LOOPLUT: 3D LOOKUP TABLES WITH PROGRESSIVE REGION REFINEMENT FOR REAL-TIME 4K IMAGE ENHANCEMENT

Anonymous authors
Paper under double-blind review

ABSTRACT

Color enhancement of 4K images must meet a quality target under a tight compute budget. Three-dimensional lookup tables (3D LUTs) dominate real-time enhancement because they decide at low resolution and apply a per-pixel lookup at full resolution. A single global LUT, however, is spatially invariant, so an underexposed shadow and a well-exposed region that share a pixel value receive identical corrections. Spatially heterogeneous demands cannot be expressed by such a mapping. We propose LoopLUT, a region-cascaded 3D LUT with progressive refinement. A global LUT performs the overall correction, followed by $K-1$ loop iterations. In each iteration a gating head predicts at low resolution the region that still needs correction, then builds a residual LUT from the color statistics of that region alone. The cascaded gates form a partition of unity, so the output is a per-pixel convex combination of the K lookup results. Fusion is therefore performed by the gates themselves, with no separate fusion module and no interpolation error accumulating across rounds. The decision stage runs at a fixed 256×256 resolution, independent of output resolution, so a 4K image costs only K pure lookups. Extensive experiments across four benchmarks show that LoopLUT improves PSNR by up to 2.81 dB over the strongest prior method, while keeping real-time throughput at 4K. The same decomposition also generalizes well to underwater enhancement datasets.

1 INTRODUCTION

Color enhancement of 4K images must preserve full-resolution detail within the compute budget of mobile and video applications. The dominant real-time route therefore decides at low resolution and executes at full resolution, as in HDRNet (Gharbi et al., 2017) and image-adaptive 3D LUTs (Zeng et al., 2022). A 3D LUT suits this route, since applying it costs one lookup per pixel.

This efficiency rests on a strong assumption, that enhancement is a single global RGB-to-RGB mapping independent of spatial position and image content. Two pixels with the same value therefore receive the same correction. Real photographic degradation, however, varies within a scene. An underexposed shadow needs brightening while an overexposed sky needs suppression, even when the two contain similar pixel values. On mixed-exposure scenes such as those of LCDP (Wang et al., 2022), a global LUT, however adaptive, can only compromise between shadows and highlights. Figure 1 shows how our cascaded gates instead split such a scene into differently corrected regions.

Existing work attacks this limitation from two directions, each leaving a gap. (i) Spatially aware operators add a weight map or local filters on top of the lookup (Wang et al., 2021; Gharbi et al., 2017; Moran et al., 2020). Their spatial decision is one-shot, predicted directly from the input, with no feedback in which an applied enhancement determines the next correction. Several also run the weight network at high resolution, so decision cost grows with output resolution. (ii) Iterative enhancement instead approaches the target in multiple steps, as in diffusion-based enhancement (Jiang et al., 2024), flow matching (Hu et al., 2025) and recurrent optical-flow refinement (Teed & Deng, 2020). In color enhancement it appears as repeated curve application (Guo et al., 2020), LUT evolution (Hu et al., 2025) and recurrent state propagation (Bai et al., 2026). Its value lies in feedback, since each round is conditioned on the previous result and harder regions can be deferred to later

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.

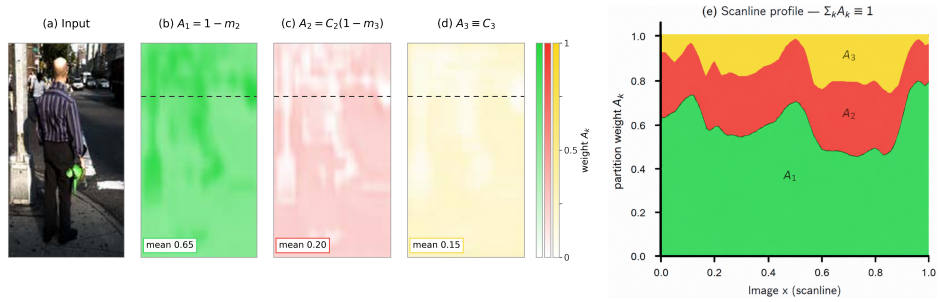


Figure 1: LoopLUT divides an image among its correction rounds. (a) An LCDP test image. (b–d) Partition weights from the trained gates of LoopLUT (K=3), shown without contrast stretching; color saturation encodes the weight. $\mathbf{A}_k(x)$ is the share of pixel x taken from the lookup of round k. The means (0.65, 0.20 and 0.15) show that the global round covers most of the image and later rounds refine ever smaller regions. Each weight is what the cascade passes to round k minus what round k passes on, $\mathbf{A}_k{=}\mathbf{C}_k(1{-}\mathbf{m}_{k+1})$, where $\mathbf{C}_k$ is the product of the gates up to round k. The weights are therefore non-negative and sum to one at every pixel, so the output is a convex blend of the K lookups. (e) Profile along the dashed scanline, confirming the unit sum and smooth transitions.

rounds. These methods, however, re-run the entire network every round, so cost grows linearly with round count. A second problem arises when several LUTs are applied in sequence, each to the output of the previous one, and their results are fused by scalar weights. The rounds are then nested, since the last already contains the effect of all earlier ones, and fusion degenerates into picking the most complete round. Spatial adaptivity, progressive per-round refinement, and a resolution-independent decision cost have not yet held together in one architecture.

We propose LoopLUT to bridge this gap. Iteration helps through feedback rather than through repeated full-resolution computation, so the loop should iterate over inexpensive low-resolution decisions, namely where to correct and which LUT to apply, rather than over full-resolution images. First, a global LUT performs the overall correction. Second, in each later round, a lightweight gating head predicts at low resolution, from the current enhancement state, the region that still requires correction. Then, a residual LUT is generated from the color statistics inside that region alone. Finally, the per-round gates multiply into a cascade that forms a partition of unity over the image, so the output is a per-pixel convex combination of the K lookup results. Because the gates themselves perform the fusion, no separate module exists for a trivial solution to bypass. All K LUTs also act directly on the original image, eliminating the compounded interpolation error of serial composition.

Our contributions are as follows.

- Region-local color correction through cascaded gates. The gates form a partition of unity and perform the fusion themselves, so the effective LUT varies per pixel and a shadow and a well-exposed region sharing a pixel value receive different corrections.
- A low-resolution decision loop. Each gate is predicted from the result of the previous round, so later rounds refine only what earlier rounds left, while the loop stays at a constant 256×256 and full-resolution work remains a pure lookup.
- A deployed system, not only a benchmark result. We export LoopLUT to ONNX and run it on an Android phone at about 100 – 200 ms per frame on CPU alone (Appendix E), and validate cross-task generality on four public benchmarks.

2 Related Work

3D LUT enhancement. Zeng et al. (2022) brought learnable 3D LUTs into the low-resolution-decision paradigm of HDRNet (Gharbi et al., 2017) and CSRNet (He et al., 2020), predicting basis-LUT coefficients so that one full-resolution lookup suffices. Later work refines the table through non-uniform lattices (Yang et al., 2022a; Zhou et al., 2026), factorized or compressed tables (Yang et al., 2022b; Zhang et al., 2022), implicit neural fields (Conde et al., 2024), attention-based fusion (Fu et al., 2024) and folded networks (Yang et al., 2024), but the operator remains global.

Local enhancement. Local methods apply region-specific corrections through spatially aware weight maps (Wang et al., 2021), context-aware 4D LUTs (Liu et al., 2023), per-pixel lookup weights (Liu et al., 2024), local parametric filters (Moran et al., 2020), multi-scale pyramids (Afifi et al., 2021) or local color distribution priors (Wang et al., 2022). Their regional decision is one-shot and predicted from the input alone, so a missed or over-corrected region is never revisited. Several also run the weight network at high resolution.

Iterative refinement. Many models reach their target through conditioned rounds, from recurrent optical flow (Teed & Deng, 2020), repeated curve application (Guo et al., 2020) and unrolled Retinex optimization (Liu et al., 2021) to multi-step diffusion (Jiang et al., 2024), flow-matching LUT evolution (Hu et al., 2025) and recurrent restoration (Bai et al., 2026). Their strength is feedback. Each round sees the previous result, so it corrects the remaining error rather than predicting the whole mapping at once. These loops, however, operate on the image or its feature maps, so every round repeats image-sized computation. LoopLUT moves the loop into the decision stage. Each round conditions its gate and residual LUT on a low-resolution rehearsal of the previous result (Section 3.3), so later rounds address only what earlier rounds left, at the cost of a 16 K-parameter gating head and one MLP at 256×256. Predicting regions round by round also makes locality part of the loop rather than a module added to a global operator.

3 METHOD

3.1 NOTATION AND OVERALL FRAMEWORK

Given a degraded input $\mathbf{I} \in [0,1]^{3\times H\times W}$, the goal is an enhanced image $\hat{\mathbf{I}}$ of the same size. A 3D LUT is a grid $\mathbf{T} \in \mathbb{R}^{3\times S\times S\times S}$ on the RGB cube (here $S{=}33$) acting by per-pixel trilinear interpolation, $\mathbf{T}(\mathbf{I})(x) = \mathrm{trilerp}\big(\mathbf{T}; \mathbf{I}(x)\big)$, which is independent of the pixel coordinate x. LoopLUT composes K such LUTs (here $K{=}3$) with $K{-}1$ spatial gates into a spatially varying mapping, while keeping full-resolution computation to lookups alone.

The decision stage (Figure 2) downsamples the input to $\mathbf{I}_{\downarrow}$ at a fixed 256×256 and passes it through a UNet backbone (Ronneberger et al., 2015) with GroupNorm (Wu & He, 2018), yielding a global color feature $\mathbf{F}_{\mathrm{g}}$ and a decoder feature map $\mathbf{D}$. The first round generates the global LUT $\mathbf{T}_1$. Each round $k = 2, \ldots, K$ then predicts a gate $\mathbf{m}_k$, cascades it into the cumulative gate $\mathbf{C}_k$, pools color statistics inside $\mathbf{C}_k$ to generate a region residual $\mathbf{\Delta}_k$, and rehearses the current enhancement state at low resolution. One backbone pass, $K{-}1$ gating heads and K MLP evaluations all run at $\leq 256 \times 256$, so the decision cost is independent of the output resolution. The execution stage performs K pure lookups on the original image and combines them per pixel with the bilinearly upsampled partition weights, so its cost grows only linearly in pixel count.

3.2 REGION CASCADE: THE GATES ARE THE FUSION

Each round's gate $\mathbf{m}_k \in (0,1)$ indicates whether a pixel still requires the k-th round of correction. With the cumulative gate $\mathbf{C}_k = \prod_{j=2}^{k} \mathbf{m}_j$, the partition weights and the output are

$$\mathbf{A}_1 = 1 - \mathbf{m}_2, \qquad \mathbf{A}_k = \mathbf{C}_k\,(1 - \mathbf{m}_{k+1}),\ \ 1 < k < K, \qquad \mathbf{A}_K = \mathbf{C}_K, \tag{1}$$

$$\hat{\mathbf{I}}(x) = \sum_{k=1}^{K} \mathbf{A}_k(x)\,\mathbf{T}_k(\mathbf{I})(x). \tag{2}$$

The telescoping structure of Equation (1) gives three properties, proved in Appendix A. First, $\sum_k \mathbf{A}_k \equiv 1$ pointwise with every $\mathbf{A}_k$ non-negative, so the output is a per-pixel convex combination of the K lookup results and cannot leave the valid range. Second, the supports contract monotonically, $\mathrm{supp}(\mathbf{A}_K) \subseteq \mathrm{supp}(\mathbf{C}_K) \subseteq \cdots \subseteq \mathrm{supp}(\mathbf{C}_2)$, so each later round refines only inside the region the previous round left marked as needing correction. Third, substituting the residual form $\mathbf{T}_k = \mathbf{T}_1 + \sum_{j=2}^{k} \mathbf{\Delta}_j$ and using the linearity of trilinear interpolation gives the exact identity

$$\hat{\mathbf{I}}(x) = \mathbf{T}^{x}_{\mathrm{eff}}(\mathbf{I})(x), \qquad \mathbf{T}^{x}_{\mathrm{eff}} = \mathbf{T}_1 + \sum_{k=2}^{K} \mathbf{C}_k(x)\,\mathbf{\Delta}_k, \tag{3}$$

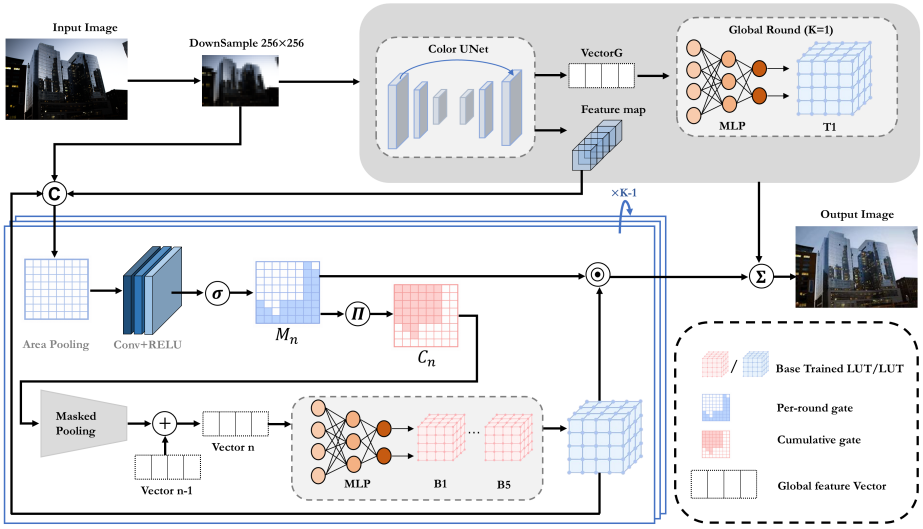


Figure 2: Overall architecture of LoopLUT ($\mathbf{M}_n$, $\mathbf{C}_n$ and $\mathbf{T}_n$ in the figure are $\mathbf{m}_k$, $\mathbf{C}_k$ and $\mathbf{T}_k$ in the text). The global round (gray) passes the 256×256 input once through the Color UNet to give the global feature $\mathbf{F}_g$ (VectorG) and the decoder feature map $\mathbf{D}$ (Feature map), which an MLP turns into the global LUT $\mathbf{T}_1$ by mixing the basis LUTs $\mathbf{B}_1, \ldots, \mathbf{B}_5$. Each region round (blue, $\times K-1$) area-pools its inputs to the $s \times s$ gate bottleneck, predicts the gate $\mathbf{M}_n$, multiplies it into the cumulative gate $\mathbf{C}_n$, and generates $\mathbf{T}_n$ from masked statistics inside $\mathbf{C}_n$. At execution the K LUTs are applied to the original image and combined per pixel by the partition weights, so the gates are the fusion. The decision stage is fixed at low resolution throughout.

so the model is exactly a per-pixel effective LUT: a global component plus region residuals whose coefficients are the cumulative gates. Equation (2) has no separate fusion step, since the partition weights are the gates themselves, and all K LUTs act on the original image, removing the compounded interpolation error of serial composition. Appendix B quantifies both defects.

3.3 Recurrent decision with progressive region refinement

Where the k-th round should refine depends on the gap remaining between the current result and the ideal one, which is unavailable at inference. LoopLUT estimates no error quantity; the gating head instead learns end to end, from inference-time inputs, which regions still need correcting:

$$\mathbf{m}_k = \Big(\sigma\big(g_k\big(\downarrow_s [\mathbf{D},\ \mathbf{I}_\downarrow,\ \mathbf{q}_{k-1},\ |\mathbf{q}_{k-1} - \mathbf{I}_\downarrow|]\big)\big)\Big) \uparrow, \tag{4}$$

where g_k is a three-layer convolutional network, independent per round (about 16 K parameters), $\downarrow_s$ is downsampling by area averaging to $s \times s$ (here s=32), and $\uparrow$ is bilinear upsampling back to the decision resolution. The bottleneck s is a structural constraint rather than an approximation for saving computation, and it bounds the spatial precision of the gate (Appendix B).

The residual is then driven by the statistics of the region it actually faces. Masked weighted pooling inside the cumulative gate gives $\mathbf{F}_k = \mathbf{F}_g + P(\mathbf{z}_k)$, where $\mathbf{z}_k$ collects the gate-weighted means $\mu_{\mathbf{C}_k}(\mathbf{D})$, $\mu_{\mathbf{C}_k}(\mathbf{I}_\downarrow)$ and $\mu_{\mathbf{C}_k}(|\mathbf{q}_{k-1} - \mathbf{I}_\downarrow|)$ together with the area fraction $\bar{\mathbf{C}}_k$, with $\mu_{\mathbf{C}}(\cdot) = \sum(\mathbf{C} \odot \cdot)/(\sum \mathbf{C} + \varepsilon)$ and P a two-layer MLP whose last layer is zero-initialized.

Closing the feedback at full resolution would destroy the efficiency skeleton, so the current enhancement state is kept inside the decision stage by a low-resolution rehearsal, $\mathbf{q}_1 = \mathbf{T}_1(\mathbf{I}_\downarrow)$ and $\mathbf{q}_k = \mathbf{q}_{k-1} + \mathbf{C}_k \odot \big(\mathbf{T}_k(\mathbf{I}_\downarrow) - \mathbf{T}_{k-1}(\mathbf{I}_\downarrow)\big)$ for $k \geq 2$, which replays the execution stage on $\mathbf{I}_\downarrow$ under the same cascade of weights. Since $\mathbf{q}_k$ feeds the next round's gate and region statistics, every decision is conditioned on the state after the previous round's correction while all feedback computation stays at 256×256.

3.4 LUT GENERATION AND TRAINING OBJECTIVE

Each round updates $\mathbf{T}_k = \text{clip}(\mathbf{T}_{k-1} + \gamma \sum_{i=1}^{n} c_{k,i}\mathbf{B}_i)$ from the identity LUT $\mathbf{T}_0$ by mixing basis LUTs (Zeng et al., 2022), with residual $\mathbf{\Delta}_k = \mathbf{T}_k - \mathbf{T}_{k-1}$ and $\mathbf{c}_k = \text{MLP}\big(\mathbf{F}_k \oplus E(k)\big)$, where $\{\mathbf{B}_i\}$ are $n{=}5$ learnable basis LUTs (about 0.54 M parameters), $E(k)$ is a sinusoidal round encoding and γ is an overall scale. Basis mixing uses two orders of magnitude fewer parameters than regressing the $3S^3$-dimensional grid directly and carries an inherent low-rank constraint.

The objective is $\mathcal{L} = \mathcal{L}_{\text{rec}} + \lambda_{\text{a}}\mathcal{L}_{\text{area}} + \lambda_{\text{tv}}\mathcal{L}_{\text{tv}} + \lambda_{\text{s}}\mathcal{L}_{\text{smooth}} + \lambda_{\text{g}}\mathcal{L}_{\text{gate}}$. The reconstruction term $\mathcal{L}_{\text{rec}}$ combines ℓ_1, SSIM (Wang et al., 2004) and contrast and saturation regularizers on the final output $\hat{\mathbf{I}}$; by Equation (2) every round of LUTs and gates is supervised through that single path, so no per-round intermediate supervision is needed. The region coverage term $\mathcal{L}_{\text{area}}$ is a two-sided hinge on the mean cumulative gate $\bar{\mathbf{C}}_k$, with a per-round decreasing upper bound τ_k that carries the inductive bias of contraction and a small floor $\tau_{\min}$ that keeps the gradient path open. $\mathcal{L}_{\text{tv}}$ and $\mathcal{L}_{\text{smooth}}$ smooth the gates and the LUTs, the latter with an added monotonicity hinge against color banding, and $\mathcal{L}_{\text{gate}}$ supplies an auxiliary gate target that exists only during training. Appendix B gives these terms in full, together with the initialization constraint imposed by the product structure and the reason a coverage bound is used in place of an error threshold.

4 EXPERIMENTS

4.1 DATASETS AND METRICS

We evaluate on four public benchmarks: LCDP (Wang et al., 2022) for mixed exposure, MIT-Adobe FiveK (Bychkovsky et al., 2011) for expert retouching, Mobile-Spec (Zhou et al., 2024) for mobile photography, and LSUI (Peng et al., 2023) for underwater enhancement. LCDP is our primary benchmark and carries all ablations. We report PSNR, SSIM (Wang et al., 2004), LPIPS (Zhang et al., 2018) with the AlexNet backbone, and CIEDE2000, all evaluated at native resolution without cropping or rescaling. Splits and the full evaluation protocol are given in Appendix C.

4.2 IMPLEMENTATION DETAILS

All experiments are run on a single NVIDIA RTX 4050 Laptop GPU (6 GB). During training the whole input image is resized to 256×256 rather than randomly cropped, because LUT generation depends on the color statistics of the entire image and cropping would make the framing protocol inconsistent between training and inference. The structural hyperparameters match those in the method section: $K{=}3$ rounds, gate bottleneck $s{=}32$, $n{=}5$ basis LUTs and LUT side length $S{=}33$. The model is trained for 100 epochs.

Compared methods. We compare against nine methods: HDRNet (Gharbi et al., 2017), RUAS (Liu et al., 2021), LCDPNet (Wang et al., 2022), AdaInt (Yang et al., 2022a), LightenDiffusion (Jiang et al., 2024), CSEC (Li et al., 2024), DnLUT (Yang et al., 2025), CanonCGT (Ko et al., 2026) and ShiftLUT (Zeng et al., 2026); the reasons for this selection are given in Appendix C.3. On LSUI we compare against methods designed for underwater enhancement (Table 4).

4.3 COMPARISON WITH STATE-OF-THE-ART METHODS

Table 1 shows that LoopLUT ranks first on all four metrics on MIT-Adobe FiveK and Mobile-Spec. Detailed metrics averaged over the three benchmarks are given in Appendix D.1 (Table 5 and Figure 9). Its PSNR exceeds the runner-up by 1.15 dB on FiveK (DnLUT, 20.77 dB) and by 2.81 dB on Mobile-Spec (AdaInt, 26.98 dB). On LCDP it ranks first on PSNR and SSIM (23.48 dB and 0.8616), ahead of CSEC and LCDPNet, which are designed for exposure correction. CanonCGT leads on the remaining two metrics, CIEDE2000 (7.03 vs. 7.53) and LPIPS (0.1264 vs. 0.1558), on which LoopLUT ranks second and third respectively.

LoopLUT is also the most consistent across datasets. Methods tailored to one degradation drop sharply outside it: CSEC and LCDPNet reach only 12.88 and 15.13 dB on FiveK, and RUAS 12.22 dB on Mobile-Spec. Averaged over the three datasets, LoopLUT ranks first on all four metrics, with 25.06 dB PSNR against 22.44 dB for the runner-up CanonCGT.

Table 1: Quantitative comparison with state-of-the-art methods on LCDP, MIT-Adobe FiveK and Mobile-Spec. In each column the **best** is bold and the second best is underlined; ↑/↓ indicates that higher/lower is better.

		LCDP				MIT-Adobe FiveK				Mobile-Spec			
Method	Venue	PSNR↑	SSIM↑	LPIPS↓	CIEDE↓	PSNR↑	SSIM↑	LPIPS↓	CIEDE↓	PSNR↑	SSIM↑	LPIPS↓	CIEDE↓
HDRNet	SIGGRAPH'17	21.75	0.8265	0.1860	10.98	20.57	0.7251	0.0960	9.10	20.34	0.7911	0.1652	6.57
RUAS	CVPR'21	18.25	0.7568	0.2065	9.11	19.29	0.7095	0.1393	9.86	12.22	0.7470	0.1833	17.26
LCDPNet	ECCV'22	23.29	0.8423	0.1587	8.43	15.13	0.6029	0.2437	17.17	25.25	0.9097	0.1404	5.21
AdaInt	CVPR'22	20.93	0.7856	0.2258	9.42	18.76	0.7235	0.1474	12.21	26.98	0.9264	0.0850	4.02
LightenDiffusion	ECCV'24	19.04	0.7743	0.1334	9.33	18.35	0.7050	0.1542	12.35	23.23	0.8383	0.2759	7.31
CSEC	CVPR'24	23.35	0.8552	0.1563	7.93	12.88	0.5668	0.2757	21.13	24.13	0.9023	0.1565	5.79
DnLUT	CVPR'25	21.02	0.7563	0.3002	8.68	20.77	0.7119	0.2431	9.05	25.19	0.8716	0.2068	5.30
CanonCGT	CVPR'26	22.92	0.8241	**0.1264**	**7.03**	19.37	0.7030	0.2086	14.15	25.04	0.9254	0.0701	4.83
ShiftLUT	CVPR'26	20.25	0.7419	0.2412	10.45	19.67	0.7122	0.1357	10.50	24.32	0.8824	0.1893	7.02
LoopLUT (ours)	–	**23.48**	**0.8616**	0.1558	7.53	**21.92**	**0.7501**	**0.0875**	**7.79**	**29.79**	**0.9680**	**0.0458**	**2.91**

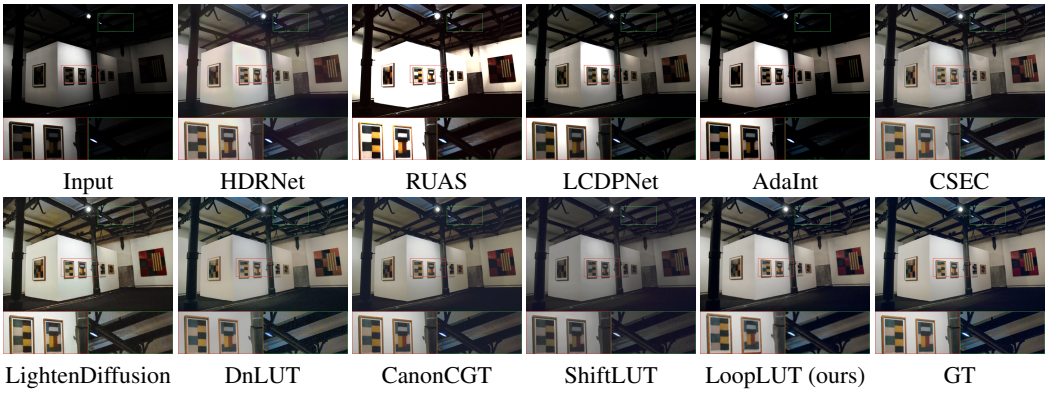


Figure 3: Qualitative comparison on the LCDP test set. Each panel shows the full output above and, below, the magnified red and green crops, placed on the two middle paintings and on the steel frame at the top; the red box examines midtone color reproduction and the green box detail recovery in dark structure. All methods use the same crops.

Figures 3–5 show the same trend. On LCDP, RUAS overexposes the white wall and HDRNet and AdaInt leave the steel frame in shadow, whereas LoopLUT brightens the frame without clipping the wall. On FiveK, HDRNet leaves a yellow-green cast and CSEC and LCDPNet fail visibly, while LoopLUT matches the target color temperature. On Mobile-Spec, RUAS turns almost the whole image white and HDRNet and LCDPNet cannot lift the facade; LoopLUT brightens the facade while keeping the sky suppressed.

Real-world photographs. To examine performance in the real world beyond standard test sets, we compare LoopLUT with CSEC and LCDPNet on two everyday smartphone photographs: LoopLUT attains the best average NIQE and BRISQUE while running 11× faster than LCDPNet and 99× faster than CSEC (Figure 11 in Appendix D.3).

4.4 Ablation Studies

Effectiveness of spatial gating. To separate spatial adaptivity from multi-round cascading, we keep the three-round cascade and the full capacity but lower the gate bottleneck from $s{=}32$ to $s{=}1$, so every round can only act on the whole image. Region gating wins on 213/218 test images with a mean per-image PSNR gain of $+3.08$ dB (Figure 6): with an identical cascade, using different LUTs in different regions accounts for the bulk of the gain.

Effectiveness of the number of rounds. We sweep $K{=}1$ to 8 and measure accuracy and latency together (Table 2). Accuracy rises with diminishing returns, the second, third and fourth rounds adding $+0.55$, $+0.31$ and $+0.13$ dB, peaks on all four metrics at $K{=}4$ (23.61 dB) and declines slowly from $K{\geq}5$, as the contracting coverage bound leaves later rounds ever smaller regions to

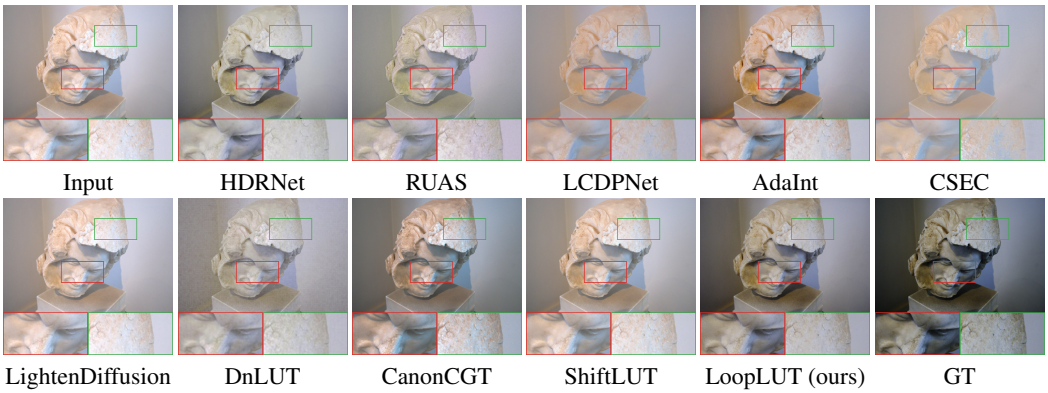


Figure 4: Qualitative comparison on the MIT-Adobe FiveK test set. Layout as in Figure 3; the red crop covers the face of a stone head and the green crop the weathered stone of its crown, examining the light-to-dark transition across the face and texture preservation on the highlight side.

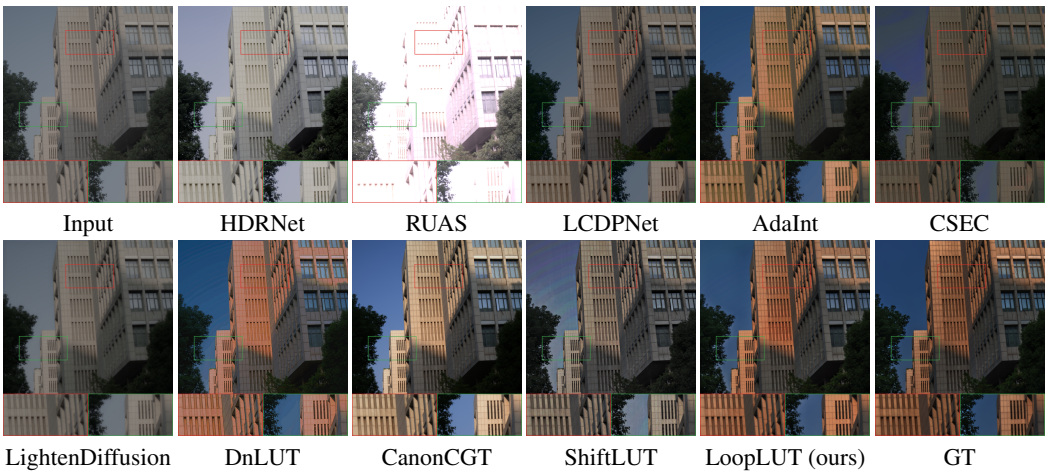


Figure 5: Qualitative comparison on the Mobile-Spec test set. Layout as in Figure 3; the red crop covers the sunlit facade and the green crop the backlit region at lower left, examining color reproduction under warm illumination and tonal separation in the backlit building and tree canopy.

correct. Cost grows linearly, by 1.1–1.5 ms per round at 1620×1080, so the gain per added millisecond falls from 0.40 dB/ms for the second round to 0.25 for the third and 0.11 for the fourth: $K{=}4$ would buy $+0.13$ dB with 24% more latency. We adopt $K{=}3$ as the last round whose gain is commensurate with its cost.

Effectiveness of progressive refinement process. Figure 7 unfolds the decision process of a test example round by round. The global round performs the overall brightening ($+1.4$ dB); the round-2 gate skips the already correctly graded sky and admits the underexposed foreground and building, contributing the bulk of the gain ($+3.2$ dB); round 3 contracts to the heaviest residual areas for a small refinement ($+0.3$ dB). PSNR rises monotonically while the corrected region contracts monotonically, consistent with the support contraction of Equation (1), which indicates that the gates allocate later capacity to positions the previous round failed to handle rather than reapplying the same correction everywhere.

4.5 Computational efficiency

LoopLUT has 2.89 M parameters (0.54 M of them basis LUTs), fp32 weights of 11.04 MB (5.53 MB in fp16), and needs 6.564 GFLOPs to process an FHD frame, which it does in 5.67 ms (176 FPS).

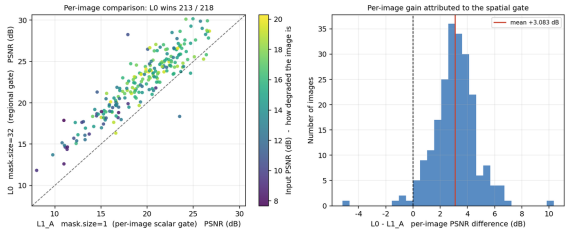


Figure 6: Ablation 1: region gate (s=32) vs. per-image scalar gate (s=1), same rounds and parameters, 218 LCDP test images. Left: per-image PSNR. Right: per-image gain.

Table 2: Ablation 2: number of rounds K (LCDP). Best in bold; †: main configuration; ms: latency at 1620×1080.

K	PSNR↑	SSIM↑	LPIPS↓	CIEDE↓	ms
1	22.62	0.8214	0.1657	7.64	2.49
2	23.17	0.8428	0.1603	7.56	3.87
3†	23.48	0.8616	0.1558	7.53	5.10
4	**23.61**	**0.8671**	**0.1539**	**7.49**	6.32
5	23.60	0.8669	0.1541	**7.49**	7.49
6	23.56	0.8658	0.1545	7.50	8.71
7	23.51	0.8639	0.1552	7.51	9.83
8	23.43	0.8607	0.1566	7.54	11.34

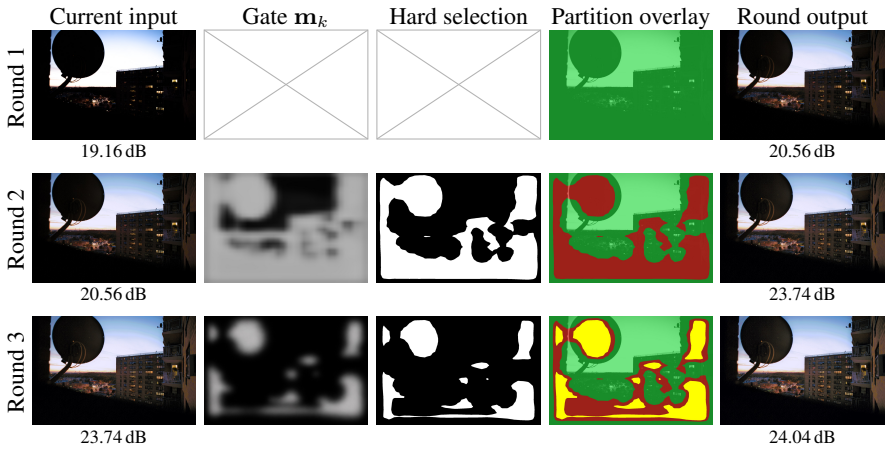


Figure 7: Ablation 3: progressive refinement on an LCDP test example. PSNR rises 19.16 (input) → 20.56 → 23.74 → 24.04 dB, and each row's "Current input" is the previous row's "Round output". Round 1 is the global round, so its overlay is entirely green. In the partition overlay, green marks pixels fixed after round 1 ($\mathbf{A}_1$), red those entering round 2 ($\mathbf{A}_2$) and yellow round 3 ($\mathbf{A}_3$); "Hard selection" is a thresholded view of the gate, shown for illustration only.

Why the cost barely grows with resolution. The decision stage (one backbone pass, $K-1$ gating heads and K MLP evaluations) runs at a constant 256×256, costing 6.135 GFLOPs and 1.84 ms at any output resolution; the execution stage is K trilinear lookups plus a per-pixel multiply-accumulate, about 69 FLOPs per pixel per round. From 1620×1080 to 4K the pixel count grows $4.7\times$ but total computation only 21% (6.497 to 7.852 GFLOPs), an asymptotic advantage at ultra-high resolution over spatially aware methods whose decision cost grows with it.

Table 3: Resolution scaling and memory behavior of LoopLUT. The decision stage is constant at 6.135 GFLOPs and 1.84 ms *at every resolution*; only the pure-lookup execution stage grows with pixel count. Traffic is the per-frame device-memory read+write volume of the current unfused implementation, and Bandwidth is that volume divided by the measured latency.

Resolution	Pixels	GFLOPs			Latency			Device memory		
	(MP)	Decide	Exec	Total	Decide (ms)	Total (ms)	Throughput (FPS)	Peak (MB)	Traffic (MB)	Bandwidth (GB/s)
1620×1080 (LCDP native)	1.75	6.135	0.362	6.497	1.84	5.10	196	93	774	159
1920×1080 (FHD)	2.07	6.135	0.429	6.564	1.84	5.67	176	104	880	163
3840×2160 (4K)	8.29	6.135	1.717	7.852	1.84	16.23	62	296	2914	188

FLOPs are not the whole story. Latency does not follow that 21%: it triples, from 5.10 to 16.23 ms. Table 3 shows why. The decision stage is 94% of the FLOPs at 1620×1080 but only 36% of the time, and at 4K it is still 78% of the FLOPs yet just 11% of the time. What grows is the execution stage, and it is bound by memory rather than arithmetic: at 4K the current implementation moves 2.9 GB per frame, or 188 GB/s. The execution stage is therefore running close to the hardware's memory limit, not leaving a design inefficiency on the table.

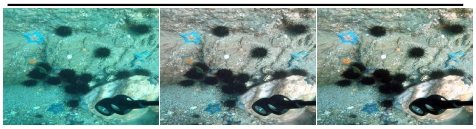
This also bounds what an optimized implementation can gain. The 2.9 GB figure is the traffic of an unfused pipeline in which every operator round-trips through device memory; a single fused kernel need only read the input image and write the output, 94.9 MB at 4K in fp16, a $31\times$ reduction.

4.6 Discussion

Generality beyond exposure. On LSUI the color shift comes from medium absorption rather than exposure. With no underwater prior and unchanged structural hyperparameters, LoopLUT reaches 25.37 dB / 0.87, above the task-specific U-shape Transformer (25.13 dB / 0.86, Table 4). It removes the cast without damaging structure (Figure 8): the global round absorbs the whole-image cast and the region rounds the depth-varying residual. Appendix D.2 shows all methods (Figure 10).

Table 4: LSUI underwater benchmark. Best in bold, second underlined; compared methods quoted from public reports.

Method	Venue	PSNR↑	SSIM↑
UGAN	ICRA'18	19.29	0.7943
UIE-DAL	CVPRW'19	16.57	0.75
Ucolor	TIP'21	21.45	0.83
U-shape Trans.	TIP'23	<u>25.13</u>	<u>0.86</u>
LoopLUT (ours)	–	**25.37**	**0.87**



Input 17.94 dB / 0.9045 | LoopLUT 30.11 dB / 0.9640 | GT

Figure 8: An LSUI test example; values under the panels are PSNR / SSIM.

What the loop contributes. A single 3D LUT is a function of color alone (Zeng et al., 2022), so two pixels with the same value receive the same correction wherever they lie. The loop lifts this restriction yet keeps a per-pixel lookup: each round adds a LUT restricted to the region its gate selects, so identical colors in different regions are mapped differently (Equation (1)). As in recurrent refinement (Teed & Deng, 2020), each gate reads the current enhancement state and selects what is still wrong. Locality is what makes this work: a per-image scalar gate with the same rounds and parameters falls 3.08 dB behind (Figure 6). The loop is also cheap, since all rounds share one 256×256 backbone pass and each extra round adds 1.1–1.5 ms (Table 2); as the coverage bound contracts, the benefit concentrates in the early rounds, so $K{=}3$ suffices.

On-device deployment. LoopLUT runs on an Android phone: we export it to ONNX and execute it with the ONNX Runtime CPU backend, using the same fp32 weights as Table 1 without quantization or pruning. The application processes a photograph in roughly 100–200 ms on CPU alone and displays the input, the enhanced output and the partition assignment (Appendix E).

5 Conclusion

We have presented LoopLUT, which performs iterative refinement through low-resolution LUT decisions rather than repeated full-resolution image updates. Cascaded gates organize K LUTs, all applied directly to the original image, into a partition of unity over that image, and fusion is carried out by the gates themselves, so that no separate fusion module exists for a trivial solution to bypass and no interpolation error is compounded through serial composition; the entire decision process runs at a constant 256×256, independent of the output resolution. The accuracy of 23.48 dB on LCDP, the best value on all four metrics on FiveK and Mobile-Spec, and the cross-task generality check on LSUI together indicate that spatial adaptivity, progressive refinement and a decision cost independent of resolution can hold simultaneously within a single architecture.

AI USE STATEMENT

Generative AI tools were used under the authors' direction to assist with refining the research hypotheses, designing the research methodology and experiments and providing feedback on them, implementing the method, translation, cleaning and reformatting datasets, interpreting results, editing the manuscript for readability, and typesetting and organizing the manuscript. All AI-assisted code, data processing, analyses and text were reviewed and verified by the authors. The authors take full responsibility for the final content of this work.

REPRODUCIBILITY STATEMENT

Code, configuration files and data-processing scripts will be released with the paper. Section 4.2 gives the structural hyperparameters and the training setup. Appendix A proves the structural properties of the cascaded partition, Appendix B gives the definition and initialization of every loss term, and Appendix C gives the dataset splits and the evaluation protocol for all four benchmarks. All experimental results of our proposed method, LoopLUT, reported in this paper can be reproduced with the released code.

REFERENCES

Mahmoud Afifi, Konstantinos G. Derpanis, Björn Ommer, and Michael S. Brown. Learning multi-scale photo exposure correction. In *CVPR*, 2021.

Runci Bai, Yucheng Xin, Pu Wang, Yongcong Wang, Chen Wu, Dianjie Lu, Guijuan Zhang, Pengwen Dai, Guangwei Gao, Siyuan Yao, and Zhuoran Zheng. Loop-mamba: A loop Mamba with degradation-aware and shared memory for old photo restoration, 2026. arXiv preprint.

Vladimir Bychkovsky, Sylvain Paris, Eric Chan, and Frédo Durand. Learning photographic global tonal adjustment with a database of input/output image pairs. In *CVPR*, 2011.

Marcos V. Conde, Javier Vazquez-Corral, Michael S. Brown, and Radu Timofte. NILUT: Conditional neural implicit 3D lookup tables for image enhancement. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, 2024.

Kang Fu, Yicong Peng, Zicheng Zhang, Qihang Xu, Xiaohong Liu, Jia Wang, and Guangtao Zhai. AttentionLUT: Attention fusion-based canonical polyadic LUT for real-time image enhancement. In *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2024.

Michaël Gharbi, Jiawen Chen, Jonathan T. Barron, Samuel W. Hasinoff, and Frédo Durand. Deep bilateral learning for real-time image enhancement. *ACM Transactions on Graphics*, 36(4), 2017.

Chunle Guo, Chongyi Li, Jichang Guo, Chen Change Loy, Junhui Hou, Sam Kwong, and Runmin Cong. Zero-reference deep curve estimation for low-light image enhancement. In *CVPR*, 2020.

Jingwen He, Yihao Liu, Yu Qiao, and Chao Dong. Conditional sequential modulation for efficient global image retouching. In *ECCV*, 2020.

Liubing Hu, Chen Wu, Anrui Wang, Dianjie Lu, Guijuan Zhang, and Zhuoran Zheng. FlowLUT: Efficient image enhancement via differentiable LUTs and iterative flow matching, 2025. arXiv preprint.

Hai Jiang, Ao Luo, Xiaohong Liu, Songchen Han, and Shuaicheng Liu. LightenDiffusion: Unsupervised low-light image enhancement with latent-retinex diffusion models. In *ECCV*, 2024.

Jinwon Ko, Keunsoo Ko, and Chang-Su Kim. CanonCGT: Reference-based color grading via canonical pivot representation. In *CVPR*, 2026.

Yiyu Li, Ke Xu, Gerhard Petrus Hancke, and Rynson W. H. Lau. Color shift estimation-and-correction for image enhancement. In *CVPR*, 2024.

Chengxu Liu, Huan Yang, Jianlong Fu, and Xueming Qian. 4D LUT: Learnable context-aware 4D lookup table for image enhancement. *IEEE Transactions on Image Processing*, 32, 2023.

Jing Liu, Qingying Li, Xiongkuo Min, Yuting Su, Guangtao Zhai, and Xiaokang Yang. Pixel-learnable 3DLUT with saturation-aware compensation for image enhancement. *IEEE Transactions on Multimedia*, 26, 2024.

Risheng Liu, Long Ma, Jiaao Zhang, Xin Fan, and Zhongxuan Luo. Retinex-inspired unrolling with cooperative prior architecture search for low-light image enhancement. In *CVPR*, 2021.

Sean Moran, Pierre Marza, Steven McDonagh, Sarah Parisot, and Gregory Slabaugh. DeepLPF: Deep local parametric filters for image enhancement. In *CVPR*, 2020.

Lintao Peng, Chunli Zhu, and Liheng Bian. U-shape transformer for underwater image enhancement. *IEEE Transactions on Image Processing*, 32, 2023.

Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-Net: Convolutional networks for biomedical image segmentation. In *Medical Image Computing and Computer-Assisted Intervention – MICCAI 2015*. Springer, 2015.

Zachary Teed and Jia Deng. RAFT: Recurrent all-pairs field transforms for optical flow. In *ECCV*, 2020.

Haoyuan Wang, Ke Xu, and Rynson W. H. Lau. Local color distributions prior for image enhancement. In *ECCV*, 2022.

Tao Wang, Yong Li, Jingyang Peng, Yipeng Ma, Xian Wang, Fenglong Song, and Youliang Yan. Real-time image enhancer via learnable spatial-aware 3D lookup tables. In *ICCV*, 2021.

Zhou Wang, Alan C. Bovik, Hamid R. Sheikh, and Eero P. Simoncelli. Image quality assessment: From error visibility to structural similarity. *IEEE Transactions on Image Processing*, 13(4), 2004.

Yuxin Wu and Kaiming He. Group normalization. In *ECCV*, 2018.

Canqian Yang, Meiguang Jin, Xu Jia, Yi Xu, and Ying Chen. AdaInt: Learning adaptive intervals for 3D lookup tables on real-time image enhancement. In *CVPR*, 2022a.

Canqian Yang, Meiguang Jin, Yi Xu, Rui Zhang, Ying Chen, and Huaida Liu. SepLUT: Separable image-adaptive lookup tables for real-time image enhancement. In *ECCV*, 2022b.

Sidi Yang, Binxiao Huang, Mingdeng Cao, Yatai Ji, Hanzhong Guo, Ngai Wong, and Yujiu Yang. Taming lookup tables for efficient image retouching. In *ECCV*, 2024.

Sidi Yang, Binxiao Huang, Yulun Zhang, Dahai Yu, Yujiu Yang, and Ngai Wong. DnLUT: Ultra-efficient color image denoising via channel-aware lookup tables. In *CVPR*, 2025.

Hui Zeng, Jianrui Cai, Lida Li, Zisheng Cao, and Lei Zhang. Learning image-adaptive 3D lookup tables for high performance photo enhancement in real-time. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 44(4), 2022.

Xiaolong Zeng, Yitong Yu, Shiyao Xiong, Jinhua Hao, Ming Sun, Chao Zhou, and Bin Wang. ShiftLUT: Spatial shift enhanced look-up tables for efficient image restoration. In *CVPR*, 2026.

Fengyi Zhang, Hui Zeng, Tianjun Zhang, and Lin Zhang. CLUT-Net: Learning adaptively compressed representations of 3DLUTs for lightweight image enhancement. In *Proceedings of the 30th ACM International Conference on Multimedia (ACM MM)*, 2022.

Richard Zhang, Phillip Isola, Alexei A. Efros, Eli Shechtman, and Oliver Wang. The unreasonable effectiveness of deep features as a perceptual metric. In *CVPR*, 2018.

Kailai Zhou, Lijing Cai, Yibo Wang, Mengya Zhang, Bihan Wen, Qiu Shen, and Xun Cao. Joint RGB-spectral decomposition model guided image enhancement in mobile photography. In *ECCV*, 2024.

Ruikai Zhou, Canqian Yang, Yiming Li, Meiguang Jin, Xu Jia, Ying Chen, and Yi Xu. Self-distilled learning of adaptive interval 3D lookup tables on real-time image enhancement. *Pattern Recognition*, 172, 2026.