OSRT: Omnidirectional Image Super-Resolution with Distortion-aware Transformer
–Supplementary File–

Fanghua Yu[1*] Xintao Wang[2*] Mingdeng Cao[2,3] Gen Li[4] Ying Shan[2] Chao Dong[1,5†]
[1]ShenZhen Key Lab of Computer Vision and Pattern Recognition
Shenzhen Institute of Advanced Technology, Chinese Academy of Sciences
[2]ARC, Tencent PCG [3]The University of Tokyo
[4]Platform Technologies, Tencent Online Video [5]Shanghai Artificial Intelligence Laboratory

Due to the lack of space in the main paper, we provide more details of the proposed OSRT in the supplementary file. In Sec. 1, we show the transformation relationships from the uniformed sphere to various projection types (ERP, Fisheye, and Perspective) and the derivation processes of each projection type. More experimental details and interpretations can be found in Sec. 2. Then we provide additional visual comparisons and visualizations under various projection types in Sec. 3.

1. Geometric Relationship

In this section, x_E, y_E and x_P, y_P refer to plane coordinates of ERP and Perspective, respectively. For an ideal sphere, θ_S, φ_S are the spherical coordinates, and x_S, y_S, z_S are the space coordinates. ρ_F, θ_F and x_F, y_F are polar coordinates and plane coordinates of Fisheye, respectively.

1.1. Transformation

ERP. For ERP, the coordinate is defined as:

$$\begin{cases} x_E = \theta_S \\ y_E = \varphi_S. \end{cases} \tag{1}$$

Fisheye. For Fisheye, the coordinate is defined as:

$$\begin{cases} \rho_F = 2 \times \arctan(\sqrt{x_S^2 + y_S^2/z_S^2})/A_F \\ \theta_F = \arctan(y_S/x_S) \\ x_S = \rho_F \times \cos(\theta_F) \\ y_S = \rho_F \times \sin(\theta_F), \end{cases} \tag{2}$$

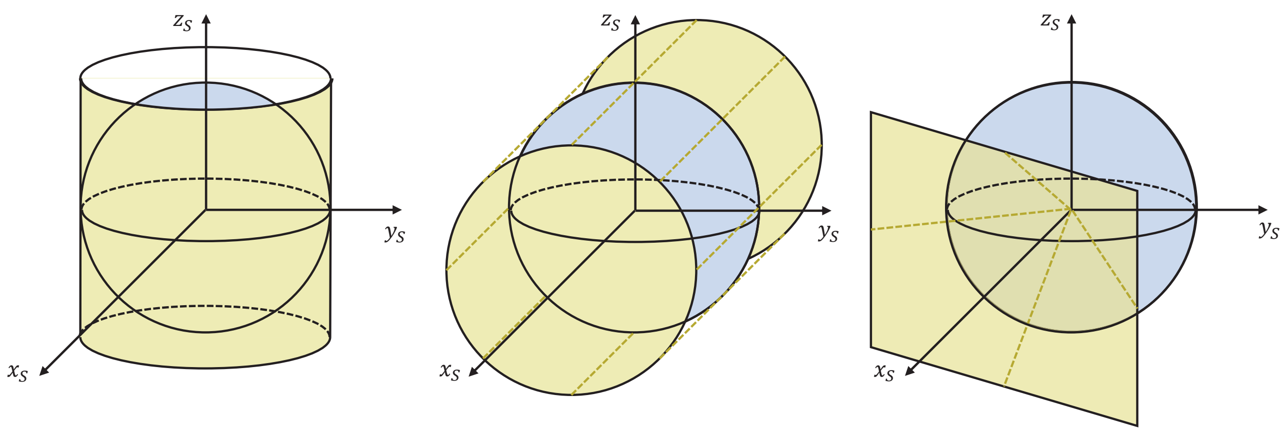
where A_F is the aperture degree of Fisheye. Specifically, when the normal vector of the Fisheye splicing plane is parallel to the z-axis, Eq. (2) can be simplified as:

$$\begin{cases} \rho_F = 2 \times (\pi/2 - \varphi_S)/A_F \\ \theta_F = \theta_S. \end{cases} \tag{3}$$

Here, we define a rotation transformation under the spherical coordinates:

$$[x_S^*, y_S^*, z_S^*]^T = M_r \cdot [x_S, y_S, z_S]^T, \tag{4}$$

where M_r is the 3D rotation matrix. $[x_S, y_S, z_S]^T$ and $[x_S^*, y_S^*, z_S^*]^T$ are the original and rotated spherical coordinates, respectively. Eq. (4) is defined to align general Fisheye to the horizontally spliced one, which is identical to add $\Delta\theta_r, \Delta\varphi_r$ on spherical polar coordinates.



(a) ERP (b) Fisheye (c) Perspective

Figure 1. Geometric illustration of three projection types. Blue and yellow refer to the spherical surface and projection plane, respectively.

Perspective. The coordinates is defined as:

$$\begin{cases} x_P = \tan(\theta_S) \\ y_P = \tan(\varphi_S)/cos(\theta_S), \end{cases} \tag{5}$$

where $x_P, y_P \in [-\tan(A_P/2), \tan(A_P/2)]$. A_P is the aperture degree of Perspective, which determines the field-of-view (FOV) of the given Perspective. Note that a perspective image only represents information on a partial area of a spherical surface.

1.2. Distortion

As mentioned in the main paper, the distortion degree of each projection type is measured by [5]:

$$\mathbf{K}(x,y) = \frac{\delta S(\theta,\varphi)}{\delta P(x,y)} = \frac{\cos(\varphi)|d\theta d\varphi|}{|dxdy|} = \frac{\cos(\varphi)}{|J(\theta,\varphi)|}, \tag{6}$$

where $\delta S(\cdot,\cdot)$ and $\delta P(\cdot,\cdot)$ represent the area on the spherical surface and the projection plane, respectively. $|didj|$ represents a plane microunit. $|J(\theta,\varphi)|$ is the Jacobian determinant from spherical coordinate to projection coordinate.

ERP distortion. From Eqs. (1) and (6), ERP stretching ratio can be derived as:

$$\mathbf{K}_{\text{ERP}}(x_E, y_E) = \cos(\varphi_S) = \cos(y_E). \tag{7}$$

Fisheye distortion. In this paragraph, we denote A_F as π. $|J_F^*(\theta_S,\varphi_S)|$ can be simplified by Eq. (3):

$$\begin{aligned} &|J_F^*(\theta_S,\varphi_S)| \\ =& \begin{vmatrix} \frac{\partial(x_F)}{\partial(\theta_S)} & \frac{\partial(x_F)}{\partial(\varphi_S)} \\ \frac{\partial(y_F)}{\partial(\theta_S)} & \frac{\partial(y_F)}{\partial(\varphi_S)} \end{vmatrix} \\ =& \begin{vmatrix} \frac{\partial(\rho_F\cos\theta_F)}{\partial(\theta_S)} & \frac{\partial(\rho_F\cos\theta_F)}{\partial(\varphi_S)} \\ \frac{\partial(\rho_F\sin\theta_F)}{\partial(\theta_S)} & \frac{\partial(\rho_F\sin\theta_F)}{\partial(\varphi_S)} \end{vmatrix} \\ =& \begin{vmatrix} \frac{\partial((1-2\varphi_S/\pi)\cos\theta_S)}{\partial(\theta_S)} & \frac{\partial((1-2\varphi_S/\pi)\cos\theta_S)}{\partial(\varphi_S)} \\ \frac{\partial((1-2\varphi_S/\pi)\sin\theta_S)}{\partial(\theta_S)} & \frac{\partial((1-2\varphi_S/\pi)\sin\theta_S)}{\partial(\varphi_S)} \end{vmatrix} \\ =& \begin{vmatrix} -(1-2\varphi_S/\pi)\sin\theta_S & -2\cos\theta_S/\pi \\ (1-2\varphi_S/\pi)\cos\theta_S & -2\sin\theta_S/\pi \end{vmatrix} \\ =& \frac{2}{\pi}(1-2\varphi_S/\pi)(\sin^2\theta_S + \cos^2\theta_S) \\ =& \frac{2}{\pi}\rho_F. \end{aligned} \tag{8}$$

From Eqs. (3), (6) and (8), the stretching ratio of horizontally spliced Fisheye can be derived as:

$$\begin{aligned} \mathbf{K}^*_{\text{Fisheye}}(x_F,y_F) &= \frac{\cos(\varphi_S)}{|J_F(\theta_S,\varphi_S)|} \\ &= \frac{\cos(\frac{\pi}{2}(1-\rho_F))}{\frac{2}{\pi}\rho_F}. \end{aligned} \tag{9}$$

Then, we can derive stretching ratio of general Fisheye from Eqs. (6), (8) and (9):

$$\begin{aligned} \mathbf{K}_{\text{Fisheye}}(x_F,y_F) &= \frac{\delta S(\theta_S,\varphi_S)}{\delta P(x_F,y_F)} \\ &= \underbrace{\frac{\delta S(\theta_S^*,\varphi_S^*)}{\delta P(x_F,y_F)}}_{\text{Projection}} \cdot \underbrace{\frac{\delta S(\theta_S,\varphi_S)}{\delta S(\theta_S^*,\varphi_S^*)}}_{\text{Rotation}} \\ &= \mathbf{K}^* \cdot \frac{\cos(\varphi_S)|d\theta_S d\varphi_S|}{\cos(\varphi_S^*)|d\theta_S^* d\varphi_S^*|} \\ &= \mathbf{K}^* \cdot \frac{\cos(\varphi_S^* + \Delta\varphi_r)}{\cos(\varphi_S^*)} \\ &= \frac{\cos(\frac{\pi}{2}(1-\rho_F) - \Delta\varphi_r)}{\frac{2}{\pi}\rho_F}, \end{aligned} \tag{10}$$

where $\Delta\varphi_r$ is a constant, which is determined by the angle between the normal vector of splicing plane and z-axis.

Perspective. From Eqs. (5) and (6), the Perspective stretching ratio can be derived as:

$$\begin{aligned} \mathbf{K}_{\text{Perspective}}(x_P,y_P) &= \frac{\cos(\varphi_S)}{|J_P(\theta_S,\varphi_S)|} \\ &= cos^3(\theta_S)cos^3(\varphi_S) \\ &= (1 + x_P^2 + y_P^2)^{-\frac{3}{2}}. \end{aligned} \tag{11}$$

2. Details and Discussions

2.1. Data Cleaning on ODI Dataset

Except for ERP downsampling, we still find other issues in both ODI-SR and SUN360 datasets. Previous datasets are downsampled by bicubic function without anti-alias design (OpenCV-Python), which introduces mottled artifacts (Fig. 2). Meanwhile, they are stored in the format of JPEG, which leads to missing details and JPEG-blocking artifacts. Storing HR images in JPEG format is harmful for both training and evaluation. To tackle these issues, we propose to apply downsampling by anti-aliased bicubic function (Pillow) and store images in a lossless format (PNG). Moreover, there are problematic ODIs in previous datasets: **1**) transforming mistakes; **2**) virtual scenarios; **3**) extremely low qualities; **4**) plane images. Consequently, we propose ODI-SR-clean and SUN360-clean datasets, the differences are shown in Tab. 1. We train and test all models on cleaned

	Original	Cleaned
Num of images in ODI-SR (training)	1200	1150
Num of images in ODI-SR (testing)	100	100
Num of images in ODI-SR (validation)	100	97
Num of images in SUN360	100	100
Downsampling function	OpenCV	Pillow
Downsampling target	ERP	Dual Fisheye
Storage format	JPEG	PNG

Table 1. Differences between the original and cleaned datasets.

Backbone network	Datasets	Training scheme	Scale	ODI-SR		SUN360	
				PSNR	SSIM	PSNR	SSIM
SwinIR	ODI-SR	N/A	×2	30.52	0.8819	31.21	0.8852
SwinIR	DF2K/ODI-SR	one-stage		30.59	0.8810	31.26	0.8841
SwinIR	DF2K-ERP/ODI-SR	one-stage		30.64	0.8821	31.33	0.8855
SwinIR	DF2K-ERP/ODI-SR	two-stage		30.54	0.8797	31.17	0.8818
OSRT	DF2K-ERP/ODI-SR	one-stage		**30.77**	**0.8846**	**31.52**	**0.8888**
SwinIR	ODI-SR	N/A	×4	27.12	0.7663	27.39	0.7707
SwinIR	DF2K/ODI-SR	one-stage		27.24	0.7708	27.59	0.7768
SwinIR	DF2K-ERP/ODI-SR	one-stage		27.31	0.7735	27.71	0.7804
SwinIR	DF2K-ERP/ODI-SR	two-stage		27.33	0.7725	27.74	0.7795
OSRT	DF2K-ERP/ODI-SR	one-stage		**27.41**	**0.7762**	**27.84**	**0.7835**

Table 2. Ablation study on data augmentation.

Method	Scale	ODI-SR		SUN 360 Panorama	
		PSNR	SSIM	PSNR	SSIM
RCAN [8]	×2	30.08	0.8723	30.56	0.8712
RCAN-local [1]		**30.28**	**0.8735**	**30.80**	**0.8740**
RCAN [8]	×4	26.85	0.7621	27.10	0.7660
RCAN-local [1]		**26.99**	**0.7622**	**27.24**	**0.7665**

Table 3. Influence of test-time local converter.

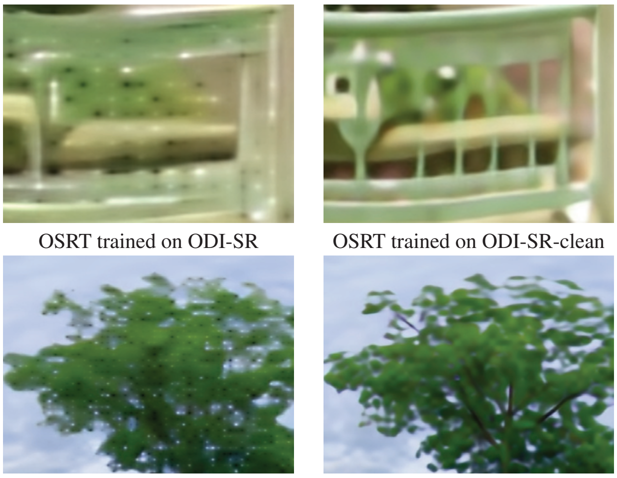

OSRT trained on ODI-SR
OSRT trained on ODI-SR-clean
OSRT trained on ODI-SR
OSRT trained on ODI-SR-clean

Figure 2. Visual comparisons of ×8 SR results trained and tested on the original and cleaned datasets.

datasets except the comparison under ERP downsampling (Sec. 4.3 in the main paper).

When comparing SR results under ERP downsampling, we train and test models on original datasets, which is identical to previous methods. Thus we can directly compare the SR results of OSRT with SR results reported by previous methods, *e.g.*, LAU-Net [2] and SphereSR [7].

2.2. Instability of RCAN

For RCAN [8] trained with Fisheye downsampling, the training process is unstable and thus the performance is degraded. We find that the instability of RCAN is caused by incompatibility between the channel attention block (CAB) and Fisheye downsampling. CAB requires global statistical features, and its training stability depends on the consistent mean value distribution of each patch [1]. However, when Fisheye downsampling is applied to an ERP image, the ERP image suffers from nonuniform downsampling, which directly increases the mean value diversity between patches. Although implementing a test-time local converter (TLC [1]) can reduce the distribution gap between the patch and the whole image (Tab. 3), it cannot reduce the distribution gap within patches. Consequently, while training ODISR models under Fisheye downsampling, blocks that require global statistical values are not recommended.

2.3. Full Ablation Results of Data Augmentation

Due to the lack of space in the main paper, we only show partial ablation results of data augmentation strategies (Tab. 4). The full results are shown in Tab. 2. Compared with fine-tuning on DF2K-ERP pre-trained models (two-stage), training on two datasets jointly (one-stage) shows better results. Moreover, the advantage of OSRT is enlarged when additional training patches are applied.

2.4. Domain Gap between Real and Pseudo ODIs

As mentioned in the main paper (Sec. 3.4), we synthesize pseudo ERP training data (DF2K-ERP) from the plain images to alleviate the over-fitting problem of large networks. Although DF2K-ERP has shown obvious benefits, there is still a domain gap between real and pseudo images. From Eq. (11), we can see that the distortion degree of Perspective is determined by the distance from the center. As the projection range is determined by FOV degree, perspective images with different FOV degrees suffer inconsistent distortions. However, we cannot obtain the distribution of FOV degrees in real-world scenarios. Thus we directly assume that all pseudo perspective images have a fixed FOV degree of 90°, which introduces a domain gap. While the inevitably domain gap is a limitation of DF2K-ERP, it still overcomes the over-fitting issue and improves the reconstruction ability.

3. Visualization

As mentioned in the main paper (Sec. 3.2), ERP downsampling leads to unrealistic ODIs. Thus we only show visualizations based on Fisheye downsampling in this section.

Additional qualitative comparison. We provide additional visual comparisons with other methods on the ODI-SR-clean testing dataset and SUN360-clean dataset in

Fig. 3. Reconstructed ERP images are compared under ERP, Fisheye, and Perspective. As shown in Fig. 3 (d) and (f), we can see that OSRT can reconstruct sharp and accurate boundaries. Besides, from Fig. 3 (a) and (c), we conclude that OSRT is skilled at reconstructing rigid textures.

Additional visualization of OSRT. To show the overall quality of OSRT reconstructed images, we project these ERP images to arbitrary projection types. Figs. 4 to 6 depict visualizations of $\times 2$, $\times 4$ and $\times 8$ SR results, respectively. Under all projection types, OSRT can reconstruct details with high fidelity (buildings in Fig. 4, tiles in Fig. 5, and grasses in Fig. 6).

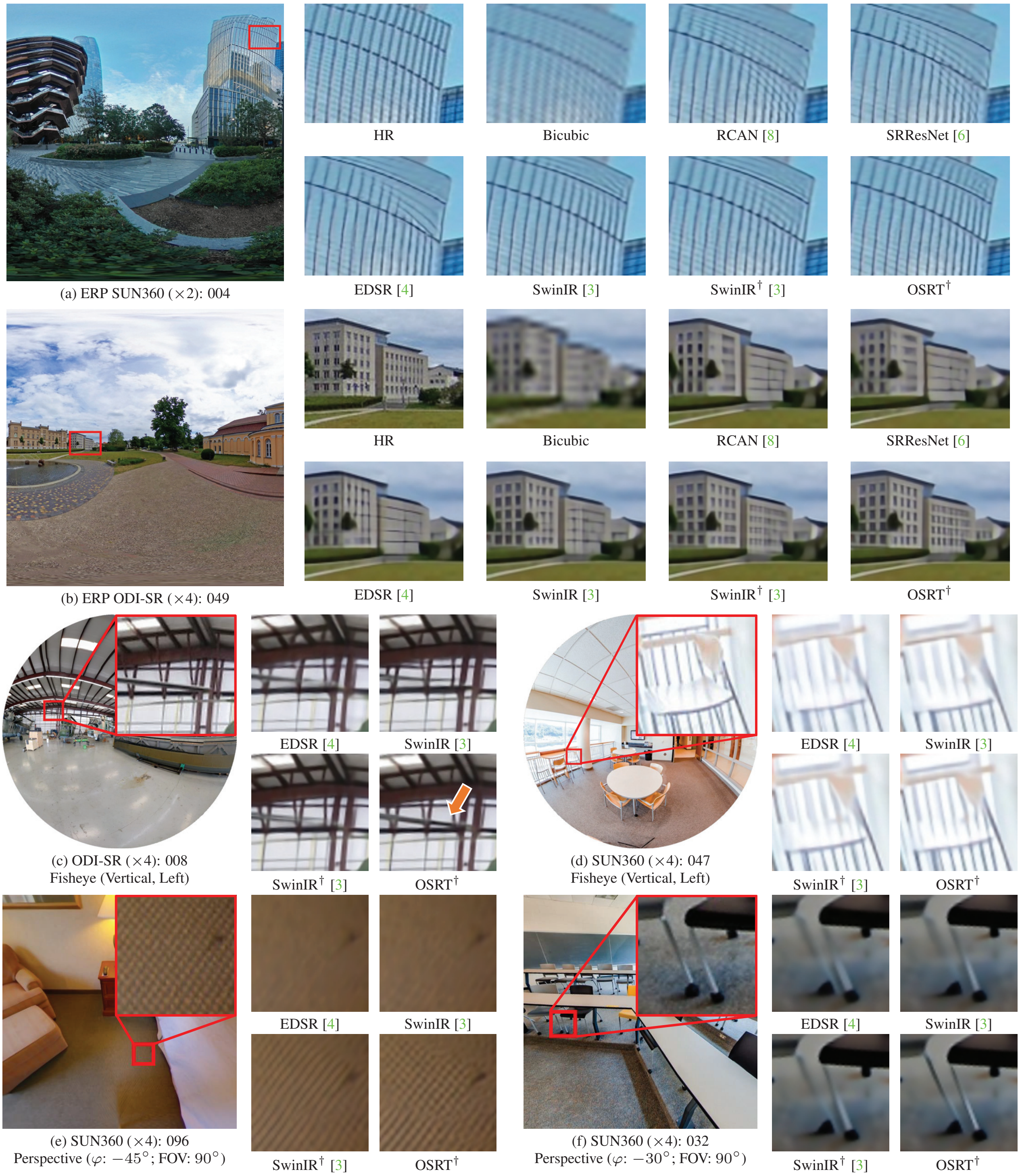


Figure 3. Visual comparisons of SR results under Fisheye downsampling. † denotes applying DF2K-ERP as augmented dataset.

Figure 4. Visualization of ×8 SR results (SUN360-062).

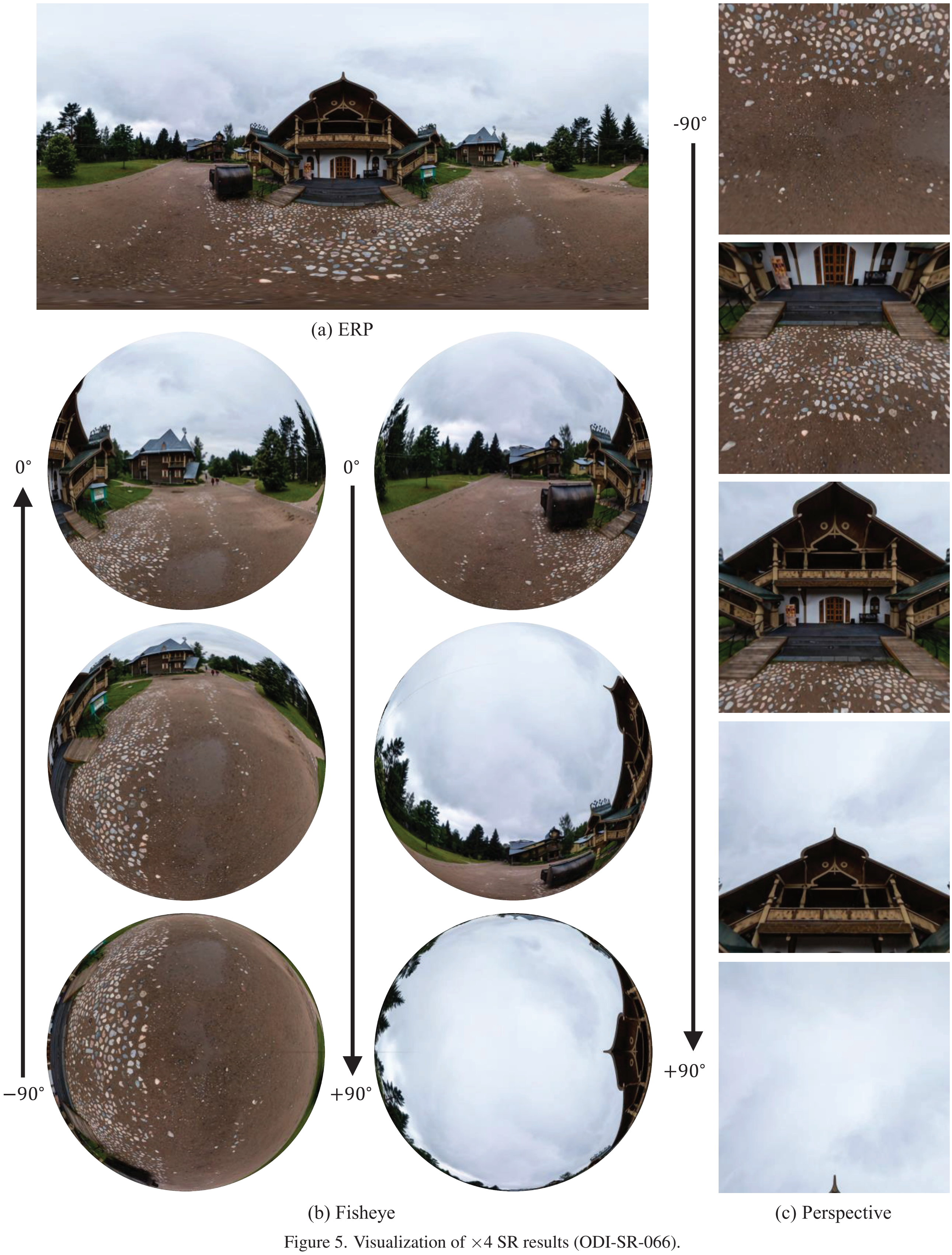


Figure 5. Visualization of $\times 4$ SR results (ODI-SR-066).

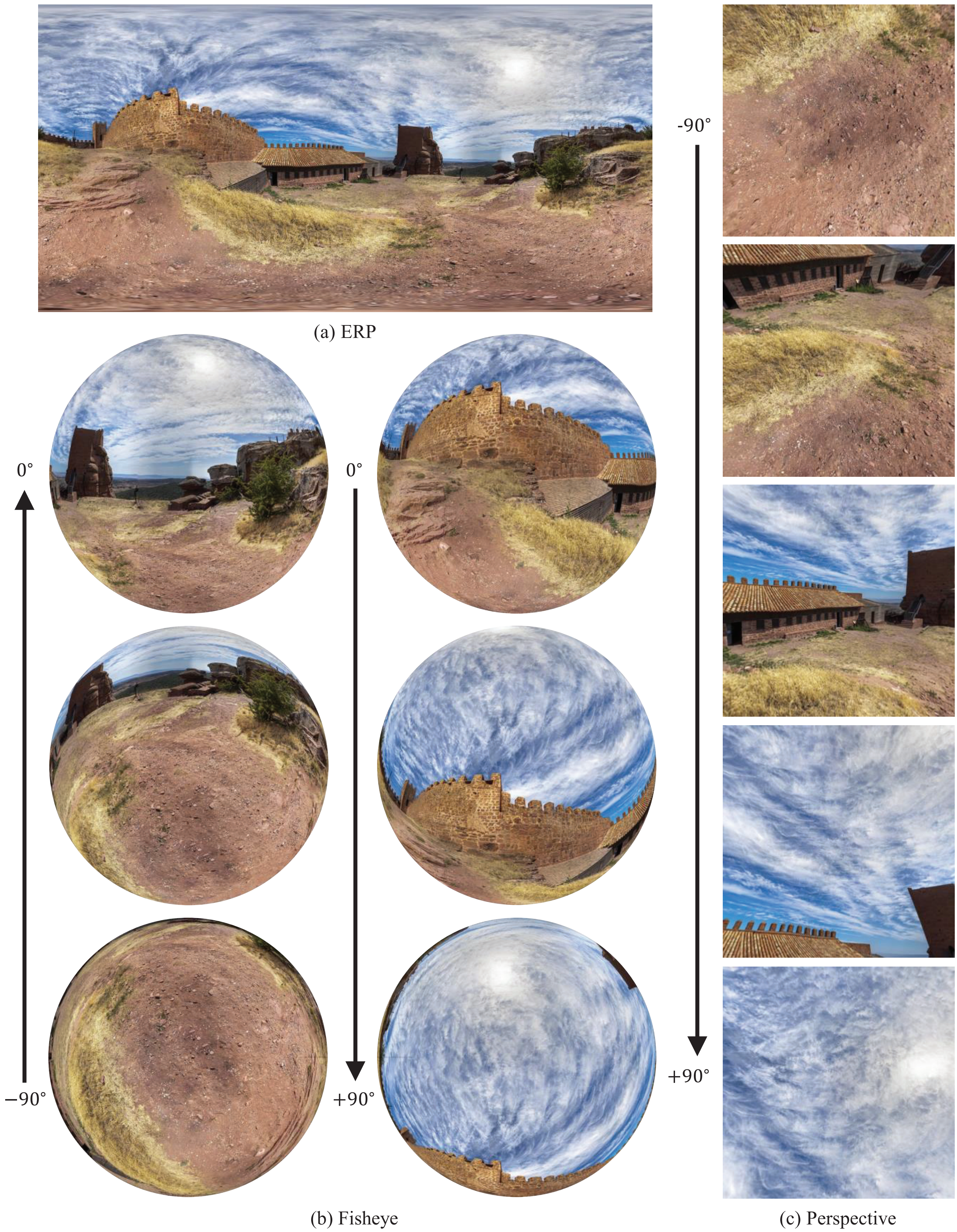


Figure 6. Visualization of ×2 SR results (SUN360-007).

References

[1] Xiaojie Chu, Liangyu Chen, Chengpeng Chen, and Xin Lu. Improving image restoration by revisiting global information aggregation. In *European Conference on Computer Vision*, pages 53–71. Springer, 2022. 3

[2] Xin Deng, Hao Wang, Mai Xu, Yichen Guo, Yuhang Song, and Li Yang. Lau-net: Latitude adaptive upscaling network for omnidirectional image super-resolution. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9189–9198, 2021. 3

[3] Jingyun Liang, Jiezhang Cao, Guolei Sun, Kai Zhang, Luc Van Gool, and Radu Timofte. Swinir: Image restoration using swin transformer. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 1833–1844, 2021. 4

[4] Bee Lim, Sanghyun Son, Heewon Kim, Seungjun Nah, and Kyoung Mu Lee. Enhanced deep residual networks for single image super-resolution. In *Proceedings of the IEEE conference on computer vision and pattern recognition workshops*, pages 136–144, 2017. 4

[5] Yule Sun, Ang Lu, and Lu Yu. Weighted-to-spherically-uniform quality evaluation for omnidirectional video. *IEEE signal processing letters*, 24(9):1408–1412, 2017. 2

[6] Xintao Wang, Ke Yu, Shixiang Wu, Jinjin Gu, Yihao Liu, Chao Dong, Yu Qiao, and Chen Change Loy. Esrgan: Enhanced super-resolution generative adversarial networks. In *Proceedings of the European conference on computer vision (ECCV) workshops*, pages 0–0, 2018. 4

[7] Youngho Yoon, Inchul Chung, Lin Wang, and Kuk-Jin Yoon. Spheresr: 360deg image super-resolution with arbitrary projection via continuous spherical image representation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5677–5686, 2022. 3

[8] Yulun Zhang, Kunpeng Li, Kai Li, Lichen Wang, Bineng Zhong, and Yun Fu. Image super-resolution using very deep residual channel attention networks. In *Proceedings of the European conference on computer vision (ECCV)*, pages 286–301, 2018. 3, 4