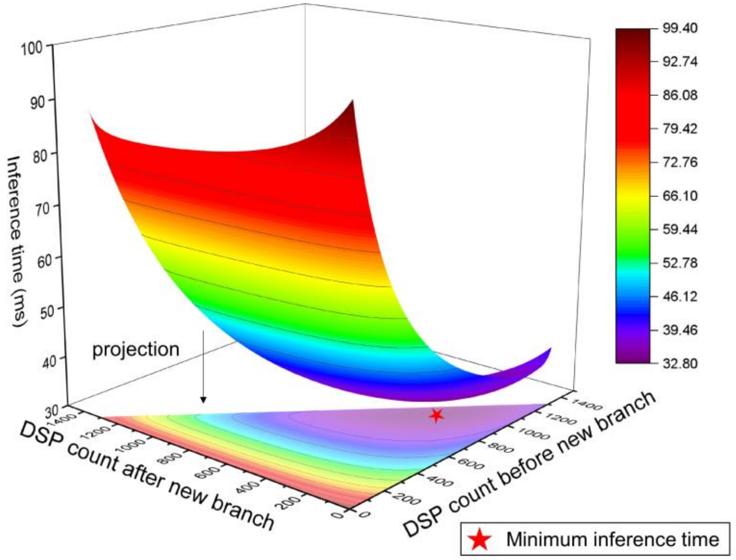
$$T_m = \frac{N_{pre}^*}{N_{pre}} \times t_{mpre} + \frac{N_{post}^*}{N_{post}} \times t_{mpost} + \frac{N_b^*}{N_b} \times t_{nb}$$

$$N_{total} = N_{pre} + N_{post} + N_b$$

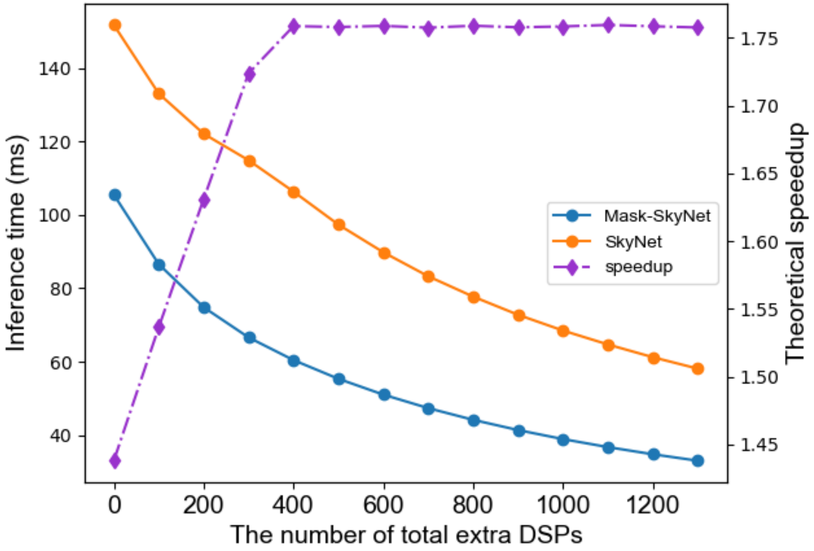
$$T = \frac{N_{pre}^*}{N_{pre}} \times t_{pre} + \frac{N_{post}^*}{N_{post}} \times t_{post}$$

$$r = \frac{T}{T_m}$$

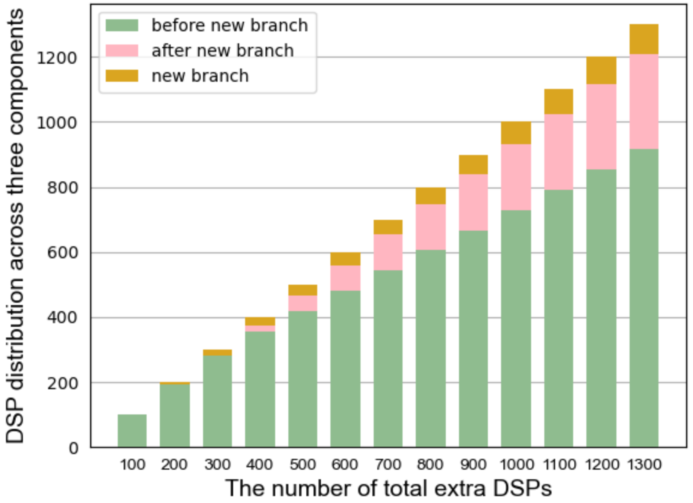
Design Space Exploration Results



DSP exploration space when extra DSP count is 1309



The relationship between the number of extra DSPs, inference time and theoretical speedup



The DSP distribution across three parts with different number of total extra DSPs.

Overview

- Background & Motivation
- Related work
    - Region proposal
    - Cascade network
- Mask-Net
    - The architecture
    - Promising features
    - Algorithm and hardware innovations
- Experiment results
- Design space exploration
- Future research directions

Future Research Directions

- Adaptive threshold
  - The threshold used in the gate in Mask-Net now is empirically selected from experiment results.
  - Adaptive threshold may select region proposals more efficiently, further reduce the computation cost, especially in case of complex background.
- Extend Mask-Net to different tasks
  - Object tracking
  - Image classification
  - Instance segmentation

Thank you!