RDRN: Recursively Defined Residual Network for Image Super-Resolution Supplementary Material

Alexander Panaetov[1[0000−0003−2309−9798]], Karim Elhadji Daou[1[0000−0003−4677−7571]], Igor Samenko[1[0000−0001−9400−312X]], Evgeny Tetin[1[0000−0001−6878−8330]], and Ilya Ivanov[1[0000−0001−7919−5143]]

Huawei, Moscow Research Center, Russia `{panaetov.alexander1, karim.daou, samenko.igor, evgeny.tetin, ivanov.ilya1}@huawei.com`

1 Results on Image SR ($\times$ 8)

We show the comparison on classical image SR ($\times$ 8, BI degradation) in Table 1.

Table 1. Quantitative results with BI degradation model. The best and second best results are highlighted in **bold** and underlined

Methods	Scale	Set5		Set14		B100		Urban100		Manga109	
		PSNR	SSIM	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM
Bicubic	×8	24.40	0.6580	23.10	0.5660	23.67	0.5480	20.74	0.5160	21.47	0.6500
SRCNN [3]	×8	25.33	0.6900	23.76	0.5910	24.13	0.5660	21.29	0.5440	22.46	0.6950
RCAN [8]	×8	27.31	0.7878	25.23	0.6511	24.98	0.6058	23.00	0.6452	25.24	0.8029
SAN [2]	×8	27.22	0.7829	25.14	0.6476	24.88	0.6011	22.70	0.6314	24.85	0.7906
DRLN [1]	×8	27.36	0.7882	25.34	0.6531	25.01	0.6057	23.06	0.6471	25.29	0.8041
HAN [6]	×8	27.33	0.7884	25.24	0.6510	24.98	0.6059	22.98	0.6347	25.20	0.8000
SwinIR [4]	×8	**27.55**	0.7941	**25.46**	**0.6568**	25.04	**0.6092**	23.17	**0.6547**	25.55	**0.8132**
RDRN(ours)	×8	27.52	**0.7953**	25.40	0.6548	**25.10**	0.6084	**23.20**	0.6511	**25.66**	0.8114
SwinIR+ [4]	×8	27.58	0.7949	**25.49**	**0.6579**	25.06	**0.6100**	23.26	**0.6575**	25.66	0.8154
RDRN+ (ours)	×8	**27.65**	**0.7971**	25.48	0.6573	**25.15**	**0.6100**	**23.38**	0.6568	**25.93**	**0.8157**

2 Additional Visual Comparison

We provide more visual comparison for $\times 2$, $\times 3$, $\times 4$ SR with BI degradation model in Figures 1, 2, 3 and for $\times 3$ SR with BD degradation model in Figure 4.

3 RDRB Implementation Details

To avoid any misunderstanding about the definition of the proposed RDRB, we provide PyTorch implementation of the basic block and the recursive step.

PyTorch Implementation of recursively defined residual block

```
class ConvAdaESA(nn.Module): # RDRB_0
    def __init__(self, channels):
        self.conv = conv(channels, channels, 3)
        self.esa  = ESA(channels)
        self.act  = activation('lrelu')
        self.norm = BatchNorm2d(channels)
        self.phi  = conv(1, 1, 1)
        self.level = 0

    def forward(self, x):
        s   = torch.std(x, dim=[1,2,3], keepdim=True)
        out = self.act(self.conv(self.norm(x)))       # BN + Conv + Act
        out = out * torch.exp(self.phi(torch.log(s))) # AdaDM
        out = self.esa(out+x)                         # Skip + ESA
        return out

def build_RDRB(block):
    class RDRB_base(nn.Module):
        def __init__(self, channels):
            self.block1 = block(channels)
            self.block2 = block(channels)
            self.conv   = conv(channels*2, channels, 1)
            self.esa    = ESA(channels)
            self.act    = activation('lrelu')
            self.level  = self.block1.level + 1
            if self.level == 3:
                self.NLSA = NonLocalSparseAttention(channels=channels)

        def forward(self, x):
            out1 = self.block1(x)
            out2 = self.block2(out1)
            out  = torch.cat([out1,out2],dim=1)
            out  = self.conv(out) + x
            out  = self.esa(self.act(out))
            if self.level == 3:
                out = self.NLSA(out)
            return out
    return RDRB_base
```

4 Results on DIV2K validation dataset

We show results for classical SR on validation DIV2K dataset in Table 2. We employ peak signal-to-noise ratio (PSNR) and structural similarity (SSIM) [7] to measure the quality of super-resolved images. All SR results are evaluated on RGB channels and on Y channel after color space transformation from RGB to YCbCr.

Model	RDRN				RDRN+			
Metric	Y channel		RGB channels		Y channel		RGB channels	
	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM
x2	36.98	0.9512	35.47	0.9437	37.03	0.9515	35.52	0.9440
x3	33.16	0.8985	31.70	0.8855	33.22	0.8992	31.76	0.8864
x4	31.11	0.8515	29.66	0.8344	31.17	0.8525	29.73	0.8354
x8	27.32	0.7333	25.89	0.7059	27.42	0.7356	25.99	0.7083

Table 2. Quantitative results of SR with BI degradation model on DIV2K validation dataset.

References

1. Anwar, S., Barnes, N.: Densely residual laplacian super-resolution. IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI) (2020)
2. Dai, T., Cai, J., Zhang, Y., Xia, S.T., Zhang, L.: Second-order attention network for single image super-resolution. In: CVPR (2019)
3. Dong, C., Loy, C.C., He, K., Tang, X.: Learning a deep convolutional network for image super-resolution. In: ECCV (2014)
4. Liang, J., Cao, J., Sun, G., Zhang, K., Gool, L.V., Timofte, R.: Swinir: Image restoration using swin transformer. arXiv preprint arXiv:2108.10257 (2021)
5. Liu, J., Zhang, W., Tang, Y., Tang, J., Wu, G.: Residual feature aggregation network for image super-resolution. In: CVPR. pp. 2356–2365. IEEE (2020)
6. Niu, B., Wen, W., Ren, W., Zhang, X., Yang, L., Wang, S., Zhang, K., Cao, X., Shen, H.: Single image super-resolution via a holistic attention network. In: ECCV (12). Lecture Notes in Computer Science, vol. 12357, pp. 191–207. Springer (2020)
7. Wang, Z., Bovik, A.C., Sheikh, H.R., Simoncelli, E.P.: Image quality assessment: from error visibility to structural similarity. IEEE Trans. Image Processing **13**(4), 600–612 (2004)
8. Zhang, Y., Li, K., Li, K., Wang, L., Zhong, B., Fu, Y.: Image super-resolution using very deep residual channel attention networks. In: ECCV (2018)
9. Zhang, Y., Tian, Y., Kong, Y., Zhong, B., Fu, Y.: Residual dense network for image super-resolution. In: CVPR (2018)

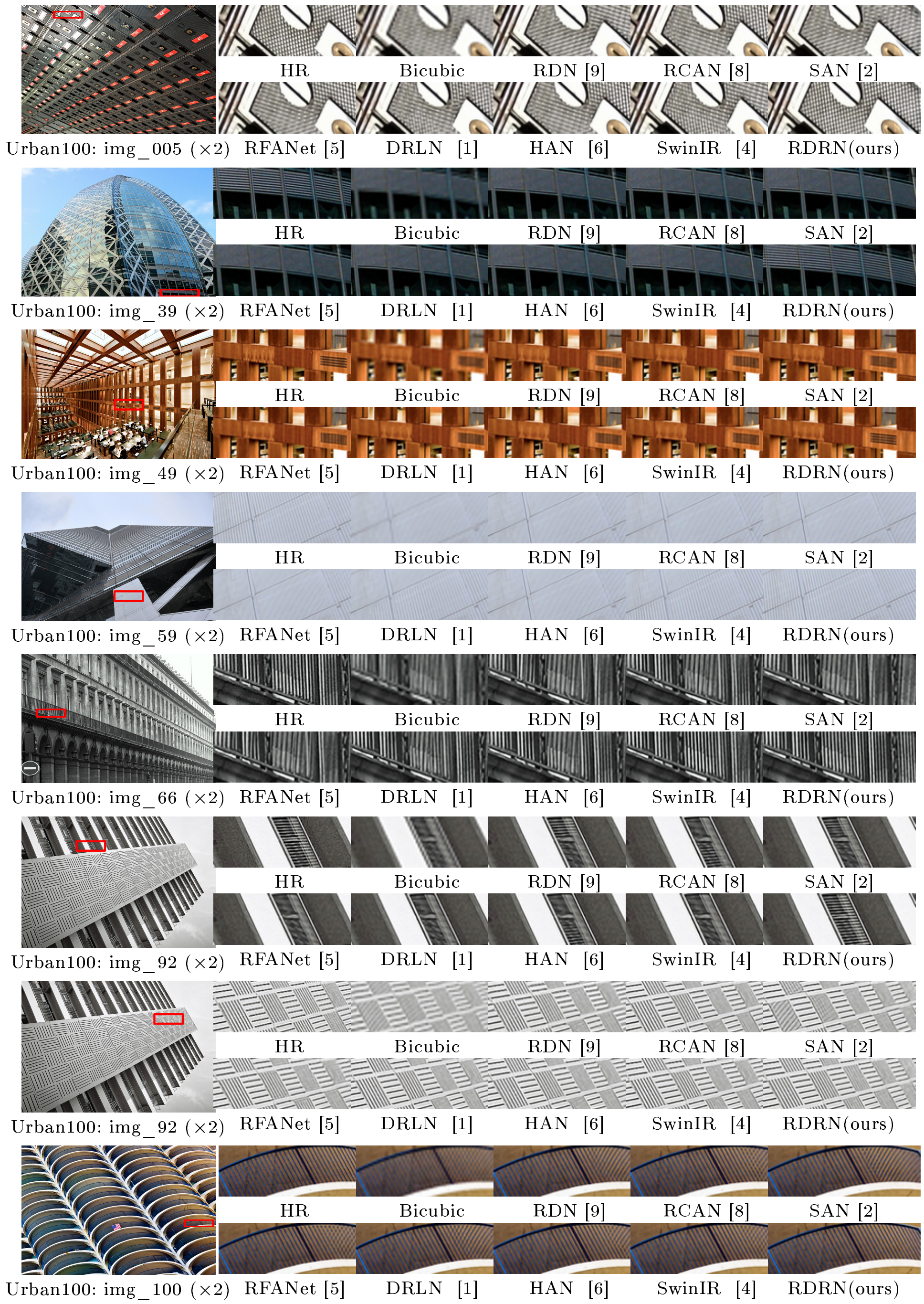

Fig. 1. Visual comparison for 2× SR with BI model.

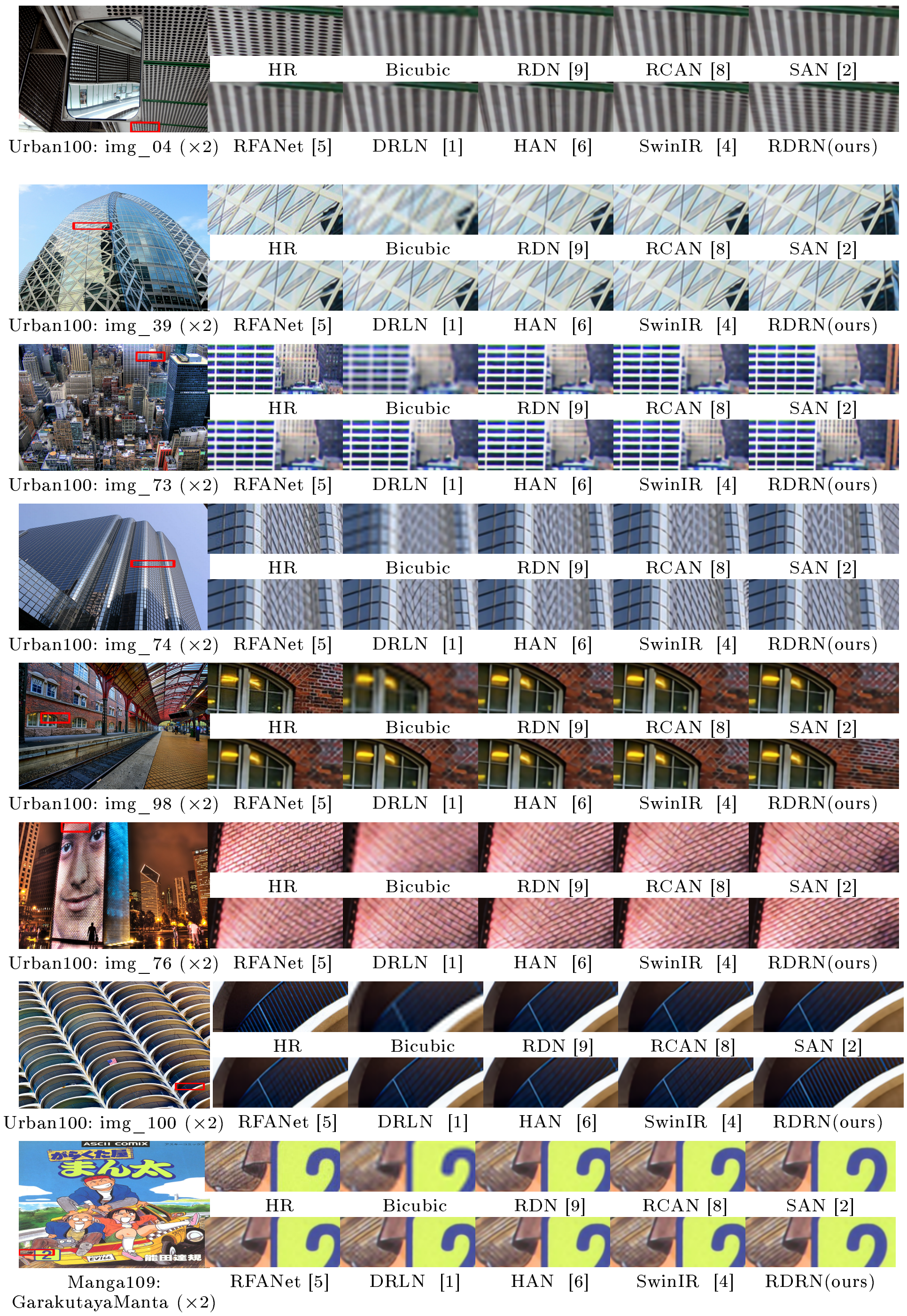

Fig. 2. Visual comparison for 3× SR with BI model.

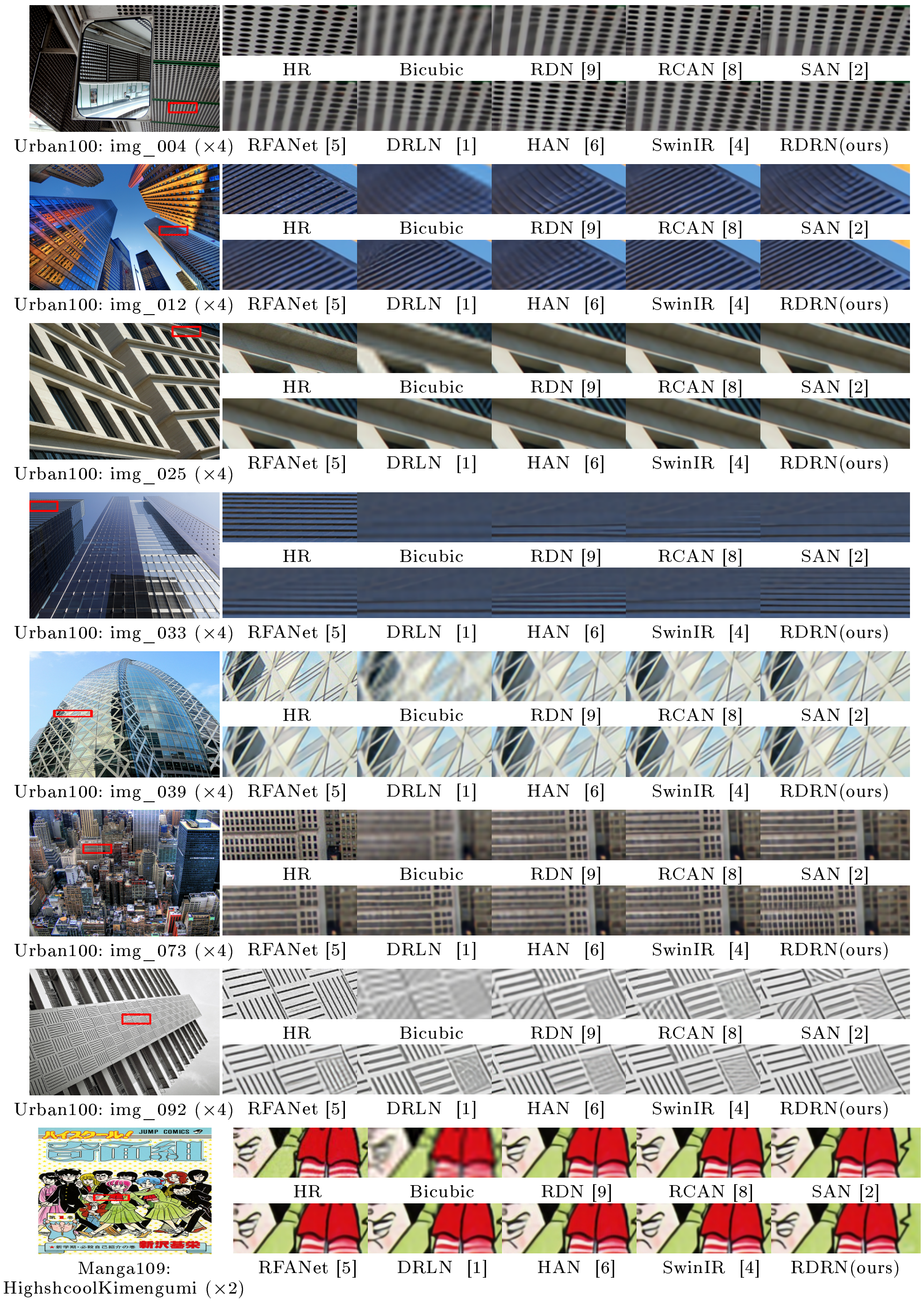

Fig. 3. Visual comparison for 4× SR with BI model.

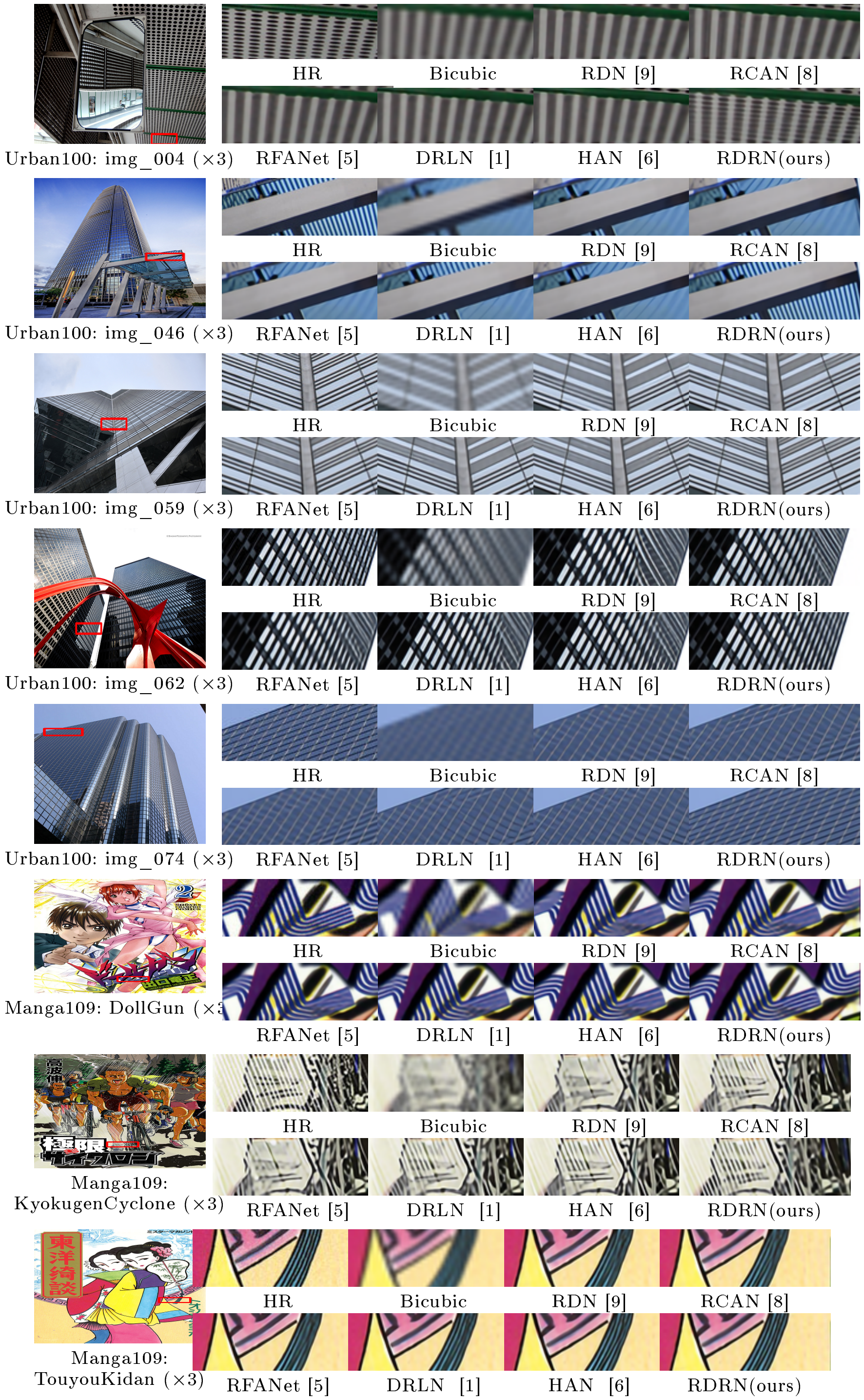

Fig. 4. Visual comparison for 3× SR with BD model.