NAFSSR: Stereo Image Super-Resolution Using NAFNet

Xiaojie Chu* Liangyu Chen* Wenqing Yu
chuxiaojie@stu.pku.edu.cn {chenliangyu,yuwenqing}@megvii.com

Winner of NTIRE 2022 Challenge on Stereo Image Super-Resolution

Introduction:

- Task: Reconstructing high-resolution details from a pair of low-resolution left and right images.
- Motivation: Both context information within a single view (i.e. intra-view information) and information between left and right image (i.e. cross-view information) are crucial.
- Contribution: A simple baseline named NAFSSR for stereo image super-resolution, by adding simple cross attention modules to state-of-the-art single image restorer (NAFNet).

Summary

- NAFSSR
  - Single View: NAFNet Block [1]
  - Cross-view: Stereo Cross Attention Module
- Training Tricks
  - Data augmentation: flip +RGB shuffle
  - Regularization: stochastic depth [2]
- Inference Tricks
  - Test-time Local Statistics Converter [3]

Code

Scan Me

- Code: github.com/megvii-research/NAFNet

References

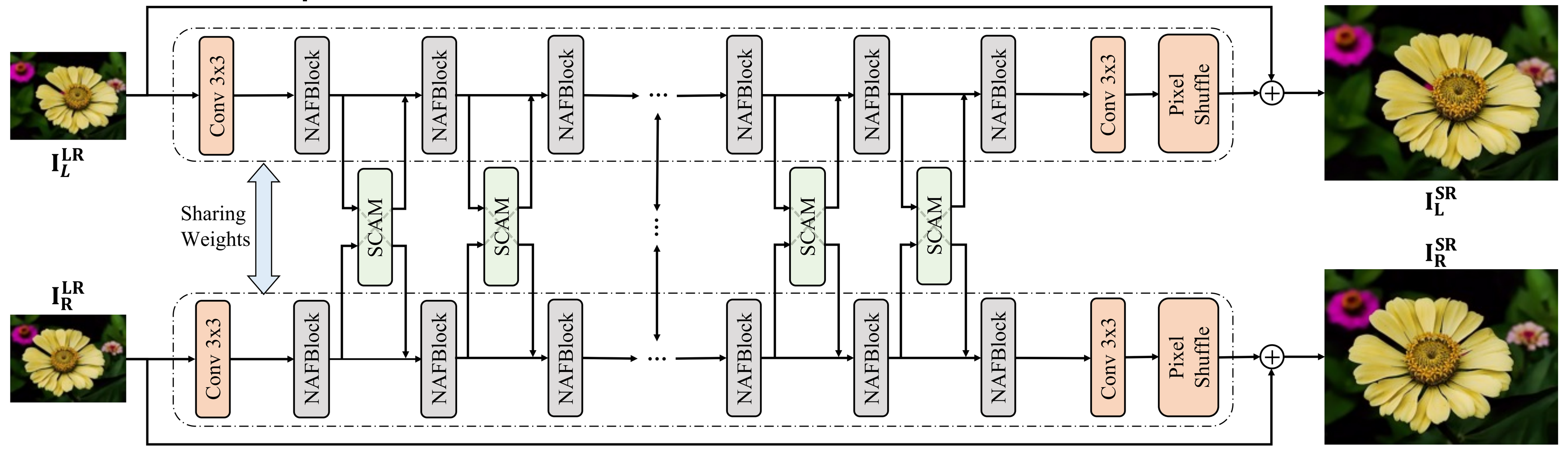
[1] Chen, Liangyu, et al. "Simple baselines for image restoration." arXiv preprint arXiv:2204.04676 (2022).

[2] Huang, Gao, et al. "Deep networks with stochastic depth." ECCV, 2016.

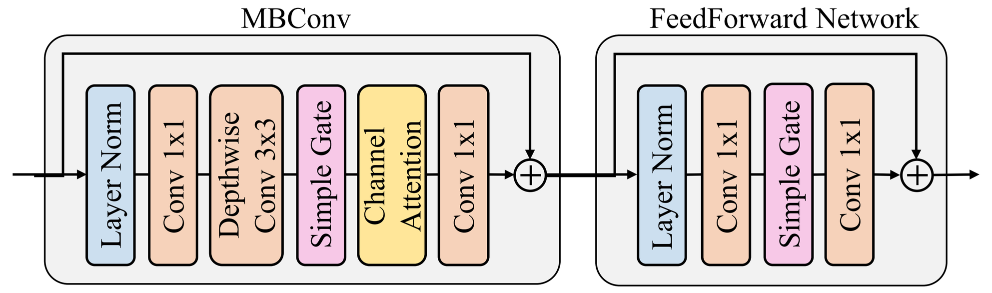
[3] Chu, Xiaojie, et al. "Revisiting Global Statistics Aggregation for Improving Image Restoration." arXiv preprint arXiv:2112.04491 (2021).

Methodology:

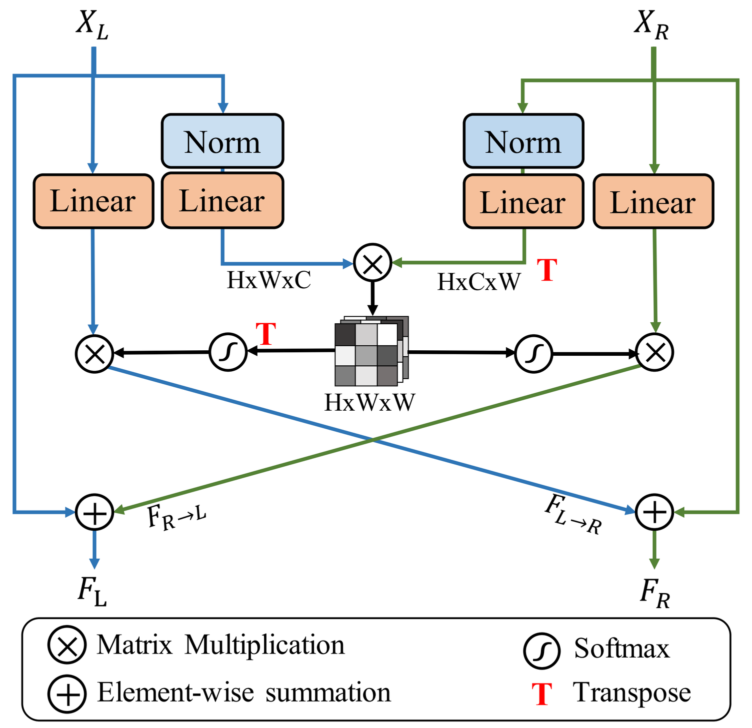
- NAFSSR Pipeline



- NAFNet's Block (NAFBlock)
  - SimpleGate($\mathbf{X}$)
    - $\mathbf{X_1} \odot \mathbf{X_2}$, where $\mathbf{X_1}, \mathbf{X_2}$ are obtained by splitting $\mathbf{X}$ along the channel dimension
  - Channel Attention
    - $\mathrm{CA}(\mathbf{X}) = \mathbf{X} * \mathbf{W}\,\mathrm{pool}(\mathbf{X})$



- Stereo Cross Attention Module (SCAM)
  - Scaled dot-Product Attention
    - $\mathrm{Attention}(\boldsymbol{Q}, \boldsymbol{K}, \boldsymbol{V}) = \mathrm{softmax}(\boldsymbol{Q}\boldsymbol{K}^T/\sqrt{C})\boldsymbol{V}$
  - Bidirectional Cross Attention
    - along the horizontal epipolar line
    - $F_{R\rightarrow L} = \mathrm{Attention}(\boldsymbol{W}_1^L \boldsymbol{X}_L,\ \boldsymbol{W}_1^R \boldsymbol{X}_R,\ \boldsymbol{W}_2^R \boldsymbol{X}_R)$
    - $F_{L\rightarrow R} = \mathrm{Attention}(\boldsymbol{W}_1^R \boldsymbol{X}_R,\ \boldsymbol{W}_1^L \boldsymbol{X}_L,\ \boldsymbol{W}_2^L \boldsymbol{X}_L)$
  - Fusion
    - $F_L = \gamma_L F_{R\rightarrow L} + X_L \ \mid\ F_R = \gamma_R F_{L\rightarrow R} + X_R$



Key Tricks:

- Strong regularization and augmentation for preventing overfitting
- Train-test inconsistency
  - Distribution of image-based features during inference differs from that of patch-based features during training
  - Converts global operation to local one, allowing it to extract representations based on local spatial region of features as in training phase.

Results:

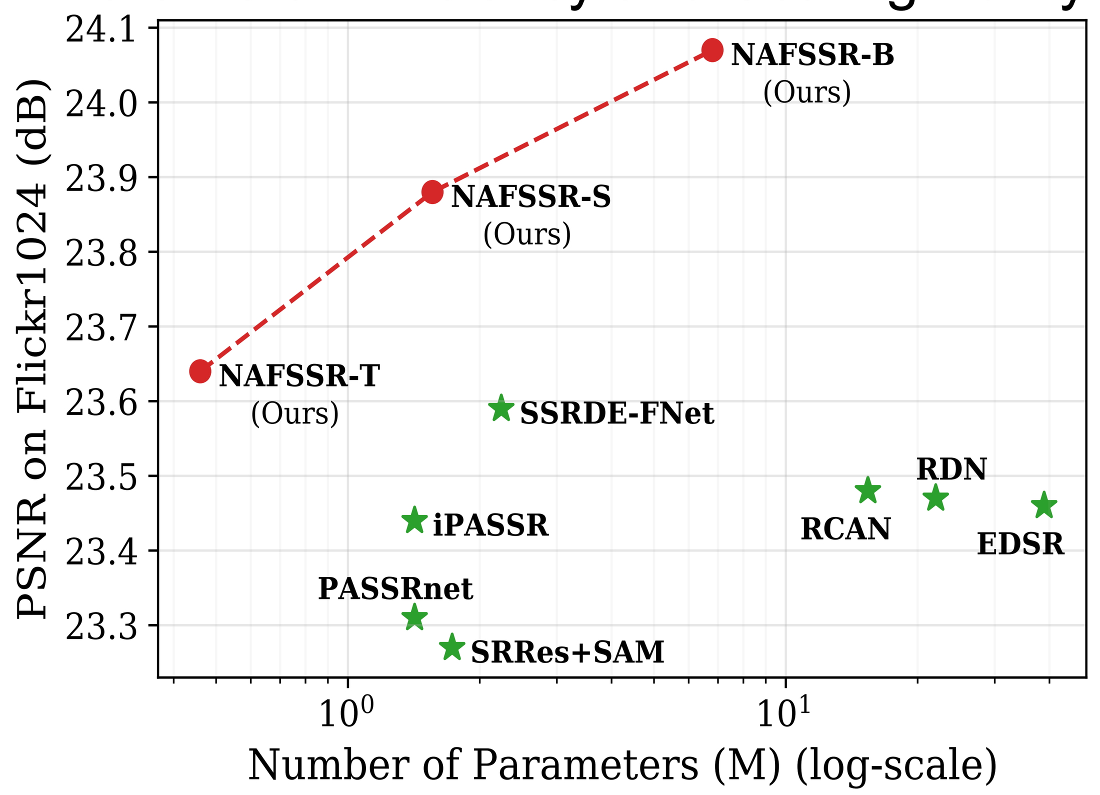
- Runtime Efficiency

Models	PSNR	Time(ms)	Speedup
SSRDEFNet [4]	23.59	238.5	1.00×
NAFSSR-T (Ours)	23.64 (+0.05)	46.7	5.11×
NAFSSR-S (Ours)	23.88 (+0.29)	91.8	2.60×
NAFSSR-B (Ours)	24.07 (+0.48)	224.9	1.06×

- Channel shuffle is complementary to flip augmentations

hflip	vflip	channel shuffle	PSNR	ΔPSNR
✗	✗	✗	23.43	-
✓	✗	✗	23.64	+0.21
✗	✓	✗	23.63	+0.20
✗	✗	✓	23.62	+0.19
✓	✓	✗	23.73	+0.30
✓	✓	✓	23.82	+0.39

- Parameter Efficiency and Scaling Ability



- More SCAM, better results

#SCAM	0	1	4	8	16	32
PSNR	23.56	23.74	23.76	23.79	23.82	23.85
ΔPSNR	-	+0.18	+0.20	+0.23	+0.26	+0.29

- Stochastic depth improves generality of models
- Solving train-test inconsistency by TLSC improves test-time performance

	Training	Test	In-distribution	Out-distribution			
Model	Stoch. Depth	TLSC	Flickr1024 [32]	KITTI 2012 [9]	KITTI 2015 [25]	Middlebury [27]	Average
NAFSSR-S	✓	✓	23.85	26.91	26.74	29.63	27.76
	✗	✓	23.82 (−0.03)	26.88 (−0.03)	26.71 (−0.03)	29.61 (−0.02)	27.73 (−0.03)
	✓	✗	23.78 (−0.07)	26.86 (−0.05)	26.67 (−0.07)	29.54 (−0.09)	27.69 (−0.07)
NAFSSR-B	✓	✓	24.10	27.05	26.89	29.93	27.96
	✗	✓	23.98 (−0.11)	26.92 (−0.13)	26.70 (−0.19)	29.78 (−0.15)	27.80 (−0.16)
	✓	✗	24.01 (−0.09)	27.00 (−0.05)	26.80 (−0.09)	29.81 (−0.12)	27.87 (−0.09)

- Visual Results