Research Square

Preprints are preliminary reports that have not undergone peer review.
They should not be considered conclusive, used to inform clinical practice,
or referenced by the media as validated information.

Regression, adversarial and diffusion architectures for single-image super-resolution of remote-sensing imagery: a statistically controlled benchmark with downstream land-use evaluation

Srinivasa Naveen Kumar
Narsimha Reddy Engineering College
Preethi Veerlapalli
preethireddyveerlapalli@gmail.com
Malla Reddy University
U. Rakesh
Geethanjali College of Engineering and Technology
Ganta Raju
Malla Reddy University
M. Archana
Malla Reddy University
Srinivasa Kumar Ch
Malla Reddy University
Gaddam Sridhar
Malla Reddy Engineering College for Women

Article

Keywords: super-resolution, remote sensing, generative adversarial network, diffusion model, perception–distortion trade-off, land-use classification

Posted Date: November 27th, 2026

DOI: https://doi.org/10.21203/rs.3.rs-11710885/v1

License: This work is licensed under a Creative Commons Attribution 4.0 International License.
Read Full License

Additional Declarations: No competing interests reported.

Abstract

Single-image super-resolution (SISR) can enhance the spatial resolution of aerial and satellite images, but generative models sacrifice pixel fidelity for perceptual realism. We compared regression (EDSR, RRDBNet), adversarial (ESRGAN) and diffusion (SR3-style) architectures under one ×4 protocol on UC Merced, with cross-dataset testing on AID, paired statistics, a frozen land-use classifier and robustness tests over training seeds, diffusion training length and realistic degradations. Regression achieved the highest fidelity (RRDB-PSNR 29.00 dB PSNR-Y), whereas ESRGAN (LPIPS 0.155) and diffusion (0.195) reduced perceptual distance relative to regression (0.294) at a fidelity cost; this ordering held on AID and across training seeds. A classifier trained on original images reached 98.6% on them and 86.2% on bicubic upsampling; its accuracy on generative outputs was statistically indistinguishable from the originals (98.8% and 98.1%) and on regression outputs slightly lower (96.4–96.9%). Doubling diffusion training did not help. Realistic blur, noise and compression reduced accuracy on all super-resolved outputs to 82–84%; short degradation-aware fine-tuning restored ESRGAN to 94.5%. Diffusion inference was about 73 times slower than ESRGAN. Fidelity and task utility favour different families, so remote-sensing super-resolution should be evaluated on both, under realistic degradations.

1 Introduction

Earth-observation (EO) imagery underpins land-use mapping, urban planning, agriculture and disaster response, but the spatial resolution of an image is fixed by the sensor, the orbit and the cost of acquisition. Very-high-resolution commercial imagery is costly and has limited revisit, whereas Sentinel-2, a freely available mission, images the entire land surface every few days at 10 m ground sampling distance. Single-image super-resolution (SISR), which estimates a high-resolution (HR) image from a single low-resolution (LR) observation, offers a way to narrow this gap computationally[1–3]. Because many HR images are consistent with a given LR input, SISR is an ill-posed inverse problem, and the way a model resolves that ambiguity determines what its output is useful for.

Deep learning has produced three broad families of SISR models. *Regression* networks, including SRCNN[4], EDSR[5], RCAN[6] and the Transformer-based SwinIR[7] and HAT[8], minimise a pixel-wise loss and therefore predict, approximately, the conditional mean of plausible HR images; they achieve a high peak signal-to-noise ratio (PSNR) but blur textures that are ambiguous given the input. *Adversarial* models, starting with SRGAN[9] and improved in ESRGAN[10] and Real-ESRGAN[11], add a discriminator[12,13] and a feature-space perceptual loss[9,10] so that outputs lie on the manifold of natural images, producing sharp textures at the expense of pixel fidelity and occasionally hallucinated structure. *Diffusion* models[14–16], applied to super-resolution in SR3[17], SRDiff[18], ResShift[19] and StableSR[20], learn to reverse a gradual noising process conditioned on the LR image and can generate several distinct HR samples, making the ambiguity of the problem explicit.

Blau and Michaeli demonstrated that distortion (e.g., mean-squared error) and perceptual quality cannot both be optimised beyond a certain bound[21]: each SISR method sits at a point on a perception–

distortion plane, and moving towards realism necessarily costs fidelity. For natural photographs, where the usual objective is a pleasing picture, the perceptual end of the plane is often preferred. Remote-sensing images are different: they are measurements, they are often consumed by algorithms rather than people, and invented detail such as spurious roofs, road markings or crop rows can mislead an analyst or a downstream model. Which point on the plane is appropriate for EO imagery is therefore an empirical question that PSNR and SSIM alone cannot answer.

The remote-sensing SR literature has embraced all three families, from CNN models such as LGCNet[22] and TransENet[23] to adversarial designs[24] and, recently, diffusion models including EDiffSR[25], SGDM[26] and DLMSR[27]; see recent surveys[2,3]. However, comparisons in this literature, including recent work in this journal[24,28], are difficult to interpret. Methods are trained with different budgets and data splits, results are reported as single averages without uncertainty, statistical tests are rarely used, perceptual metrics are often omitted for regression models and fidelity metrics for generative ones, and few studies examine whether super-resolved images retain the semantic content on which EO applications rely[29]. Sample diversity, a defining characteristic of diffusion models, is seldom quantified.

Among these families, regression models are trained with L1 or L2 losses and evaluated primarily by PSNR and SSIM[30]; the ESRGAN perceptual loss is calculated on pre-activation features of an ImageNet-trained VGG network[10,31]; and diffusion SR has branched out rapidly, from neural-operator diffusion for continuous-scale remote-sensing SR[32] and meta-learned implicit decoders for cross-domain SR[33] to the many methods reviewed by Moser *et al.*[34]. Following LGCNet[22] and TransENet[23], most remote-sensing SR studies synthesise LR images by bicubic downsampling of scene-classification datasets such as UC Merced[35] and AID[36], although multi-sensor datasets with real LR–HR pairs have recently become available for Sentinel-2[37] and in WorldStrat[38]. Learned perceptual metrics such as LPIPS[39] are standard in natural-image SR but have been applied inconsistently in remote sensing[3].

This work addresses these gaps with a controlled benchmark. Our contributions are:

- An identical training and evaluation protocol for representative regression (EDSR, RRDB-PSNR), adversarial (ESRGAN) and diffusion (SR3-style residual diffusion) SISR models at ×4 on UC Merced, with cross-dataset testing on AID, implemented in a single notebook from which every number in this paper is regenerated.
- A statistical treatment of image-quality results: per-image PSNR, SSIM and LPIPS with 95% bootstrap confidence intervals, paired Wilcoxon signed-rank tests with Holm correction and rank-biserial effect sizes.
- A task-based evaluation in which a land-use classifier trained on original HR imagery is applied unchanged to the output of each method, with McNemar tests, quantifying how much class-relevant information super-resolution preserves.
- An analysis of diffusion sample diversity and of whether per-pixel sample spread signals reconstruction error, together with a per-class analysis across 21 land-use categories and a

complexity comparison.

- Three robustness experiments that check the main design choices: repeated training seeds, doubled diffusion training and realistic blur–noise–JPEG degradation with degradation-aware fine-tuning.

Figure 1. Overview of the experimental pipeline. HR images are degraded by ×4 bicubic downsampling; five methods reconstruct them; outputs are assessed by image-quality metrics, by a fixed land-use classifier and, for diffusion, by sample uncertainty; all comparisons are made with paired statistics.

2 Results

2.1 Training behaviour

Training took 0.24 h for EDSR, 1.24 h for RRDB-PSNR, 1.14 h for ESRGAN fine-tuning and 2.74 h for the diffusion model. Validation PSNR for the pixel-loss models, perceptual and discriminator losses during ESRGAN training and the diffusion noise-prediction loss are shown in Supplementary Fig. S5. We obtain the final validation PSNR-Y of 30.26 dB for EDSR and 30.32 dB for RRDB-PSNR, both of which plateau at the end of training. As seen in the adversarial fine tuning, the validation PSNR of the ESRGAN generator dropped from its RRDB-PSNR initialisation to about 26.28 dB and fluctuated, which is natural because the objective of the fine tuning is not pixel fidelity anymore, but texture realism. The perceptual loss steadily went down and the discriminator loss was kept low (final 0.109), indicating that the discriminator separated real and generated patches with high confidence throughout.

2.2 Image quality on UC Merced and AID

The mean PSNR-Y, SSIM-Y and LPIPS of each method on the 420 UC Merced test images and on 300 AID images for cross-dataset evaluation are provided with 95% bootstrap confidence intervals in Table 1.

Table 1
×4 super-resolution on the UC Merced test set (n = 420) and, for models trained on UC Merced only, on the AID cross-dataset test set (n = 300).

Method	PSNR-Y (dB) ↑	SSIM-Y ↑	LPIPS ↓
UC Merced test set (in-domain, n = 420)			
Bicubic	26.83 [26.40, 27.27]	0.6997 [0.6868, 0.7123]	0.4882 [0.4759, 0.5010]
EDSR	28.96 [28.54, 29.40]	0.7899 [0.7797, 0.7996]	0.2950 [0.2819, 0.3087]
RRDB-PSNR	**29.00 [28.58, 29.44]**	**0.7900 [0.7796, 0.8001]**	0.2945 [0.2813, 0.3082]
ESRGAN	24.86 [24.55, 25.17]	0.6811 [0.6675, 0.6943]	**0.1552 [0.1489, 0.1619]**
SR3	26.32 [25.90, 26.76]	0.6965 [0.6828, 0.7099]	0.1954 [0.1868, 0.2044]
AID test set (cross-dataset, n = 300)			
Bicubic	29.43 [28.74, 30.13]	0.7455 [0.7294, 0.7617]	0.4625 [0.4459, 0.4790]
EDSR	30.93 [30.24, 31.63]	0.8049 [0.7913, 0.8183]	0.3052 [0.2897, 0.3209]
RRDB-PSNR	**30.94 [30.25, 31.64]**	**0.8057 [0.7923, 0.8191]**	0.3061 [0.2905, 0.3218]
ESRGAN	26.27 [25.83, 26.72]	0.7061 [0.6895, 0.7220]	**0.1999 [0.1917, 0.2086]**
SR3	28.30 [27.65, 28.95]	0.7238 [0.7060, 0.7415]	0.2201 [0.2084, 0.2318]

Mean over test images with 95% bootstrap confidence interval. Best value per column and dataset in bold.

Table 1 summarises in-domain results. Relative to bicubic interpolation (26.83 dB), EDSR improved PSNR-Y by 2.13 dB and RRDB-PSNR by 2.17 dB. ESRGAN reached 24.86 dB, 4.14 dB below its own initialisation, and the diffusion model reached 26.32 dB. RRDB-PSNR achieved the highest PSNR-Y and SSIM-Y. For perceptual similarity the ordering was different: ESRGAN had the lowest LPIPS (0.1552), compared with 0.2950 for EDSR and 0.4882 for bicubic.

2.3 Statistical comparisons

Because all methods are evaluated on the same images, Table 2 compares them pairwise on per-image values.

Table 2
Paired comparisons on the UC Merced test set: mean per-image difference Δ, Holm-adjusted Wilcoxon p-value and rank-biserial effect size r.

Comparison	ΔPSNR	p	r	ΔSSIM	p	r	ΔLPIPS	p	r
EDSR – Bicubic	2.132	< 0.001	0.98	0.0901	< 0.001	0.99	-0.1933	< 0.001	-1.00
RRDB-PSNR – Bicubic	2.170	< 0.001	0.99	0.0903	< 0.001	0.99	-0.1938	< 0.001	-1.00
ESRGAN – Bicubic	-1.972	< 0.001	-0.82	-0.0187	< 0.001	-0.29	-0.3331	< 0.001	-1.00
SR3 – Bicubic	-0.508	< 0.001	-0.40	-0.0032	0.258	0.06	-0.2928	< 0.001	-1.00
RRDB-PSNR – EDSR	0.038	< 0.001	0.45	0.0002	< 0.001	0.56	-0.0005	0.018	-0.13
ESRGAN – EDSR	-4.104	< 0.001	-1.00	-0.1088	< 0.001	-1.00	-0.1398	< 0.001	-0.98
SR3 – EDSR	-2.640	< 0.001	-1.00	-0.0934	< 0.001	-1.00	-0.0996	< 0.001	-0.91
ESRGAN – RRDB-PSNR	-4.142	< 0.001	-1.00	-0.1090	< 0.001	-1.00	-0.1393	< 0.001	-0.98
SR3 – RRDB-PSNR	-2.678	< 0.001	-1.00	-0.0935	< 0.001	-1.00	-0.0991	< 0.001	-0.92
SR3 – ESRGAN	1.464	< 0.001	0.72	0.0154	< 0.001	0.36	0.0403	< 0.001	0.72

Positive ΔPSNR/ΔSSIM and negative ΔLPIPS favour the first method. All 10 pairs per metric are shown; p-values are Holm-adjusted within each metric.

Table 2 reports paired tests. Of the 30 pairwise comparisons (10 pairs × 3 metrics), 29 were significant after Holm correction. The PSNR advantage of RRDB-PSNR over ESRGAN (Δ = 4.142 dB, r = 1.00) was significant ($p < 0.001$), and ESRGAN's LPIPS difference relative to RRDB-PSNR (Δ = -0.1393) was significant ($p < 0.001$, r = -0.98). Effect sizes near ± 1 indicate that the direction of the difference was consistent across almost all images rather than driven by a few.

2.4 Perception–distortion trade-off

Figure 2 places the methods on the perception–distortion plane. On UC Merced the Pareto-optimal methods, those not beaten on both axes by any other, were RRDB-PSNR, ESRGAN and SR3; Bicubic and EDSR were dominated. Pixel-loss models lie at the high-PSNR end and generative models towards the low-LPIPS end, consistent with the theoretical trade-off[21].

2.5 Qualitative comparison

Figure 3. Visual comparison on UC Merced (central 64 × 64 crop of the 256 × 256 output). Rows are selected automatically: the images with the largest LPIPS difference between EDSR and ESRGAN plus the image with median EDSR PSNR, to avoid selection by eye. Numbers give PSNR-Y (dB) / LPIPS for each crop.

Figure 3 shows representative outputs; rows were chosen by the fixed rule in the caption rather than by visual inspection.

2.6 Per-class analysis

Gains are not uniform across land-use types (Supplementary Fig. S6). For EDSR the gain over bicubic was largest for *runway* (3.85 dB) and smallest for *forest* (0.71 dB). The classes with the smallest regression gains are dominated by stochastic natural texture (*forest*, *agricultural*, *river*: EDSR + 0.71 to + 0.94 dB), and in these classes the generative models lost most fidelity relative to bicubic (ESRGAN − 3.68 to -2.68 dB) while gaining most in perceptual similarity: on *forest*, for example, LPIPS fell from 0.528 for EDSR to 0.241 for ESRGAN. Classes with regular man-made structure (*runway*, *buildings*, *freeway*, *airplane*) showed the largest regression gains (EDSR + 3.07 to + 3.85 dB), and there the fidelity penalty of the generative models was much smaller (ESRGAN − 1.03 to + 0.13 dB, diffusion + 0.28 to + 1.51 dB relative to bicubic). The largest single loss was ESRGAN on *beach* (-7.25 dB), where large smooth surfaces received synthetic texture. Edges and repeated geometric patterns are well constrained by the LR input and can be recovered by all learned models, whereas foliage, crop rows, sand and water surfaces are not; in those classes the generative models synthesise plausible texture that improves appearance but not pixel agreement. Per-class values for all methods are provided with the released data.

2.7 Cross-dataset generalisation to AID

The lower part of Table 1 applies the same evaluation to 300 AID images using the UC Merced models without adaptation; Supplementary Fig. S7 shows example outputs.

On AID, the best PSNR-Y was obtained by RRDB-PSNR and the lowest LPIPS by ESRGAN. Gains over bicubic were EDSR 1.50 dB, RRDB-PSNR 1.51 dB, ESRGAN − 3.16 dB, SR3 -1.13 dB.

Paired tests on AID (provided with the released data) gave the same pattern as on UC Merced. All comparisons were significant after Holm correction except EDSR versus RRDB-PSNR in LPIPS ($p = 0.064$). The diffusion model's SSIM-Y was significantly below that of bicubic interpolation on AID, whereas the two did not differ on UC Merced.

As on UC Merced, the smallest regression gains on AID occurred in texture-dominated natural classes (meadow, forest, mountain, bareland; EDSR + 0.45 to + 1.00 dB), and the largest ESRGAN losses relative to bicubic in classes dominated by smooth natural surfaces (desert, bareland, beach, pond; -10.49 to -5.33 dB). Per-class values are provided with the released data.

2.8 Downstream land-use classification

Table 3 reports the accuracy of the frozen classifier on the original test images and on the output of each method, with McNemar tests against HR and against bicubic interpolation.

Table 3
Top-1 accuracy of a ResNet-18 trained on original HR UC Merced images and applied unchanged to each input.

Input to classifier	Accuracy (%)	95% CI	p vs HR	p vs Bicubic
HR	98.6	[97.4, 99.5]	—	—
Bicubic	86.2	[82.9, 89.3]	< 0.001	—
EDSR	96.9	[95.2, 98.3]	0.092	< 0.001
RRDB-PSNR	96.4	[94.5, 98.1]	0.022	< 0.001
ESRGAN	98.8	[97.6, 99.8]	1.000	< 0.001
SR3	98.1	[96.7, 99.3]	0.688	< 0.001

p-values from exact McNemar tests on the same 420 test images.

Table 3 and Supplementary Fig. S8 show that the classifier reached 98.6% on HR images and 86.2% on bicubic-upsampled images, a loss of 12.4 percentage points. Every learned method recovered most of this loss (all $p < 0.001$ versus bicubic). The two generative methods came closest to HR: ESRGAN reached 98.8% and the diffusion model 98.1%, and neither differed significantly from HR (McNemar $p = 1.000$ and $p = 0.688$). The regression models, despite their higher PSNR, reached 96.9% (EDSR) and 96.4% (RRDB-PSNR); RRDB-PSNR was significantly below HR and EDSR was not (EDSR $p = 0.092$; RRDB-PSNR $p = 0.022$). The differences among learned methods are small, 2.4 percentage points at most, with overlapping confidence intervals, and generative and regression methods were not tested directly against each other. The evidence is therefore that generative outputs are at least as informative for this classifier as regression outputs, not that they are significantly better.

2.9 Diffusion sample diversity and uncertainty

Only the first 16 test images were used in the diversity analysis, and all these are in the agricultural class, which means that it describes sampling behaviour with one type of texture, but not the entire set. Eight stochastic samples (η_DDIM = 1) were drawn for each image. They differed from each other by a mean pairwise LPIPS of 0.1117, with a mean per-pixel standard deviation of 7.7 grey levels. Averaging the 8 samples raised PSNR-Y from 25.98 to 27.30 dB while LPIPS worsened from 0.3337 to 0.4019. This is a re-expression of the perception–distortion trade-off in one model: the sample mean is similar to the regression estimate. However, on the same images, deterministic sampling (η_DDIM = 0) gave an LPIPS of 0.296, lower than that of a single stochastic sample (0.334), suggesting some residual noise at full stochastic sampling with 50 steps. The per-pixel spread of samples had a weak correlation with the absolute error of a sample (Spearman $\rho = 0.079$; Supplementary Fig. S9). In this setting, variance of the samples is thus not a good indicator of error in the reconstruction.

2.10 Computational cost

Supplementary Table S6 shows that the diffusion model required 375 times the inference time of EDSR and 73 times that of ESRGAN, because every output requires 50 sequential U-Net evaluations.

2.11 Robustness of the findings

2.11.1 Training randomness

Supplementary Table S7 summarises the independently trained runs, and Fig. 4a shows each run on the perception–distortion plane.

Seed-to-seed variation was small relative to the differences between families. The standard deviation of PSNR-Y across runs was 0.018 dB for EDSR and 0.594 dB for ESRGAN, against a gap of 3.44 dB between their means. For LPIPS, the standard deviations were 0.0017 and 0.0025 against a gap of 0.138. Every ESRGAN run had lower LPIPS and lower PSNR-Y than every EDSR run, so the perception–distortion ordering does not depend on the seed. Classifier accuracy on ESRGAN outputs was statistically indistinguishable from HR in all 3 runs (98.6 ± 0.2%). EDSR outputs reached 96.8 ± 0.1%, and were not significantly below HR in any run either ($p = 0.057–0.092$), so the accuracy advantage of the adversarial family over regression is consistent in direction but small.

2.11.2 Longer diffusion training

Supplementary Table S8 compares the diffusion model after 25k and 50k training iterations, and Fig. 4b shows its learning curve.

Doubling the training budget did not improve the diffusion model. Relative to 25k iterations, PSNR-Y changed by -1.406 dB ($p < 0.001$) and LPIPS by 0.0234 ($p = 0.002$) on UC Merced, and by -1.987 dB and 0.0940 on AID. On the learning-curve subset, LPIPS was essentially flat for the first 10k additional iterations (0.163 → 0.161 at 35k) and then rose to 0.207, while PSNR-Y fell from 26.52 to 25.29 dB. With 1,470 training images, this pattern is consistent with over-fitting of the from-scratch diffusion model; restarting the optimiser for the extension may also have contributed. The main results therefore use the 25k model. After the extension, the diffusion model's LPIPS was higher than ESRGAN's by 0.0636 ($p < 0.001$), and its PSNR-Y differed from ESRGAN's by 0.06 dB ($p = 0.152$). The perceptual advantage of ESRGAN over the diffusion model is therefore not an artefact of under-training the diffusion model within this regime.

2.11.3 Realistic degradation

Table 4 reports all methods on the realistically degraded UC Merced test images. For reference, the classifier reached 98.6% on the original HR images.

Table 4
×4 super-resolution of realistically degraded inputs (Eq. 15) on the UC Merced test set (n = 420). Models with suffix -R were fine-tuned on realistic degradations.

Method	PSNR-Y (dB)	SSIM-Y	LPIPS	Accuracy (%)	p vs HR
Bicubic	25.42 [25.04, 25.82]	0.6230	0.6294 [0.6178, 0.6414]	77.1	< 0.001
EDSR	25.79 [25.43, 26.15]	0.6378	0.4932 [0.4802, 0.5061]	84.3	< 0.001
RRDB-PSNR	25.79 [25.43, 26.16]	0.6380	0.4963 [0.4836, 0.5089]	83.8	< 0.001
ESRGAN	23.54 [23.25, 23.84]	0.5476	0.3945 [0.3840, 0.4053]	81.7	< 0.001
SR3	24.87 [24.52, 25.24]	0.5818	0.4152 [0.4043, 0.4264]	81.9	< 0.001
EDSR-R	**26.52 [26.15, 26.90]**	**0.6837**	0.4415 [0.4277, 0.4557]	86.9	< 0.001
ESRGAN-R	23.18 [22.89, 23.46]	0.5602	**0.3002 [0.2928, 0.3080]**	**94.5**	< 0.001
SR3-R	23.52 [23.17, 23.87]	0.5453	0.3383 [0.3280, 0.3491]	90.0	< 0.001

Mean with 95% bootstrap CI; best value per column in bold; p from exact McNemar tests against the HR images.

Realistic degradation hurt every bicubic-trained model. Relative to bicubic inputs, PSNR-Y fell by 3.18 dB for EDSR, 3.21 dB for RRDB-PSNR, 1.32 dB for ESRGAN and 1.46 dB for the diffusion model. The advantage of the regression models over bicubic interpolation shrank from about 2.13 dB to 0.37 dB; both generative models remained below bicubic interpolation in PSNR-Y (-1.88 and − 0.55 dB, against − 1.97 and − 0.51 dB on bicubic inputs). Classifier accuracy dropped to 84.3%, 83.8%, 81.7% and 81.9% for these four models, against 77.1% for bicubic interpolation and 98.6% for HR. Degradation-aware fine-tuning helped each family along its own axis. EDSR-R gained 0.73 dB PSNR-Y ($p < 0.001$) and improved LPIPS by 0.052; ESRGAN-R and SR3-R reduced LPIPS by 0.094 and 0.077 (both $p < 0.001$) at a further cost in PSNR-Y (-0.36 and − 1.35 dB). The largest effect of fine-tuning was on task utility: accuracy rose to 86.9% for EDSR-R ($p = 0.108$ versus EDSR), 94.5% for ESRGAN-R ($p < 0.001$) and 90.0% for SR3-R ($p < 0.001$), although all remained significantly below HR. The fine-tuned regression model had the highest PSNR-Y and a fine-tuned generative model the lowest LPIPS, so the perception–distortion ordering of the families is preserved under realistic degradation (Fig. 4c).

Supplementary Table S9 separates the three degradation factors.

For EDSR, the most damaging single factor was JPEG compression (-2.86 dB relative to bicubic inputs); for ESRGAN it was noise (-2.35 dB). Under noise alone ESRGAN fell 3.99 dB below bicubic interpolation, consistent with the adversarial generator amplifying noise into spurious texture. Fine-tuning on the combined degradation lowered LPIPS under noise and JPEG compression for both EDSR-R and ESRGAN-

R, but raised it slightly under blur. The fine-tuned models therefore trade some performance on the factors they rarely see in isolation for robustness to the combined degradation.

Supplementary Fig. S10 shows example outputs on realistically degraded inputs.

3 Discussion

The results reproduce the perception–distortion trade-off[21] on remote-sensing imagery with statistical support. The two regression models performed almost equally: RRDB-PSNR exceeded EDSR by only 0.04 dB on UC Merced and 0.01 dB on AID, and both outperformed bicubic interpolation by 2.13–2.17 dB PSNR-Y and about 0.09 SSIM-Y. The generative models occupied the opposite end of the plane. ESRGAN had the lowest LPIPS (0.155, about 53% of that of the regression models), and the diffusion model the second lowest (0.195). Both scored below bicubic interpolation in PSNR-Y, although the SSIM-Y of the diffusion model did not differ from that of bicubic interpolation ($p = 0.26$). On UC Merced the Pareto-optimal methods were RRDB-PSNR, ESRGAN and SR3. Neither generative model dominated the other: the diffusion model traded part of the perceptual advantage of ESRGAN for 1.46 dB higher PSNR-Y and 0.015 higher SSIM-Y. The same ordering on AID, which differs in sensor, geography and ground resolution, indicates that the trade-off is a property of the model families rather than of one dataset, and the repeated-seed experiment (Section 2.11) confirms that it does not depend on the training seed.

That a sharper, more realistic reconstruction has lower PSNR than simple interpolation is not a failure of the generative models. It is the expected consequence of sampling one plausible texture instead of averaging over all plausible textures, as the per-class analysis makes explicit. In classes whose fine structure is unconstrained by the LR input (forest, agricultural fields, rivers), regression gained at most 0.9 dB over bicubic, whereas the generative models lost 1.5 to 3.7 dB but substantially reduced LPIPS (on forest, from 0.528 for EDSR to 0.241 for ESRGAN). In classes with regular geometry, all learned models recovered structure and the generative fidelity penalty was small. For practitioners, this means that the fidelity penalty of generative SR is concentrated in natural land covers, which are exactly where synthesised detail is least verifiable.

Image-quality metrics alone would favour the regression models. Bicubic upsampling cost 12.4 percentage points of accuracy, and every learned method recovered most of that loss. The generative outputs came closest to HR (98.8% and 98.1%) and were statistically indistinguishable from it, consistently across 3 ESRGAN training runs. The regression outputs were only slightly lower (96.4–96.9%), significantly so only for RRDB-PSNR; EDSR did not differ significantly from HR in any of its 3 training runs. The effect is consistent in direction but small. A plausible explanation is that the classifier, trained on real HR images, uses high-frequency texture cues, such as canopy grain or the spacing of mobile homes, which regression models smooth by averaging and generative models restore with statistically similar texture. Two cautions apply. The generative models were not shown to be significantly better than the regression models, and a texture typical of a class can be synthesised without being correct at the pixel level: the classifier measures semantic plausibility, not fidelity.

Generative SR can therefore support scene-level interpretation, but object-level detail, such as counts of buildings or vehicles, should not be derived from it without independent verification. We recommend that task-based evaluation, as advocated for object detection[29], accompany image-quality metrics in remote-sensing SR studies, because the two can rank methods differently.

The realistic-degradation experiment shows that bicubic benchmarks, including the main experiment here, flatter every SR family. With blur, noise and compression applied to the input, the PSNR-Y advantage of EDSR over bicubic interpolation fell from 2.13 dB to 0.37 dB, and classifier accuracy on the outputs of bicubic-trained models fell from 96–99% to 82–84%. Short degradation-aware fine-tuning, at a small fraction of the original training cost, recovered part of this loss and again split along the perception–distortion line: the regression model regained fidelity, and the adversarial model regained the most task utility (94.5% accuracy). For operational use, the training degradation should match the sensor, and results reported only on bicubic downsampling are likely to be optimistic.

The diffusion model was the slowest method, at 2.07 s per 256 × 256 image, about 73 times slower than ESRGAN and 375 times slower than EDSR, yet it did not improve on ESRGAN in perceptual quality or downstream accuracy. Its distinctive capability is sampling several reconstructions, but in our configuration the spread of these samples was only weakly related to reconstruction error, and fully stochastic samples were perceptually worse than deterministic ones. Doubling the training iterations did not help, and performance degraded with longer training, so the iteration count is not the bottleneck; training from scratch on 1,470 images is the more likely limitation. Recent approaches that start from large pre-trained priors[20,26,27], shorten the sampling chain[19,28] or use more efficient architectures[25] are designed to address exactly these weaknesses. On the evidence here, a compact adversarial model offers the best balance of realism, task utility and cost for operational remote-sensing SR at ×4, whereas diffusion models become attractive when pre-trained priors or calibrated uncertainty are available.

Several limitations remain. Training randomness for the regression and adversarial families, the training budget of the diffusion model and the use of bicubic degradation alone were addressed directly, but the realistic inputs are still synthesised from HR images, and real sensor LR–HR pairs[37,38] remain the definitive test. RRDB-PSNR and the diffusion model were trained once, the latter because each additional run costs about 2.7 GPU hours, so their single-run results should be read with the seed-to-seed spread of the other families in mind. All models are smaller than in their original papers (12 rather than 23 RRDB blocks), and the diffusion model is trained from scratch without a pre-trained prior[20,26,27]; pre-trained diffusion priors, which are designed for small datasets, may change its standing. A single scene classifier was used, and methods were compared against HR and bicubic but not against each other; object detection and segmentation, which depend on finer detail, may rank methods differently. Only ×4 on three-band RGB imagery was evaluated, so multispectral bands, larger factors and non-integer scales are not covered. LPIPS is trained on natural-image judgements and may not reflect the preferences of EO analysts; a user study with remote-sensing experts would complement it. Sample diversity was analysed on 16 test images from a single class, which shows the effects qualitatively but does not establish them across land-use types. Finally, UC Merced and AID consist of aerial and Google Earth imagery at sub-

metre to metre resolution, so transfer to coarser satellite sensors such as Sentinel-2 remains to be shown.

Natural extensions are to evaluate on real multi-sensor LR–HR pairs such as WorldStrat[38]; to include Transformer regression baselines[7,8] and pre-trained latent diffusion priors[20] under the same statistical protocol; to extend the downstream evaluation to segmentation and object detection with direct pairwise tests between methods; and to calibrate diffusion sample variance so that it can be reported as a quantitative uncertainty estimate.

We compared regression, adversarial and diffusion architectures for ×4 remote-sensing image super-resolution under a single, statistically tested protocol. The regression models (EDSR, RRDB-PSNR) achieved the highest fidelity, 2.1–2.2 dB PSNR-Y above bicubic interpolation. ESRGAN and the diffusion model produced the most realistic images, at a fidelity cost concentrated in natural land covers, and classifier accuracy on their outputs was statistically indistinguishable from that on the original imagery, while accuracy on regression outputs was slightly lower. These orderings were unchanged on an independent dataset and across training seeds. Under an equal compute budget, ESRGAN offered the best balance of realism, task utility and speed. The diffusion model was about 73 times slower without a measurable advantage, did not benefit from doubled training, and its sample spread was a weak indicator of error. Realistic blur, noise and compression reduced the benefit of every family, with classifier accuracy falling to 82–84%; brief degradation-aware fine-tuning recovered part of the loss, most of all for task utility with ESRGAN (94.5%). We recommend that remote-sensing SR studies report fidelity and perceptual metrics with uncertainty, include a task-based evaluation, test under realistic degradations, and treat generative detail as plausible rather than measured.

4 Methods

Supplementary Table S1 lists the methods evaluated in this study and the configuration used for each.

4.1 Problem formulation

Let $y \in \mathbb{R}^{H\times W\times 3}$ be an HR image and x its LR observation. We adopt the standard synthetic degradation used in remote-sensing SR benchmarks[22,23],

$x = (y \otimes k_{bic})\downarrow_s,\ s = 4$ (1)

where $\otimes k_{bic}$ denotes anti-aliased bicubic filtering and $\downarrow_s$ decimation by the scale factor s. An SISR model is a mapping $f_\theta : x \mapsto \hat{y}$, or, for generative models, a conditional distribution $p_\theta(\hat{y} \mid x)$ from which samples are drawn. Because Eq. (1) discards high frequencies, $p(y \mid x)$ is broad; regression models approximate its mean, whereas adversarial and diffusion models attempt to sample from it.

4.2 Datasets and experimental protocol

UC Merced Land Use[35] contains 2,100 RGB aerial images of 256 × 256 pixels at 0.3 m (1 ft) resolution from the USGS National Map, 100 per class across 21 land-use classes. We split each class, with a fixed seed, into 70% training, 10% validation and 20% test images, yielding 1470 training, 210 validation and 420 test images; the split is stratified so that every class contributes equally to the test set. The few UC Merced images that are not exactly 256 × 256 are centre-cropped to the nearest multiple of the scale factor. UC Merced therefore consists of aerial orthophotographs rather than satellite images; together with the Google Earth imagery of AID, it is used here as representative very-high-resolution remote-sensing data, following common practice in remote-sensing SR[22,23].

AID[36] contains 10,000 images of 600 × 600 pixels from Google Earth covering 30 scene classes, with ground resolution varying between about 0.5 and 8 m. AID is used only for testing: we draw 10 images per class (300 images) with a fixed seed and centre-crop each to 256 × 256. Because AID differs from UC Merced in sensor, geography, resolution and class set, it measures cross-dataset generalisation of models trained only on UC Merced.

The two datasets and their roles are summarised in Supplementary Table S2.

Supplementary Fig. S1 illustrates the degradation on two UC Merced test images. At ×4 the LR input retains the layout of the scene, but fine periodic structure such as crop rows and canopy texture is no longer resolvable, and bicubic interpolation cannot restore it. This is the information gap that the SR methods attempt to close.

4.3 Regression models

EDSR follows the baseline configuration of Lim et al.[5]: a head convolution, 16 residual blocks with 64 channels and no batch normalisation, a global skip connection and two ×2 pixel-shuffle upsampling stages. RRDB-PSNR uses the RRDBNet generator of ESRGAN[10] with 12 residual-in-residual dense blocks (64 channels, growth 32, residual scaling 0.2) and nearest-neighbour upsampling followed by convolutions. We use 12 rather than the original 23 RRDB blocks so that all models fit a single GPU session; Section 3 discusses the implications. Both models are trained with the L1 loss

$$L_{pix} = \| f_\theta(x) - y \|_1 \quad (2)$$

which is more robust to outliers than L2 and is standard for PSNR-oriented SR[5,6].

4.4 Adversarial model

ESRGAN shares the RRDBNet generator G and is initialised from RRDB-PSNR, which stabilises adversarial training[10]. A VGG-style discriminator D with batch normalisation and strided convolutions scores 128 × 128 patches. Following the relativistic average GAN (RaGAN)[13], with C(·) the raw discriminator output, the discriminator and generator adversarial losses are

$$L_D = - E_y[\log \sigma(C(y) - E_{\hat{y}} C(\hat{y}))] - E_{\hat{y}}[\log(1 - \sigma(C(\hat{y}) - E_y C(y)))] \quad (3)$$

$$L_{adv} = -E_y[\log(1 - \sigma(C(y) - E_{\hat{y}} C(\hat{y})))] - E_{\hat{y}}[\log \sigma(C(\hat{y}) - E_y C(y))] \quad (4)$$

where σ is the logistic function and $\hat{y} = G(x)$. The perceptual loss compares pre-activation conv5_4 features φ of an ImageNet-trained VGG19[31],

$$L_{perc} = \| \varphi(G(x)) - \varphi(y) \|_1 \quad (5)$$

and the total generator objective uses the ESRGAN weights[10]:

$$L_G = L_{perc} + \lambda L_{adv} + \eta L_{pix}, \lambda = 5\times10^{-3}, \eta = 10^{-2} \quad (6)$$

4.5 Diffusion model

The diffusion model follows SR3[17] with two modifications that improve sample efficiency under a limited training budget: the network models the residual $r_0 = 2(y - u)$, where $u = up_{bic}(x)$ is the bicubic-upsampled LR image, as in SRDiff[18], and a cosine noise schedule $\bar{\alpha}_t$ is used[16]. The forward process is

$$q(r_t \mid r_0) = N(r_t ; \sqrt{\bar{\alpha}_t}\, r_0, (1 - \bar{\alpha}_t) I), t = 1, \ldots, T, T = 1000 \quad (7)$$

A conditional U-Net ε_θ (64 base channels, channel multipliers (1, 2, 2, 4), two residual blocks per level, sinusoidal time embedding, self-attention at 16 × 16 resolution and in the bottleneck) receives the concatenation of r_t and $2u - 1$ and is trained to predict the injected noise:

$$L_{diff} = E_{t, r_0, \varepsilon} \| \varepsilon - \varepsilon_\theta(\sqrt{\bar{\alpha}_t}\, r_0 + \sqrt{(1 - \bar{\alpha}_t)}\, \varepsilon, u, t) \|_2^2, \varepsilon \sim N(0, I) \quad (8)$$

An exponential moving average (EMA, decay 0.999) of the weights is used for sampling. Sampling uses DDIM[15] with S = 50 steps; with predicted clean residual $\hat{r}_0 = (r_t - \sqrt{(1 - \bar{\alpha}_t)}\, \varepsilon_\theta)/\sqrt{\bar{\alpha}_t}$, each step is

$$r_{t'} = \sqrt{\bar{\alpha}_{t'}}\, \hat{r}_0 + \sqrt{(1 - \bar{\alpha}_{t'} - \sigma_t^2)}\, \varepsilon_\theta + \sigma_t z, \sigma_t = \eta_{DDIM} \sqrt{((1 - \bar{\alpha}_{t'})/(1 - \bar{\alpha}_t))}\, \sqrt{(1 - \bar{\alpha}_t/\bar{\alpha}_{t'})} \quad (9)$$

where t' is the next time step and $z \sim N(0, I)$. The final image is $\hat{y} = u + \hat{r}_0/2$. We use deterministic sampling ($\eta_{DDIM} = 0$, fixed seed) for all quality and downstream metrics, so that the diffusion model is evaluated as a single-output estimator like the others, and stochastic sampling ($\eta_{DDIM} = 1$) for the diversity analysis.

Supplementary Figs. S2 and S3 summarise the three families and the order of operations in one training iteration and at inference for the two generative models; pseudocode is given in Supplementary Algorithms S1–S3.

4.6 Image-quality metrics

Following common practice[5,6,23], PSNR and SSIM are computed on the luminance (Y) channel of YCbCr after removing an s-pixel border. With MSE the mean squared error between Y channels in [0, 255],

$$\mathrm{PSNR} = 10 \log_{10} (255^2 / \mathrm{MSE}) \quad (10)$$

$$\mathrm{SSIM}(a, b) = [(2\mu_a \mu_b + C_1)(2\sigma_{ab} + C_2)] / [(\mu_a^2 + \mu_b^2 + C_1)(\sigma_a^2 + \sigma_b^2 + C_2)] \quad (11)$$

where local statistics use an 11 × 11 Gaussian window (σ = 1.5), $C_1 = (0.01 \cdot 255)^2$ and $C_2 = (0.03 \cdot 255)^2$[30], averaged over the image. Perceptual similarity is measured with LPIPS[39] (AlexNet backbone, v0.1 linear weights), computed on RGB images in [− 1, 1]; lower is better. LPIPS correlates better with human judgements of texture realism than PSNR or SSIM[39], and together they place each method on the perception–distortion plane[21]. Inference time is measured per 256 × 256 output on the same GPU with explicit synchronisation.

4.7 Downstream land-use classification

To measure how much class-relevant information survives super-resolution, we fine-tune an ImageNet-initialised ResNet-18[40] on the *original HR* UC Merced training images (2000 iterations, batch 32, AdamW, learning rate 3×10^{-4} with cosine decay, label smoothing 0.1, flips and 90° rotations, inputs resized to 224 × 224). The classifier is then frozen and applied to the HR test images and to the outputs of each SR method on the same test images. Because the classifier never sees super-resolved images, a drop in accuracy relative to HR reflects information lost or corrupted by the SR method rather than adaptation to its artefacts.

4.8 Statistical analysis

All metrics are computed per test image, so methods are compared on identical inputs. For each method and metric we report the mean, standard deviation and a 95% percentile bootstrap confidence interval of the mean from B = 10,000 resamples[41]:

$$\mathrm{CI}_{95} = [q_{0.025}(\{ \bar{m}^*_b \}_{b = 1}^B), q_{0.975}(\{ \bar{m}^*_b \}_{b = 1}^B)] \quad (12)$$

where $\bar{m}^*_b$ is the mean of the b-th resample of the n test images with replacement. Pairwise differences between methods are tested with the two-sided Wilcoxon signed-rank test[42] on per-image values, and p-values within each metric are adjusted for the ten pairwise comparisons with the Holm procedure[43]. Effect sizes are reported as the matched-pairs rank-biserial correlation,

$$r_{rb} = (\Sigma_{d_i > 0} R_i - \Sigma_{d_i < 0} R_i) / \Sigma_i R_i \quad (13)$$

where d_i is the per-image difference and R_i the rank of $|d_i|$ among non-zero differences; r_{rb} ranges from − 1 to 1. Classification accuracies are compared with the exact McNemar test[44] on the discordant pairs n_{01} and n_{10},

$$p = \min\{ 1, 2 \cdot \min\{ P(B \le n_{01}), P(B \ge n_{01}) \} \}, \; B \sim \mathrm{Binomial}(n_{01} + n_{10}, \tfrac{1}{2}) \quad (14)$$

A significance level of 0.05 is used throughout, after correction where stated. The complete evaluation protocol is summarised in Supplementary Algorithm S4.

4.9 Robustness experiments

Three further experiments test whether the conclusions depend on the main design choices. They re-use the trained models and follow the evaluation protocol of Supplementary Algorithm S4 unchanged.

4.9.1 Training randomness

EDSR and ESRGAN were retrained with additional random seeds, giving three independent runs per family; the first run used seed 2026. Each seed changes the network initialisation, the order and position of training patches, and the augmentations. To isolate the variability of adversarial training, the additional ESRGAN runs start from the same RRDB-PSNR initialisation as the first run. For every run we report test-set means and, across runs, their mean and standard deviation. We also check whether the ranking of the families and the downstream conclusions hold for every seed.

4.9.2 Diffusion training budget

Training of the diffusion model was continued from 25,000 to 50,000 iterations, starting from its final EMA weights. A fresh AdamW optimiser was used, with a 500-iteration warm-up to half the original peak learning rate followed by cosine decay, and the EMA was continued throughout. A learning curve is recorded every 5,000 iterations on two test images per class (42 images), using the same 50-step deterministic DDIM sampler. The final model is evaluated on the full test sets and compared with the 25k model by paired tests.

4.9.3 Realistic degradation

Bicubic downsampling ignores the blur of real optics, sensor noise and lossy compression. We therefore also generate LR inputs with a randomised degradation in the spirit of practical blind-SR models[11]:

$$x = \mathrm{JPEG}_q\left((y \otimes k_{\sigma_1,\sigma_2,\theta}) \downarrow_s + n \right),\ n \sim \mathcal{N}(0, \sigma_n^2 I) \quad (15)$$

Here k is an anisotropic Gaussian kernel with $\sigma_1, \sigma_2 \sim U(0.6, 2.0)$ pixels and orientation $\theta \sim U(0, \pi)$, $\downarrow_s$ is bicubic ×4 downsampling, $\sigma_n \sim U(1, 8)$ grey levels and the JPEG quality $q \sim U\{65, \ldots, 95\}$. Each test image receives one fixed draw, so all methods see identical inputs. All bicubic-trained models are evaluated on these inputs. In addition, each family is fine-tuned on degradations drawn on the fly from Eq. (15): EDSR for 10,000 iterations at learning rate 10^{-4}, ESRGAN for 5,000 iterations with a fresh discriminator, and the diffusion model for 5,000 iterations. These degradation-aware models are denoted EDSR-R, ESRGAN-R and SR3-R. To attribute the effect of each factor, the fast models are also evaluated on single-factor degradations: isotropic blur ($\sigma = 1.5$), noise ($\sigma_n = 5$) and JPEG compression ($q = 75$), each combined with bicubic downsampling. The frozen land-use classifier is applied to every output, as in the main experiment.

4.10 Implementation details

Supplementary Table S3 specifies the four network architectures, and Supplementary Table S4 gives their training configurations.

A small number of UC Merced images are not exactly 256 × 256 pixels; after centre-cropping to a multiple of the scale factor they measure, for example, 252 × 256. Such images are processed at their native size by all methods. Because the diffusion U-Net downsamples three times, its input is reflect-padded to the next multiple of eight and the output is cropped back to the original size before any metric is computed. For the downstream classifier, every image is resized bilinearly to 224 × 224 in both training and testing.

4.11 Environment and reproducibility

All experiments were run end to end in a single Jupyter notebook on Kaggle with one Tesla T4 GPU (16 GB), using Python 3.12, PyTorch 2.10 with CUDA 12.8, automatic mixed precision and random seed 2026. Training took 5.4 GPU hours for the main benchmark and 6.8 GPU hours for the robustness experiments. The notebook acquires both datasets automatically, stores the split by class, file name and a hash of the decoded pixels so that the identical split is recovered from any mirror, and checkpoints every stage atomically so that an interrupted run resumes where it stopped. All tables and figures in this article were generated programmatically from the notebook outputs, so the reported numbers are exactly those computed. The software environment, data-acquisition procedure and fault-tolerant workflow are described in Supplementary Note 2, Supplementary Table S5 and Supplementary Fig. S4.

Declarations

Competing Interests

The authors declare that there is no conflict of interest.

Ethical Approval

This study used only publicly available remote-sensing image datasets (UC Merced Land Use and AID) and did not involve human or animal participants; therefore, ethical approval was not required.

Funding

The authors did not receive any funding.

Author Contribution

Srinivasa Naveen Kumar G, U. Rakesh conceived the study, developed the methodology, implemented the software, performed the formal analysis and drafted the original manuscript. Preethi Veerlapalli, Ganta

Raju, M. Archana supervised the work, validated the results, and Srinivasa Kumar Ch, Gaddam Sridhar critically reviewed and edited the manuscript. All authors read and approved the final manuscript.

Acknowledgements

None.

Data Availability

UC Merced Land Use is publicly available from the University of California, Merced (http://weegee.vision.ucmerced.edu/datasets/landuse.html) and AID from https://captain-whu.github.io/AID/; both are also mirrored on Kaggle. The data split (class, file name and pixel hash of every image), per-image metrics, per-class results, the AID paired tests.

Code Availability

The complete notebook used to train the models, run the evaluation and generate all tables and figures is available at on request.

References

1. Wang, Z., Chen, J. & Hoi, S. C. H. Deep learning for image super-resolution: a survey. *IEEE Trans. Pattern Anal. Mach. Intell.* **43**, 3365–3387 (2021).
2. Wang, P., Bayram, B. & Sertel, E. A comprehensive review on deep learning based remote sensing image super-resolution methods. *Earth-Sci. Rev.* **232**, 104110 (2022).
3. Qi, Y. et al. Advancing image super-resolution techniques in remote sensing: a comprehensive survey. *ISPRS J. Photogramm Remote Sens.* **231**, 68–100 (2026).
4. Dong, C., Loy, C. C., He, K. & Tang, X. Image super-resolution using deep convolutional networks. *IEEE Trans. Pattern Anal. Mach. Intell.* **38**, 295–307 (2016).
5. Lim, B., Son, S., Kim, H., Nah, S. & Lee, K. M. Enhanced deep residual networks for single image super-resolution. In Proc. IEEE Conf. Computer Vision and Pattern Recognition Workshops 136–144 (2017).
6. Zhang, Y. et al. Image super-resolution using very deep residual channel attention networks. In Proc. European Conf. Computer Vision 286–301 (2018).
7. Liang, J. et al. SwinIR: image restoration using Swin Transformer. In Proc. IEEE/CVF Int. Conf. Computer Vision Workshops 1833–1844 (2021).
8. Chen, X., Wang, X., Zhou, J., Qiao, Y. & Dong, C. Activating more pixels in image super-resolution transformer. In Proc. IEEE/CVF Conf. Computer Vision and Pattern Recognition 22367–22377

(2023).

9. Ledig, C. et al. Photo-realistic single image super-resolution using a generative adversarial network. In Proc. IEEE Conf. Computer Vision and Pattern Recognition 4681–4690 (2017).
10. Wang, X. et al. ESRGAN: enhanced super-resolution generative adversarial networks. In Proc. European Conf. Computer Vision Workshops 63–79 (2018).
11. Wang, X., Xie, L., Dong, C. & Shan, Y. Real-ESRGAN: training real-world blind super-resolution with pure synthetic data. In Proc. IEEE/CVF Int. Conf. Computer Vision Workshops 1905–1914 (2021).
12. Goodfellow, I. et al. Generative adversarial nets. *Adv. Neural. Inf. Process. Syst.* **27**, 2672–2680 (2014).
13. Jolicoeur-Martineau, A. The relativistic discriminator: a key element missing from standard GAN. In Int. Conf. Learning Representations (2019).
14. Ho, J., Jain, A. & Abbeel, P. Denoising diffusion probabilistic models. *Adv. Neural. Inf. Process. Syst.* **33**, 6840–6851 (2020).
15. Song, J., Meng, C. & Ermon, S. Denoising diffusion implicit models. In Int. Conf. Learning Representations (2021).
16. Nichol, A. Q. & Dhariwal, P. Improved denoising diffusion probabilistic models. In Proc. Int. Conf. Machine Learning 8162–8171 (2021).
17. Saharia, C. et al. Image super-resolution via iterative refinement. *IEEE Trans. Pattern Anal. Mach. Intell.* **45**, 4713–4726 (2023).
18. Li, H. et al. SRDiff: single image super-resolution with diffusion probabilistic models. *Neurocomputing* **479**, 47–59 (2022).
19. Yue, Z., Wang, J. & Loy, C. C. ResShift: efficient diffusion model for image super-resolution by residual shifting. In *Adv. Neural. Inf. Process. Syst.* **36** (2023).
20. Wang, J., Yue, Z., Zhou, S., Chan, K. C. K. & Loy, C. C. Exploiting diffusion prior for real-world image super-resolution. *Int. J. Comput. Vis.* **132**, 5929–5949. https://doi.org/10.1007/s11263-024-02168-7 (2024).
21. Blau, Y. & Michaeli, T. The perception-distortion tradeoff. In Proc. IEEE Conf. Computer Vision and Pattern Recognition 6228–6237 (2018).
22. Lei, S., Shi, Z. & Zou, Z. Super-resolution for remote sensing images via local–global combined network. *IEEE Geosci. Remote Sens. Lett.* **14**, 1243–1247 (2017).
23. Lei, S., Shi, Z. & Mo, W. Transformer-based multistage enhancement for remote sensing image super-resolution. *IEEE Trans. Geosci. Remote Sens.* **60**, 1–11. https://doi.org/10.1109/TGRS.2021.3136190 (2022).
24. Liu, D., Zhong, L., Wu, H., Li, S. & Li, Y. Remote sensing image super-resolution reconstruction by fusing multi-scale receptive fields and hybrid transformer. *Sci. Rep.* **15**, 2140 (2025).
25. Xiao, Y. et al. EDiffSR: an efficient diffusion probabilistic model for remote sensing image super-resolution. *IEEE Trans. Geosci. Remote Sens.* **62**, 5601514.

https://doi.org/10.1109/TGRS.2023.3341437 (2024).

26. Wang, C. & Sun, W. Semantic guided large scale factor remote sensing image super-resolution with generative diffusion prior. *ISPRS J. Photogramm Remote Sens.* **220**, 125–138 (2025).
27. Lu, L., Huang, Z., Bao, Y., Wan, L. & Li, Z. Multi-level priors-guided diffusion-based remote sensing image super-resolution. *ISPRS J. Photogramm Remote Sens.* **228**, 756–770 (2025).
28. Shi, Y., Zhang, X., Jia, Y. & Zhao, J. Multi-scale adversarial diffusion network for image super-resolution. *Sci. Rep.* **15**, 11690 (2025).
29. Shermeyer, J. & Van Etten, A. The effects of super-resolution on object detection performance in satellite imagery. In Proc. IEEE/CVF Conf. Computer Vision and Pattern Recognition Workshops (2019).
30. Wang, Z., Bovik, A. C., Sheikh, H. R. & Simoncelli, E. P. Image quality assessment: from error visibility to structural similarity. *IEEE Trans. Image Process.* **13**, 600–612 (2004).
31. Simonyan, K. & Zisserman, A. Very deep convolutional networks for large-scale image recognition. In Int. Conf. Learning Representations (2015).
32. Xu, Z., Tang, Y., Xu, B. & Li, Q. NeurOp-Diff: continuous remote sensing image super-resolution via neural operator diffusion. In Proc. IEEE/CVF Int. Conf. Computer Vision 12491–12501 (2025).
33. Zhang, Q. et al. Cross-domain continuous-scale remote sensing image super-resolution via meta-weight learning. *Sci. Rep.* **16**, 6073 (2026).
34. Moser, B. B. et al. Diffusion models, image super-resolution, and everything: a survey. *IEEE Trans. Neural Netw. Learn. Syst.* 1–21 (2024).
35. Yang, Y. & Newsam, S. Bag-of-visual-words and spatial extensions for land-use classification. In Proc. 18th ACM SIGSPATIAL Int. Conf. Advances in Geographic Information Systems 270–279 (2010).
36. Xia, G. S. et al. AID: a benchmark data set for performance evaluation of aerial scene classification. *IEEE Trans. Geosci. Remote Sens.* **55**, 3965–3981 (2017).
37. Nguyen-Vi, K., Bui-Quoc, B. & Kamel, N. Deep learning-based Sentinel-2 super-resolution via channel attention and high-frequency feature enhancement. *Front. Remote Sens.* **6**, 1644460 (2025).
38. Cornebise, J., Oršolić, I. & Kalaitzis, F. Open high-resolution satellite imagery: the WorldStrat dataset — with application to super-resolution. In Advances in Neural Information Processing Systems 35, Datasets and Benchmarks Track (2022).
39. Zhang, R., Isola, P., Efros, A. A., Shechtman, E. & Wang, O. The unreasonable effectiveness of deep features as a perceptual metric. In Proc. IEEE Conf. Computer Vision and Pattern Recognition 586–595 (2018).
40. He, K., Zhang, X., Ren, S. & Sun, J. Deep residual learning for image recognition. In Proc. IEEE Conf. Computer Vision and Pattern Recognition 770–778 (2016).
41. Efron, B. & Tibshirani, R. J. *An Introduction to the Bootstrap* (Chapman & Hall/CRC, 1993).
42. Wilcoxon, F. Individual comparisons by ranking methods. *Biometrics Bull.* **1**, 80–83 (1945).

43. Holm, S. A simple sequentially rejective multiple test procedure. *Scand. J. Stat.* **6**, 65–70 (1979).
44. McNemar, Q. Note on the sampling error of the difference between correlated proportions or percentages. *Psychometrika* **12**, 153–157 (1947).

Figures

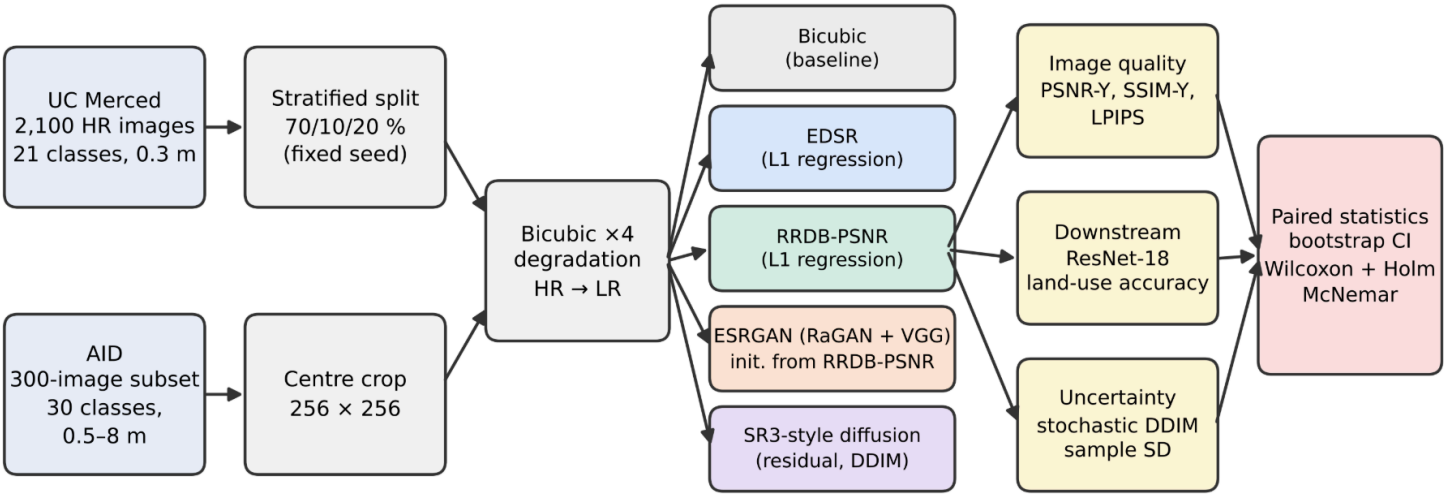


Figure 1

Overview of the experimental pipeline. HR images are degraded by ×4 bicubic downsampling; five methods reconstruct them; outputs are assessed by image-quality metrics, by a fixed land-use classifier and, for diffusion, by sample uncertainty; all comparisons are made with paired statistics.

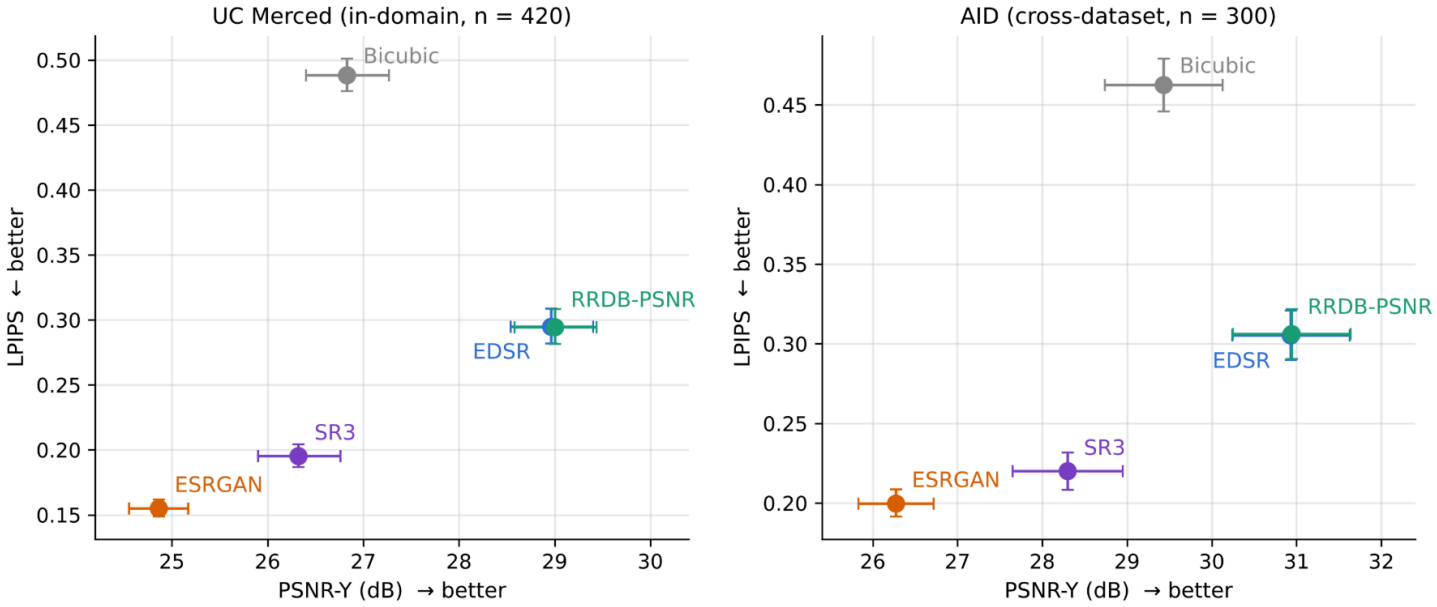


Figure 2

Perception–distortion plane. Each point is a method's mean PSNR-Y (horizontal, higher is better) and LPIPS (vertical, lower is better) with 95 % bootstrap confidence intervals, on UC Merced (left) and AID (right). The lower-right corner is ideal and is unattainable beyond the bound of Blau and Michaeli[21].

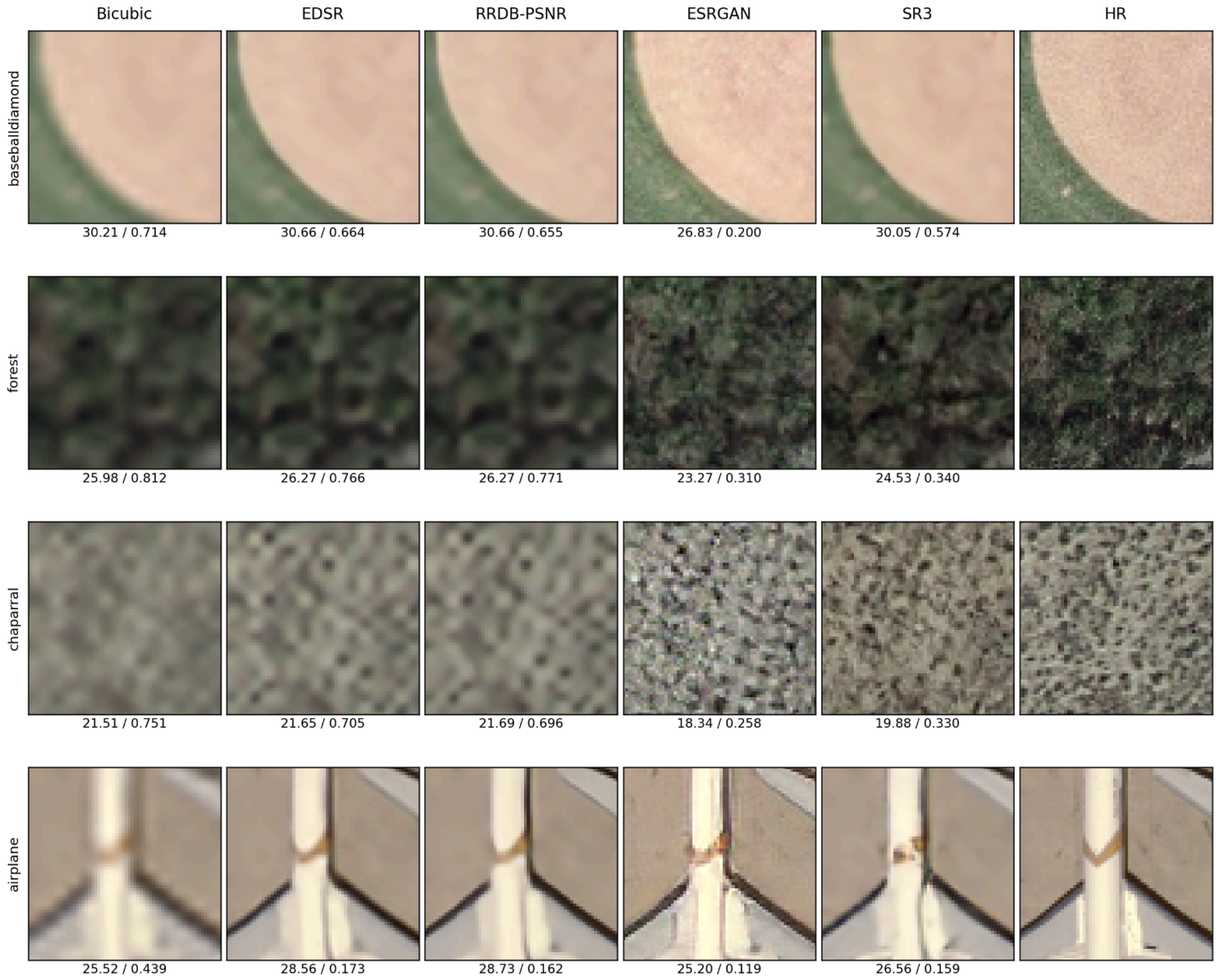


Figure 3

Visual comparison on UC Merced (central 64 × 64 crop of the 256 × 256 output). Rows are selected automatically: the images with the largest LPIPS difference between EDSR and ESRGAN plus the image with median EDSR PSNR, to avoid selection by eye. Numbers give PSNR-Y (dB) / LPIPS for each crop.

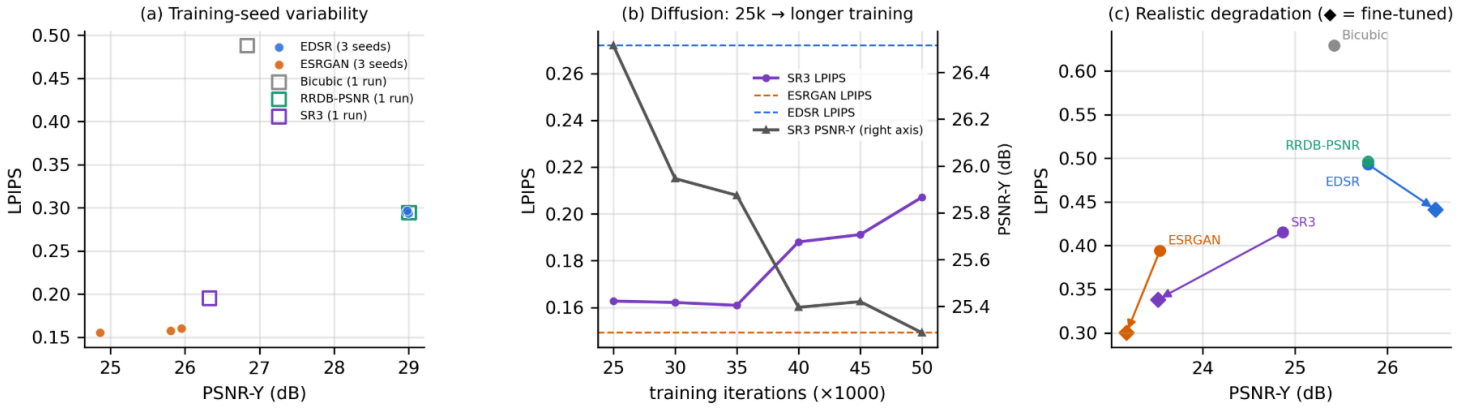


Figure 4

Robustness of the findings. (a) UC Merced PSNR-Y and LPIPS of each independently trained run; circles show repeated seeds and squares single runs. (b) Learning curve of the diffusion model when training is continued, with EDSR and ESRGAN LPIPS on the same images as dashed references. (c) Methods on realistically degraded inputs; arrows lead from each bicubic-trained model to its degradation-aware fine-tuned version (diamonds).

Supplementary Files

This is a list of supplementary files associated with this preprint. Click to download.

- SupplementaryInformation.docx