FASSST: Fast Attention Based Single-Stage Segmentation Net for Real-Time Instance Segmentation

Yuan Cheng[1,2], Rui Lin[2], Peining Zhen[1], Tianshu Hou[1], Chiu Wa Ng[2], Hai-Bao Chen[1], Hao Yu[3], Ngai Wong[2]

[1] Shanghai Jiao Tong University
[2] The University of Hong Kong
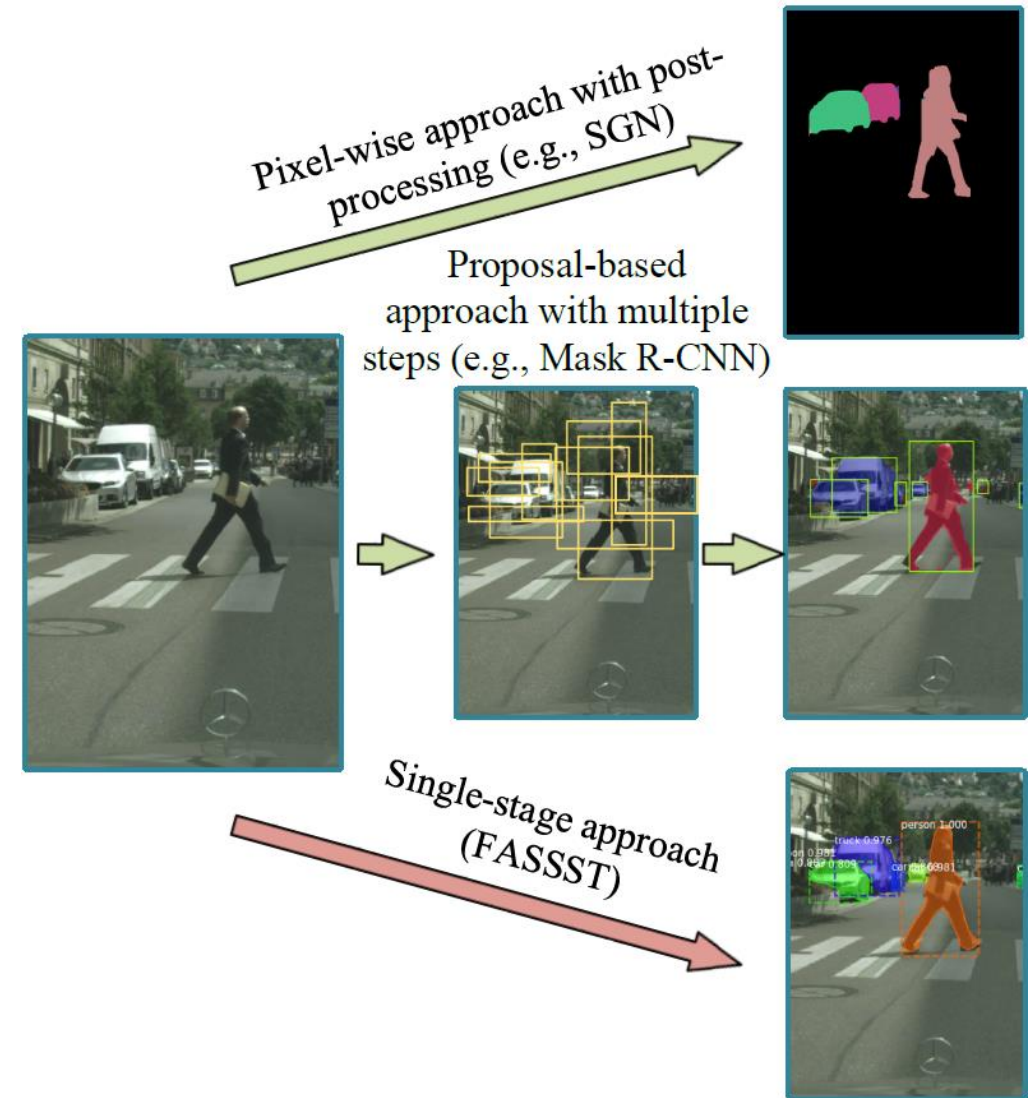[3] Southern University of Science and Technology

CONTENT

1. Background

2. FASSST

3. Experiments

CONTENT

Background

Can automobiles accurately and quickly **locate, classify and segment** instances in real time?

Pixel-wise approach with post-processing (e.g., SGN)

Proposal-based approach with multiple steps (e.g., Mask R-CNN)

Single-stage approach (FASSST)

CONTENT

FASSST Framework

We design FASSST (Fast Attention-based Single-Stage Segmentation NeT) for real-time instance segmentation. Using an **instance attention module (IAM),** FASSST quickly locates target instances and segments with **ROI feature fusion (RFF)** aggregating ROI features from pyramid mask layers.
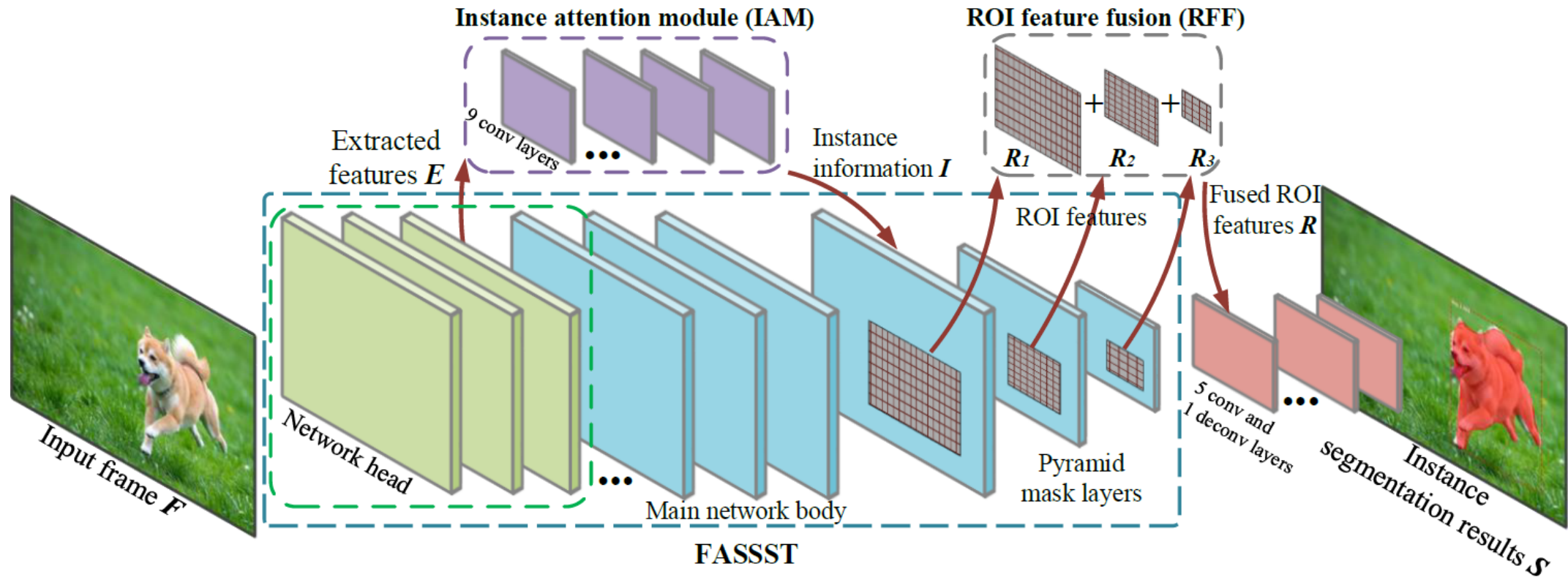
Working Flow

FASSST quickly locates target instances and segments ROIs by following steps:

➢ Step 1, $E = extr(F)$
➢ Step 2, $I = attn(E)$
➢ Step 3, $R = fuse(R_1, R_2, R_3)$
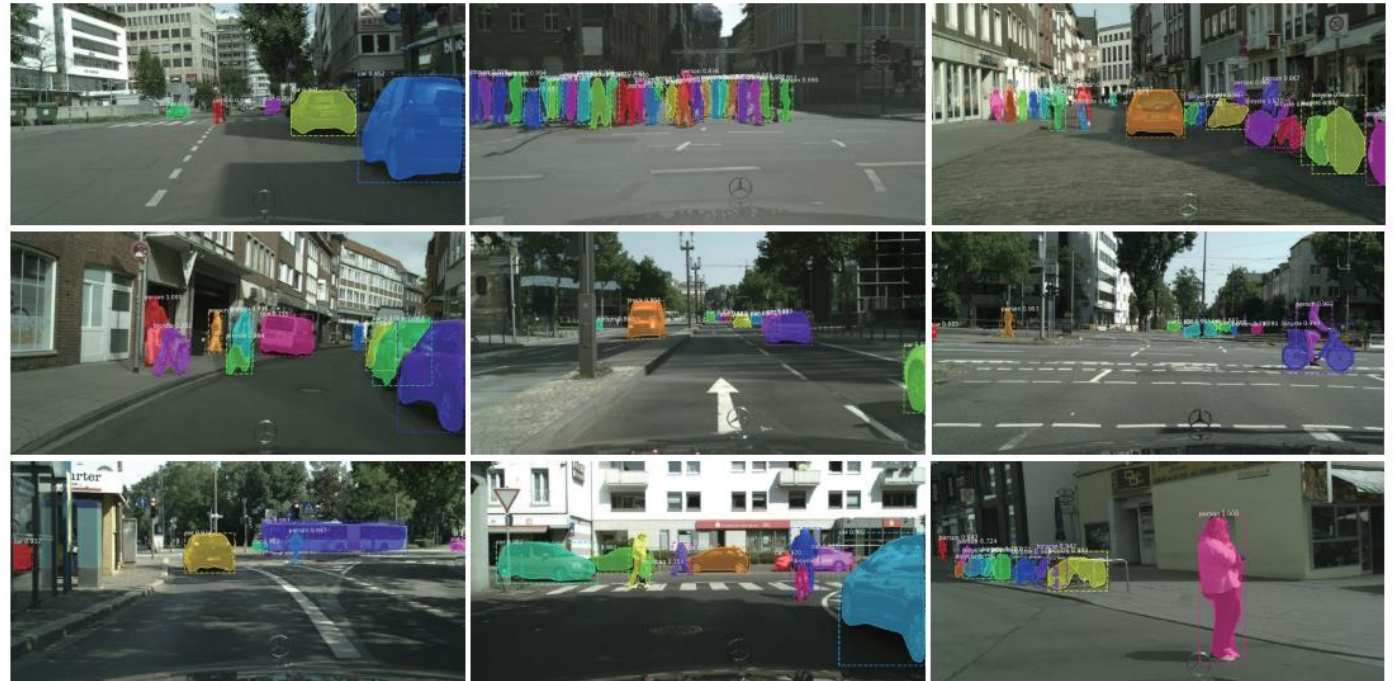➢ Step 4, get the final instance segmentation results S from R
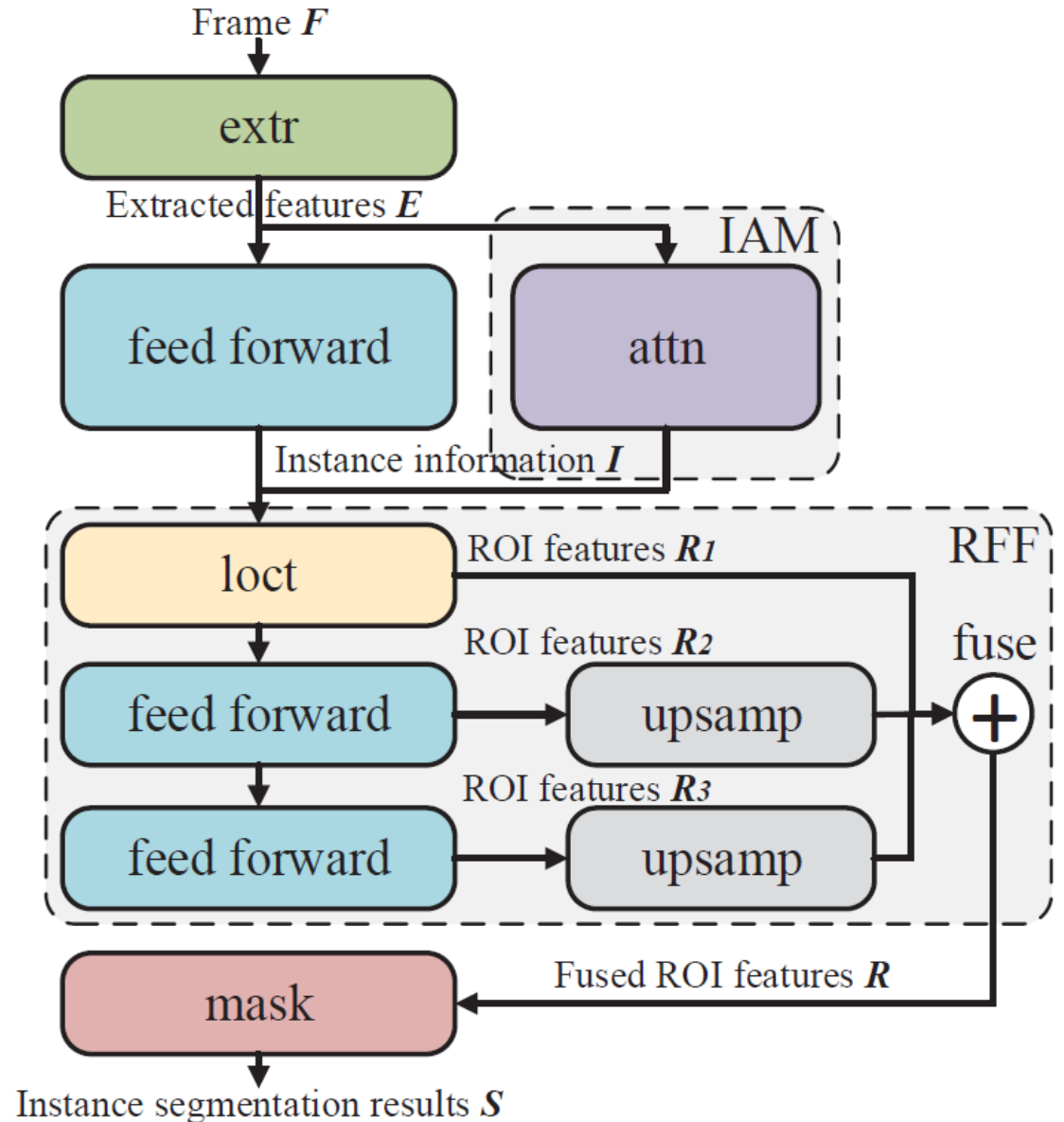
CONTENT

1. Background

2. FASSST

3. *Experiments*

Visual Results

On COCO and CityScapes datasets, all existing instances can be **located, classified and segmented** precisely in real time.

Accuracy Results

FASSST achieves competitive accuracy using **more compact backbone (MobileNet-54-V2)** for the main network body.

➢ The average AP of FASSST on COCO reaches **34.2**, which outperforms various state-of-the-arts and is only slightly lower than Mask R-CNN and SOLO.

Category	Approach	Backbone	AP	AP$_{50}$	AP$_{75}$	AP$_S$	AP$_M$	AP$_L$
Pixel-wise	SGN [22]	-	25.0	44.9	25.8	-	-	-
	SSAP [12]	ResNet-101-FPN	29.4	48.1	28.8	-	28.6	-
Proposal-based	FCIS [18]	ResNet-101-C5-dilated	29.2	49.5	-	7.1	31.3	50.0
	FCIS+++ [18]	ResNet-101-C5-dilated	33.6	54.5	37.9	-	-	-
	MNC [9]	ResNet-101-C4	24.6	44.3	24.8	4.7	25.9	43.6
	Mask R-CNN [13]	ResNet-101-FPN	35.7	58.0	37.8	15.5	38.1	52.4
Single-stage	ExtremeNet [32]	Hourglass-104	18.9	44.5	13.7	10.4	20.4	28.3
	YOLACT [2]	ResNet-101-FPN	31.2	50.6	32.8	12.1	33.3	47.1
	SOLO [28]	ResNet-101-FPN	37.8	59.5	40.4	16.4	40.6	54.2
	SipMask [17]	ResNet-101-FPN	32.8	53.4	34.3	9.3	35.6	54.0
	CenterMask [17]	ResNet-50-FPN	32.9	-	-	12.9	34.7	48.7
	PolarMask [30]	ResNet-101-FPN	30.4	51.9	31.0	13.4	32.4	42.8
Proposed	**FASSST**	MobileNet-54-V2	34.2	56.4	38.1	14.9	36.7	53.8

Performance Results

➢ FASSST reaches 59.2FPS on COCO and is 5.7× faster than Mask R-CNN.

➢ Requires FLOPs (71.6G) and storage (36.3MB) on COCO, which are 3.8× and 6.7× smaller than the Mask R-CNN.

➢ FASSST achieves 47.5FPS on CityScapes and 2.2× speedup YOLACT.

➢ Requires FLOPs (112.8G) and storage (41.3MB) on CityScapes, which are 1.9× and 4.6× smaller than YOLACT.

Approach	FPS	FLOPs (G)	Storage (MB)
SSAP [12]	5.5	-	-
FCIS [18]	6.2	364.1	207.0
Mask R-CNN [13]	10.3	273.6	242.3
RetinaMask [20]	6.8	358.3	423.6
MS R-CNN [14]	11.5	-	-
YOLACT-550 [2]	41.7	97.3	121.8
SOLO [28]	22.5	-	422.0
PolarMask-400 [30]	23.1	248.7	409.3
FASSST	59.2	71.6	36.3

Approach	FPS	FLOPs (G)	Storage (MB)
SegNet [1]	2.4	604.7	112.0
SSAP [12]	3.4	-	-
Mask R-CNN [13]	6.9	463.5	245.6
YOLACT-700 [2]	21.7	214.3	192.0
PolarMask-800 [30]	18.3	324.8	705.4
FASSST-768	47.5	112.8	43.7

Thank You!

Contact: Yuan Cheng

cyuan328@sjtu.edu.cn