Pairwise Distance Distillation for Unsupervised Real-World Image Super-Resolution Supplementary Material

Yuehan Zhang[1], Seungjun Lee[2], and Angela Yao[1]

[1] National University of Singapore
{zyuehan,ayao}@comp.nus.edu.sg
[2] Korea University
9penguin9@korea.ac.kr

The supplementary features the following sections:

- Training details in Sec. A.
- Details of low-level characteristics change in Sec. B.
- Comparison with StableSR in Sec. C.
- Supplemental ablations in Sec. D:
  - Sec. D.a: Perception-distortion trade-off related to R_{intra} and R_{inter}.
  - Sec. D.b: Influence of degradation for synthesizing labeled data.
  - Sec. D.c: Importance of Gram Matrix in R_{inter}.
  - Sec. D.d: Rationale for using VGG features for distance calculation.
- Details of color correction in Sec. E.
- Visual comparisons in Sec. F.
- Limitations in Sec. G.

A Training Details

Balance of Loss Terms. Here, we specify the weights of terms in Eq. (10), Eq. (11) and Eq. (12) in the main paper. The weights α_{1-3} in Eq. (10) are to balance the objectives in supervised training for labeled data, $\{\omega_i\}_{i=0,1,2,3}$ in Eq. (11) are for balancing the focus of ℓ_1 loss on wavelet channels {LL,HL,LH,HH} accordingly, and λ_{1-3} of Eq. (12) balance the optimization aims in unsupervised training. The values of α_{1-3} and $\{\omega_i\}_{i=0,1,2,3}$ are shown in Tab. 4, and we use $\lambda_3 = 0.01$ in Eq. (12). The discussion of λ_{1-2} is in Sec. D.a.

Table 4: Weights (α_{1-3}) of loss terms in Eq. (10) and weights (λ_{0-3}) for wavelet channels in Eq. (11). We use the same weight 0.1 for channel {HL,LH}.

	α_1	α_2	α_3	ω_0	ω_{1-2}	ω_3
Value	0.5	1.0	0.01	1.0	0.1	0.001

Choice of VGG19 feature layers. In the main paper, we use VGG19 to extract feature maps from output high-resolution images for the calculation of

R_{intra} and R_{inter} in Eq. (6) and Eq. (8). In Tab. 5, we provide the choices of Φ_{ij} and in two equations the weights of them. We use multi-level features to capture more comprehensive representations. We choose slightly deeper feature maps for R_{intra} than R_{inter} to focus more on content-related information.

Table 5: Choices of VGG19 feature layers for distance calculation of R_{intra} and R_{inter} and layer weights respectively.

R_{intra}	conv2_2	conv3_4	conv4_4	conv5_4
Weights	0.1	0.3	1.0	1.0
R_{inter}	conv2_1	conv3_1	conv4_1	conv5_1
Weights	0.25	0.25	0.25	0.25

Others. The EMA implementation of our method requires a warm-up process to create an initial difference between the generalist and specialist predictions before optimizing for R_{intra} and R_{inter}. As such, the model is first trained with the supervised loss in Eq. (10) only for k iterations. For all experiments in the main paper, we set $k = 5$.

Computation cost. All methods in the main paper are trained with two RTX A5000 GPUs. The Naive Distillation (ND) requires ∼11 hours for training and the EMA version of our method requires ∼14 hours respectively. Our method, as a learning framework, does not affect inference speed. The inference time depends on the chosen model.

B Low-Level Characteristics Change

We note that visualizations and KL Divergence are tools to provide evidence for closer low-level characteristics rather than evaluate the model performance.

Fig. 5 of the main paper visualizes distributions of low-level characteristics following [3]. Specifically, we exact feature maps represented by an SRGAN [2] model pretrained with bicubic-interpolated data. Such feature maps are called Deep Degradation Representations in [3], which contain indicative information for low-level characteristics such as blurriness or sharpness. In our paper, we call them low-level features to avoid ambiguous reuse of the word "degradation". For visualization, we split low-level features into 50×50 patches and project flattened patch features to 50-dimensional vectors through PCA. We further visualize the projected features in the 2D plane through t-SNE.

As stated in [3], visualization is far from perfect. When real-world and synthetic datasets have very different spatial layouts, it is hard to exhibit meaningful distribution shifts of low-level characteristics through visualization. Thus, we calculate the KL Divergence between low-level features to show how the difference between distributions changes quantitatively. Fig.6 in the main paper shows the effect of our EMA version. The specialist M_S is initialized with generalized RealESRGAN and gets specialization for bicubic interpolation through

supervised training on labeled synthetic data. We use unlabeled real-world predictions from pretrained M_S and M_S after adaptation. We use labeled synthetic predictions from adapted M_S, featuring high image quality. We calculate the KL Divergence between the labeled and unlabeled predictions before or after adaptation. The extraction of low-level features follows the same procedure as in Fig.5.

In Fig. 10, we supplement the KL Divergence comparisons for the static version of our method, which also shows a closer distribution after adaptation.

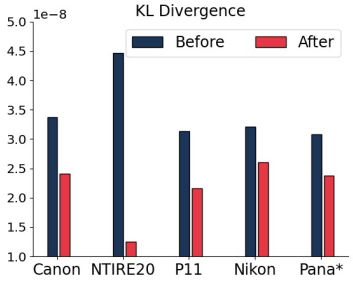


Fig. 10: KL Divergence change brought by the static version of our method. The KL Divergence decreases after applying our method (red bars).

(a) LR **(b)** StableSR **(c)** Ours

Fig. 11: Visual comparison to StableSR.

C Comparison with StableSR

Tab. 2 in the main paper lists the existing state-of-the-art methods that are able for direct comparisons. Here we compare with StableSR [4], a recent diffusion model for RWSR. We did not include StableSR in the main paper for two reasons:

1. StableSR uses a much higher number of parameters than our model (152.7M versus 16.7M). The significantly larger model capacity makes the comparison unfair;
2. The testing settings in the paper of StableSR require a restricted input size, which is different from the standard settings we follow in the main paper.

We test StableSR on the RealSR dataset following the standard testing settings. We follow the official testing script[3] to make predictions for *full* input images with arbitrary spatial size. As shown in Tab. 6, we offer evaluation metrics used in Tab. 2 together with that in the StableSR paper. Our methods achieve competitive performances despite a nearly ten times lighter model size,

[3] https://github.com/IceClear/StableSR

outscoring StableSR in four out of six evaluation metrics. Comparing visually, our method predicts fewer hallucinations and clearer borders, as shown in Fig. 11.

Besides, we emphasize that our method provides a novel learning perspective for unsupervised RWSR instead of designing new model architectures, such as diffusion-based models.

Table 6: Quantitative comparison with StableSR following standard test settings. The best scores are **bold**. Our method scores better on four of six metrics for each tested dataset.

Dataset	Canon					
Metrics	PSNR↑	SSIM↑	LPIPS↓	NRQM↑	CLIP-IQA↑	MUSIQ↑
StableSR	24.84	0.7584	**0.2431**	5.8396	**0.6181**	61.75
Ours	**26.13**	**0.7626**	0.2517	**6.1323**	0.5532	**62.67**
Dataset	Nikon					
Metrics	PSNR↑	SSIM↑	LPIPS↓	NRQM↑	CLIP-IQA↑	MUSIQ↑
StableSR	24.9	**0.7376**	0.2742	5.5872	**0.6155**	57.94
Ours	**25.07**	0.7294	**0.2713**	**6.1276**	0.5739	**61.18**

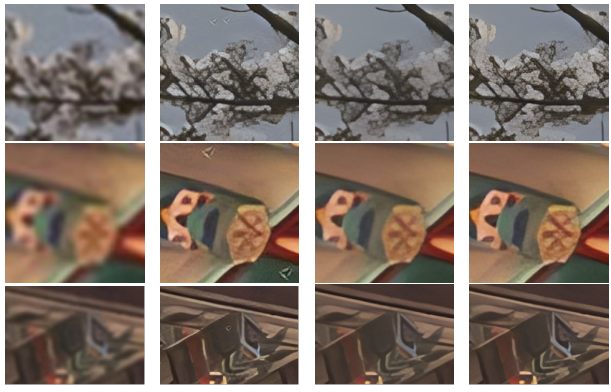
D Supplemental Ablations

D.a Perception-Fidelity Trade-Off

Ablation in Sec. 4.4 of the main paper shows the different preferences of R_{intra} and R_{inter} towards fidelity and perceptual quality, where we experiment on the EMA version of our method. Tab. 7 complements corresponding experiments on the static version of our method. We have similar observations as the EMA version experiments, where using one of R_{intra} or R_{inter} only yields high scores regarding fidelity (PSNR) or perception (NRQM). We further show visual examples in Fig. 12, where using both R_{inter} and R_{intra} can produce sharper patterns without artifacts.

Table 7: Ablation of R_{intra} and R_{inter} on static version of our method. The best and second best results are **bolded** and underlined.

Mth.	R_{intra}	R_{inter}	Canon		NTIRE20	
			PSNR	NRQM	PSNR	NRQM
a.	✓		**26.47**	5.0157	**25.89**	5.8333
b.		✓	24.81	**6.3381**	25.16	**6.4379**
Ours	✓	✓	<u>25.92</u>	<u>5.8479</u>	**25.89**	<u>6.2762</u>



(a) LR **(b)** R_{inter} **(c)** R_{intra} **(d)** Ours

Fig. 12: Visual examples for methods in Tab. 7. Using R_{inter} only yields artifact points, while R_{intra} only produce blurry patterns.

As we mentioned in the main paper, the ratio of weights for R_{intra} and R_{inter}, represented as $ra = \lambda_1/\lambda_2$, relates to a trade-off between fidelity and perceptual quality of the trained model, where $\lambda_{1,2}$ are weights from Eq. (12). Fig. 13 demonstrates examples of this trade-off phenomenon. With increasing proportion of R_{intra} ($ra = 0.1$ to $ra = 1$), the perceptual score (NRQM) decreases while the fidelity score (PSNR) increases.

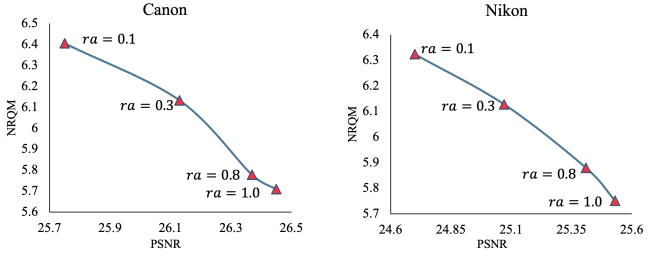


Fig. 13: Perception-fidelity trade-off experimented on the EMA version of our method, taking BSRGAN as the pre-trained model.

D.b Synthetic Degradation of X^L

Synthetic degradation of X^L is set to bicubic interpolation in the main paper, which is simple and excludes other degradations suspected to overlap with the degradation of unlabeled data. We explore the method's sensitivity to synthetic operation and substitute the bicubic interpolation by randomly selecting the downsampling kernel from bilinear, area, and bicubic with equal probability. As

shown in Tab. 8, changing the synthetic degradation to more complex interpolations will not eliminate the improvements over the pretrained generalist model brought by our method, while extending the type of synthetic degradations does slightly decrease the overall performance. Our method uses the high-quality predictions of synthetic data as references. We speculate that using inputs with more complex synthetic degradation will increase the difficulty of learning high-quality predictions. Thus, we finally choose degradation in $\{X^L\}$ to be a simple bicubic interpolation.

Table 8: Ablation of synthetic process for $\{X^L\}$. Comb. represents a random selection from a combination of {bicubic, area, bilnear}. The best and second best scores are **bolded** and underlined.

L_{ct}	Canon		NTIRE20		P11	
	PSNR	NRQM	PSNR	NRQM	PSNR	NRQM
Bicubic	<u>25.94</u>	<u>6.0647</u>	**26.42**	**6.2263**	**27.40**	**5.3722**
Comp.	**25.98**	5.9655	<u>25.97</u>	<u>6.1987</u>	<u>27.32</u>	<u>5.3720</u>
Generalist	24.74	**6.0649**	25.08	6.1213	26.73	5.2530

D.c Designs in R_{inter}

We provide ablation on the Gram Matrix in distance calculation for R_{inter}. Here, we skip the Gram Matrix and use the ℓ_1 differences between features as distances to enforce consistency. As shown in the last column of Tab. 9, skipping Gram Matrix yields a significant performance drop.

Table 9: Ablation on R_{inter}: Method ℓ_1 skips Gram Matrix calculation. Ablation on VGG: Alex. replaces VGG with AlexNet.

	Ours	ℓ_1	Alex.
PSNR ↑	**26.42**	25.72	24.73
LPIPS ↓	**0.2475**	0.2683	0.2882

D.d Rationale of Using VGG Feature

As explained in Sec. 3.3 in the main paper, we choose VGG feature space for distance calculation because of its ability to capture not only semantic but also low-level characteristics. Besides, its wide use in perceptual loss promises that VGG representations are suitable for optimizing SR models. Thus, we project predictions into the VGG feature space to enforce the consistency relationships.

Fig. 14 is a schematic diagram exhibiting relationships between model predictions in the VGG feature space. Regardless of input degradation, the low-level characteristics of predictions from the generalist are similar, and we make the same assumption for the specialist model; In contrast, the low-level characteristics of predictions from different models differ.

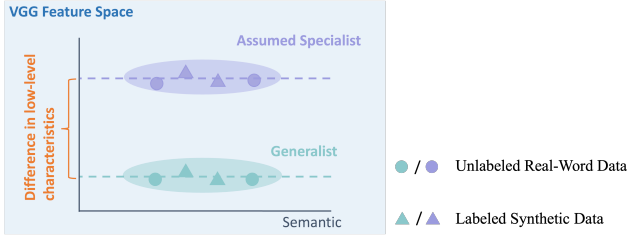


Fig. 14: Schematic diagram of the relationships between specialist and generalist predictions. Predictions from the same model have similar low-level characteristics regardless of the input degradation; predictions from different models differ in low-level characteristics. The diagram is for explanation and does not represent the true feature space.

Here, we further demonstrate the rationality of using VGG features by comparing them with two alternations. First, we exploit the low-level features used for exhibiting changes in low-level characteristics. As explained in Sec. B, pretrained SRGAN extracts representations named low-level features in this paper. These features focus on low-level characteristics and produce indicative visualizations in a low-dimensional embedding space [3]. However, being effective for visualization does not ensure they are suitable for optimizing an SR model.

If the low-level features are ideal representations of low-level characteristics for optimization, then enforcing similarity between such features of real-world and synthetic predictions from the specialist M_S should close their low-level characteristics and improve the quality of real-world predictions. In our preliminary experiments, we extracted low-level features from model predictions and passed them through the Gram Matrix to obtain features for optimization. We minimized the difference between the above features of real-world and synthetic predictions from M_S, along with the supervised loss $\mathcal{L}_L$ on labeled predictions. The loss functions are as follows:

$$\mathcal{L}_{llf} = \|\text{Gram}(F_{srgan}(\hat{Y}_S^U)) - \text{Gram}(F_{srgan}(\hat{Y}_S^L))\|_{\mathcal{F}} + \mathcal{L}_L, \tag{12}$$

where F_{srgan} refers to the model for extracting low-level features and $\|\cdot\|_{\mathcal{F}}$ represents Frobenius norm. However, it is observed that this optimization led to collapsed model predictions during experimentation. As a result, collecting representations of low-level characteristics for optimization is a non-trivial task, and we plan to consider it as a future work.

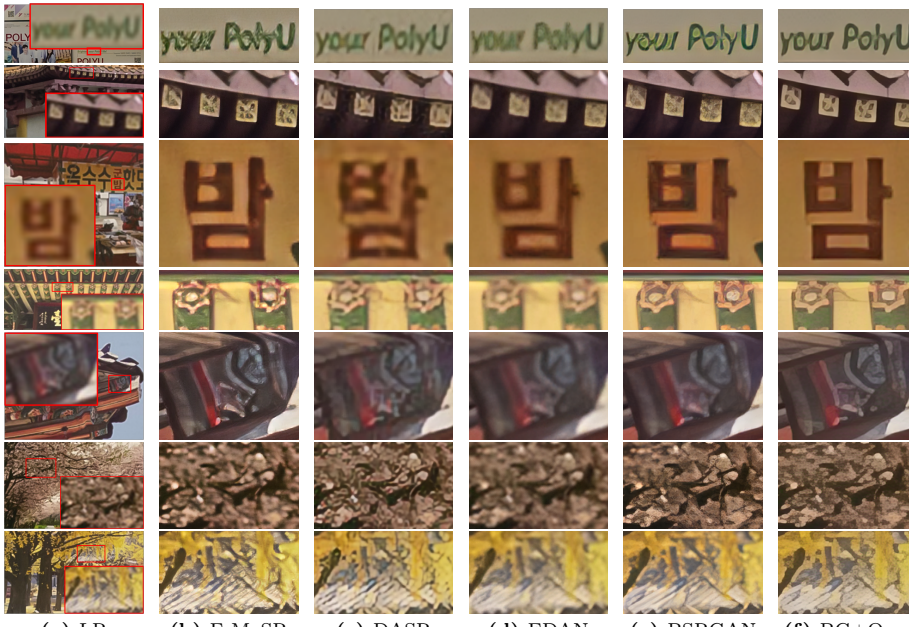
Second, we tried an alternative classification model, Alexnet [1], to explore whether features from an arbitrary classification model can serve our optimization. However, replacing VGG with Alexnet cannot achieve competitive performance, as shown in Tab. 9. We speculate that the architecture and depth of the classification network will affect the features' compatibility with the optimization process. As VGG is commonly used in the SR context, we use it directly in the main paper.

E Color Correction

We rectify the color of output images by normalizing the mean and variance of each color channel with those of the corresponding input channels. For the RGB format HR prediction $\hat{Y}_S^U$ of LR input X^U, the kth color channel $\hat{Y}_S^U[k]$ is corrected as follows:

$$\hat{Y}_S^U[k] = \frac{\hat{Y}_S^U[k] - \mu(\hat{Y}_S^U[k])}{\sigma(\hat{Y}_S^U[k])} \cdot \sigma(X^U[k]) + \mu(X^U[k]), \tag{13}$$

where $\mu(\cdot)$ and $\sigma(\cdot)$ computes mean and standard deviation of the color channels.



(a) LR **(b)** FeMaSR **(c)** DASR **(d)** EDAN **(e)** BSRGAN **(f)** BG+Ours

Fig. 15: Visual comparisons with state-of-the-art methods on RealSR-Canon and RealSR-Nikon datasets.

F Visual Comparison

Visual Comparisons with SOTAs are provided in Fig. 15. Our method produces clear patterns with fewer artifacts.

G Limitation

As discussed in Sec. D.a, the preference of models trained by our method is related to weights for R_{intra} and R_{inter}, which are hyperparameters need tuning. The balanced ratios can differ for different pre-trained models and different datasets. Although tuning for the balance weights is feasible, an automatic balance of the two regularizers through a certain training algorithm is more convenient. We leave it to future work.

References

1. Krizhevsky, A., Sutskever, I., Hinton, G.E.: Imagenet classification with deep convolutional neural networks. Advances in neural information processing systems **25** (2012)
2. Ledig, C., Theis, L., Huszár, F., Caballero, J., Cunningham, A., Acosta, A., Aitken, A., Tejani, A., Totz, J., Wang, Z., et al.: Photo-realistic single image super-resolution using a generative adversarial network. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 4681–4690 (2017)
3. Liu, Y., Liu, A., Gu, J., Zhang, Z., Wu, W., Qiao, Y., Dong, C.: Discovering distinctive" semantics" in super-resolution networks. arXiv preprint arXiv:2108.00406 (2021)
4. Wang, J., Yue, Z., Zhou, S., Chan, K.C., Loy, C.C.: Exploiting diffusion prior for real-world image super-resolution. arXiv preprint arXiv:2305.07015 (2023)