SPARK: SPATIAL POLICY-DRIVEN ADAPTIVE REINFORCEMENT LEARNING FOR KNOWLEDGE DISTILLATION

Anonymous authors
Paper under double-blind review

ABSTRACT

Low-bit quantization enables the deployment of image restoration (IR) networks on resource-constrained devices, but the resulting rounding noise disproportionately degrades high-frequency regions such as edges and fine textures. Existing knowledge distillation (KD) methods apply the distillation signal uniformly across spatial locations and thus ignore the varying reconstruction difficulty within an image. To address this limitation, we propose SPARK (Spatial Policy-driven Adaptive Reinforcement learning for Knowledge distillation), a framework that allocates distillation effort with a lightweight reinforcement learning (RL) policy. At each training step, a difficulty feature extractor computes four signals, namely Laplacian variance, pixel variance, student reconstruction error, and the teacher-student gap, and a compact policy CNN maps them to a stochastic spatial weight map that modulates the KD loss during quantization-aware training (QAT). SPARK is task-agnostic, adds no inference cost, and integrates into existing QAT pipelines without architectural changes. Experiments on low-light enhancement, denoising, and super-resolution with four teacher-student configurations show that SPARK consistently outperforms post-training quantization (PTQ), QAT, and state-of-the-art KD methods, yielding the reconstruction quality closest to the full-precision teacher.

1 INTRODUCTION

Deep neural networks dominate low-level vision tasks such as super-resolution (SR), denoising, dehazing, and low-light enhancement (LLE) (Zhou et al., 2025; Rasool et al., 2025; Zhang & Yan, 2025). This progress has come at a high computational cost, since state-of-the-art (SOTA) image restoration (IR) networks require billions of multiply-accumulate operations per image, which hinders deployment on smartphones, security cameras, drones, and embedded vision systems (Rasool et al., 2025; Mardieva et al., 2024). Low-bit quantization, which uses 8 or fewer bits, reduces this cost by replacing floating-point weights and activations with compact integer representations, yielding substantial memory, latency, and energy savings (Hong et al., 2022; Tu et al., 2023).

However, quantization degrades output quality far more in IR than in high-level tasks such as classification, detection, or segmentation (Hong et al., 2022). Pixel-regression objectives are highly sensitive to the rounding noise of low-bit operators, and the absence of a softmax-style smoothing stage lets small numerical errors propagate directly to the output. Quantized IR models therefore fall short of their full-precision counterparts (Tu et al., 2023), with the largest drops on edges, fine textures, and repetitive structures, which are precisely the regions that human observers attend to most.

Knowledge distillation (KD) is a well-established approach for training lightweight models under the supervision of a stronger teacher (Hinton et al., 2015; He et al., 2020; Zhang & Yan, 2025). However, existing KD formulations for IR apply a single distillation loss averaged uniformly over all pixels. Smooth regions, such as skies or homogeneous surfaces, are easy for both networks to reconstruct, so matching the teacher there provides little useful gradient and can dominate the optimization at the expense of harder regions. Edges and textures, in contrast, occupy only a small fraction of the pixels, yet they account for a disproportionate share of the perceptual error and are

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

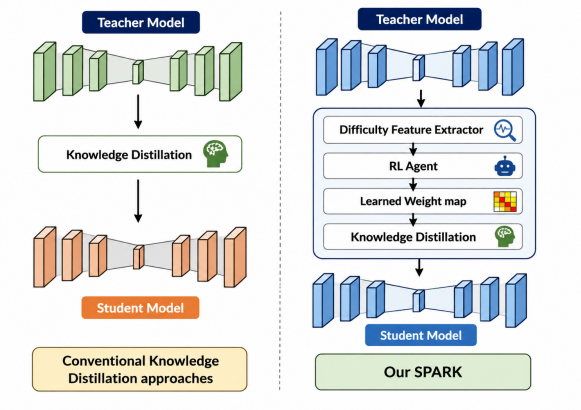

Figure 1: Conventional KD versus SPARK. Conventional KD (left) applies uniform supervision across all spatial regions, whereas SPARK (right) uses an RL policy that generates a spatially adaptive weight map to focus distillation on challenging regions such as edges and fine textures.

the regions most sensitive to quantization noise. Consequently, a uniformly weighted loss underemphasizes the most important locations.

Hand-crafted alternatives such as edge masks (Zhu et al., 2019), gradient-based weighting, and fixed difficulty heuristics (He et al., 2020; Zhang & Yan, 2025) commit in advance to a single notion of where to focus and remain static throughout training. The appropriate spatial allocation of distillation effort is instead a state-dependent decision, since it depends on the current capacity of the student, the residual gap to the teacher, the local image statistics, and the degradation at hand, all of which evolve during training. SPARK therefore formulates spatial KD weighting as a decision problem and learns it with a lightweight reinforcement learning (RL) policy.

Concretely, a Spatial KD Policy observes four difficulty features of the current sample at each training step, namely Laplacian variance (edges), pixel variance (texture), student reconstruction error, and the teacher-student gap, and emits a stochastic spatial weight map. This map weights a mean squared error (MSE) distillation loss between the quantized student and the frozen full-precision teacher, while an L1 reconstruction loss anchors the student to the ground truth (GT). The sampled action is detached before it enters the KD loss, so the policy is trained by a reward tied to reconstruction quality rather than by pixel-wise gradients, which allows it to explore allocations that a purely differentiable objective would not reach. The policy consists of three convolutional layers on an 8×8 grid and is discarded after training, so it adds no inference cost and requires no change to the student architecture. Figure 1 contrasts conventional KD with SPARK. The main contributions of this work are summarized as follows.

- The spatial uniformity of existing KD losses is identified as a key bottleneck in low-bit IR, motivating spatially adaptive distillation for quantization-robust restoration.
- SPARK is introduced as an RL-driven spatial KD policy that learns where distillation matters most during QAT. Rather than weighting all pixels equally, SPARK maps interpretable difficulty features to a per-region weight map that concentrates teacher supervision on edges and textures, where quantization errors are most damaging. Its actions are sampled from a Gaussian distribution and detached from the gradient graph, which decouples policy learning from the supervised objective and stabilizes training. Since the policy operates only during training, SPARK incurs no computational overhead at inference. To the best of our knowledge, SPARK is the first RL-based spatial weighting strategy for KD in low-bit IR, and it generalizes across LLE, denoising, and SR without task-specific modification.
- Experiments on three IR tasks with four teacher-student configurations show that SPARK consistently outperforms both uniform KD with QAT and state-of-the-art KD methods. Ablation studies further validate the learned spatial policy and the choice of policy grid. Analysis of the learned weight maps shows that the policy assigns higher weights to edges and fine details than to flat regions, and that these weights correlate positively with per-

patch student error, whereas a fixed teacher-error heuristic is negatively correlated with it (Appendix B).

2 Related Work

Knowledge distillation for low-level vision. KD transfers knowledge from a teacher network to a lightweight student and has been widely explored for IR. He et al. (2020) proposed FAKD, which aligns spatial affinity matrices derived from teacher and student feature maps. Li et al. (2025) proposed MiPKD, which narrows the teacher-student semantic gap for SR through feature mixing and stochastic block mixing in the latent space. Zhou et al. (2025) introduced DCKD, which adapts the distilled solution space to the learning state of the student through dynamic contrastive regularization and aligns pixel-level category distributions with a distribution mapping module. These methods nevertheless treat all spatial regions equally and lack a principled mechanism for directing distillation effort toward harder regions, a limitation that becomes critical under the constrained capacity of low-bit QAT (Makhov et al., 2025).

Reinforcement learning in vision. RL introduces adaptive decision-making into vision systems where deterministic supervision is insufficient (Le et al., 2022). In high-level vision, RL has been applied to neural architecture search (Wu et al., 2019), attention, active object localization, and early-exit policies for efficient inference (Le et al., 2022). In low-level vision, Yu et al. (2018) proposed RL-Restore, which trains an agent to select restoration operations sequentially from a predefined toolbox for compound, unknown degradations, and ReLLIE (Zhang et al., 2021) formulates LLE as sequential curve estimation with a lightweight policy trained on non-reference rewards. The use of RL to guide distillation itself, namely to decide where and how strongly a student should mimic its teacher, remains largely unexplored, particularly under low-bit QAT.

3 Proposed Method

In SPARK (Figure 2), a compact quantized student is trained to match the reconstruction quality of a larger pretrained teacher. Instead of distilling uniformly, an RL policy assigns a relevance weight to each image region according to its reconstruction difficulty.

3.1 Teacher and Student Networks

Fig. 2 illustrates the overall teacher-student framework. The teacher is a high-capacity IR model pretrained to convergence and frozen throughout training. Given a degraded input, it produces an output $\hat{I}_t$ with the same spatial size as the ground truth I_{GT}, which serves as a soft supervisory signal for the student. The student is either a quantized copy of the teacher architecture (self-quantized) or a lightweight quantized network, trained with QAT for deployment on resource-constrained hardware. Fake quantization is applied to weights and activations in the forward pass to mimic deployment behavior, while full-precision gradients are retained for optimization. The student processes the same input and produces $\hat{I}_s$, as shown in Fig. 2.

3.2 Difficulty Feature Extractor

Four complementary difficulty maps are computed without gradient flow, so the extractor acts purely as an observation. The Laplacian variance $D_1 = \mathrm{Var}(\Delta \hat{I}_s)$ highlights high-frequency regions where strong edges make reconstruction intrinsically difficult. The pixel variance $D_2 = \mathrm{Var}(\hat{I}_s)$ measures local texture complexity, since highly textured areas demand more representational capacity. The student error $D_3 = |\hat{I}_s - I_{GT}|$ localizes regions that the student currently fails to recover, and the teacher-student gap $D_4 = |\hat{I}_t - \hat{I}_s|$ identifies regions with the greatest potential for knowledge transfer. The maps are stacked into $D = [D_1, D_2, D_3, D_4] \in \mathbb{R}^{4\times H\times W}$ per sample. Together, they couple local structural difficulty (D_1, D_2) with the evolving capability of the student (D_3, D_4), which lets the policy adapt its allocation as training progresses.

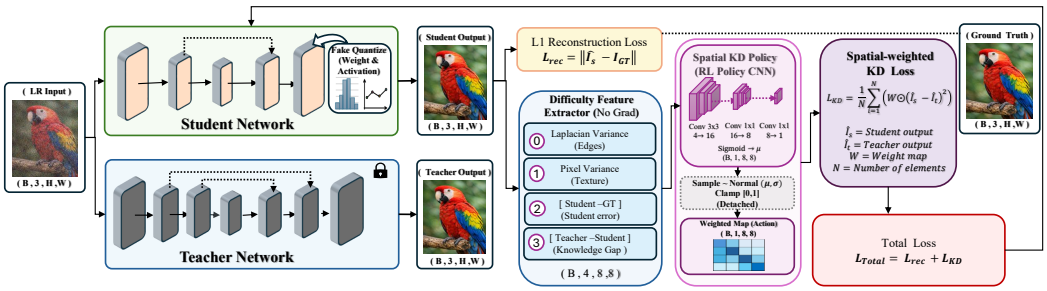

Figure 2: Overview of SPARK. The difficulty feature extractor summarizes the teacher and student outputs, the RL policy CNN converts these features into a spatial weight map, and the resulting spatially weighted KD loss is combined with the reconstruction loss to form the training objective.

3.3 Spatial Policy Network

Spatial KD weighting is cast as a contextual bandit, that is, a Markov decision process of horizon one, since a single weight map is chosen per iteration and no state is carried across iterations. A policy-gradient formulation is adopted since the action space is continuous and high-dimensional, and a stochastic policy supports exploration without discretizing importance scores.

State. The state is the difficulty tensor average-pooled to the policy grid, $s = \mathrm{AvgPool}_{8\times 8}(D) \in \mathbb{R}^{4\times 8\times 8}$.

Action. A three-layer CNN f_θ predicts the per-cell mean through a sigmoid, and the policy factorizes over the 64 cells as an isotropic Gaussian,

$$\mu_\theta(s) = \mathrm{sig}\big(f_\theta(s)\big), \qquad \pi_\theta(a \mid s) = \prod_{i=1}^{64} \mathcal{N}\big(a_i \,;\, \mu_\theta(s)_i, \sigma^2\big), \tag{1}$$

where $\sigma = 0.1$ is a fixed exploration level rather than a predicted quantity. Each sampled action is clipped to $[0, 1]$ to form the weight map $W \in [0, 1]^{8\times 8}$, while its log-probability is evaluated before clipping so that the score function matches the sampling distribution. Sampling also keeps the policy from collapsing prematurely into a rigid deterministic weighting.

Reward. The reward measures the reconstruction improvement over uniform weighting,

$$r = \alpha\, \Delta\mathrm{PSNR}(\hat{I}_s, I_{GT}) + \beta\, \Delta\mathrm{SSIM}(\hat{I}_s, I_{GT}), \tag{2}$$

where ΔPSNR and ΔSSIM denote the change relative to a student output obtained without spatial weighting, and α and β balance the two terms. A positive reward indicates that the sampled weight map improved reconstruction over the uniform baseline, whereas a negative reward penalizes harmful configurations.

Policy update. The policy maximizes $J(\theta) = \mathbb{E}[r(s, a)]$ with REINFORCE and a baseline,

$$\nabla_\theta J(\theta) = \mathbb{E}\big[(r - b)\, \nabla_\theta \log \pi_\theta(a \mid s)\big], \tag{3}$$

where b is a running mean of previous rewards, updated as $b \leftarrow \eta\, b + (1-\eta)\, r$ after each policy step, so it reduces gradient variance without biasing the gradient. The student and the policy are updated in a single loop but optimize disjoint objectives, since W is detached before entering the KD loss. The student is driven only by $\mathcal{L}_{\mathrm{total}}$ and the policy only by the reward, which prevents the policy from trivially zeroing all weights to shrink the KD loss. Algorithm 1 in Appendix A summarizes one training iteration.

3.4 Training Objective

A pixel-wise L1 loss anchors the student to the ground truth in every region, and the policy weight map modulates an MSE distillation loss toward the teacher,

$$\mathcal{L}_{\mathrm{rec}} = \frac{1}{N}\big\|\hat{I}_s - I_{GT}\big\|_1, \qquad \mathcal{L}_{\mathrm{KD}} = \frac{1}{N}\sum_{i=1}^{N} \tilde{W}_i \big(\hat{I}_{s,i} - \hat{I}_{t,i}\big)^2, \tag{4}$$

Table 1: Comparison on LLE (LOLv1) and denoising (SIDD). The teacher is full-precision MIRNet, and the INT8 students are self-quantized MIRNet and DnCNN. Best results among the quantized students are in **bold**.

Approach	Student	LLE (LOLv1)			Denoising (SIDD)		
		PSNR↑	SSIM↑	LPIPS↓	PSNR↑	SSIM↑	LPIPS↓
Full precision (teacher)	None	24.14	0.801	0.150	39.72	0.959	0.320
PTQ	Self-quantized MIRNet	20.10	0.751	0.194	28.42	0.510	0.577
QAT		20.32	0.762	0.185	29.15	0.555	0.539
KD + QAT		21.45	0.778	0.179	29.57	0.571	0.527
FAKD + QAT		20.20	0.760	0.189	31.80	0.687	0.399
DCKD + QAT		19.92	0.751	0.192	31.72	0.680	0.402
SLKD + QAT		21.67	0.780	0.178	32.14	0.700	0.389
SPARK (ours)		**22.01**	**0.785**	**0.176**	**33.89**	**0.807**	**0.325**
PTQ	Quantized DnCNN	18.47	0.719	0.261	28.12	0.508	0.590
QAT		19.01	0.720	0.211	28.45	0.520	0.579
KD + QAT		19.45	0.742	0.204	28.87	0.542	0.550
FAKD + QAT		20.88	0.761	0.191	29.36	0.558	0.541
DCKD + QAT		19.84	0.751	0.198	29.25	0.550	0.556
SLKD + QAT		20.97	0.764	0.190	29.34	0.563	0.530
SPARK (ours)		**21.32**	**0.778**	**0.181**	**31.71**	**0.677**	**0.399**

where N is the number of output elements and $\tilde{W}$ is the detached weight map W resized to the output resolution. The student minimizes $\mathcal{L}_{\text{total}} = \mathcal{L}_{\text{rec}} + \mathcal{L}_{\text{KD}}$, which concentrates distillation pressure on edges, textures, and high-error regions while smooth, well-reconstructed regions contribute less, so the quantized student spends its limited capacity where it matters most.

4 Experiments

4.1 Experimental Setup

Datasets and metrics. LLE is evaluated on LOLv1 (Wei et al., 2018), which contains 485 training and 15 test images captured under diverse low-light conditions. Denoising is evaluated on SIDD (Abdelhamed et al., 2018), with 30,000 noisy-clean pairs for training and 1,280 pairs for testing. SR is evaluated on Urban100 (Huang et al., 2015), which contains 100 high-resolution images of urban scenes rich in buildings, windows, and repetitive structures. Restoration quality is reported with PSNR, SSIM, and LPIPS (Zhang et al., 2018).

Configurations and baselines. Four teacher-student configurations are considered. For LLE and denoising, a full-precision MIRNet (Zamir et al., 2022) teacher distills into an INT8 self-quantized MIRNet or an INT8 DnCNN (Zhang et al., 2017). For SR, a full-precision RFDN (Kong et al., 2022) teacher distills into an INT8 self-quantized RFDN or an INT8 SRCNN (Dong et al., 2015). SPARK is compared with PTQ, QAT, and four distillation methods combined with QAT, namely KD (Hinton et al., 2015), FAKD (He et al., 2020), DCKD (Zhou et al., 2025), and SLKD (Zhang & Yan, 2025).

Implementation details. The student is trained from scratch with $\mathcal{L}_{\text{total}}$ while the teacher remains frozen. Optimization uses Adam with momentum parameters 0.9 and 0.999 and an initial learning rate of 10^{-4}, decayed by cosine annealing over 500,000 steps, on non-overlapping 128×128 patches with a batch size of 4 and random flips and 90° rotations. The framework is implemented in PyTorch (Paszke et al., 2019) with its native quantization toolkit, and all experiments run on a single NVIDIA GeForce RTX 3080 GPU with 12 GB of memory. The policy network is discarded after training, so the deployed student incurs no additional parameters or latency beyond the INT8 backbone, and Appendix C reports its FP32 and INT8 inference throughput. Appendix E provides further quantization details.

4.2 Quantitative Results

Table 1 reports LLE and denoising results. SPARK ranks first on every metric for both INT8 students. With the self-quantized MIRNet student, it attains 22.01 dB PSNR, 0.785 SSIM, and 0.176

Table 2: Comparison on Urban100 SR at $\times 2$ and $\times 4$. The teacher is full-precision RFDN, and the INT8 students are self-quantized RFDN and SRCNN. Best results among the quantized students are in **bold**.

Scale	Approach	Self-quantized RFDN			Quantized SRCNN		
		PSNR↑	SSIM↑	LPIPS↓	PSNR↑	SSIM↑	LPIPS↓
$\times 2$	Full precision (teacher)	32.17	0.928	0.178		N/A	
	PTQ	30.10	0.725	0.257	27.11	0.851	0.321
	QAT	31.04	0.738	0.244	27.85	0.868	0.310
	KD + QAT	31.33	0.772	0.229	28.14	0.872	0.286
	FAKD + QAT	31.45	0.805	0.216	28.20	0.879	0.278
	DCKD + QAT	31.11	0.796	0.228	28.00	0.871	0.282
	SLKD + QAT	31.57	0.809	0.216	28.28	0.879	0.277
	SPARK (ours)	**32.01**	**0.825**	**0.191**	**28.65**	**0.884**	**0.265**
$\times 4$	Full precision (teacher)	26.11	0.785	0.312		N/A	
	PTQ	23.36	0.654	0.426	22.56	0.690	0.489
	QAT	24.01	0.681	0.415	22.70	0.693	0.458
	KD + QAT	24.56	0.699	0.411	23.32	0.711	0.445
	FAKD + QAT	25.07	0.712	0.405	23.45	0.715	0.430
	DCKD + QAT	25.09	0.710	0.414	23.52	0.719	0.430
	SLKD + QAT	25.19	0.712	0.401	23.54	0.715	0.429
	SPARK (ours)	**25.89**	**0.719**	**0.394**	**23.78**	**0.720**	**0.418**

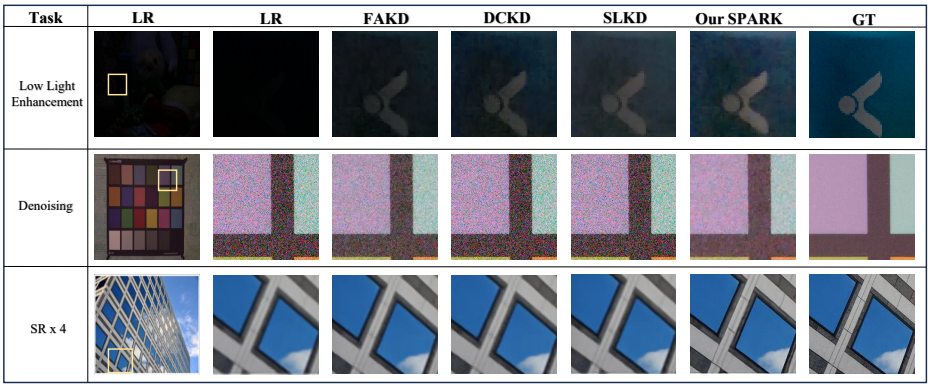

Figure 3: Qualitative comparison of SPARK with FAKD + QAT, DCKD + QAT, and SLKD + QAT on LLE (LOLv1), denoising (SIDD), and $\times 4$ SR (Urban100). The last column shows the GT.

LPIPS on LOLv1 and 33.89 dB PSNR, 0.807 SSIM, and 0.325 LPIPS on SIDD, the smallest gap to the full-precision teacher among all compared methods. KD, FAKD, DCKD, and SLKD partially recover the fidelity lost to quantization, yet none matches SPARK on either task, which indicates that spatially uniform distillation leaves a consistent margin unexploited. PTQ degrades sharply and trails SPARK by 5.47 dB on SIDD with the MIRNet student, underlining the fragility of quantization-unaware training under pixel-level objectives.

Table 2 reports SR results on Urban100. At $\times 2$, SPARK reaches 32.01 dB PSNR, 0.825 SSIM, and 0.191 LPIPS with the RFDN student, exceeding PTQ by 1.91 dB and narrowing the gap to the teacher to 0.16 dB. At $\times 4$, where quantization most damages the high-frequency structures that dominate reconstruction, the margin over PTQ widens to 2.53 dB for RFDN, and SPARK still improves over PTQ by 1.22 dB for SRCNN. SPARK achieves the highest PSNR and SSIM and the lowest LPIPS at both scales and for both students, which shows that the learned spatial policy transfers teacher fidelity more faithfully than uniform distillation.

4.3 Qualitative Results

Figure 3 compares SPARK with FAKD, DCKD, and SLKD. For LLE, the baselines struggle to recover brightness and structural detail from severely underexposed inputs, whereas SPARK recovers

Table 3: Ablation of the spatial policy and the policy grid size on LLE (LOLv1) and denoising (SIDD). Best results are in **bold**.

Approach	Student	LLE (LOLv1)			Denoising (SIDD)		
		PSNR↑	SSIM↑	LPIPS↓	PSNR↑	SSIM↑	LPIPS↓
RL policy (4×4 grid)	Self-quantized MIRNet	21.94	0.780	0.183	33.80	0.799	0.329
RL policy (8×8 grid)		**22.01**	**0.785**	**0.176**	**33.89**	**0.807**	**0.325**
RL policy (16×16 grid)		21.97	0.781	0.181	33.85	0.801	0.325
No policy (uniform L1 KD)		21.45	0.778	0.179	29.57	0.571	0.527
No policy (uniform MSE KD)		21.48	0.778	0.185	29.63	0.572	0.531
RL policy (4×4 grid)	Quantized DnCNN	21.26	0.765	0.190	31.62	0.676	0.405
RL policy (8×8 grid)		**21.32**	**0.778**	**0.181**	**31.71**	**0.677**	**0.399**
RL policy (16×16 grid)		21.28	0.771	0.185	31.68	0.676	0.402
No policy (uniform L1 KD)		19.45	0.742	0.204	28.87	0.542	0.550
No policy (uniform MSE KD)		19.46	0.749	0.218	28.91	0.554	0.567

Table 4: Ablation of the spatial policy and the policy grid size for SR on Urban100 at $\times 2$ and $\times 4$. Best results are in **bold**.

Scale	Approach	Self-quantized RFDN			Quantized SRCNN		
		PSNR↑	SSIM↑	LPIPS↓	PSNR↑	SSIM↑	LPIPS↓
$\times 2$	RL policy (4×4 grid)	31.84	0.823	0.199	28.59	0.881	0.270
	RL policy (8×8 grid)	**32.01**	**0.825**	**0.191**	**28.65**	**0.884**	**0.265**
	RL policy (16×16 grid)	31.91	0.823	0.195	28.62	0.882	0.268
	No policy (uniform L1 KD)	31.33	0.772	0.229	28.14	0.872	0.286
	No policy (uniform MSE KD)	31.41	0.780	0.230	28.16	0.876	0.291
$\times 4$	RL policy (4×4 grid)	25.80	0.702	0.402	23.64	0.711	0.426
	RL policy (8×8 grid)	**25.89**	**0.719**	**0.394**	**23.78**	**0.720**	**0.418**
	RL policy (16×16 grid)	25.85	0.704	0.397	23.66	0.715	0.422
	No policy (uniform L1 KD)	24.56	0.699	0.411	23.32	0.711	0.445
	No policy (uniform MSE KD)	24.58	0.701	0.422	23.40	0.711	0.457

scene content and contrast closest to the GT. For denoising, the baselines leave residual noise and color distortion in smooth regions, whereas SPARK produces cleaner outputs with more uniform color and sharper edges. For $\times 4$ SR, the baselines over-smooth and alias high-frequency structures such as diagonal architectural edges, whereas SPARK reconstructs finer structure and more accurate geometry.

4.4 Ablation Study

Two ablations vary one component while all other settings are fixed. The first removes the policy and applies a spatially uniform L1 or MSE distillation loss, and the second varies the policy grid over 4×4, 8×8, and 16×16. Tables 3 and 4 report the results. Removing the policy degrades every metric relative to the 8×8 policy for all students, tasks, and scales, with the largest drop on denoising, where SSIM falls from 0.807 to 0.571 for the MIRNet student. The learned spatial weighting, rather than the choice of distillation loss, is therefore the main source of the gains. The 8×8 grid performs best in every setting, likely since a coarser 4×4 grid cannot localize supervision finely enough, whereas a finer 16×16 grid enlarges the action space and increases policy variance without further benefit. The margins between grid sizes are small, indicating that SPARK is robust to this choice within a reasonable range. Appendix B complements these results with an analysis of the learned weight maps, where the policy weights correlate positively with per-patch student error on SIDD (Pearson $r = 0.158$), whereas a fixed teacher-error heuristic is anti-correlated ($r = -0.121$).

5 Conclusion

This work presents SPARK, a spatial policy-driven RL framework for KD in low-bit IR. Unlike existing KD methods that supervise all regions uniformly, SPARK uses a lightweight policy to produce a spatially adaptive weight map from interpretable difficulty features, concentrating distillation on edges, textures, and high-error regions. The policy is learned from a reconstruction-quality reward,

which enables exploration of spatial allocations beyond those reachable by purely differentiable objectives. Experiments on LLE, denoising, and SR show that SPARK consistently outperforms SOTA KD methods across student architectures and scales, and ablations confirm that the learned spatial weighting drives the improvement over uniform L1 and MSE distillation. Appendix D discusses the remaining limitations, including the reliance on paired training data and the evaluation at INT8 only. SPARK thus establishes a foundation for spatially adaptive distillation in low-level vision and opens a direction for RL-guided compression of IR networks.

AI USE STATEMENT

Generative AI tools were used in this work only to correct grammar and improve the readability of the manuscript. Generative AI tools were not used for research ideation, methodology design, implementation, data generation, or the analysis and interpretation of results, and the remaining required-disclosure tasks are not applicable to this work. All edited text was reviewed by the authors, who take full responsibility for the final content of this work.

REFERENCES

Abdelrahman Abdelhamed, Stephen Lin, and Michael S Brown. A high-quality denoising dataset for smartphone cameras. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 1692–1700, 2018.

Chao Dong, Chen Change Loy, Kaiming He, and Xiaoou Tang. Image super-resolution using deep convolutional networks. *IEEE transactions on pattern analysis and machine intelligence*, 38(2): 295–307, 2015.

Zibin He, Tao Dai, Jian Lu, Yong Jiang, and Shu-Tao Xia. Fakd: Feature-affinity based knowledge distillation for efficient image super-resolution. In *2020 IEEE International Conference on Image Processing (ICIP)*, pp. 518–522. IEEE, 2020.

Geoffrey Hinton, Oriol Vinyals, and Jeff Dean. Distilling the knowledge in a neural network. *arXiv preprint arXiv:1503.02531*, 2015.

Cheeun Hong, Heewon Kim, Sungyong Baik, Junghun Oh, and Kyoung Mu Lee. Daq: Channel-wise distribution-aware quantization for deep image super-resolution networks. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pp. 2675–2684, 2022.

Jia-Bin Huang, Abhishek Singh, and Narendra Ahuja. Single image super-resolution from transformed self-exemplars. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 5197–5206, 2015.

Fangyuan Kong, Mingxi Li, Songwei Liu, Ding Liu, Jingwen He, Yang Bai, Fangmin Chen, and Lean Fu. Residual local feature network for efficient super-resolution. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 766–776, 2022.

Ngan Le, Vidhiwar Singh Rathour, Kashu Yamazaki, Khoa Luu, and Marios Savvides. Deep reinforcement learning in computer vision: a comprehensive survey. *Artificial Intelligence Review*, 55(4):2733–2819, 2022.

Simiao Li, Yun Zhang, Wei Li, Hanting Chen, Wenjia Wang, Bingyi Jing, Shaohui Lin, and Jie Hu. Knowledge distillation with multi-granularity mixture of priors for image super-resolution. In *International Conference on Learning Representations*, volume 2025, pp. 88752–88768, 2025.

Denis Makhov, Ruslan Ostapets, Irina Zhelavskaya, and Dehua Song. Prefilt: Prefiltering for fully quantized image restoration neural networks. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 4042–4051, 2025.

Sevara Mardieva, Shabir Ahmad, Sabina Umirzakova, MJ Aashik Rasool, and Taeg Keun Whangbo. Lightweight image super-resolution for iot devices using deep residual feature distillation network. *Knowledge-Based Systems*, 285:111343, 2024.

Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, et al. Pytorch: An imperative style, high-performance deep learning library. *Advances in neural information processing systems*, 32, 2019.

MJ Aashik Rasool, Shabir Ahmed, SMA Sharif, Mardieva Sevara, and Taeg Keun Whangbo. Pixelcraftsr: Efficient super-resolution with multi-agent reinforcement for edge devices. *Sensors*, 25 (7):2242, 2025.

Zhijun Tu, Jie Hu, Hanting Chen, and Yunhe Wang. Toward accurate post-training quantization for image super resolution. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 5856–5865, 2023.

Chen Wei, Wenjing Wang, Wenhan Yang, and Jiaying Liu. Deep retinex decomposition for low-light enhancement. *arXiv preprint arXiv:1808.04560*, 2018.

Bichen Wu, Xiaoliang Dai, Peizhao Zhang, Yanghan Wang, Fei Sun, Yiming Wu, Yuandong Tian, Peter Vajda, Yangqing Jia, and Kurt Keutzer. Fbnet: Hardware-aware efficient convnet design via differentiable neural architecture search. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 10734–10742, 2019.

Ke Yu, Chao Dong, Liang Lin, and Chen Change Loy. Crafting a toolchain for image restoration by deep reinforcement learning. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 2443–2452, 2018.

Syed Waqas Zamir, Aditya Arora, Salman Khan, Munawar Hayat, Fahad Shahbaz Khan, Ming-Hsuan Yang, and Ling Shao. Learning enriched features for fast image restoration and enhancement. *IEEE transactions on pattern analysis and machine intelligence*, 45(2):1934–1948, 2022.

Kai Zhang, Wangmeng Zuo, Yunjin Chen, Deyu Meng, and Lei Zhang. Beyond a gaussian denoiser: Residual learning of deep cnn for image denoising. *IEEE transactions on image processing*, 26 (7):3142–3155, 2017.

Richard Zhang, Phillip Isola, Alexei A Efros, Eli Shechtman, and Oliver Wang. The unreasonable effectiveness of deep features as a perceptual metric. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 586–595, 2018.

Rongkai Zhang, Lanqing Guo, Siyu Huang, and Bihan Wen. Rellie: Deep reinforcement learning for customized low-light image enhancement. In *Proceedings of the 29th ACM international conference on multimedia*, pp. 2429–2437, 2021.

Yongheng Zhang and Danfeng Yan. Knowledge distillation for image restoration: simultaneous learning from degraded and clean images. In *ICASSP 2025-2025 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 1–5. IEEE, 2025.

Yunshuai Zhou, Junbo Qiao, Jincheng Liao, Wei Li, Simiao Li, Jiao Xie, Yunhang Shen, Jie Hu, and Shaohui Lin. Dynamic contrastive knowledge distillation for efficient image restoration. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pp. 10861–10869, 2025.

Yousong Zhu, Chaoyang Zhao, Chenxia Han, Jinqiao Wang, and Hanqing Lu. Mask guided knowledge distillation for single shot detector. In *2019 IEEE international conference on multimedia and expo (ICME)*, pp. 1732–1737. IEEE, 2019.

Appendix

This appendix provides the full training procedure (Appendix A), an analysis of the learned weight map (Appendix B), inference-efficiency measurements (Appendix C), discussion of limitations (Appendix D), and quantization protocol (Appendix E).

A Training Procedure

Algorithm 1 summarizes one training iteration of SPARK. The student and the policy are updated in the same loop but optimize disjoint objectives, since the weight map is detached before it enters the KD loss. The frozen teacher provides the soft supervisory signal, and the difficulty feature extractor runs without gradients.

Algorithm 1: One training iteration of SPARK.

Input: Paired data $\{(I_{\text{in}}, I_{GT})\}$, frozen teacher T, quantized student S_ϕ, policy f_θ, exploration level σ, reward weights α and β, baseline decay η
Output: Trained student S_ϕ
Initialize ϕ and θ, and set the running baseline $b \leftarrow 0$;
for *each training step* k **do**
  Sample a batch (I_{in}, I_{GT});
  $\hat{I}_t \leftarrow T(I_{\text{in}})$; // frozen teacher, no gradient
  $\hat{I}_s \leftarrow S_\phi(I_{\text{in}})$; // fake-quantized forward pass
  Compute difficulty maps $D_1, \ldots, D_4$; // no gradient
  $s \leftarrow \text{AvgPool}_{8\times 8}([D_1, D_2, D_3, D_4])$;
  $\mu_\theta(s) \leftarrow \text{sig}(f_\theta(s))$;
  Sample $a \sim \mathcal{N}(\mu_\theta(s), \sigma^2\mathbf{I})$ and set $W \leftarrow \text{clip}(a, 0, 1)$;
  $\tilde{W} \leftarrow \text{detach}(\text{resize}(W))$; // to output resolution
  $\mathcal{L}_{\text{rec}} \leftarrow \frac{1}{N}\|\hat{I}_s - I_{GT}\|_1$;
  $\mathcal{L}_{\text{KD}} \leftarrow \frac{1}{N}\sum_i \tilde{W}_i\,(\hat{I}_{s,i} - \hat{I}_{t,i})^2$;
  Update ϕ with $\nabla_\phi(\mathcal{L}_{\text{rec}} + \mathcal{L}_{\text{KD}})$;
  $r \leftarrow \alpha\,\Delta\text{PSNR} + \beta\,\Delta\text{SSIM}$ relative to uniform weighting;
  Update θ with $(r - b)\,\nabla_\theta \log \pi_\theta(a \mid s)$;
  $b \leftarrow \eta\, b + (1 - \eta)\, r$;
end

B Analysis of the Learned Weight Map

B.1 Qualitative Comparison

Figure 4 examines whether SPARK allocates distillation effort where it is needed rather than uniformly, by comparing the learned weight map with a uniform map on two representative SIDD crops. The learned policy produces a structured map that assigns high weight to text, edges, and fine detail and suppresses flat, homogeneous regions, whereas the uniform map spends equal supervision on regions that are already easy to reconstruct. The policy thus recovers an interpretable spatial prior aligned with the high-frequency content where low-bit quantization is most damaging.

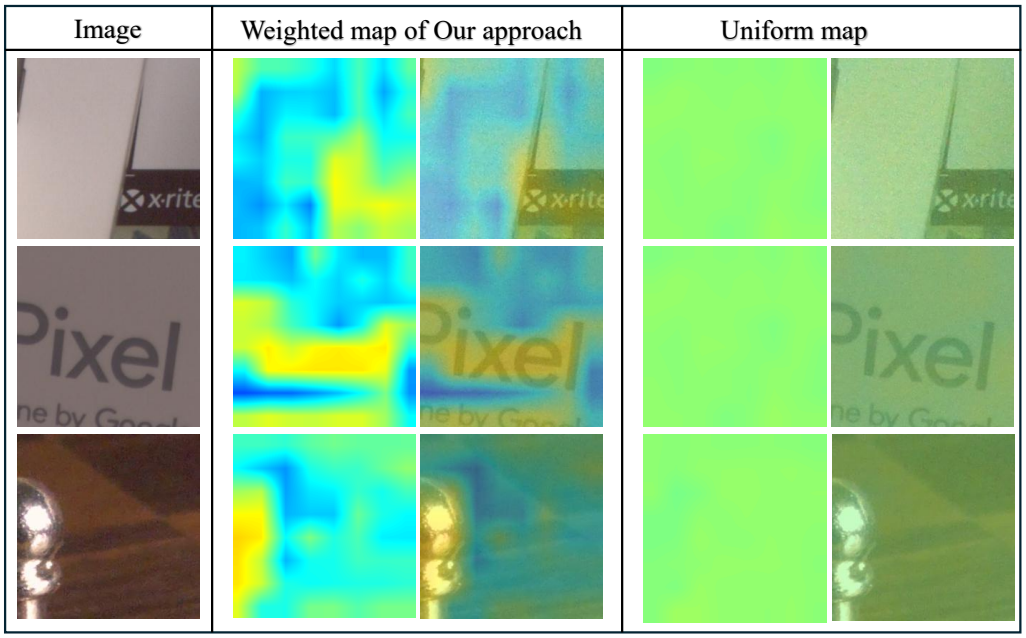

Figure 4: Learned weight maps compared with a uniform map on SIDD. For each sample, the panels show the input image, the learned weight map and its overlay, and the uniform map and its overlay. The learned policy assigns high weight to edges and text and suppresses flat regions, whereas the uniform map treats all locations equally.

B.2 Correlation with Reconstruction Difficulty

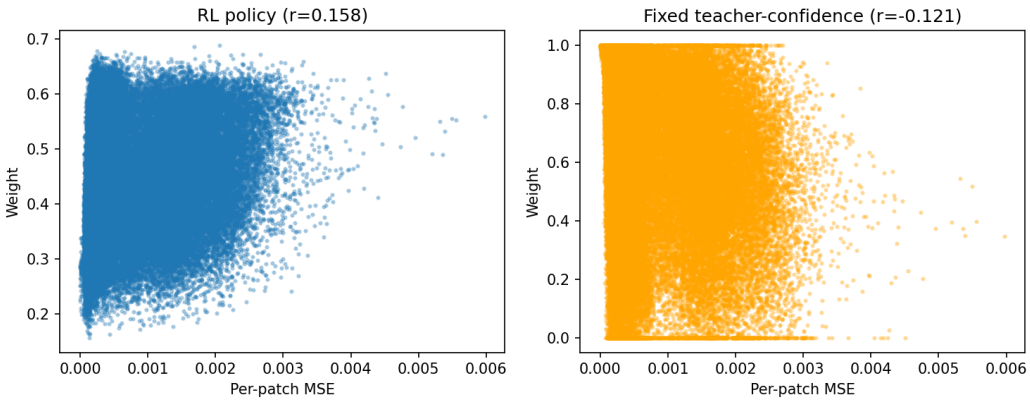

Figure 5: Per-patch correlation between the assigned weight and the reconstruction error of the student on SIDD. The learned policy correlates positively with student error (Pearson $r = 0.158$), whereas the fixed teacher-error heuristic is anti-correlated ($r = -0.121$).

To test whether the learned weighting captures more than a simple heuristic, the policy is compared with a fixed, training-free baseline that weights KD supervision inversely to the teacher's own error, $w_{\text{fixed}} = 1 - \text{normalize}(|\hat{I}_t - I_{GT}|)$, computed on the same 8×8 grid. This baseline encodes a natural hypothesis, namely that regions where the imperfect teacher deviates most from the ground truth should be discounted during distillation. For both schemes, the Pearson and Spearman correlations between each patch's weight and the per-patch reconstruction error of the student, $\|\hat{I}_s - I_{GT}\|_2^2$, are

measured over 81,920 patches from the 1,280 SIDD validation images (Figure 5). The learned policy correlates positively with this difficulty ($r = 0.158$, $\rho = 0.214$) and therefore assigns higher weight to harder regions, whereas the fixed baseline correlates negatively ($r = -0.121$, $\rho = -0.157$). Regions where the teacher is unreliable are thus not the regions where the student needs the most emphasis, since the two networks differ substantially in capacity and the errors of the teacher do not predict where the smaller student struggles. The fixed map is not under-expressive, as its weights vary more widely than those of the policy (standard deviation 0.246 against 0.097), yet it points in the wrong direction, which shows that useful allocation depends on the structure of the variation rather than its magnitude. The correlation gap remains consistent across independent training checkpoints, and the learned map varies meaningfully (standard deviation 0.097, range $[0.16, 0.69]$) rather than collapsing to a constant output.

C Inference Efficiency

Table 5 reports the inference throughput of each backbone in frames per second (FPS) under FP32 and INT8 precision, measured on RTX 3080 with 256×256 inputs. Since the policy network is discarded after training, the deployed student runs at the INT8 rate with no additional overhead. INT8 quantization yields a consistent speed-up across all four architectures, ranging from $1.89\times$ for DnCNN to $2.99\times$ for RFDN.

Table 5: Inference throughput in frames per second (FPS) for each backbone under FP32 and INT8 precision. Higher is better.

Model	**FPS (FP32)**	**FPS (INT8)**	**Speed-up**
MIRNet	7.56	17.64	$2.33\times$
DnCNN	91.00	172.00	$1.89\times$
RFDN	21.41	64.00	$2.99\times$
SRCNN	95.65	194.04	$2.03\times$

D Discussion and Limitations

Experiments on LLE, denoising, and SR show that SPARK consistently surpasses uniform KD and SOTA KD baselines under identical INT8 quantization and narrows the gap to the full-precision teacher. These results support the central hypothesis of this work, namely that the spatial allocation of distillation effort is a state-dependent decision that benefits from being learned rather than a fixed property of the task. By conditioning on interpretable difficulty features, namely local edge and texture statistics together with the student error and the teacher-student gap, the policy grounds its decisions in signals that are meaningful for restoration rather than in opaque latent representations. The stochastic Gaussian formulation is intended to maintain enough exploration to avoid trivial or degenerate weight configurations, although the effect of the exploration level σ is not ablated in this study.

Beyond the aggregate metrics, the weight-map analysis in Appendix B helps explain why the method works. The learned policy concentrates supervision on edges, text, and fine structure, which is the high-frequency content where low-bit rounding noise is most damaging, and its per-patch weight correlates positively with student error, whereas a fixed teacher-error heuristic correlates negatively. This distinction matters for model compression more broadly, since it indicates that the regions where a large teacher is itself unreliable are not the regions where a small quantized student most needs guidance, and that a learned, student-aware allocation captures a signal that this teacher-derived rule misses. To the best of our knowledge, SPARK is the first method to formulate spatial distillation for low-bit IR as an RL problem, and the same formulation may transfer to other capacity-constrained distillation settings.

Several limitations remain. First, the reward relies on global reconstruction metrics such as PSNR and SSIM, which may not fully reflect perceptual quality in every degradation regime, and a region-level or perceptual reward could provide more targeted feedback to the policy. Second, both the difficulty features and the reward reference the ground truth, so SPARK assumes paired training data

and does not directly extend to unpaired or real-world settings without clean targets, which would require no-reference difficulty signals and non-reference rewards in the spirit of ReLLIE (Zhang et al., 2021). Third, the evaluation is limited to INT8 quantization, and the behavior of SPARK at 4 bits or lower remains to be tested. Fourth, difficulty extraction and reward evaluation add computation during training, which this study does not quantify. All of these constraints concern training alone, since the policy network is discarded after training and the deployed student keeps its original quantized cost. Perceptual and region-level rewards, no-reference operation, and lower bit-widths therefore form a promising path toward a more general framework for RL-guided compression of restoration networks.

E QUANTIZATION PROTOCOL

This appendix specifies the quantization configuration shared by every quantized model in Tables 1 to 4. PTQ, QAT, KD + QAT, FAKD + QAT, DCKD + QAT, SLKD + QAT, and SPARK all operate on an identical quantized graph with identical quantizer settings, so the compared methods differ only in their training objective.

Framework and backend. Quantization uses the native PyTorch toolkit (`torch.ao.quantization`) in FX graph mode with the `fbgemm` backend. Fake quantization simulates INT8 arithmetic in the forward pass, and gradients propagate through the rounding operation with the straight-through estimator .

Weights. The weights of all convolutional layers are quantized to signed 8-bit integers with per-channel symmetric quantization along the output-channel axis. PTQ uses a per-channel min-max observer, and QAT uses a moving-average per-channel min-max observer. Per-channel granularity is adopted since weight ranges in IR networks vary considerably across output channels, and a single per-tensor scale would waste resolution on channels with narrow ranges.

Activations. Activations are quantized to unsigned 8-bit integers with per-tensor asymmetric quantization over the full 8-bit range, that is, the `reduce_range` option of the backend is disabled. PTQ uses a histogram observer that selects the clipping range minimizing the quantization error of each tensor. QAT and all distillation variants use a moving-average min-max observer with averaging constant 0.01, which tracks the evolving activation statistics during training.

Calibration and observer freezing. PTQ calibrates all observers on randomly selected 256×256 patches from the training split of each dataset, and no test image is used at any stage. During QAT, observers are updated for the first $20{,}000$ iterations and then frozen, after which the quantization parameters remain fixed and only the network weights continue to be optimized. The same calibration patches, freezing step, and random seed are shared by all compared methods.

First and last layers. The first convolution, which maps the input image to features, and the last convolution, which produces the restored image, are quantized to INT8 with the same settings as all other layers. The input image in $[0, 1]$ is quantized with a fixed scale of $1/255$ and zero point 0, which represents the 8-bit source images without loss. The final output after the global residual connection is dequantized and clamped to $[0, 1]$ before metric computation. For SRCNN, bicubic upsampling of the low-resolution input is performed in FP32 before the quantized network, following the original design (Dong et al., 2015).

Non-linear and element-wise operations. Residual additions, multiplications, and concatenations are implemented with quantized functional operators, each with its own output observer, so that every merge point receives an independent scale. Sigmoid gates, which appear in the dual attention units of MIRNet and the enhanced spatial attention of RFDN, use fixed quantization parameters with scale $1/256$ and zero point 0, matching their bounded output range. The softmax in the selective kernel feature fusion of MIRNet is executed in FP32 between dequantization and quantization stubs, and LeakyReLU and PReLU layers are quantized with a dedicated output observer. Pixel-shuffle upsampling in RFDN operates directly on quantized tensors, since it only rearranges values.

Evaluation protocol. PSNR and SSIM are computed after the output is rounded to 8-bit integers, with a data range of 255. SSIM uses an 11×11 Gaussian window with standard deviation 1.5. For LLE and denoising, metrics are computed on RGB channels. For SR, a border of s pixels is cropped for scale factor s, and metrics are computed on the Y channel of the YCbCr space. LPIPS uses the AlexNet backbone with inputs rescaled to $[-1, 1]$. SIDD metrics are averaged over the $1{,}280$ validation patches of size 256×256, and LOLv1 metrics are averaged over the 15 full-resolution test images.