NTIRE 2025 Efficient SR Challenge Factsheet
-Reparameterized Pixel Attention Network for Efficient Super-Resolution-

Yawei Li[1], Jinbiao Zhao[2], Zhao Zhang[3]
Hefei University of Technology
Hefei, China
jarvis65536@gmail.com[1], sotakingzhao@gmail.com[2], cszzhang@gmail.com[3]

1. Team details

- NanoSR
- Yawei Li
- Addr: Hefei University of Technology, Hefei, China
Tel: (+86) 15032403279
Email: jarvis65536@gmail.com
- Rest of the team members: Jinbiao Zhao, Zhao Zhang
- Codalab user name: jarvis_kyrie
- Best scoring entries of the team during the development/validation phase: entry #2 on 03/02/25
- Link to the codes/executables of the solution(s): `https://github.com/kyrie2to11/NTIRE2025_ESR_NanoSR.git`

2. Method Details

- **Network Architecture:** Our network architecture is inspired by SPAN [2] and PAN [4]. While maintaining the overall design of SPAN, we replace the SPAB block with the RepBlock. The RepBlock consists of a feature extractor using reparameterized convolution and a reparameterized pixel attention module. During training, the RepBlock operates in a complex mode to achieve better quality performance but can be equivalently transformed into a simple mode with fewer parameters and FLOPs. The detailed network architecture is illustrated in Fig. 2.
- **Reparameterized Convolution:** Reparameterized convolution plays a crucial role in improving the performance of efficient CNN-based super-resolution networks. We employ the RepMBConv introduced in PlainUSR [3], and this RepMBConv forms all the convolutions in the RepBlock. In addition, RepMBConv is derived from MobileNetV3 [1] Block (MBConv). The architecture of RepMBConv is depicted in Fig. 1.

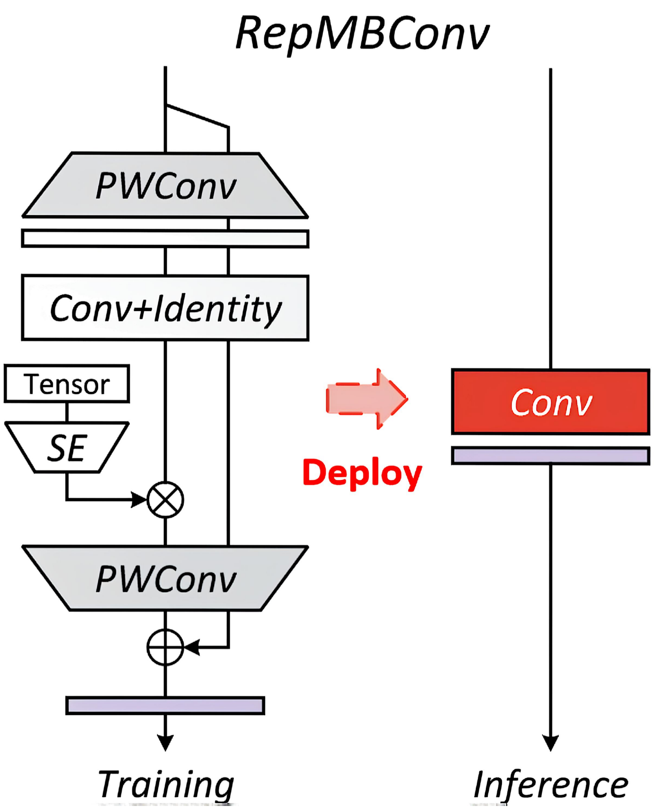

Figure 1. Detail of RepMBConv.

- **Implementation Details:** We train the model using all 85,791 image pairs from the DIV2K and LSDIR datasets. Each image pair is cropped into 480×480 sub-patches for training. During each training batch, 64 HR RGB patches of size 128×128 are randomly cropped and augmented with random flipping and ro-

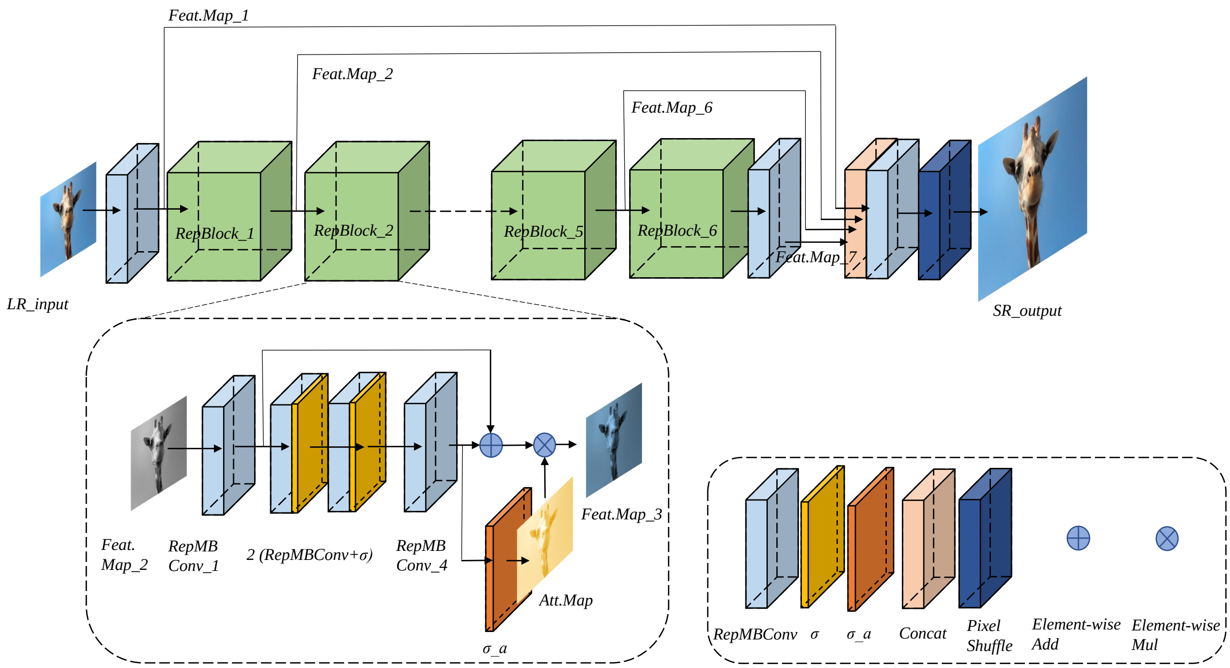

Figure 2. Network Architecture of NanoSR. Zoom in for better viewing.

tation. The optimization objective is the ℓ_1 loss, and we use the AdamW optimizer ($\beta_1 = 0.9$, $\beta_2 = 0.99$) to train NanoSR. The learning rate is initialized at 5×10^{-4} and halved at $\{250\text{k}, 400\text{k}, 450\text{k}, 475\text{k}\}$ iterations within a total of 500k iterations. The proposed method is implemented using the PyTorch framework on a single NVIDIA RTX 4090 GPU.

- **Total Method Complexity:** This section presents the validation results, including PSNR, SSIM, total method complexity, number of parameters, FLOPs, GPU memory consumption, and runtime. For detailed information, refer to Tab. 1.

Table 1. Model Performance Comparison

Model	Val PSNR	Val SSIM	Val Time [ms]	Params [M]	FLOPs [G]	Acts [M]	Mem [M]	Conv
NanoSR_train	26.97	0.78	31.62	2.423	156.48	472.25	1067.08	100
NanoSR_inference	26.97	0.78	5.87	0.551	36.02	88.08	965.48	28

References

[1] Andrew Howard, Mark Sandler, Grace Chu, Liang-Chieh Chen, Bo Chen, Mingxing Tan, Weijun Wang, Yukun Zhu, Ruoming Pang, Vijay Vasudevan, et al. Searching for mobilenetv3. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 1314–1324, 2019.

[2] Cheng Wan, Hongyuan Yu, Zhiqi Li, Yihang Chen, Yajun Zou, Yuqing Liu, Xuanwu Yin, and Kunlong Zuo. Swift parameter-free attention network for efficient super-resolution. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6246–6256, 2024.

[3] Yan Wang, Yusen Li, Gang Wang, and Xiaoguang Liu. Plainusr: Chasing faster convnet for efficient super-resolution. In *Proceedings of the Asian Conference on Computer Vision*, pages 4262–4279, 2024.

[4] Hengyuan Zhao, Xiangtao Kong, Jingwen He, Yu Qiao, and Chao Dong. Efficient image super-resolution using pixel attention. In *Computer Vision–ECCV 2020 Workshops: Glasgow, UK, August 23–28, 2020, Proceedings, Part III 16*, pages 56–72. Springer, 2020.