Recursive Generalization Transformer for Image Super-Resolution

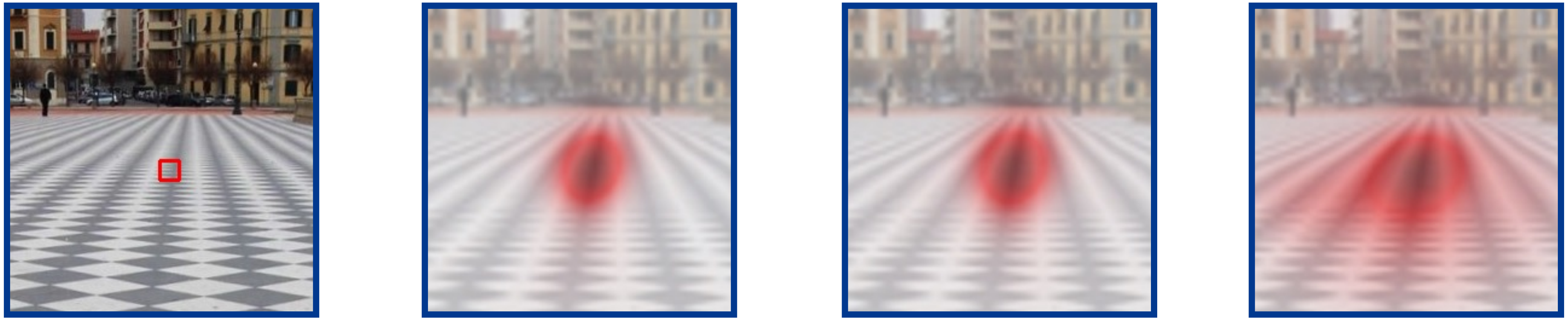
Zheng Chen[1], Yulun Zhang[1*†], Jinjin Gu[2, 3], Linghe Kong[1*], Xiaokang Yang[1]

[1]Shanghai Jiao Tong University, [2]Shanghai AI Laboratory, [3]The University of Sydney

Introduction

Motivation

- **Window Attention:** efficiency through window-based attention in Transformer models.
- **Global Information:** crucial for image reconstruction.
- **Larger window:** increases the field of view, though with higher model complexity.

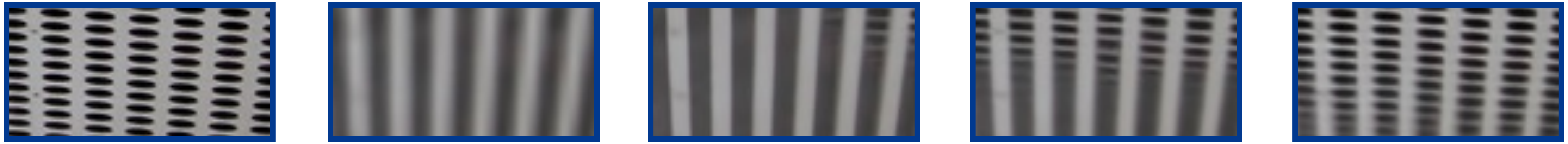


Model: Params(M)/FLOPs(G) **SwinIR:** 11.90/215.32 **CAT:** 16.60/360.67 **RGT(ours):** 10.20/193.08

Introduction

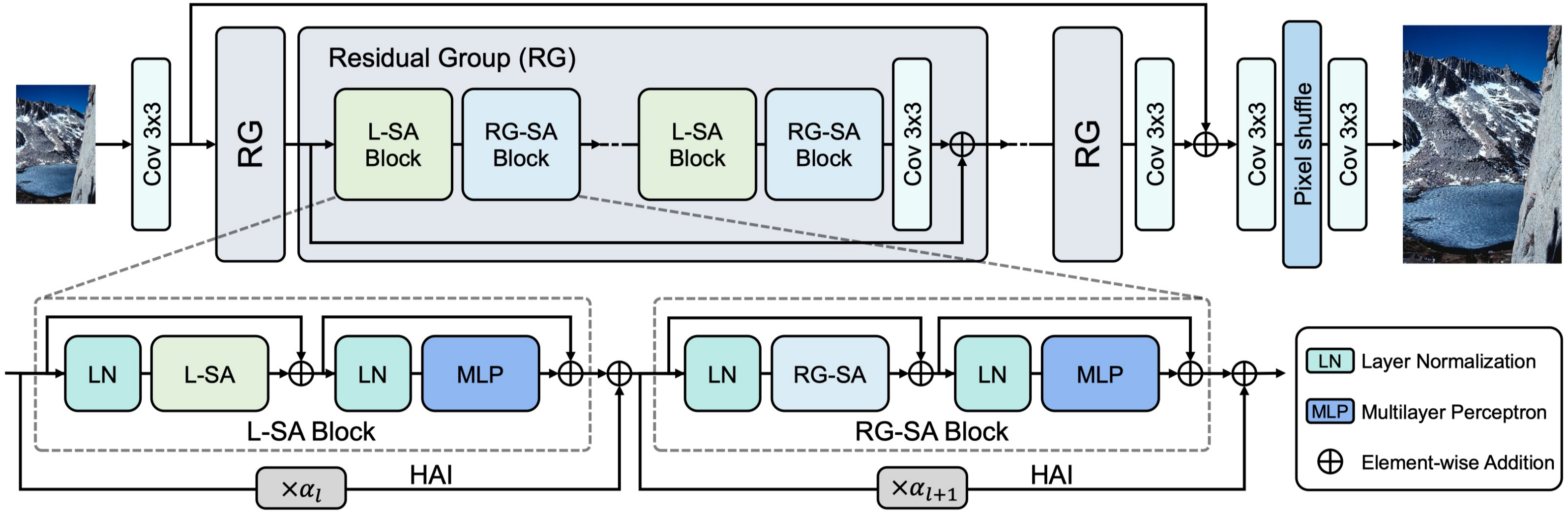
Motivation

- **Demand:** develop a method for image SR to capture global information effectively.
- **RGT:** we design the **R**ecursive **G**eneralization **T**ransformer, which can capture global spatial information and is suitable for high-resolution images.
- **Better Performance:** RGT achieves superior SR performance quantitatively and visually.



HR | **LR:** PSNR/SSIM | **SwinIR:** 33.81/0.9427 | **CAT:** 34.26/0.9440 | **RGT(ours):** 34.47/0.9467

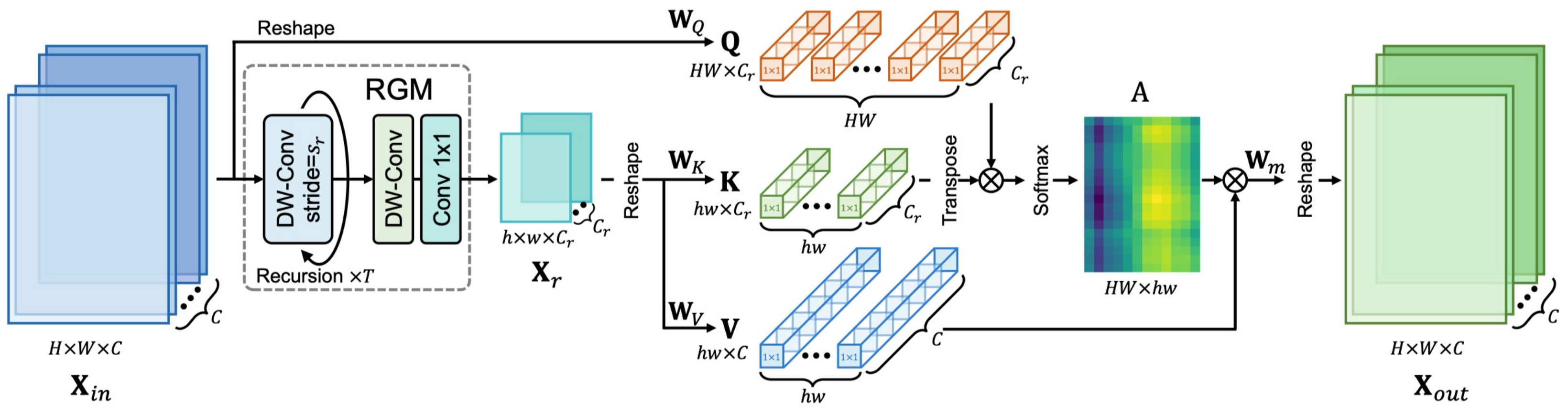
Method



Architecture

- **RGT:** utilizes two attention mechanisms: local self-attention (L-SA) and recursive-generalization self-attention (RG-SA), coupled with Hybrid Adaptive Integration (HAI).
- **RG-SA:** models global dependencies with linear complexity.
- **HAI:** integrates different modules, combining global and local information.

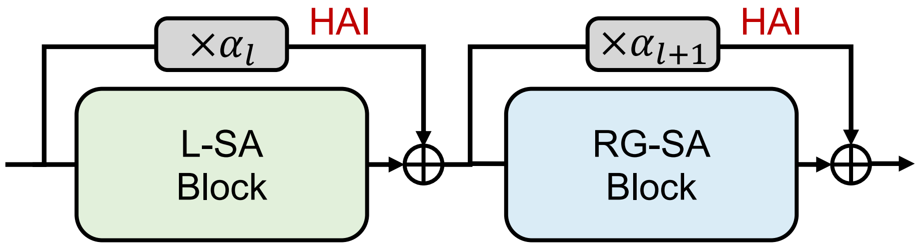
Method

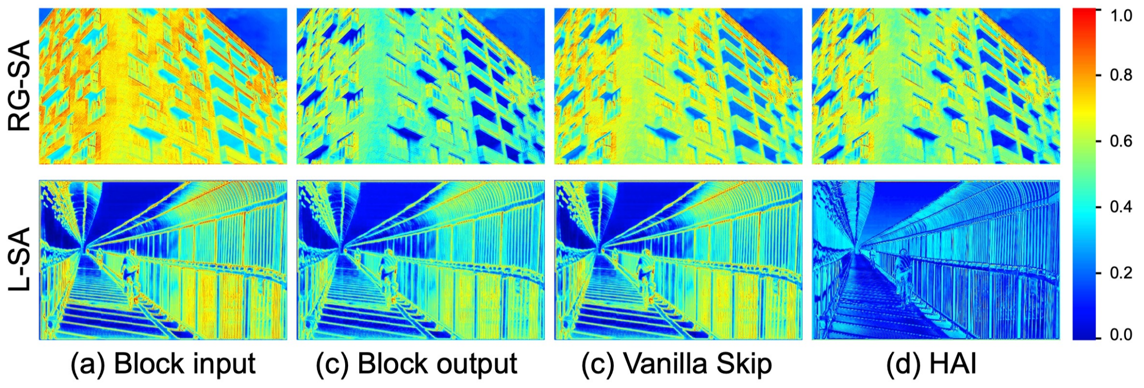


RG-SA

- **Recursive Generalization Module (RGM):** aggregates image features of any resolution into representative maps. The recursive time is $T = log_{s_r}(\frac{H}{W})$.
- **Cross-Attention:** calculates between input features and representative maps.
- **Complexity:** $\mathcal{O}(HWC^2)$, linearly related to image resolution.

Method





HAI

- **Hybrid Adaptive Integration (HAI):** acts on the outside of each block, where input features are adaptively adjusted by a learnable adapter α.
- **Module Integration:** couples global and local modules.
- **Visual Results:** feature maps from different modules are adaptively fused through HAI.

Experiments

L-SA	RG-SA	HAI	Params(M)	FLOPs(G)	PSNR(dB)	SSIM
✓			10.69	229.42	33.43	0.9396
✓	✓		10.04	183.08	33.52	0.9405
✓	✓	✓	10.05	183.08	33.68	0.9414

Ablation: break-down

- **Baseline:** Transformer with only local self-attention.
- Demonstrate the effectiveness of the RG-SA and HAI.

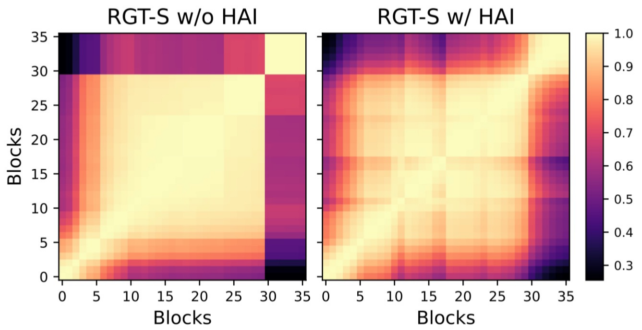
Method	Recursion	c_r	Params(M)	FLOPs(G)	PSNR(dB)	SSIM
w/o Recur		0.5	10.05	274.54	33.57	0.9412
w/o Scale	✓	1	11.37	189.62	33.54	0.9404
RGT-S	✓	0.5	10.05	183.08	33.68	0.9414

Ablation: RG-SA

- Recursive operation reduces the FLOPs by 30%.
- Channel scaling mitigates the redundancy between channels.

Experiments

Method	Vanilla Skip	α	Params (M)	FLOPs (G)	PSNR (dB)	SSIM
w/o HAI			10.04	183.08	33.52	0.9405
w/ Skip	✓		10.04	183.08	32.71	0.9339
w/ HAI	✓	✓	10.05	183.08	33.68	0.9414



Method	Params(M)	FLOPs(G)	PSNR(dB)	SSIM
L-SA only	10.69	229.42	33.43	0.9396
L-SA w/ HAI	10.69	229.42	33.44	0.9400

Ablation: HAI

- Vanilla skip connection degrades the model performance.
- HAI adaptively adjusts the input features, obtaining 0.16 dB gain.

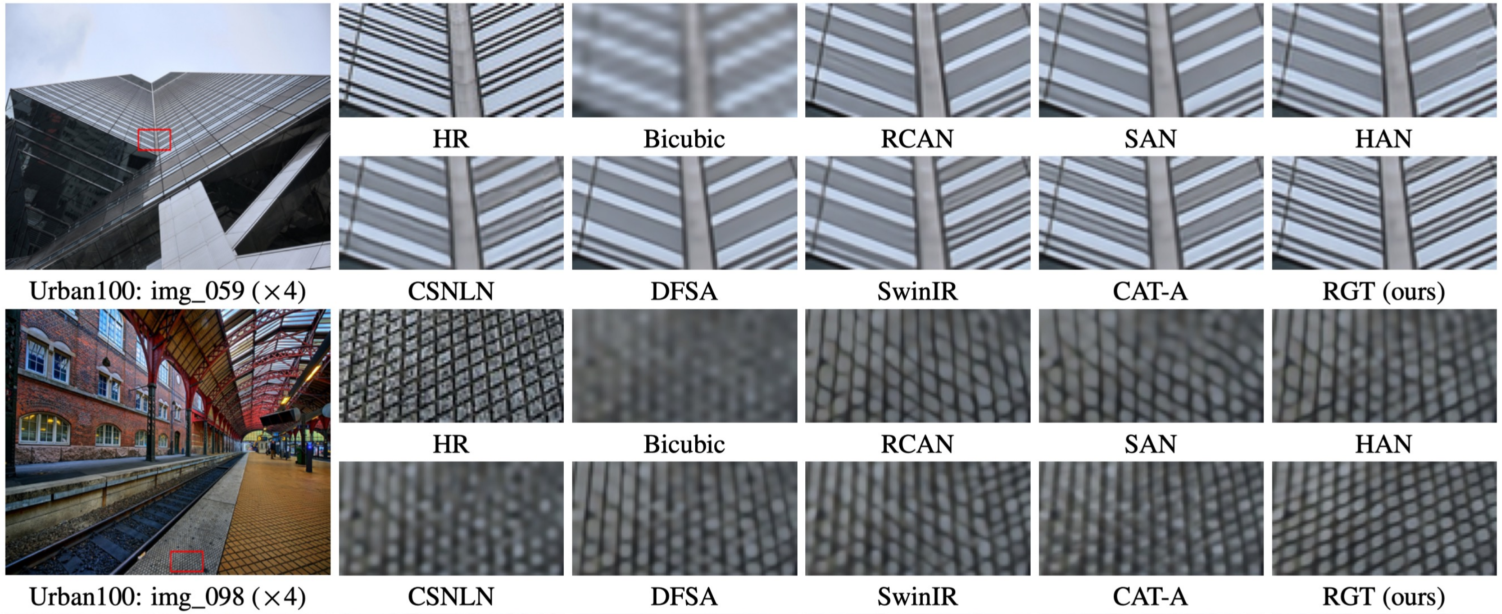
Experiments

Method	Scale	Set5 PSNR	Set5 SSIM	Set14 PSNR	Set14 SSIM	B100 PSNR	B100 SSIM	Urban100 PSNR	Urban100 SSIM	Manga109 PSNR	Manga109 SSIM
EDSR (Lim et al., 2017)	×2	38.11	0.9602	33.92	0.9195	32.32	0.9013	32.93	0.9351	39.10	0.9773
RCAN (Zhang et al., 2018a)	×2	38.27	0.9614	34.12	0.9216	32.41	0.9027	33.34	0.9384	39.44	0.9786
SRFBN (Li et al., 2019)	×2	38.11	0.9609	33.82	0.9196	32.29	0.9010	32.62	0.9328	39.08	0.9779
SAN (Dai et al., 2019)	×2	38.31	0.9620	34.07	0.9213	32.42	0.9028	33.10	0.9370	39.32	0.9792
HAN (Niu et al., 2020)	×2	38.27	0.9614	34.16	0.9217	32.41	0.9027	33.35	0.9385	39.46	0.9785
CSNLN (Mei et al., 2020)	×2	38.28	0.9616	34.12	0.9223	32.40	0.9024	33.25	0.9386	39.37	0.9785
NLSA (Mei et al., 2021)	×2	38.34	0.9618	34.08	0.9231	32.43	0.9027	33.42	0.9394	39.59	0.9789
CRAN (Zhang et al., 2021)	×2	38.31	0.9617	34.22	0.9232	32.44	0.9029	33.43	0.9394	39.75	0.9793
DFSA (Magid et al., 2021)	×2	38.38	0.9620	34.33	0.9232	32.50	0.9036	33.66	0.9412	39.98	0.9798
ELAN (Zhang et al., 2022)	×2	38.36	0.9620	34.20	0.9228	32.45	0.9030	33.44	0.9391	39.62	0.9793
SwinIR (Liang et al., 2021)	×2	38.42	0.9623	34.46	0.9250	32.53	0.9041	33.81	0.9427	39.92	0.9797
CAT-A (Chen et al., 2022c)	×2	38.51	0.9626	34.78	0.9265	32.59	0.9047	34.26	0.9440	40.10	0.9805
RGT-S (ours)	×2	38.56	0.9627	34.77	0.9270	32.59	0.9050	34.32	0.9457	40.18	0.9805
RGT (ours)	×2	38.59	0.9628	34.83	0.9272	32.62	0.9050	34.47	0.9467	40.34	0.9808
RGT+ (ours)	×2	38.62	0.9629	34.88	0.9275	32.64	0.9053	34.63	0.9474	40.45	0.9810
EDSR (Lim et al., 2017)	×4	32.46	0.8968	28.80	0.7876	27.71	0.7420	26.64	0.8033	31.02	0.9148
RCAN (Zhang et al., 2018a)	×4	32.63	0.9002	28.87	0.7889	27.77	0.7436	26.82	0.8087	31.22	0.9173
SRFBN (Li et al., 2019)	×4	32.47	0.8983	28.81	0.7868	27.72	0.7409	26.60	0.8015	31.15	0.9160
SAN (Dai et al., 2019)	×4	32.64	0.9003	28.92	0.7888	27.78	0.7436	26.79	0.8068	31.18	0.9169
HAN (Niu et al., 2020)	×4	32.64	0.9002	28.90	0.7890	27.80	0.7442	26.85	0.8094	31.42	0.9177
CSNLN (Mei et al., 2020)	×4	32.68	0.9004	28.95	0.7888	27.80	0.7439	27.22	0.8168	31.43	0.9201
NLSA (Mei et al., 2021)	×4	32.59	0.9000	28.87	0.7891	27.78	0.7444	26.96	0.8109	31.27	0.9184
CRAN (Zhang et al., 2021)	×4	32.72	0.9012	29.01	0.7918	27.86	0.7460	27.13	0.8167	31.75	0.9219
DFSA (Magid et al., 2021)	×4	32.79	0.9019	29.06	0.7922	27.87	0.7458	27.17	0.8163	31.88	0.9266
ELAN (Zhang et al., 2022)	×4	32.75	0.9022	28.96	0.7914	27.83	0.7459	27.13	0.8167	31.68	0.9226
SwinIR (Liang et al., 2021)	×4	32.92	0.9044	29.09	0.7950	27.92	0.7489	27.45	0.8254	32.03	0.9260
CAT-A (Chen et al., 2022c)	×4	33.08	0.9052	29.18	0.7960	27.99	0.7510	27.89	0.8339	32.39	0.9285
RGT-S (ours)	×4	32.98	0.9047	29.18	0.7966	27.98	0.7509	27.89	0.8347	32.38	0.9281
RGT (ours)	×4	33.12	0.9060	29.23	0.7972	28.00	0.7513	27.98	0.8369	32.50	0.9291
RGT+ (ours)	×4	33.16	0.9066	29.28	0.7979	28.03	0.7520	28.09	0.8388	32.68	0.9303

Quantitative

- **Two variants:** RGT-S and RGT, with different computational complexity.
- Compare our methods with some recent state-of-the-art methods.
- Our proposed RGT outperforms other methods on **all** datasets with **all** scaling factors.

Experiments



Visual

- RGT can alleviate the blurring artifacts better and recover more image details.
- Visual results further demonstrate the effectiveness of our method.

Method	EDSR	RCAN	HAN	CSNLN	SwinIR	CAT-A	RGT-S (ours)	RGT (ours)
Params(M)	43.09	15.59	16.07	6.57	11.90	16.60	10.20	13.37
FLOPs(G)	823.34	261.01	269.1	84,155.24	215.32	360.67	193.08	251.07
Urban100	26.64	26.82	26.85	27.22	27.45	27.89	27.89	27.98
Manga109	31.02	31.22	31.42	31.43	32.03	32.39	32.38	32.50

Model Size

Better trade-off between complexity and performance.

Conclusion

Contribution

- We propose the Recursive Generalization Transformer (RGT) for accurate image SR.
- We design the recursive-generalization self-attention (RG-SA) to model global dependency with linear complexity.
- We design the hybrid adaptive integration (HAI) for global and local integration.

Project Paper Homepage

Poster

- Time: Thu 9 May 4:30 p.m. - 6:30 p.m.
- Session: Halle B #287

Thanks!