DnLUT: Ultra-Efficient Color Image Denoising via Channel-Aware Lookup Tables

Sidi Yang[1]   Binxiao Huang[1†]   Yulun Zhang[2]   Dahai Yu[3]   Yujiu Yang[4†]   Ngai Wong[1]

[1] The University of Hong Kong   [2] Shanghai Jiaotong University
[3] TCL Corporate Research   [4] Tsinghua University

{yangsidi99, huanghx7}@connect.hku.hk, yulun100@gmail.com, dahai.yu@tcl.com
yang.yujiu@sz.tsinghua.edu.cn, nwong@eee.hku.hk

Abstract

While deep neural networks have revolutionized image denoising capabilities, their deployment on edge devices remains challenging due to substantial computational and memory requirements. To this end, we present DnLUT, an ultra-efficient lookup table-based framework that achieves high-quality color image denoising with minimal resource consumption. Our key innovation lies in two complementary components: a Pairwise Channel Mixer (PCM) that effectively captures inter-channel correlations and spatial dependencies in parallel, and a novel L-shaped convolution design that maximizes receptive field coverage while minimizing storage overhead. By converting these components into optimized lookup tables post-training, DnLUT achieves remarkable efficiency - requiring only 500KB storage and 0.1% energy consumption compared to its CNN contestant DnCNN, while delivering 20× faster inference. Extensive experiments demonstrate that DnLUT outperforms all existing LUT-based methods by over 1dB in PSNR, establishing a new state-of-the-art in resource-efficient color image denoising. The project is available at https://github.com/Stephen0808/DnLUT.

1. Introduction

Color images pervade our digital world - from social media and photography to scientific imaging and medical diagnostics. These images are inherently more vulnerable to noise contamination than their grayscale counterparts, not only due to their threefold data size during acquisition, storage, and transmission, but also because human perception exhibits heightened sensitivity to color distortions compared to luminance variations.

The advent of deep neural networks (DNNs) has dramatically advanced the field of color image denoising[2, 15,
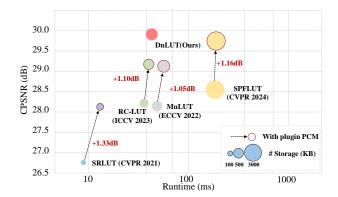

Figure 1. Model comparison in terms of color peak signal-to-noise ratio (CPSNR), runtime, and storage. The CPSNR and runtime are calculated on the CBSD68 dataset with Gaussian noise level $\sigma = 25$ using Qualcomm Snapdragon 8 Gen2. Our method outperforms state-of-the-art LUT-based methods with the highest CPSNR, at a low storage requirements and reduced runtime. Additionally, our PCM module serves as a versatile plug-in module that enhances existing methods' performance by over 1dB.

27, 30, 32]. State-of-the-art models leverage sophisticated architectures with numerous convolutional layers or transformer blocks, achieving unprecedented denoising quality. However, these advances come at a steep cost - the computational complexity and memory requirements of such models render them impractical for edge devices. This limitation is particularly acute given that most edge devices lack specialized hardware accelerators such as graphics processing units (GPUs) or tensor processing units (TPUs), creating a significant gap between algorithmic capabilities and practical deployability.

Recent works have explored lookup tables (LUTs) as an elegant solution for efficient image enhancement. LUTs replace complex runtime computations with simple array indexing operations through direct addressing. This approach typically involves training a neural network model, then con-

---

[†] Corresponding authors.

verting it into LUTs by exhaustively mapping input-output relationships. During inference, the system performs rapid table lookups instead of costly computations, effectively leveraging DNN capabilities while circumventing hardware constraints. However, LUT-based methods face a fundamental challenge: the storage requirement grows exponentially with the input dimensionality. This has led to the adoption of 4D LUTs (using four pixel values for indexing) as a practical compromise between storage and performance.

Current LUT-based methods predominantly employ 2×2 convolutional kernels processed independently for each channel[10, 12, 14, 16], creating spatial-wise LUTs for single-channel processing. While this approach works well for tasks like super-resolution, it proves inadequate for color image denoising where noise typically affects all RGB channels simultaneously. Channel correlation plays a pivotal role in modeling noise characteristics, a critical factor for achieving effective restoration. Alternative approaches either use 1×1 convolutions with three-channel depth for RGB information (creating 3D channel-wise LUTs)[13, 25] or attempt to use larger $H \times W \times 3$ kernels. However, these solutions either disrupt noise distribution patterns or lead to prohibitive storage requirements.

To address these challenges, we propose DnLUT, a novel lookup table-based framework specifically designed for efficient color image denoising. Our approach centers on two key innovations:

First, we introduce the Pairwise Channel Mixer (PCM), which simultaneously processes spatial and channel information. PCM strategically combines RGB channels into three pairs (RG, GB, BR), processing them through parallel branches using a 1×2 convolution kernel with depth 2. This design efficiently captures channel correlations while maintaining manageable storage requirements. The PCM module can also serve as a plug-in module for existing LUT-based methods, consistently improving their performance by approximately 1dB.

Second, we develop an innovative L-shaped convolution kernel that addresses the limitations of conventional rotation-based approaches. After taking the channel correlation into account, as discussed in [12, 14], there is still a need to enlarge the spatial receptive field. While previous methods like SR-LUT[10] use rotation ensemble training to expand receptive fields of spatial-wise LUT, they often introduce redundant pixel usage and significant storage overhead. Our L-shaped kernel design eliminates pixel overlap during rotations, enabling conversion to more efficient 3D LUTs while maintaining the same effective receptive field size as 4D LUTs.

Extensive experimental validation demonstrates the superiority of DnLUT over existing approaches:

- We achieve up to 1.3dB CPSNR improvement over SPFLUT[14] on standard Gaussian denoising bench-

marks. For real-world denoising tasks, we outperform CBM3D[3] by nearly 5dB.
- Our method maintains exceptional efficiency, requiring only 0.1% of DnCNN[30]'s energy consumption and 500KB storage.
- The PCM module serves as a versatile performance enhancer, bringing over 1dB CPSNR improvement to existing LUT-based methods.
- Our L-shaped convolution design reduces storage requirements by 17× while preserving receptive field coverage.

These results establish DnLUT as a significant advance in practical, resource-efficient color image denoising, bridging the gap between algorithmic capability and deployability on edge devices.

2. Related works

2.1. Color Image denoising

A fundamental characteristic of color images is the strong correlation among RGB channels[5]. Research has consistently shown that processing these channels independently yields suboptimal results compared to joint processing approaches[19]. This insight has driven numerous innovations in color image denoising.

A landmark development was CBM3D[3], which operates in luminance-chrominance space (*i.e.*, YC_bC_r). The method performs BM3D[4] grouping once on the luminance channel Y and leverages this grouping for collaborative filtering of chrominance channels U and V. Building on this concept, [11] introduced a radial basis function mixture for more sophisticated handling of channel correlations. Another notable approach, MC-WNNM[24], extended WNNM[7] by incorporating a weight matrix to adaptively balance the data fidelity across channels based on their noise characteristics.

The deep learning era, initiated by DnCNN[30], transformed the field of color image denoising. Subsequent approaches[2, 15, 26, 27, 32] have increasingly sophisticated architectures that process all RGB channels simultaneously through complex networks of convolution layers or transformer blocks. While these models achieve remarkable denoising quality, their substantial computational and memory requirements present significant deployment challenges, particularly for resource-constrained edge devices.

2.2. Replacing CNN with LUT

The challenge of deploying deep neural networks on edge devices has sparked interest in lookup table (LUT) based alternatives. Inspired by the prevalence of color LUTs in image signal processors, [28] pioneered an approach using learnable basis LUTs combined with a weight prediction network for image enhancement. [25] further developed this

concept, proposing a more efficient channel-wise LUT system. However, these early channel-wise LUTs, processing single pixels per channel, proved inadequate for the complex task of denoising.

A significant breakthrough came with [10], which introduced a method to convert CNNs with limited receptive fields into spatial-wise LUTs for super-resolution. This approach trains networks on restricted pixel neighborhoods (*i.e.*, 2×2) and converts them to LUTs by exhaustively mapping input-output relationships. During inference, the system efficiently retrieves pre-computed outputs through coordinate indexing. Follow-up works[12, 16, 17] enhanced this framework using serial LUTs to expand receptive fields, achieving substantial performance improvements.

To manage the exponential growth of LUT size with input dimensionality, these methods typically process each input channel independently. While this approach proves effective for super-resolution, where spatial information dominates, it falls short for color image denoising, which requires preserving both spatial and channel information to model noise distribution effectively. The limitations of existing channel-wise and spatial-wise LUT approaches highlight the need for a more sophisticated solution that can handle the unique challenges of color image denoising while maintaining computational efficiency.

3. Method

3.1. Preliminary

The foundation of LUT-based image restoration was established by SR-LUT[10], which introduces a constrained receptive field approach for network training (*i.e.*, 2 × 2). Post-training, the network's output values are systematically cached into a LUT by exhaustively traversing all possible input pixel value combinations. During inference, input patches of size 2 × 2 are converted to indices of a 4D LUT to retrieve corresponding output values. To manage storage requirements, the 4D index space is uniformly subsampled. Additionally, a rotation ensemble strategy rotates each 2×2 input patch four times, effectively expanding the receptive field to 3 × 3. Recent works have focused on increasing the effective receptive field through various techniques, including multiple LUTs[12, 14, 17], shift aggregations[17], and diverse kernel patterns[8, 12, 16].

Our work advances this foundation by specifically addressing the challenges of LUT-based color image denoising, a critical application for edge devices.

A notable limitation of existing LUT-based approaches is their treatment of channel depth, typically set to 1. For color images, this means each RGB channel independently accesses cached output values without inter-channel interaction. This design overlooks crucial RGB channel correlations and spatial-channel-wise degradation patterns inher-ent in color image denoising. While a straightforward solution might be to employ vanilla convolution kernel patterns, the exponential growth of LUT size with index dimension makes this impractical. For instance, a kernel of size 2 × 2 with depth 3 would require approximately 582 TB of storage, even with subsampling compression. This constraint presents a fundamental dilemma: **when aiming to capture more spatial information, we have to sacrifice the richness of channel information.** Moreover, while 4D spatial-wise LUTs show promise, they typically require multiple LUTs (analogous to stacked convolution layers), creating significant storage overhead, particularly problematic for L1 cache access.

Table 1. LUT size versus receptive field (RF) and channel numbers. * means subsampled LUT size

RF	Depth	LUT Dim.	LUT Size*
1×1	1	1D	17 B
1×1	3	3D	4.9 KB
2×2	1	4D	83.5 KB
1×2	3	6D	24.1 MB
2×2	3	12D	582.6 TB
$k \times k$	c	$k \times k \times c$D	$(2^4 + 1)^{k \times k \times C}$ B

3.2. Overview

To address these fundamental challenges, we introduce two complementary innovations: 1) a pairwise channel mixer that leverages 4D LUT efficiency while modeling spatial-channel correlations; and 2) a novel L-shaped kernel featuring inherent rotation non-overlapping properties, enabling conversion to 3D LUT while maintaining 4D LUT-equivalent receptive field coverage.

We first build the DnNet for training, as shown in Fig. 2. Our proposed framework enhances traditional spatial-wise LUT capabilities through these two key modules: the pairwise channel mixer (PCM) establishes robust channel-spatial correlations in three RGB channels while the L-shaped kernel expands the effective receptive field with minimal computational overhead independently in each channel. The architecture combines PCM with L-shaped convolution in two stages. The first stage, inspired by [14], generates multi-channel features from LUT groups, which are then concatenated in the fusion module. The second stage integrates both proposed modules. Importantly, all components are designed for efficient conversion to 3D or 4D LUTs during inference, collectively referred to as DnLUT.

3.3. Pairwise Channel Mixer

Designing effective kernel patterns for LUT-based methods presents significant challenges. Fig. 3 illustrates various kernel pattern categories. Traditional approaches have
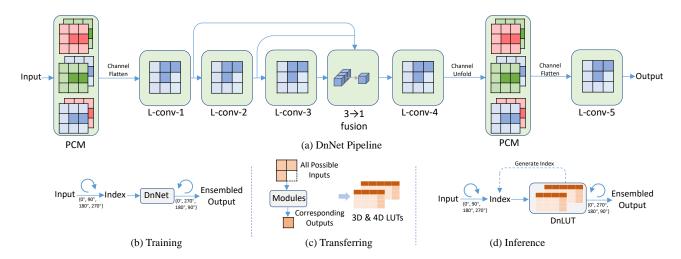
Figure 2. System architecture of DnLUT: (a) The DnNet pipeline integrates pairwise channel mixers and L-shaped convolutions, with multi-scale fusion enhancing receptive field coverage. Channel dimensions are flattened for parallel processing in L-shaped operations, then unfolded for PCM input. (b) Input pixels undergo four rotations (0°, 90°, 180°, 270°) during processing, with outputs averaged for enhanced results. (c) Post-training, all possible input combinations are processed through DnNet modules, with outputs cached in optimized 3D or 4D LUTs. (d) During inference, input pixels are efficiently processed through multiple LUTs, with each LUT's outputs informing subsequent LUT indices, culminating in final denoised pixel values.

focused on single-dimension kernels due to LUT index dimension constraints, leading to two major limitations in color image denoising. Spatial-wise kernels, while capturing spatial information, neglect channel correlations, resulting in color distortion and suboptimal noise modeling. Conversely, channel-wise kernels establish strong RGB channel connections but ignore spatial relationships, leading to persistent artifacts. While combining these kernels in a cascade network might seem promising, this approach fails to process channel and spatial information simultaneously, resulting in suboptimal noise distribution modeling.

Our PCM addresses these limitations through a novel architecture for color image denoising. The module first reorganizes RGB channels into three pairwise combinations: RG, GB, and BR. These pairs are processed through parallel branches using a kernel with 1×2 spatial dimensions and depth 2 in the initial layer, followed by cascaded 1×1 convolution layers. Each convolution operation processes four pixel values to produce one channel output, enabling efficient conversion to 4D LUTs post-training. The cached LUTs (*i.e.*, $LUT_{RG}, LUT_{GB}, LUT_{GR}$) operate in parallel, with their predictions concatenated. For an input anchor (I_R, I_G, I_B), the output values (V_R, V_G, V_B) are computed as:

$$
\begin{aligned}
(V_R, V_G, V_B) = \mathrm{Cat}(&LUT_{RG}[I_{0,0,R}][I_{0,1,R}][I_{0,0,G}][I_{0,1,G}], \\
&LUT_{GB}[I_{0,0,G}][I_{0,1,G}][I_{0,0,B}][I_{0,1,B}], \\
&LUT_{BR}[I_{0,0,B}][I_{0,1,B}][I_{0,0,R}][I_{0,1,R}]),
\end{aligned}
\tag{1}
$$

where H, W, C of each $I_{H,W,C}$ denotes height, width, and channel index. PCM effectively balances spatial and
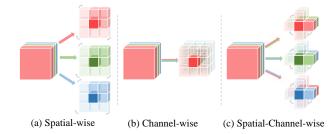

Figure 3. Taxonomy of kernel patterns for LUT-based methods. Dark cubes indicate rotation points, while medium-dark regions show involved pixel positions during one rotation.

channel-wise processing, capturing inter-channel correlations crucial for color fidelity while preserving spatial relationships through the 1×2 kernel design and refining features via cascaded 1×1 convolutions.

3.4. Rotation Non-overlapping Kernel

The rotation ensemble strategy is fundamental to LUT-based methods, using four rotations (0°, 90°, 180°, 270°) to enable asymmetric convolution kernels to cover larger symmetric areas. By combining outputs from all rotations, the approach enhances performance without increasing LUT index dimensionality. As shown in Fig. 4, traditional methods like SR-LUT and MuLUT use this strategy to expand receptive fields to 3×3 and 5×5 respectively.

However, this conventional approach has two critical limitations. First, approximately half of the pixels within the receptive field are accessed multiple times during lookup operations, as illustrated in Fig. 4's tables. This redundancy

indicates inefficient kernel pattern design. Second, stacking these kernels to create multiple LUTs for expanded receptive fields results in significant 4D LUT storage overhead.

To address these inefficiencies, we introduce an innovative L-shaped kernel design. During each rotation, the kernel processes two additional pixels (beyond the center pixel) without overlap, ensuring each surrounding pixel contributes exactly once to the output. This design maximizes pixel value utilization within the receptive field while enabling conversion to 3D LUTs instead of 4D LUTs, dramatically reducing storage requirements without sacrificing coverage.

3.5. PCM Plug-in Module

To demonstrate the versatility of our pairwise channel mixer, we developed PCM as a plug-in module compatible with existing LUT-based methods. The inference of plug-in algorithm is summarized in Alg. 1. This module seamlessly integrates with current architectures, enhancing their ability to capture channel-spatial correlations without requiring structural modifications. Compared to state-of-the-art methods like SPFLUT, the PCM plug-in adds only 12% runtime and 8% storage overhead while delivering over 1dB performance improvement.

Algorithm 1 Inference of Pairwise Channel Mixer Module

Require: Input image I of size $H \times W \times 3$
Require: 4D Lookup Table LUT of PCM
Require: Padding size p
  1: Initialize output image O of size $H \times W \times 3$
  2: **for** $r = 0$ to 3 **do**
  3:     Rotate image I by $r \times 90$ degrees to get I_{rot}
  4:     Pad I_{rot} with p pixels on each side to get I'_{rot}
  5:     **for** $h = 0$ to $H - 1$ **do**
  6:         **for** $w = 0$ to $W - 1$ **do**
  7:             **for** $c = 0$ to 2 **do**
  8:                 $p_1 \leftarrow I'_{\text{rot}}[h, w, c]$
  9:                 $p_2 \leftarrow I'_{\text{rot}}[h, w + 1, c]$
 10:                 $p_3 \leftarrow I'_{\text{rot}}[h, w, (c + 1) \mod 3]$
 11:                 $p_4 \leftarrow I'_{\text{rot}}[h, w + 1, (c + 1) \mod 3]$
 12:                 $index \leftarrow \text{Quantize}(p_1, p_2, p_3, p_4)$
 13:                 $value \leftarrow LUT[index]$
 14:                 $O[h, w, c] \leftarrow O[h, w, c] + \text{Interpolate}(value)$
 15:             **end for**
 16:         **end for**
 17:     **end for**
 18:     Rotate O back by $(4 - r) \times 90$ degrees to get O_{rot}
 19:     $O \leftarrow O + O_{\text{rot}}$
 20: **end for**
 21: $O \leftarrow O/4$
 22: **return** Output image O

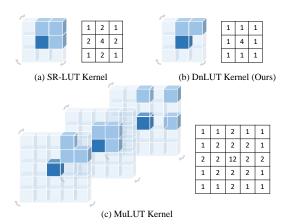(a) SR-LUT Kernel   (b) DnLUT Kernel (Ours)

(c) MuLUT Kernel

Figure 4. Comparison of spatial-wise kernel designs. Left patterns show kernel configurations, while right tables quantify lookup frequencies during output retrieval.

4. Experiments

4.1. Implementation details

We implement DnLUT using the Adam optimizer with a cosine annealing learning rate schedule and mean squared error (MSE) loss function. For Gaussian denoising, we train networks for 2×10^5 iterations with a batch size of 12, increasing to 32 for real-world denoising scenarios. Following established practices[10, 12], we uniformly sample LUTs at 2^4 intervals. To minimize indexing artifacts, final predictions utilize 4D simplex interpolation and 3D tetrahedral interpolation. We further refine cached LUTs through 2000 iterations of LUT-aware fine-tuning[12] on the training dataset.

4.2. Gaussian Color Image denoising

Following [27], we train our model using four comprehensive datasets: DIV2K[1], Flickr2K[23], BSD400[20], and WaterlooED[18]. Training data is augmented with additive Gaussian noise at three levels ($\sigma \in \{15, 25, 50\}$). We evaluate performance on four standard benchmarks: CBSD68[21], Kodak24, Urban100[9], and McMaster[31], using identical noise levels.

4.3. Real-world Color Image denoising

To validate real-world performance, we train DnLUT on the SIDD training dataset and evaluate on both SIDD validation and DnD datasets. The SIDD dataset comprises indoor scenes captured by five smartphones under varying noise conditions, providing 320 image pairs for training and 1,280 for validation. The DnD dataset offers 50 high-resolution images with realistic noise patterns for comprehensive evaluation.

Table 2. Quantitative comparison on Gaussian color image denoising benchmark. The table presents CPSNR(dB) values.

Category	Method	CBSD68			Kodak24			Urban100			McMaster		
		$\sigma = 15$	$\sigma = 25$	$\sigma = 50$	$\sigma = 15$	$\sigma = 25$	$\sigma = 50$	$\sigma = 15$	$\sigma = 25$	$\sigma = 50$	$\sigma = 15$	$\sigma = 25$	$\sigma = 50$
LUT-based	SR-LUT[10]	29.76	26.71	22.41	30.35	27.16	22.65	29.38	26.04	21.60	31.18	28.01	23.35
	MuLUT[12]	30.52	28.11	24.85	31.31	29.02	25.28	30.25	27.67	23.75	32.28	29.88	26.36
	RC-LUT[16]	30.68	28.12	25.04	31.57	29.07	25.89	30.33	27.80	23.86	32.51	29.89	26.50
	SPF-LUT[14]	30.97	28.56	25.33	31.86	29.58	26.26	30.89	28.26	24.22	31.77	30.44	26.91
	DnLUT(Ours)	32.41	29.88	26.03	33.02	30.24	26.74	32.12	28.87	25.01	32.88	30.44	27.12
Classical	CBM3D[3]	33.52	30.71	27.38	34.28	31.68	29.02	33.93	31.36	27.93	34.06	31.66	28.51
	MC-WNNM[24]	31.98	29.32	26.98	33.23	30.89	28.67	30.23	29.23	27.00	31.23	30.20	27.55
DNN	DnCNN[30]	33.90	31.24	27.95	34.60	32.14	28.96	32.98	30.81	27.59	33.45	31.52	28.62
	SwinIR[15]	34.42	31.78	28.56	35.34	32.89	29.79	35.61	33.20	30.22	35.13	32.90	29.82

Table 3. Quantitative comparison on real-world color image denoising. The table presents CPSNR(dB) and SSIM values. Methods on DnD dataset are validated on the online platform. We could only get the PSNR rather than CPSNR.

Datasets	Method	SR-LUT[10]	MuLUT[12]	RC-LUT[16]	SPF-LUT[14]	DnLUT(Ours)	CBM3D[3]	MC-WNNM[24]	DnCNN[30]
SIDD	CPSNR	29.38	33.24	33.88	34.91	35.44	30.14	29.45	36.45
	SSIM	0.634	0.830	0.839	0.865	0.875	0.702	0.682	0.900
DnD	PSNR*	33.69	35.11	35.43	36.22	36.67	33.12	32.34	37.11
	SSIM	0.839	0.868	0.875	0.911	0.922	0.823	0.817	0.932

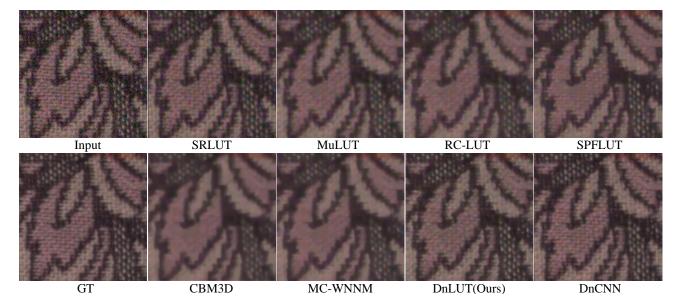Figure 5. Qualitative comparison on synthetic datasets.

Figure 6. Qualitative comparison on real-world datasets.

Table 4. PCM can be easily incorporated to the existing LUT-based methods. It equips the model with the ability of capturing the channel correlations which brings more than 1dB on the widely used benchmark. The noise level σ of Gaussian noise is set to 25. The table presents CPSNR(dB) values.

Method	CBSD68	Kodak24	Urban100	McMaster	SIDD
SR-LUT[10]	26.71	27.16	26.04	28.01	29.38
PCM+SR-LUT	28.04(+1.33)	28.55(+1.39)	27.33(+1.29)	28.78(+0.77)	32.33(+2.96)
MuLUT[12]	28.11	29.01	27.67	29.88	33.24
PCM+MuLUT	29.17(+1.05)	29.92(+0.91)	28.74(+1.07)	30.33(+0.44)	34.33(+1.09)
RC-LUT[16]	28.12	29.07	27.80	29.89	33.88
PCM+RC-LUT	29.22(+1.10)	30.09(+1.02)	28.77(+0.97)	30.54(+0.65)	34.67(+0.79)
SPF-LUT[14]	28.56	29.58	28.26	30.44	34.91
PCM+SPF-LUT	29.72(+1.16)	30.63(+1.05)	29.43(+1.17)	30.96(+0.52)	35.56(+0.65)

4.4. Quantitative results

We evaluate performance using color peak signal-to-noise ratio(CPSNR) and structural similarity index(SSIM) metrics, comparing against state-of-the-art methods across three categories: LUT-based methods (SR-LUT[10], MuLUT[12], RC-LUT[16], SPF-LUT[14]), classical approaches (CBM3D[3], MC-WNNM[24]), and DNN-based solutions (DnCNN[30], SwinIR[15]).

As shown in Tab. 2, DnLUT achieves superior performance in the LUT-based category, surpassing SPF-LUT by over 1dB in Gaussian noise removal while requiring only 17% of its storage. Results from Tab. 3 demonstrate DnLUT's effectiveness in real-world scenarios, outperforming classical methods CBM3D[3] and MC-WNNM[24] by approximately 5dB. Though the Gaussian noise is independent across RGB channels, the network can still leverage inter-channel natural distribution in images to suppress the noise. For example, if a pixel in the R channel has a high value due to noise, but the corresponding pixels in the G and B channels do not, the network can infer that the high value is likely noise and suppress it.

4.5. Qualitative results

Visual comparisons in Figs. 5 & 6 illustrate DnLUT's superior denoising capabilities. Existing LUT-based methods exhibit noticeable color distortion, particularly evident in Fig. 5's first row. In contrast, DnLUT achieves visual quality comparable to computation-intensive methods like DnCNN and CBM3D. Fig. 6 further demonstrates DnLUT's advantage in real-world scenarios, where both LUT-based and classical methods struggle with color fidelity and often over-smooth details. DnLUT successfully preserves color consistency while retaining fine image details.

4.6. Efficiency Evaluation

We conduct comprehensive efficiency analysis across three key metrics: theoretical energy cost, runtime performance, and storage requirements (Tab. 5).

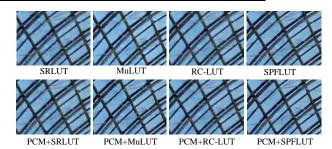**Energy cost.** Following AdderSR[22], we analyze multipli-

Figure 7. Visualization of LUT-based method with plug-in PCM.

cation and addition operations across different data types to estimate total energy consumption. DnLUT achieves 70% energy savings compared to SPF-LUT while delivering superior denoising quality. The efficiency gap widens dramatically when compared to DNN-based methods, with DnLUT requiring only 0.1% of DnCNN's energy consumption.

Runtime. We implement LUT-based methods using standard JAVA API on the ANDROID platform. DnLUT maintains the rapid inference capabilities characteristic of SR-LUT[10], demonstrating clear advantages over classical and DNN-based approaches. For reference, DnCNN running on mobile CPU requires 20× more processing time. DnLUT's performance can be further optimized through device-specific implementations, such as FPGA deployments.

Storage. DnLUT requires approximately 500 KB storage, well within typical L2-cache limits of modern chips. This efficient memory footprint enables rapid lookup operations with minimal dependency overhead.

4.7. Effectiveness of Plug-in PCM

Tab. 4 demonstrates PCM's versatility as a performance enhancer for existing LUT-based methods. When integrated with current approaches, PCM consistently delivers significant improvements across all benchmark datasets. The visual comparisons in Fig. 7 show marked reduction in color distortion, particularly in smooth image regions, resulting in superior perceptual quality.

Table 5. Efficiency evaluation of three categories color image denoising methods. We use the Qualcomm Snapdragon 8 Gen 2 as the mobile platform. The energy cost is calculated on the 512×512 resolution color images.

Category	Method	Platform	Runtime(ms) 256×256	Runtime(ms) 512×512	Energy Cost (pJ)	Storage (KB)
LUT-based	SR-LUT[10]	Mobile	6	19	149.98M	82
	MuLUT[12]	Mobile	21	80	899.88M	489
	RC-LUT[6]	Mobile	18	78	612.92M	326
	SPF-LUT[14]	Mobile	72	257	2.32G	30,178
	DnLUT(ours)	Mobile	23	88	687.34M	518
Classical	CBM3D[29]	PC	3,189	14,343	4.82G	-
	MC-WNNM[24]	PC	74,290	293,424	89.23G	-
DNN	DnCNN[30]	Mobile	543	1,923	542.53G	2,239
	SwinIR[15]	Mobile	73,234	274,453	12.03T	45,499

4.8. Ablation study

We conduct ablation studies using Gaussian color image denoising benchmark to validate DnLUT's key components.

4.8.1. Channel-spatial-wise convolution

Channel indexing[13, 25] possesses a 1×1 receptive field to extract channel information. We evaluate various combinations of channel-wise, spatial-wise, and our proposed channel-spatial-wise kernels using a consistent architecture comprising one channel indexing module, two L-shaped convolution modules, and one PCM component.

Results in Tab. 6 reveal that naive combination of channel-wise and spatial-wise convolutions yields suboptimal performance. Pure channel-spatial-wise convolution, while promising, struggles with limited receptive field coverage. The integration of spatial-wise and channel-spatial-wise convolutions produces substantial improvements. Notably, adding channel-wise convolution offers minimal additional benefit, as channel-spatial-wise convolution already captures comprehensive color information.

4.8.2. L-shaped convolution

To validate our L-shaped convolution design, we compare it against standard 2×2 spatial-wise kernels (denoted as S in Tab. 7, $\sigma = 1$). While the conventional kernel shows marginal performance advantages, our L-shaped design achieves comparable results while enabling 3D LUT conversion, reducing storage requirements by 17×. It paves the way of deploying multi-layer spatial-wise convolutions.

5. Conclusion

This paper presents DnLUT, a novel approach to color image denoising that bridges the gap between high-performance deep learning models and resource-constrained edge devices. Addressing the limitations of existing LUT-based methods, DnLUT introduces two key innovations. First, the Pairwise Channel Mixer (PCM) models channel and spa-

Table 6. Different combinations of channel-wise, spatial-wise, and our channel-spatial-wise kernels. These kernels are represented by channel indexing[13], L-shaped kernel, and PCM. The table presents CPSNR(dB) values.

Channel	Spatial	Channel-spatial	CBSD68
	✓		29.77
		✓	30.15
✓	✓		30.21
	✓	✓	31.41
✓	✓	✓	31.47

Table 7. Comparison of L-shaped kernel and 2×2 spatial-wise kernel. The table presents CPSNR(dB) values.

Method	CBSD68	Kodak	McMaster	LUT Size(KB)
PCM + L	26.63	27.05	27.96	494 + 24
PCM + S	26.71	27.16	28.01	494 + 408

tial correlations simultaneously by processing channel pairs, enabling comprehensive color modeling without excessive storage costs. Second, an L-shaped convolution design rethinks rotation-based receptive field expansion, eliminating redundant pixel usage and converting 4D to 3D LUTs, reducing storage by 17× while maintaining spatial coverage. These advancements allow DnLUT to outperform state-of-the-art LUT-based methods by over 1dB in denoising quality while consuming just 0.1% of DnCNN's energy.

References

[1] Eirikur Agustsson and Radu Timofte. Ntire 2017 challenge on single image super-resolution: Dataset and study. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, 2017. 5

[2] Hanting Chen, Yunhe Wang, Tianyu Guo, Chang Xu, Yiping Deng, Zhenhua Liu, Siwei Ma, Chunjing Xu, Chao Xu, and Wen Gao. Pre-trained image processing transformer. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 12299–12310, 2021. 1, 2

[3] Kostadin Dabov, Alessandro Foi, Vladimir Katkovnik, and Karen Egiazarian. Color image denoising via sparse 3d collaborative filtering with grouping constraint in luminance-chrominance space. In *2007 IEEE international conference on image processing*, pages I–313. IEEE, 2007. 2, 6, 7

[4] Kostadin Dabov, Alessandro Foi, Vladimir Katkovnik, and Karen Egiazarian. Image denoising by sparse 3-d transform-domain collaborative filtering. *IEEE Transactions on image processing*, 16(8):2080–2095, 2007. 2

[5] Jingjing Dai, Oscar C Au, Lu Fang, Chao Pang, Feng Zou, and Jiali Li. Multichannel nonlocal means fusion for color image denoising. *IEEE Transactions on circuits and systems for video technology*, 23(11):1873–1886, 2013. 2

[6] 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 (TOG)*, 36(4):1–12, 2017. 8

[7] Shuhang Gu, Lei Zhang, Wangmeng Zuo, and Xiangchu Feng. Weighted nuclear norm minimization with application to image denoising. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2862–2869, 2014. 2

[8] Binxiao Huang, Jason Chun Lok Li, Jie Ran, Boyu Li, Jiajun Zhou, Dahai Yu, and Ngai Wong. Hundred-kilobyte lookup tables for efficient single-image super-resolution. *arXiv preprint arXiv:2312.06101*, 2023. 3

[9] 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*, pages 5197–5206, 2015. 5

[10] Younghyun Jo and Seon Joo Kim. Practical single-image super-resolution using look-up table. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 691–700, 2021. 2, 3, 5, 6, 7, 8

[11] Stamatios Lefkimmiatis. Non-local color image denoising with convolutional neural networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3587–3596, 2017. 2

[12] Jiacheng Li, Chang Chen, Zhen Cheng, and Zhiwei Xiong. Mulut: Cooperating multiple look-up tables for efficient image super-resolution. In *European Conference on Computer Vision*, pages 238–256. Springer, 2022. 2, 3, 5, 6, 7, 8

[13] Jiacheng Li, Chang Chen, Zhen Cheng, and Zhiwei Xiong. Toward dnn of luts: Learning efficient image restoration with multiple look-up tables. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2024. 2, 8

[14] Yinglong Li, Jiacheng Li, and Zhiwei Xiong. Look-up table compression for efficient image restoration. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 26016–26025, 2024. 2, 3, 6, 7, 8

[15] Jingyun Liang, Jiezhang Cao, Guolei Sun, Kai Zhang, Luc Van Gool, and Radu Timofte. Swinir: Image restoration using swin transformer. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 1833–1844, 2021. 1, 2, 6, 7, 8

[16] Guandu Liu, Yukang Ding, Mading Li, Ming Sun, Xing Wen, and Bin Wang. Reconstructed convolution module based look-up tables for efficient image super-resolution. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 12217–12226, 2023. 2, 3, 6, 7

[17] Cheng Ma, Jingyi Zhang, Jie Zhou, and Jiwen Lu. Learning series-parallel lookup tables for efficient image super-resolution. In *European Conference on Computer Vision*, pages 305–321. Springer, 2022. 3

[18] Kede Ma, Zhengfang Duanmu, Qingbo Wu, Zhou Wang, Hongwei Yong, Hongliang Li, and Lei Zhang. Waterloo exploration database: New challenges for image quality assessment models. *IEEE Transactions on Image Processing*, 26 (2):1004–1016, 2016. 5

[19] Julien Mairal, Michael Elad, and Guillermo Sapiro. Sparse representation for color image restoration. *IEEE Transactions on image processing*, 17(1):53–69, 2007. 2

[20] D. Martin, C. Fowlkes, D. Tal, and J. Malik. A database of human segmented natural images and its application to evaluating segmentation algorithms and measuring ecological statistics. In *Proc. 8th Int'l Conf. Computer Vision*, pages 416–423, 2001. 5

[21] David Martin, Charless Fowlkes, Doron Tal, and Jitendra Malik. A database of human segmented natural images and its application to evaluating segmentation algorithms and measuring ecological statistics. In *Proceedings eighth IEEE international conference on computer vision. ICCV 2001*, pages 416–423. IEEE, 2001. 5

[22] Dehua Song, Yunhe Wang, Hanting Chen, Chang Xu, Chunjing Xu, and DaCheng Tao. Addersr: Towards energy efficient image super-resolution. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 15648–15657, 2021. 7

[23] Radu Timofte, Eirikur Agustsson, Luc Van Gool, Ming-Hsuan Yang, and Lei Zhang. Ntire 2017 challenge on single image super-resolution: Methods and results. In *Proceedings of the IEEE conference on computer vision and pattern recognition workshops*, pages 114–125, 2017. 5

[24] Jun Xu, Lei Zhang, David Zhang, and Xiangchu Feng. Multichannel weighted nuclear norm minimization for real color image denoising. In *Proceedings of the IEEE international conference on computer vision*, pages 1096–1104, 2017. 2, 6, 7, 8

[25] Sidi Yang, Binxiao Huang, Mingdeng Cao, Yatai Ji, Hanzhong Guo, Ngai Wong, and Yujiu Yang. Taming lookup tables for efficient image retouching. *arXiv preprint arXiv:2403.19238*, 2024. 2, 8

[26] Syed Waqas Zamir, Aditya Arora, Salman Khan, Munawar Hayat, Fahad Shahbaz Khan, Ming-Hsuan Yang, and Ling

Shao. Multi-stage progressive image restoration. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 14821–14831, 2021. 2

[27] Syed Waqas Zamir, Aditya Arora, Salman Khan, Munawar Hayat, Fahad Shahbaz Khan, and Ming-Hsuan Yang. Restormer: Efficient transformer for high-resolution image restoration. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 5728–5739, 2022. 1, 2, 5

[28] 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):2058–2073, 2020. 2

[29] 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*, pages 6493–6501, 2022. 8

[30] 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. 1, 2, 6, 7, 8

[31] Lei Zhang, Xiaolin Wu, Antoni Buades, and Xin Li. Color demosaicking by local directional interpolation and nonlocal adaptive thresholding. *Journal of Electronic imaging*, 20(2):023016–023016, 2011. 5

[32] Yi Zhang, Dasong Li, Xiaoyu Shi, Dailan He, Kangning Song, Xiaogang Wang, Hongwei Qin, and Hongsheng Li. Kbnet: Kernel basis network for image restoration. *arXiv preprint arXiv:2303.02881*, 2023. 1, 2