DoFP-Aligned Lookup Tables for Real-Time Polarization Demosaicking

Yidong Luo[1,2] **Chenggong Li**[3] **Yunfeng Song**[1,2] **Mengyuan Liu**[2]
Junchao Zhang[3] **Xin Yuan**[2,†]

[1]Zhejiang University, Hangzhou, China
[2]School of Engineering, Westlake University, Hangzhou, China
[3]School of Automation, Central South University, Changsha, China
{luoyidong,songyunfeng,liumengyuan,xyuan}@westlake.edu.cn
{244603040,junchaozhang}@csu.edu.cn

Abstract

Real-time polarization imaging with off-the-shelf color division-of-focal-plane (DoFP) cameras requires demosaicking that is both accurate and deployment-friendly. Interpolation methods are lightweight but limited in polarization fidelity, whereas learning-based methods improve quality at much higher online cost. We propose DoFP-LUT, a sensor-structured LUT framework and, to our knowledge, the first LUT-based framework for real-time color DoFP polarization demosaicking. It leverages the deterministic and memory-light nature of LUT inference while aligning lookup corrections with DoFP-specific residual structures. Unlike generic LUTs designed for RGB restoration, DoFP-LUT targets post-initialization residuals that are coupled across analyzer orientations, Stokes relations, and local spatial structures. It factorizes teacher-guided correction into shared analyzer correction, polarization redistribution, and high-frequency compensation, compiling each component into compact low-dimensional LUTs offline. Online inference then requires only nearest-neighbor lookup and fixed write-back. Experiments show that DoFP-LUT achieves a favorable quality–efficiency trade-off, improving polarization reconstruction while preserving deterministic, memory-light inference for real-time polarization imaging. Code is available here.

1 Introduction

Polarization imaging provides physical cues beyond intensity, including surface orientation, material properties, and scene geometry [36], and has been widely used in autonomous perception [38], remote sensing [22], biomedical imaging [26], and computational photography [20]. A color division-of-focal-plane (DoFP) camera places a color-polarization filter array (CPFA) over the sensor so that each pixel measures one color through one micro-polarizer orientation. We call the intensity image associated with a given micro-polarizer orientation an analyzer image. Such cameras capture color and polarization information in a single shot [21, 23, 37, 14], making them suitable for dynamic scenes and real-time applications. However, the captured measurement is mosaicked across both color and polarization dimensions, and reliable polarization cues such as degree of linear polarization (DoLP) and angle of polarization (AoP) can only be recovered after demosaicking.

Despite its practical importance, color DoFP demosaicking remains challenging because the sensing pattern introduces coupled aliasing across color, analyzer orientation, and local spatial structure. Existing methods reveal a clear quality–efficiency trade-off. Interpolation-based methods [18, 19]

†Corresponding author.

40th Conference on Neural Information Processing Systems (NeurIPS 2026).

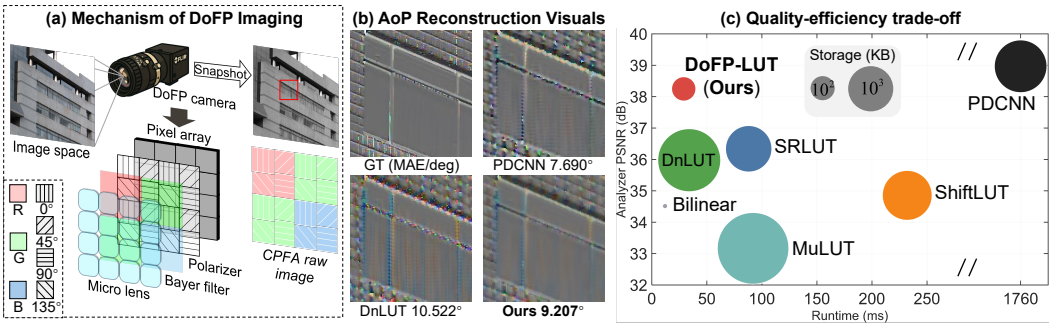

Figure 1: (a) A color DoFP sensor couples Bayer color sampling with micro-polarizer analyzers, producing a mosaicked measurement that requires joint color-polarization demosaicking. (b) Qualitative comparison of AoP visualizations among ground truth, PDCNN [35], DnLUT adaptation [33], and DoFP-LUT. Directly applying a generic LUT design fails to recover stable polarization structures, while DoFP-LUT better preserves polarization cues (lower Mean Angle Error, MAE) under lookup-based inference. (c) Quality-efficiency comparison, where the horizontal axis denotes embedded-hardware runtime and bubble size indicates model or table storage.

are lightweight, deterministic, and easy to deploy, but their local smoothness assumptions are often insufficient for recovering fine structures and stable polarization cues under color-polarization aliasing. Learning-based methods [37, 4] improve reconstruction quality by exploiting stronger color-polarization priors, but their online neural inference, model storage, and energy cost are less attractive for near-sensor or edge-side polarization imaging. This gap is particularly important for practical DoFP systems, where demosaicking may need to run continuously under limited memory, latency, and power budgets rather than on desktop GPUs. Therefore, real-time polarization imaging still lacks a demosaicker that can better balance polarization fidelity and deployment efficiency.

Lookup tables (LUTs) [2, 9] offer an appealing route to efficient inference because of their low latency, simple execution pattern, and predictable runtime. In DoFP demosaicking, LUT inference can be viewed as compiling post-initialization residual correction into efficient table queries: a lightweight initializer first lifts the mosaicked measurement into full analyzer estimates, and the remaining residuals are shaped by the sensing pattern and Stokes relations. Recent LUT-based methods [33, 34] further show that lookup inference can go beyond fixed global mappings and approximate more complex image restoration pipelines through enriched spatial indexing patterns or channel-wise lookup strategies. While such designs are effective for generic RGB restoration, they are not directly suited to color DoFP demosaicking, where the channels are physically coupled by analyzer orientations and Stokes recombination. As a result, the residuals are governed not only by local appearance statistics, but also by analyzer-domain interpolation artifacts, Stokes-related pairwise coupling, and residual polarization details. A monolithic LUT that jointly indexes these factors would lead to high dimensionality and sparse bins, whereas a compact generic index may fail to capture polarization-specific correction modes required for reliable DoFP reconstruction.

Motivated by this observation, we propose **DoFP-LUT**, a DoFP-aligned structured LUT framework for color polarization demosaicking. The key idea is to design lookup corrections around how analyzer images form Stokes quantities. Since S_0 is computed as the half-sum of the four analyzer measurements, while S_1 and S_2 are determined by two analyzer-pair differences, initialization errors affect polarization measurements through both common analyzer artifacts and pairwise imbalance. DoFP-LUT therefore decomposes the correction into three compact lookup components: shared analyzer correction suppresses common interpolation artifacts, polarization redistribution adjusts the two Stokes-related analyzer pairs, and high-frequency compensation refines the remaining residuals in provisional S_1/S_2 maps. This decomposition keeps each table low-dimensional while progressively injecting local spatial statistics and polarization-channel relations into table-only inference. As illustrated in Fig. 1, generic LUT transfer preserves efficient lookup but is not structurally matched to DoFP residuals, whereas DoFP-LUT targets a better quality–efficiency trade-off through polarization-structured lookup. To summarize, the main contributions of this paper are as follows:

- We present DoFP-LUT, to our knowledge the first LUT-based framework for deployment-friendly color DoFP demosaicking, with lookup indices structured by the sensing process.
- We formulate DoFP demosaicking as a structured post-initialization lookup problem and factorize analyzer correction into shared correction, polarization redistribution, and high-frequency compensation components.
- We develop a teacher-guided LUT compilation strategy that converts polarization-specific residual corrections into compact low-dimensional tables, enabling table-only online inference with a favorable quality–efficiency trade-off.

2 Related Work

2.1 Polarization Demosaicking

Early polarization demosaicking methods mainly relied on interpolation [18, 30, 19, 11, 10] and model-based reconstruction [23, 1, 16, 15]. These methods are lightweight, interpretable, and easy to deploy, but their local spatial priors are often insufficient under severe color-polarization aliasing, especially for fine structures and polarization cues such as DoLP and AoP. Recent learning-based methods [29, 25, 21, 28, 37, 3, 4, 12, 13] improve reconstruction quality by learning stronger color-polarization priors, and compact CNN variants can be efficient on GPU platforms. However, online convolution, model storage, and energy demand remain less attractive for embedded or edge-side polarization imaging systems that require predictable low-latency processing. Existing methods therefore suggest a practical trade-off: lightweight approaches are deployment-friendly but limited in polarization fidelity, whereas stronger learning-based methods improve quality with higher deployment overhead.

2.2 LUT-Based Low-Level Vision

Lookup tables have long been attractive for image processing because of their low latency and predictable runtime. In recent years, learnable LUTs have evolved from classical color mapping to increasingly flexible low-level vision pipelines. Early works mainly showed that lookup-based inference can go beyond fixed global mappings when lightweight conditioning signals are introduced, such as spatial-aware 3D LUTs, adaptive-interval LUTs, separable LUTs, and context-aware 4D LUTs [27, 31, 32, 8]. More recent efforts further extend LUTs toward restoration tasks, including super-resolution, LUT compression, and denoising [2, 6, 9, 7, 5, 33, 34]. In particular, DnLUT and the recent ShiftLUT explicitly develop LUTs along the line of enlarging effective receptive fields and strengthening restoration capacity while controlling storage and inference overhead [33, 34].

Although this line of work exhibits a clear development trajectory, it has not been systematically analyzed and developed for polarization imaging or DoFP demosaicking. This gap is important because representing polarization is not simply another RGB restoration problem: interpolation-induced analyzer errors and polarization-state deviations coexist and obey different correction patterns. Therefore, for polarization LUT design, deployment efficiency and reconstruction quality should be pursued together, but the first step is to make lookup inference truly compatible with the structure of polarization imaging itself. Thus, the key gap is not merely the absence of LUTs in polarization demosaicking, but the mismatch between generic LUT indices and the analyzer/Stokes-structured residuals induced by DoFP sensing. This motivates a DoFP-structured LUT formulation that preserves lookup efficiency while aligning indexing and correction with polarimetric structure.

3 Methodology

3.1 Problem Setting

We propose **DoFP-LUT**, a deployment-oriented framework for color polarization demosaicking. Unlike conventional RGB ISP or restoration tasks such as denoising or super-resolution, color polarization demosaicking is not only a problem of recovering degraded image content, but also a problem of disentangling coupled sampling effects across color, analyzer orientation, and local spatial structure. After interpolation-based initialization, the dominant recoverable errors mainly appear as structured

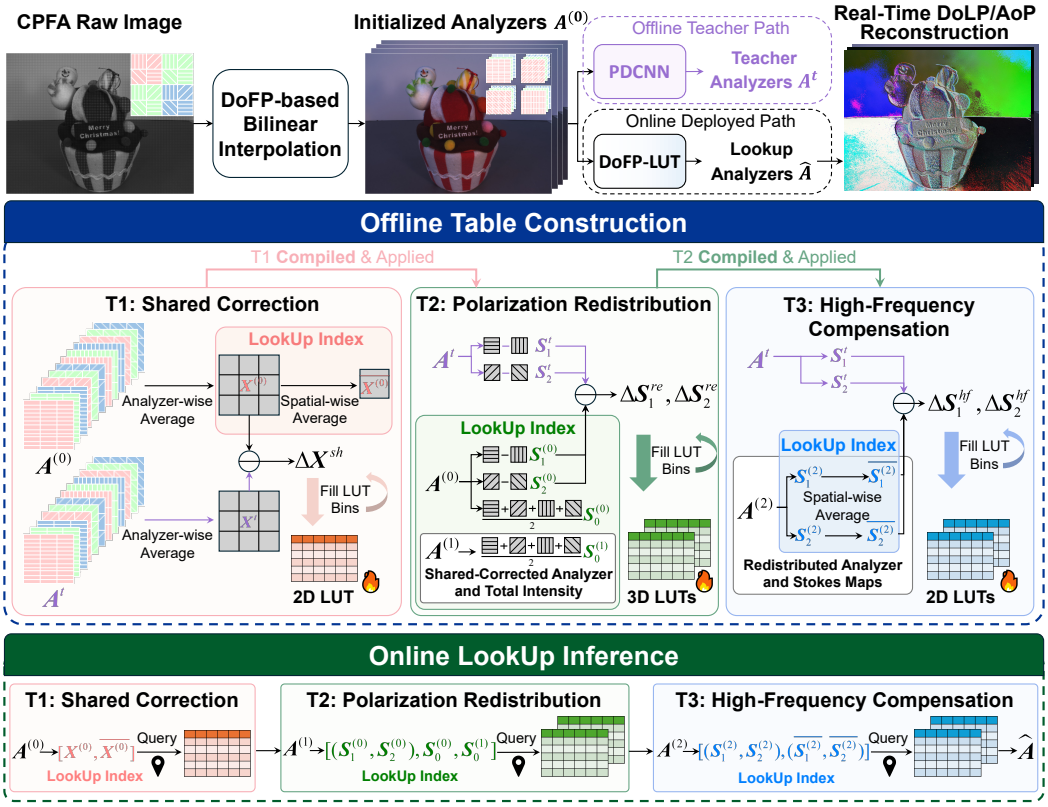

Figure 2: Overview of DoFP-LUT. Given a CPFA input, DoFP-based bilinear initialization produces initial analyzer estimates. Offline, PDCNN provides teacher analyzers to compile three structured LUT components: shared correction, polarization redistribution, and high-frequency compensation. Online, the compiled LUTs are sequentially queried and their outputs are written back by fixed algebraic rules to obtain corrected analyzers, from which Stokes, DoLP, and AoP are reconstructed.

analyzer-domain artifacts. Motivated by this observation, DoFP-LUT organizes demosaicking as a lookup-based analyzer correction problem before deriving the final Stokes representation.

Starting from a mosaicked color DoFP image, a DoFP-tailored bilinear initialization [17] produces analyzer estimates $\boldsymbol{A}^{(0)} \in \mathbb{R}^{4\times3\times H\times W}$, where the four analyzer orientations correspond to $0°$, $45°$, $90°$, and $135°$. From the analyzers, the Stokes quantities are computed as

$$S_0 = \tfrac{1}{2}(A_0 + A_{45} + A_{90} + A_{135}), \qquad S_1 = A_0 - A_{90}, \qquad S_2 = A_{45} - A_{135}, \tag{1}$$

from which DoLP and AoP are computed by

$$\mathrm{DoLP} = \frac{\sqrt{S_1^2+S_2^2}}{S_0}, \qquad \mathrm{AoP} = \tfrac{1}{2}\,\mathrm{atan2}(S_2, S_1). \tag{2}$$

These relations show that initialization errors in analyzer images affect the final polarization measurements through both intensity-like aggregation and analyzer-pair differences.

The main difficulty is that the analyzer-domain residuals after initialization are structurally mixed. Some errors are broadly shared across analyzer branches, while others are tied to the pairwise differences that determine S_1 and S_2. Treating all residuals with a single homogeneous lookup rule would therefore mix different correction behaviors into one table. DoFP-LUT instead uses a structured lookup formulation, where the analyzer correction is factorized into complementary table-driven components. The detailed lookup states, table construction, and inference procedure are described next.

3.2 Structured LUT Compilation and Lookup Inference

As shown in Fig. 2, DoFP-LUT couples offline table construction with online lookup inference in a unified framework. After DoFP-tailored bilinear initialization, a single-step CNN-based demosaicker, PDCNN [35], produces teacher analyzers $\boldsymbol{A}^t$ for compiling table targets, while the deployed DoFP-LUT route replaces the teacher with compact lookup tables for efficient inference. Given initialized analyzers $\boldsymbol{A}^{(0)}$, the deployed correction pipeline can be summarized by the following composition:

$$\hat{\boldsymbol{A}} = \mathcal{F}_{\mathrm{hf}} \circ \mathcal{F}_{\mathrm{red}} \circ \mathcal{F}_{\mathrm{sh}}(\boldsymbol{A}^{(0)}), \tag{3}$$

where $\mathcal{F}_{\mathrm{sh}}$, $\mathcal{F}_{\mathrm{red}}$, and $\mathcal{F}_{\mathrm{hf}}$ denote shared correction, polarization redistribution, and high-frequency compensation, respectively. Each component extracts a lookup index, queries a precompiled LUT by nearest-neighbor lookup, and applies the queried correction through fixed algebraic write-back.

Following the common LUT terminology, we use *lookup index* to denote the compact feature tuple used to access a table. After quantization, it is mapped to a discrete bin index b for table filling or lookup. During offline construction, the teacher-guided residual used to fill a bin is called the *target correction*; after averaging, the stored table value is the *LUT entry*; during online inference, the value retrieved from the table is the *queried correction*. We use three LUT components because the teacher-guided analyzer residual is not a single homogeneous correction mode: it contains a shared component across analyzer branches, a pairwise redistribution component constrained by the Stokes structure, and a residual component in the provisional S_1/S_2 maps. This factorization allows each LUT to model a simpler and more compact correction rule.

Shared correction. The first component models the correction mode broadly shared across analyzer branches. From the initialized analyzers $\boldsymbol{A}^{(0)}$, we construct a shared source $\boldsymbol{X}^{(0)}$ by analyzer-wise aggregation and compute its local spatial average $\bar{\boldsymbol{X}}^{(0)}$. The teacher analyzers $\boldsymbol{A}^t$ are aggregated in the same way to obtain $\boldsymbol{X}^t$. The lookup index and table target are

$$I^{\mathrm{sh}} = [\boldsymbol{X}^{(0)}, \bar{\boldsymbol{X}}^{(0)}], \qquad \Delta \boldsymbol{X}^{\mathrm{sh}} = \boldsymbol{X}^t - \boldsymbol{X}^{(0)}. \tag{4}$$

This yields a 2D LUT indexed by I^{sh}. During online inference, the queried shared correction is broadcast to all analyzer branches, producing the shared-corrected analyzers $\boldsymbol{A}^{(1)}$. This step removes the dominant common interpolation artifact before considering analyzer-pair differences.

Polarization redistribution. The second component redistributes correction inside Stokes-related analyzer pairs. Instead of repeating the Stokes definitions, we exploit the pairwise structure implied by Eq. (1): $\boldsymbol{S}_1$ and $\boldsymbol{S}_2$ encode the imbalance between the two analyzer pairs. For each pair component $k \in \{1, 2\}$, the lookup index and target are

$$I_k^{\mathrm{re}} = [\boldsymbol{S}_k^{(0)}, \boldsymbol{S}_0^{(0)}, \boldsymbol{S}_0^{(1)}], \qquad \Delta \boldsymbol{S}_k^{\mathrm{re}} = \boldsymbol{S}_k^t - \boldsymbol{S}_k^{(0)}, \qquad k \in \{1, 2\}. \tag{5}$$

Here, $\boldsymbol{S}_k^{(0)}$ describes the original pairwise polarization structure, $\boldsymbol{S}_0^{(0)}$ provides the initial intensity context, and $\boldsymbol{S}_0^{(1)}$ reflects how the shared correction has changed the total intensity level. Since $k = 1$ and $k = 2$ correspond to two different analyzer pairs, this component is implemented as two 3D LUTs. During online inference, the queried redistribution corrections are written back symmetrically to the corresponding analyzer pairs, yielding the redistributed analyzers $\boldsymbol{A}^{(2)}$. This component focuses on how the shared analyzer correction should be redistributed within Stokes-related analyzer pairs, rather than on local detail refinement.

High-frequency compensation. The third component compensates for the remaining residual details in the provisional S_1/S_2 maps. After polarization redistribution, we compute provisional Stokes components from $\boldsymbol{A}^{(2)}$. For each $k \in \{1, 2\}$, the lookup index and target are defined as

$$I_k^{\mathrm{hf}} = [\boldsymbol{S}_k^{(2)}, \bar{\boldsymbol{S}}_k^{(2)}], \qquad \Delta \boldsymbol{S}_k^{\mathrm{hf}} = \boldsymbol{S}_k^t - \boldsymbol{S}_k^{(2)}, \qquad k \in \{1, 2\}. \tag{6}$$

Here, $\bar{\boldsymbol{S}}_k^{(2)}$ acts as a local low-frequency reference, so the pair $[\boldsymbol{S}_k^{(2)}, \bar{\boldsymbol{S}}_k^{(2)}]$ indexes residual local deviations in the provisional Stokes map. Thus, this component targets remaining high-frequency polarization details after redistribution, rather than the pairwise imbalance handled by the previous

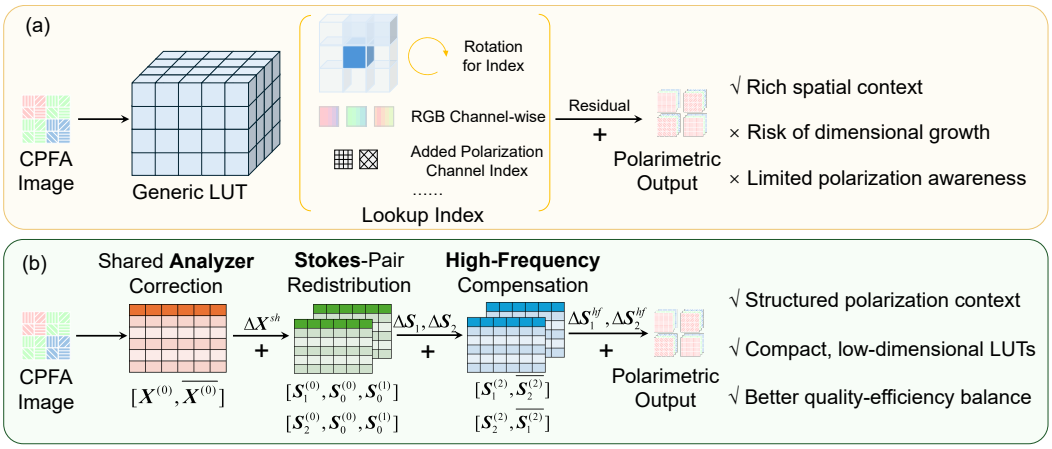

Figure 3: Design rationale of DoFP-LUT. Generic LUTs provide useful spatial lookup context but are not directly aligned with polarization demosaicking. DoFP-LUT instead factorizes the correction into compact, polarization-structured LUTs, improving the quality–efficiency trade-off.

component. This component is implemented as two 2D LUTs, one for $\boldsymbol{S}_1$ and one for $\boldsymbol{S}_2$. During online inference, the queried compensation corrections are written back to the corresponding analyzer pairs, producing the final corrected analyzers $\hat{\boldsymbol{A}}$.

All LUTs are constructed with the same bin-averaging rule. For a lookup component with index I_i and target correction Δ_i, each training pixel is assigned to a discrete bin according to its quantized index, and the corresponding target correction is accumulated into that bin:

$$\boldsymbol{L}(b) = \frac{\sum_i \Delta_i \, \mathbb{I}[\mathcal{B}(I_i) = b]}{\sum_i \mathbb{I}[\mathcal{B}(I_i) = b] + \epsilon}, \tag{7}$$

where $\mathcal{B}(\cdot)$ denotes the quantization operator. Each bin therefore stores the average teacher-guided correction over training pixels with the same quantized index.

In our implementation, lookup indices are uniformly quantized with an interval of 3, which is selected on the validation split to balance reconstruction accuracy, bin occupancy, and storage cost. During inference, DoFP-LUT uses nearest-neighbor lookup: each continuous lookup index is mapped to its nearest discrete bin, and the corresponding table entry is retrieved as the queried correction. Empty bins use a zero-correction fallback, which leaves the current estimate unchanged for that component and avoids unstable extrapolation. At test time, DoFP-LUT therefore performs only index extraction, nearest-neighbor LUT query, and fixed algebraic write-back.

3.3 Design Rationale of Structured LUT Factorization

The structured design is motivated by both DoFP residual structure and the statistical preference of LUT inference. A LUT is most reliable when its index is compact, its bins are sufficiently populated, and its stored corrections correspond to a relatively homogeneous mapping. Fig. 3 contrasts generic LUT transfer with the proposed DoFP-structured factorization.

Existing LUT restoration methods are effective in their native tasks because their indices match the corresponding residual structures, often through enriched spatial patterns for RGB restoration. Directly applying such designs to color DoFP demosaicking is less suitable: a compact generic index may miss analyzer/Stokes-specific correction modes, while a monolithic index over all relevant variables would increase dimensionality and sparsity.

DoFP-LUT addresses this mismatch by decomposing correction into low-dimensional tables. The shared and high-frequency components are 2D LUTs, while polarization redistribution uses two 3D LUTs for the two Stokes-related analyzer pairs. If each index dimension contains B bins, the tables scale as $\mathcal{O}(B^2)$ or $\mathcal{O}(B^3)$ instead of forming a single high-dimensional table. The sequential dependency from $\boldsymbol{A}^{(0)}$ to $\boldsymbol{A}^{(1)}$, $\boldsymbol{A}^{(2)}$, and $\hat{\boldsymbol{A}}$ further transfers intermediate polarization information across components. This factorization also provides a compact DoFP-structured context. Local

Table 1: Quantitative comparison of quality/efficiency. Runtime and energy cost are measured on an RK3588S edge platform with four Arm Cortex-A76 Cores under input resolution 512×512. Storage denotes deployed storage, including model weights or lookup tables required for inference.

	Method	PSNR(dB) / SSIM ↑			MAE(°) ↓	Runtime(ms) ↓	Storage(KB) ↓	Energy Cost(pJ) ↓
		Analyzer	S_0	DoLP	AoP	RK3588S, 4×Arm Cortex-A76 Cores		
Interp.	Bilinear	34.86/0.9449	35.21/0.9547	29.29/0.7790	10.893	12	–	29.26M
	EARI	36.01/0.9469	36.44/0.9573	29.91/0.7881	10.642	42	–	72.32M
DNN	PDCNN	38.97/0.9701	40.22/0.9768	32.36/0.8543	7.690	1760	1441	2.87G
	PIDSR	40.66/0.9776	41.84/0.9865	34.54/0.8805	5.947	5082	30066	64.90G
LUT	SRLUT	36.34/0.9482	36.87/0.9600	30.07/0.7991	10.334	88	1094	755.54M
	MuLUT	33.17/0.9247	33.55/0.9342	28.60/0.7756	11.214	92	2650	1.75G
	RCLUT	36.52/0.9503	36.88/0.9603	30.26/0.8099	10.228	266	1304	1.01G
	DnLUT	35.98/0.9462	36.64/0.9574	29.69/0.8021	10.522	34	2072	1.45G
	ShiftLUT	35.21/0.9483	36.67/0.9598	29.37/0.8041	10.739	232	1280	212.17M
Ours	**DoFP-LUT**	38.24/0.9657	39.12/0.9730	31.10/0.8236	9.207	29	149	54.58M

statistics such as $\bar{\boldsymbol{X}}^{(0)}$ and $\bar{\boldsymbol{S}}_k^{(2)}$ provide spatial context, while analyzer aggregation and Stokes-related pairwise quantities provide polarization context. Thus, DoFP-LUT conditions each queried correction on both local statistics and intermediate polarization states without constructing a high-dimensional monolithic table or running an online neural network.

4 Experiments

4.1 Experimental Setup

Datasets. We evaluate DoFP-LUT on public color-polarization datasets [18, 23, 24, 37] and our self-collected dataset, covering diverse scenes, materials, textures, illumination conditions, and polarization responses. All data are organized and split at the scene level to avoid training-test leakage. For each dataset, we reserve 14 representative scenes for testing, resulting in 70 held-out test scenes in total. This balanced test selection prevents any single dataset from dominating the evaluation and better reflects generalization across different capture sources. The remaining scenes are used for training and validation, where the training split is used to train the PDCNN teacher and compile LUTs, and the validation split is used to select implementation hyperparameters such as the quantization interval. All CPFA measurements are generated using the same color DoFP sampling pattern, with full-resolution analyzer images at 0°, 45°, 90°, and 135° used as reconstruction targets.

Compared methods. We compare DoFP-LUT with three groups of polarization demosaicking methods. The first group includes interpolation-based DoFP-tailored bilinear [17] and EARI [18], which represent lightweight deployment-friendly baselines. The second group includes network-based PDCNN [35] and PIDSR [37]. PIDSR is a Restormer-based high-performing method and serves as a strong reconstruction-quality reference, but its deployment cost is considerably higher. The third group includes generic LUT-based restoration methods adapted to color DoFP demosaicking, including SRLUT [2], MuLUT [6], RCLUT [9], DnLUT [33], and ShiftLUT [34] when applicable. For fairness, all adapted LUT baselines use the same training split, CPFA sampling pattern, interpolation initialization, reconstruction targets, and deployment protocol as DoFP-LUT. We also adapt their input/output interfaces and channel handling to support polarization analyzer channels while keeping their original lookup principles unchanged. These baselines test whether generic LUT designs remain effective after polarization-compatible adaptation.

Metrics. We report reconstruction quality in both image and polarization domains. For analyzer, S_0, and DoLP reconstruction, we report PSNR and SSIM. For AoP, we report mean absolute error (MAE) in degrees with angular periodicity considered. For deployment efficiency, we report runtime, deployed storage, and energy cost. Deployed storage denotes the model weights or lookup tables required at inference time, excluding offline accumulators used only during LUT construction. Runtime and energy cost is measured on an RK3588S edge platform with four Arm Cortex-A76 Cores under the same input resolution 512×512 and evaluation protocol.

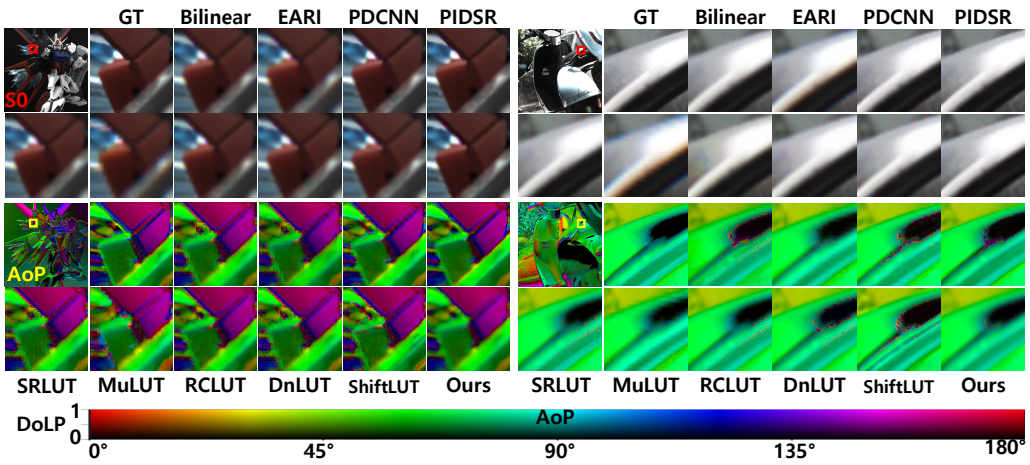

Figure 4: Qualitative comparison on S_0 and HSV-based polarization visualization, where hue encodes AoP and value encodes DoLP. DoFP-LUT better preserves intensity structures and polarization cues, especially around edges and high-frequency textures. Zoomed-in regions are shown for clarity.

4.2 Main Comparison

Table 1 reports the quantitative comparison across interpolation-based methods, DNN-based methods, generic LUT adaptations, and the proposed DoFP-LUT. The results show a clear trade-off among reconstruction quality, edge-platform runtime, deployed storage, and energy cost. Interpolation methods are lightweight and fast, but their polarization fidelity is limited because they mainly rely on local spatial smoothness and do not explicitly correct coupled analyzer/Stokes residuals. DNN-based methods achieve the strongest reconstruction quality, especially the Restormer-based PIDSR, but their online neural inference introduces much higher deployment cost on the representative edge platform. Generic LUT methods remain efficient after being adapted to the color DoFP setting, but their reconstruction quality is still limited compared with polarization-specific learning-based methods. This indicates that simply modifying the input/output interfaces and channel handling of existing LUT designs is not sufficient for color DoFP demosaicking. Although these methods preserve the deployment advantage of lookup inference, their lookup indices are originally designed for generic RGB restoration and are not fully matched to the coupled analyzer and Stokes structures in polarization demosaicking.

In contrast, DoFP-LUT achieves the best overall performance among LUT-based methods while maintaining compact storage, efficient runtime, and low energy cost. It consistently improves over interpolation and generic LUT baselines in both image-domain and polarization-domain metrics, demonstrating the benefit of the proposed polarization-structured lookup formulation. Compared with DNN-based methods, DoFP-LUT does not aim to reach the highest possible reconstruction quality; instead, it provides a more deployment-friendly operating point for resource-constrained polarization imaging. These results support the design rationale in Fig. 3: LUT inference is effective for DoFP demosaicking when the lookup formulation is aligned with analyzer-domain artifacts, Stokes-related redistribution, and residual polarization details.

Fig. 4 shows representative visual comparisons on S_0 and HSV-based polarization visualization, where hue encodes AoP and value encodes DoLP. Interpolation and generic LUT adaptations tend to produce unstable polarization structures around edges and high-frequency textures, even when some local intensity artifacts are reduced. DoFP-LUT better preserves both S_0 structures and polarization cues, suggesting that the proposed lookup formulation is more consistent with the analyzer/Stokes coupling in color DoFP demosaicking.

4.3 Ablation Study

Component contribution. We evaluate the contribution of each LUT component in Table 2 and Fig. 5. All variants start from the same DoFP-tailored bilinear initialization. T1, T2, and T3 denote

Init.	T1	T2	T3	PSNR(dB) ↑		MAE(°) ↓
				S_0	DoLP	AoP
✓				34.86	29.29	10.893
✓	✓			38.17	29.28	10.229
✓	✓	✓		38.73	30.32	10.131
✓			✓†	37.96	30.55	9.667
✓	✓	✓	✓	**39.12**	**31.10**	**9.207**

Table 2: Component ablation.

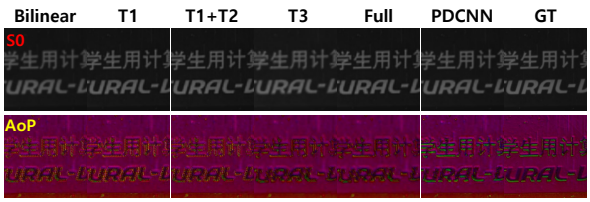

Figure 5: Visual ablation on S_0 and AoP.

Table 3: Ablation on lookup formulation under matched teacher supervision.

Formulation	Indexing Strategy	Dim.	Storage(KB) ↓	DoLP PSNR(dB) ↑	AoP MAE(°) ↓
Single-table compact	Merged low-D states	4D	1685	29.47	10.790
Single-table monolithic	Joint all states	7D	1.91×10^9	30.44	9.871
DoFP-LUT	**Structured multi-table**	**2D+2×3D+2×2D**	**149**	**31.10**	**9.207**

shared analyzer correction, polarization redistribution, and high-frequency compensation, respectively. T1 substantially improves S_0 reconstruction and reduces AoP error, indicating that shared analyzer correction suppresses dominant common interpolation artifacts. Adding T2 after T1 further improves DoLP, showing the benefit of Stokes-related polarization redistribution. We also include a standalone T3 variant, denoted by †, where the high-frequency compensation table is compiled directly from the bilinear initialization-to-teacher residual. Although this variant improves DoLP and AoP, it bypasses analyzer-domain shared correction and Stokes-pair redistribution, leaving part of the structured interpolation error insufficiently corrected. In the full DoFP-LUT, T3 is compiled after T2 and refines the remaining residuals, leading to the best overall performance. The visual progression in Fig. 5 is consistent with the quantitative trend in Table 2.

LUT formulation. We further examine whether the gain comes from the proposed structured lookup formulation rather than simply increasing LUT capacity. For a controlled comparison, both single-table baselines use the same DoFP-tailored initialization and PDCNN teacher as DoFP-LUT. Their lookup states and target corrections are constructed to approximate the three correction roles of DoFP-LUT as closely as possible: the compact variant merges these cues into a lower-dimensional single table, while the monolithic variant jointly indexes all selected states in one high-dimensional table. As shown in Table 3, the compact single-table design is easier to store but lacks sufficient correction capacity, whereas the monolithic design improves reconstruction but becomes impractical due to prohibitive storage. DoFP-LUT achieves better polarization reconstruction with much smaller storage by factorizing the correction into structured low-dimensional tables. Additional bin-coverage analysis in Appendix B shows that all factorized LUTs are fully populated and no zero-correction fallback is triggered on the 70 held-out evaluation scenes, further supporting the stability of the proposed lookup formulation. Together with the generic LUT comparisons in Table 1, these results confirm that the proposed design is not merely a larger LUT, but a more suitable discrete residual approximation for DoFP demosaicking.

4.4 Discussion

DoFP-LUT is not intended to replace large learning-based demosaickers when high reconstruction quality is the only objective. Instead, it targets real-time polarization imaging, where latency, storage, energy, and polarization fidelity must be balanced. Our RK3588S measurements are representative edge-platform profiling rather than a claim of integration into a specific camera hardware pipeline. The experiments show that generic LUT designs cannot be made fully effective by only adapting their input/output channels, since their indices remain insufficiently matched to analyzer/Stokes residuals. DoFP-LUT addresses this mismatch by decomposing teacher-guided correction into shared analyzer correction, polarization redistribution, and high-frequency compensation, improving polarization reconstruction while preserving compact table-only inference.

Beyond average runtime, DoFP-LUT also provides a predictable inference pattern. Once the tables are compiled, the online path contains no convolution, attention, iterative optimization, or input-adaptive network execution. Each spatial location follows the same sequence of index extraction, nearest-

neighbor lookup, and fixed algebraic write-back. This deterministic execution is desirable for near-sensor polarization processing, where stable latency, small deployed storage, and simple scheduling can be as important as average reconstruction accuracy. More broadly, LUTs for computational imaging should not be treated only as generic acceleration tools. For structured sensors such as DoFP cameras, lookup indices should encode variables that govern dominant reconstruction errors.

4.5 Limitations

DoFP-LUT compiles sensor-structured corrections into compact quantized tables, making the deployed tables specific to the sensing configuration used during compilation. The overall three-component formulation can be applied to other DoFP cameras, but the resulting tables depend on the CPFA layout, sensor response, teacher residuals, and quantization interval. A new sensor or substantial domain shift may therefore require recompilation, while teacher limitations and collisions between quantized states can bound the achievable correction accuracy. Our multi-source experiments test scene and source variation under a common 4×4 CPFA, leaving cross-layout and cross-sensor robustness for future study. Moreover, the RK3588S results characterize a representative edge-CPU implementation rather than a camera ISP or dedicated accelerator.

5 Conclusion

We presented **DoFP-LUT**, a DoFP-aligned lookup-table framework for real-time color polarization demosaicking. Instead of directly transferring generic LUT designs, DoFP-LUT factorizes the correction according to analyzer-domain artifacts, Stokes-related pairwise coupling, and residual polarization details, resulting in compact low-dimensional LUTs with deterministic online inference. Experiments show that this polarization-structured lookup formulation improves reconstruction over interpolation and generic LUT adaptations while maintaining a favorable quality–efficiency trade-off on embedded hardware. We hope this work encourages future LUT-based computational imaging methods to design lookup indices and factorization strategies around the physical structure of the sensing process, rather than treating LUTs only as generic acceleration tools for efficient inference.

Acknowledgements

This work was supported by the National Natural Science Foundation of China (U2541205), Hunan Provincial Research and Development Project (2025QK3019), National Key Research and Development Program of China (2024YFF0505603), National Natural Science Foundation of China (62271414), Zhejiang "Pioneer" and "Leading Goose" R&D Program (2024SDXHDX0006), and the 2023 International Sci-tech Cooperation Projects under the purview of the "Innovation Yongjiang 2035" Key R&D Program (2024Z126).

Competing interests. The authors declare no competing interests.

References

[1] Ronan Dumoulin, Pierre-Jean Lapray, Jean-Baptiste Thomas, and Ivar Farup. Impact of training data on LMMSE demosaicing for colour-polarization filter array. In *2022 16th International Conference on Signal-Image Technology & Internet-Based Systems*, pages 275–280, 2022.

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

[3] Chenggong Li, Yidong Luo, Caiyun Wu, Junchao Zhang, Degui Yang, and Dangjun Zhao. Demosaicking customized diffusion model for snapshot polarization imaging. *Optics and Laser Technology*, 188:112868, 2025.

[4] Chenggong Li, Yidong Luo, Junchao Zhang, and Degui Yang. Polarization uncertainty-guided diffusion model for color polarization image demosaicking. In *Proc. of the AAAI Conference on Artificial Intelligence*, volume 40, pages 6028–6036, 2026.

[5] Huanan Li, Juntao Guan, Rui Lai, Sijun Ma, Lin Gu, and Zhangming Zhu. TinyLUT: Tiny look-up table for efficient image restoration at the edge. In *Advances in Neural Information Processing Systems*, volume 37, pages 85340–85359, 2024.

[6] 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, 2022.

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

[8] 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*, 32:4742–4756, 2023.

[9] 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 *Proc. of the IEEE/CVF International Conference on Computer Vision*, pages 12217–12226, 2023.

[10] Yang Lu, Weihong Ren, Yiming Su, Zhen Zhang, Junchao Zhang, and Jiandong Tian. Polarization image demosaicking based on homogeneity space. *Optics and Lasers in Engineering*, 178:108179, 2024.

[11] Yang Lu, Jiandong Tian, Yiming Su, Yidong Luo, Junchao Zhang, and Chunhui Hao. A hybrid polarization image demosaicking algorithm based on inter-channel correlation. *IEEE Transactions on Computational Imaging*, 10:1400–1413, 2024.

[12] Yidong Luo, Chenggong Li, Yunfeng Song, Ping Wang, Boxin Shi, Junchao Zhang, and Xin Yuan. PolarAPP: Beyond polarization demosaicking for polarimetric applications. In *European Conference on Computer Vision*, pages 115–133, 2026.

[13] Yidong Luo, Caiyun Wu, Chenggong Li, Ping Wang, Xin Yuan, Kailun Yang, and Junchao Zhang. UP-Dem: Deep unrolling convolutional sparse coding for color polarization image demosaicking. *IEEE Transactions on Image Processing*, 35:10152–10166, 2026.

[14] Yidong Luo, Junchao Zhang, and Chenggong Li. Cpifuse: Toward realistic color and enhanced textures in color polarization image fusion. *Information Fusion*, 120:103111, 2025.

[15] Yidong Luo, Junchao Zhang, Jianbo Shao, Jiandong Tian, and Jiayi Ma. Learning a non-locally regularized convolutional sparse representation for joint chromatic and polarimetric demosaicking. *IEEE Transactions on Image Processing*, 33:5029–5044, 2024.

[16] Yidong Luo, Junchao Zhang, and Di Tian. Sparse representation-based demosaicking method for joint chromatic and polarimetric imagery. *Optics and Lasers in Engineering*, 164:107526, 2023.

[17] Ryota Maeda. Polanalyser: Polarization image analysis tool. `https://github.com/elerac/polanalyser`, 2019.

[18] Miki Morimatsu, Yusuke Monno, Masayuki Tanaka, and Masatoshi Okutomi. Monochrome and color polarization demosaicking using edge-aware residual interpolation. In *Proc. of the IEEE International Conference on Image Processing*, pages 2571–2575, 2020.

[19] Miki Morimatsu, Yusuke Monno, Masayuki Tanaka, and Masatoshi Okutomi. Monochrome and color polarization demosaicking based on intensity-guided residual interpolation. *IEEE Sensors Journal*, 21(23):26985–26996, 2021.

[20] Manasi Muglikar, Leonard Bauersfeld, Diederik Paul Moeys, and Davide Scaramuzza. Event-based shape from polarization. In *Proc. of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1547–1556, 2023.

[21] Vy Nguyen, Masayuki Tanaka, Yusuke Monno, and Masatoshi Okutomi. Two-step color-polarization demosaicking network. In *Proc. of the IEEE International Conference on Image Processing*, pages 1011–1015, 2022.

[22] Matteo Ottaviani, Gabriel Harris Myers, and Nan Chen. Global sensitivity analysis of simulated remote sensing polarimetric observations over snow. *Atmospheric Measurement Techniques*, 17:4737–4756, 2024.

[23] Simeng Qiu, Qiang Fu, Congli Wang, and Wolfgang Heidrich. Linear polarization demosaicking for monochrome and colour polarization focal plane arrays. *Computer Graphics Forum*, 40(6):77–89, 2021.

[24] Muhamad Daniel Ariff Bin Abdul Rahman, Yusuke Monno, Masayuki Tanaka, and Masatoshi Okutomi. Polarization denoising and demosaicking: Dataset and baseline method. In *Proc. of the IEEE International Conference on Image Processing*, pages 2724–2729, 2025.

[25] Yuanyuan Sun, Junchao Zhang, and Rongguang Liang. Color polarization demosaicking by convolutional neural network. *Optics Letters*, 46(17):4338–4341, 2021.

[26] Abudusalamu Tuniyazi, Tingkui Mu, Xiaosa Jiang, Xuechan Lang, Qiuxia Li, Haodong Yu, Feng Han, Jiang Ban, and Bin Qin. Snapshot polarized light scattering spectrometric fiberscopy for early cancer detection. *Laser & Photonics Reviews*, 18(12):2400486, 2024.

[27] 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 *Proc. of the IEEE/CVF International Conference on Computer Vision*, pages 2471–2480, 2021.

[28] Tianyi Wang, Xiangyue Zhang, Junlin Li, Yubo Zheng, Jingyu Ru, and Hongli Xu. SwinCPD: A swin transformer-based network for color-polarization demosaicing. *IEEE Sensors Journal*, 26(1):637–651, 2026.

[29] Sijia Wen, Yinqiang Zheng, Feng Lu, and Qinping Zhao. Convolutional demosaicing network for joint chromatic and polarimetric imagery. *Optics Letters*, 44(22):5646–5649, 2019.

[30] Rongyuan Wu, Yongqiang Zhao, Ning Li, and Seong G Kong. Polarization image demosaicking using polarization channel difference prior. *Optics Express*, 29(14):22066–22079, 2021.

[31] 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 *Proc. of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 17522–17531, 2022.

[32] 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 *European Conference on Computer Vision*, pages 201–217, 2022.

[33] 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 *Proc. of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7582–7591, 2025.

[34] 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 *Proc. of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 29959–29968, 2026.

[35] Junchao Zhang, Jianbo Shao, Haibo Luo, Xiangyue Zhang, Bin Hui, Zheng Chang, and Rongguang Liang. Learning a convolutional demosaicing network for microgrid polarimeter imagery. *Optics Letters*, 43(18):4534–4537, 2018.

[36] Yongqiang Zhao, Xinbo Qiao, Ning Li, and Quan Pan. Polarization vision. *SCIENTIA SINICA Informationis*, 54(7):1620–1645, 2024.

[37] Shuangfan Zhou, Chu Zhou, Youwei Lyu, Heng Guo, Zhanyu Ma, Boxin Shi, and Imari Sato. PIDSR: Complementary polarized image demosaicing and super-resolution. In *Proc. of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 16081–16090, 2025.

[38] Zhen Zhu, Xiaobo Li, Jingsheng Zhai, and Haofeng Hu. PODB: A learning-based polarimetric object detection benchmark for road scenes in adverse weather conditions. *Information Fusion*, 108:102385, 2024.

A Supplementary Details of PDCNN

A.1 Teacher Selection for LUT Compilation

DoFP-LUT uses PDCNN as the offline teacher for table construction. This choice is not made because PDCNN is the strongest available demosaicking network, but because it is a suitable *LUT-compilable teacher*. The teacher for DoFP-LUT should not only be accurate, but also compilable: its residual behavior should be expressible by compact lookup indices and stable under bin averaging. Very deep, transformer-based, or heavily multi-stage networks may improve reconstruction quality by exploiting nonlocal dependencies and hidden feature interactions, but such correction behavior is difficult to summarize with low-dimensional LUT states. Pixels falling into the same quantized bin may still require different corrections, which weakens the reliability of the averaged LUT entry. In contrast, PDCNN is a compact convolutional demosaicker with more local residual behavior, making it better matched to the representation capacity of structured LUTs.

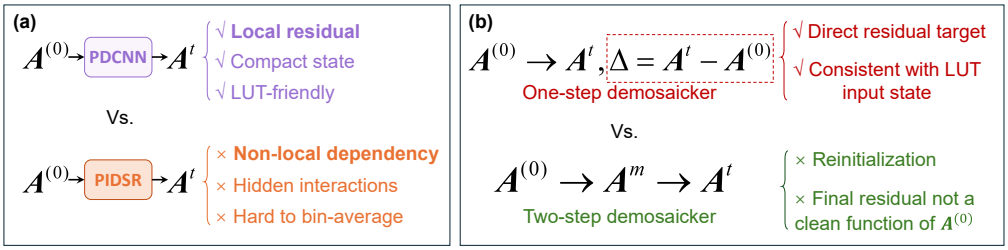

Figure 6: Rationale for using PDCNN as the LUT teacher. (a) A compact one-step PDCNN provides local residual mappings that are more compatible with low-dimensional LUT indices, whereas deeper or nonlocal models may produce corrections that are difficult to summarize by bin averaging. (b) A one-step demosaicker teacher gives a direct mapping from initialized analyzers to teacher analyzers, while a two-step teacher demosaicker introduces intermediate states that are not directly available in the deployed LUT path.

The one-step property is also important. Given the initialized analyzers $\boldsymbol{A}^{(0)}$, PDCNN directly predicts teacher analyzers $\boldsymbol{A}^t$:

$$\boldsymbol{A}^t = \mathcal{N}_{\mathrm{PDCNN}}(\boldsymbol{A}^{(0)}). \tag{8}$$

This gives a direct residual target from $\boldsymbol{A}^{(0)}$ to $\boldsymbol{A}^t$, which matches the deployed LUT path. By contrast, a two-step teacher introduces an intermediate reconstruction,

$$\boldsymbol{A}^t = \mathcal{N}_2(\boldsymbol{A}^m), \qquad \boldsymbol{A}^m = \mathcal{N}_1(\boldsymbol{A}^{(0)}), \tag{9}$$

where $\boldsymbol{A}^m$ is not directly available during table-only inference unless the first-stage behavior is also compiled. If the second stage refines or reinitializes the intermediate output, the final residual is no longer a clean and stable mapping from the original initialization state. This would create a mismatch between offline teacher residuals and online LUT states. Therefore, we use a one-step PDCNN teacher throughout LUT construction.

A.2 PDCNN Architecture and Training Details

PDCNN is implemented as a compact same-resolution residual CNN. As shown in Fig. 7, it processes the 12-channel bilinear-initialized analyzer stack through sequential convolutional blocks, connects intermediate features to the final fusion layer through hierarchical skip connections, and predicts an analyzer residual:

$$\boldsymbol{A}^t = \boldsymbol{A}^{(0)} + \mathcal{R}_\theta(\boldsymbol{A}^{(0)}), \tag{10}$$

where $\mathcal{R}_\theta$ denotes the residual CNN. All blocks operate without spatial downsampling, so the predicted residual remains pixel-aligned with the initialization. This alignment is important for LUT compilation, where each table entry represents a local correction associated with a quantized lookup state. The teacher output $\boldsymbol{A}^t$ is used only during offline LUT construction and is not executed during DoFP-LUT inference.

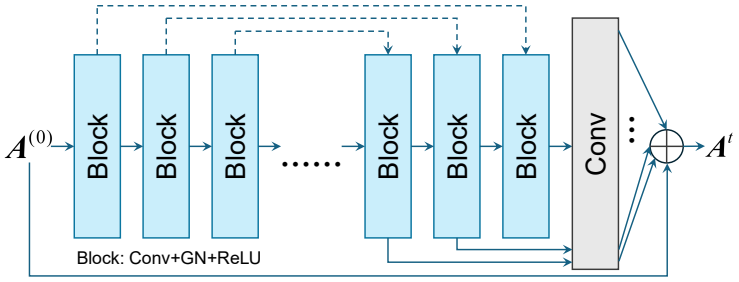

Figure 7: Architecture of the PDCNN teacher.

Table 4: LUT bin coverage and fallback statistics. All factorized LUT components are fully populated during offline compilation, and no zero-correction fallback is triggered on the 70 held-out evaluation scenes. It indicates the proposed structured factorization keeps the lookup space dense and reliable.

Component	Shape	Total bins	Stored occupancy	Eval. unique queried bins	Zero-correction fallback
T1	33×33	1,089	1,089 / 100.00%	263 / 24.15%	0 / 0.000000%
T2	$33 \times 33 \times 33$	35,937	35,937 / 100.00%	7,000 / 19.48%	0 / 0.000000%
T3	33×33	1,089	1,089 / 100.00%	362 / 33.24%	0 / 0.000000%

The teacher is trained on 500 training scenes after holding out 70 scenes for testing. We use multi-domain supervision over analyzer, Stokes, DoLP, and AoP quantities. Let $\boldsymbol{A}^{gt}$ denote the ground-truth analyzer stack, and let $\boldsymbol{S}^t$, $\boldsymbol{S}^{gt}$, DoLP^t, DoLP^{gt}, AoP^t, and AoP^{gt} be computed from the predicted and ground-truth analyzers. We use the Charbonnier penalty

$$\rho(x) = \sqrt{x^2 + \epsilon_c^2}. \tag{11}$$

ϵ_c is set to 10^{-5}. The analyzer, Stokes, and DoLP losses are defined as

$$\mathcal{L}_A = \left\langle \rho(\boldsymbol{A}^t - \boldsymbol{A}^{gt}) \right\rangle, \quad \mathcal{L}_S = \left\langle \rho(\boldsymbol{S}^t - \boldsymbol{S}^{gt}) \right\rangle, \quad \mathcal{L}_D = \left\langle \rho(\mathrm{DoLP}^t - \mathrm{DoLP}^{gt}) \right\rangle, \tag{12}$$

where $\langle \cdot \rangle$ denotes averaging over pixels and channels. For AoP, we use a periodic angular loss:

$$\mathcal{L}_\theta = \left\langle \rho\big(1 - \cos\big(2(\mathrm{AoP}^t - \mathrm{AoP}^{gt})\big)\big) \right\rangle, \tag{13}$$

where the factor of 2 accounts for the π-periodicity of AoP. The total teacher loss is

$$\mathcal{L}_{\mathrm{PDCNN}} = \mathcal{L}_A + \mathcal{L}_S + \mathcal{L}_D + 0.5\mathcal{L}_\theta. \tag{14}$$

B LUT Bin Coverage and Fallback Analysis

A key requirement for reliable LUT inference is that the lookup space should be sufficiently populated during offline compilation and well covered during evaluation. If a table is high-dimensional or poorly matched to the residual structure, many bins may remain empty, and test-time queries may frequently trigger zero-correction fallback. Such behavior would make the LUT correction unstable and would weaken the reliability of the averaged table entries. We therefore analyze the bin coverage of the proposed factorized LUTs.

Table 4 reports the bin statistics of the three LUT components used in DoFP-LUT. The statistics are computed after offline table construction and on the 70 held-out evaluation scenes. For each component, we report the table shape, the number of total bins, the number of bins populated during LUT compilation, the number of unique bins queried by nearest-neighbor lookup during evaluation, and the number of zero-correction fallback events. For T2 and T3, the reported statistics are computed over the corresponding S_1/S_2 LUTs with the same table shape.

The results show that all LUT components are fully populated after offline compilation. This is important because each stored LUT entry is obtained by averaging teacher-guided residual corrections within the same quantized bin. Full stored occupancy indicates that the proposed low-dimensional factorization avoids the sparse-bin issue that would arise in a high-dimensional monolithic LUT. During evaluation, only a subset of the compiled bins is queried, but all evaluation queries fall into

Table 5: Additional ablation on quantization interval, RGB-wise indexing and teacher choice. The default DoFP-LUT uses interval $= 3$, one-step PDCNN teacher, and no additional RGB-wise index.

Variant	S_0 PSNR(dB) ↑	DoLP PSNR(dB) ↑	AoP MAE(°) ↓	Storage(KB) ↓
Interval $= 1$	39.20	31.16	9.198	8515
Interval $= 2$	39.19	31.15	9.197	1106
Interval $= 3$ (default)	39.12	31.10	9.207	149
Interval $= 4$	38.75	30.22	9.393	21
RB/G-wise indexing	39.27	30.98	9.218	3416
DnLUT-style RGB indexing	39.28	30.99	9.219	1139
TCPDNet (one-step), teacher	40.30	32.55	7.190	24904
TCPDNet (one-step), LUT	39.03	31.28	9.270	149
PIDSR (one-step), teacher	41.40	34.26	6.323	23660
PIDSR (one-step), LUT	38.46	30.61	9.611	149
TCPDNet (two-stage), teacher	40.98	33.61	6.610	33188
TCPDNet (two-stage), LUT	36.29	29.86	10.321	2*149
PIDSR (two-stage), teacher	41.84	34.54	5.947	30066
PIDSR (two-stage), LUT	36.77	30.54	9.813	2*149

Table 6: Matched deployment comparison on RK3588S.

Method	S_0 PSNR	DoLP PSNR	AoP MAE (°)	Runtime (ms)	Storage (KB)	Energy (pJ)
Pruned and quantized PDCNN	38.86	30.79	9.350	461	133	1.23G
DoFP-LUT	**39.12**	**31.10**	**9.207**	**29**	149	**54.58M**

valid populated bins. As a result, the zero-correction fallback is never triggered on the held-out scenes. This suggests that DoFP-LUT does not rely on fallback behavior to handle unseen states. Instead, the lookup queries remain within the covered residual space learned during table construction.

This analysis further supports the design rationale of DoFP-LUT. The factorization into shared analyzer correction, polarization redistribution, and high-frequency compensation does not merely reduce storage; it also makes each lookup component compact enough to be densely populated and stable under nearest-neighbor lookup. Therefore, compared with a monolithic high-dimensional LUT, the proposed structured LUT formulation provides a more reliable discrete approximation of teacher-guided residual correction.

C Additional Ablation Studies

We further evaluate three implementation choices of DoFP-LUT in Table 5: the quantization interval, additional RGB-wise indexing strategies, and the teacher used for LUT compilation. All LUT variants use the same training split, validation split, CPFA sampling pattern, and table-only inference pipeline unless otherwise specified. Storage denotes the deployed LUT storage required for inference; neural-teacher rows report reconstruction quality only.

Quantization interval. The first four rows study the effect of the quantization interval. Smaller intervals provide finer bins and slightly better reconstruction quality, but they substantially increase table storage. Although intervals 1 and 2 achieve marginally better metrics than interval 3, their storage costs are much higher. We therefore choose interval $= 3$ as the default setting because it gives a better balance between polarization reconstruction and deployed storage. Interval 4 further reduces storage, but its coarser quantization noticeably degrades DoLP and AoP accuracy.

RGB-wise indexing. We also test two RGB-wise indexing variants. The RB/G-wise variant uses one indexing strategy for the red and blue channels and another for the green channel, motivated by the different sampling densities of Bayer color channels. The DnLUT-style variant introduces an RGB channel-correlation index inspired by channel-aware LUT designs. Both variants slightly improve S_0 but do not improve DoLP/AoP, while increasing storage. This suggests that the dominant residuals in color DoFP demosaicking are better captured by analyzer aggregation, Stokes-related pairwise structure, and local polarization deviations than by additional RGB-wise channel indexing.

Table 7: Tables of supplementary deployment profiling.

Generalization on Snapdragon 8 Elite Gen 5

Method	ms ↓	Method	ms ↓
Bilinear	5.44	MuLUT	15.54
EARI	7.15	RCLUT	35.99
PDCNN	1259.75	DnLUT	9.21
PIDSR	4233.11	ShiftLUT	28.40
SRLUT	12.35	DoFP-LUT	**7.93**

RK3588S latency distribution

Method	Mean $\pm$ std. (ms)	P50 (ms)	P95 (ms)
PDCNN	1760 ± 16	1752	1774
DnLUT	33.76 ± 0.40	33.73	34.13
ShiftLUT	232 ± 9	231	238
DoFP-LUT	29.45 ± 0.61	29.41	30.02

Table 8: Implementation details for generic LUT adaptations.

Method	Adaptation, quantization, and aggregation	Storage (KB)
SRLUT	Four shifted 4-D 2×2 raw-CPFA indices, 16 phase branches, and 12-D analyzer residual; 5 levels with nearest-bin lookup; average four shifts, without rotation.	1094
MuLUT	Complementary 4-D 2×2 raw-CPFA indices, 16 phase branches, and 12-D analyzer residual; 9 levels with nearest-bin lookup; average complementary patterns, without rotation.	2650
RCLUT	2-D analyzer pairs, 3-D context gate, and 4-D region refinement; 17-level pair/gate and 9-level feature tables with linear interpolation; gated cascaded refinement, without rotation.	1304
DnLUT	2-D cross-polarization pairs followed by 2-D cross-color pairs; 17 levels in both stages with bilinear interpolation; average overlapping pair outputs, without rotation.	2072
ShiftLUT	12-channel shift/depthwise/pointwise/upsampling lookup cascade; 6-bit MSB and 2-bit LSB with adaptive sampling ($\epsilon = 0.45$); native shift aggregation, without rotation.	1280

Stronger teacher. We also evaluate TCPDNet and PIDSR as stronger teachers in both one-step and two-stage forms. TCPDNet is a purely convolutional model with more parameters than PDCNN, providing a higher-capacity CNN comparison; PIDSR additionally uses global representations. To construct their one-step variants, we apply the same adaptation to both networks: retain one reconstruction stage and train it to map the full-resolution bilinear initialization directly to full-resolution analyzer outputs, without an intermediate reconstruction. The one-step TCPDNet yields a compiled LUT set comparable to the PDCNN-guided set and slightly improves DoLP, whereas the stronger one-step PIDSR yields a weaker LUT set. The original two-stage networks have different input and output resolutions, while our LUT corrections are pixel aligned; we therefore compile a separate set of three LUTs for each stage rather than represent the full mapping with a single set. The second-stage LUT set receives the first-stage set's output rather than the teacher's exact intermediate output, so first-stage approximation errors can propagate. Both two-stage variants underperform their one-step counterparts despite stronger neural teachers. Thus, teacher accuracy alone does not determine LUT quality; corrections must also remain consistent within quantized lookup bins.

Matched-storage PDCNN. We apply 20% unstructured pruning, 600-step FP32 distillation fine-tuning, and static W8A8 post-training quantization to PDCNN. Its dense INT8 ONNX artifact occupies 133 KB, comparable to DoFP-LUT's 149 KB. As Table 6 shows, it still requires 461 ms and 1.23G pJ on the same RK3588S, about 15.9$\times$ the latency and 22.6$\times$ the energy of DoFP-LUT, while its S_0 and DoLP PSNR are lower and its AoP MAE is higher. This comparison shows that DoFP-LUT's advantage persists at a comparable deployment storage budget.

D Supplementary Deployment Profiling

Table 7 reports additional CPU-only, batch-1 latency measurements at 512×512. Snapdragon 8 Elite Gen 5 results are reported separately from the RK3588S results in Table 1. For the RK3588S distribution, we use five untimed warm-up runs and the 70 held-out scenes. DoFP-LUT's P95 latency is 30.02 ms, only 2.1% above its median. The standard RK3588S profiling interfaces do not consistently isolate memory-access and arithmetic energy across the compared methods, so we report end-to-end energy without assigning component-wise percentages.

E Generic LUT Adaptation Details

Table 8 records the adaptation choices used for the generic LUT baselines. All methods use the same data split, CPFA sampling pattern, bilinear initialization, analyzer reconstruction target, and

Table 9: Results by data source. Entries are DoLP PSNR (dB) / AoP MAE (°).

Source	PDCNN	PIDSR	SRLUT	MuLUT	DnLUT	RCLUT	ShiftLUT	DoFP-LUT
Monno	33.68/11.36	35.26/8.42	30.01/15.30	29.13/16.36	29.88/15.32	31.26/14.26	30.68/14.79	31.98/13.34
Monno2	33.77/11.91	38.33/8.36	31.37/16.39	29.09/14.65	31.31/16.51	31.27/16.01	32.15/17.24	32.01/14.59
PIDSR	32.20/5.53	32.53/4.08	31.15/7.03	28.77/9.34	30.02/7.48	30.46/7.56	27.77/8.12	31.70/6.68
Qiu	33.48/3.33	36.84/3.16	31.02/4.42	29.69/5.44	30.99/4.39	31.33/4.63	29.74/4.40	32.34/3.70
Self-collected	28.67/6.33	29.73/5.71	26.80/8.53	26.32/10.28	26.24/8.91	26.98/8.68	26.51/9.14	27.47/7.72
Average	32.36/7.69	34.54/5.95	30.07/10.33	28.60/11.21	29.69/10.52	30.26/10.23	29.37/10.74	**31.10/9.21**

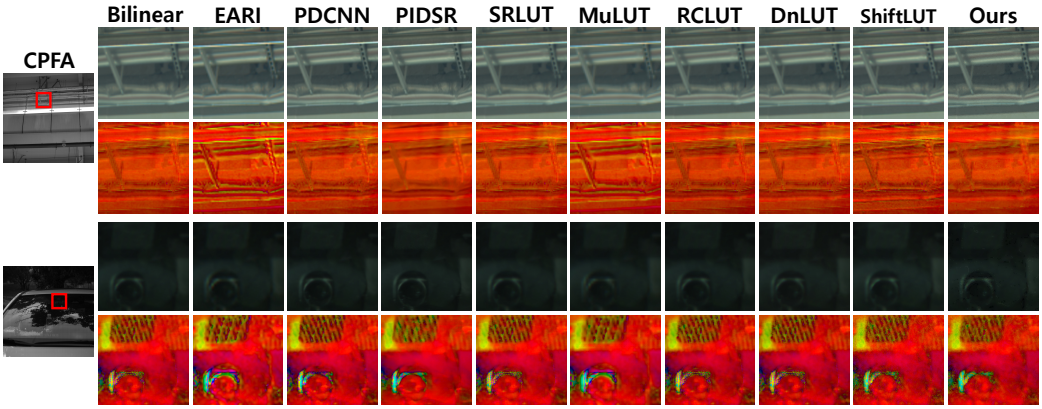

Figure 8: Real-camera demosaicking results on color DoFP captures. We show S_0 and HSV-based polarization visualization, where hue encodes AoP and value encodes DoLP. Since real captures do not provide ground truth, this figure is used for qualitative evaluation of real-camera generalization.

deployment platform as DoFP-LUT. We change only the input/output interface needed to predict the 12 analyzer channels and retain each method's defining lookup and aggregation principle. No rotation ensemble is used because rotating a CPFA patch changes its analyzer-phase semantics.

F Results by Data Source

Each source contributes 14 held-out scenes. Table 9 shows source-wise DoLP PSNR and AoP MAE. DoFP-LUT has the best aggregate DoLP/AoP result among LUT methods, although the source-wise variation also shows that no method dominates every source. All quantitative experiments use the same 4×4 CPFA pattern, so these results evaluate variation across data sources rather than across sensor layouts.

G Real-Camera Generalization on DoFP Captures

We further evaluate DoFP-LUT on real color DoFP camera captures. Unlike the synthetic CPFA experiments in the main paper, real captures do not provide full-resolution ground-truth analyzer images. Therefore, we do not report PSNR or SSIM on these data. Instead, we use real-camera results to examine whether the proposed lookup formulation can be directly applied to practical raw CPFA inputs and whether the reconstructed polarization cues remain visually stable.

For each real scene, we feed the raw CPFA measurement into the same DoFP-tailored bilinear initialization and demosaicking pipeline used in the main experiments. Fig. 8 shows representative results on S_0 and HSV-based polarization visualization, where hue encodes AoP and value encodes DoLP. Interpolation-based reconstruction tends to produce unstable polarization structures around edges and weakly polarized regions. Generic LUT adaptation reduces some local artifacts but still shows fragmented AoP patterns. In contrast, DoFP-LUT better preserves coherent polarization transitions and object boundaries, suggesting that the proposed analyzer/Stokes-structured correction transfers to real DoFP captures.

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: The main contributions and the scope of the proposed method have been elaborated in the abstract and introduction.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: The Discussion clarifies that DoFP-LUT is not intended to replace large neural demosaickers when maximum reconstruction quality is the only goal. It also states that the current design depends on a fixed teacher and fixed quantization, and suggests adaptive table construction, table compression, and hardware-specific scheduling as future directions.

3. **Theory assumptions and proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

   Answer: [N/A]

   Justification: The paper does not present formal theoretical theorems or proofs. The mathematical content consists of problem formulation, Stokes relations, lookup indices, and LUT construction equations.

4. **Experimental result reproducibility**

   Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

   Answer: [Yes]

   Justification: The paper specifies the datasets, scene-level split strategy, CPFA sampling pattern, reconstruction targets, metrics, compared method groups, LUT construction rule, quantization interval, nearest-neighbor lookup, empty-bin handling, and deployment evaluation protocol. These details provide a reasonable path to reproduce the main algorithmic and experimental claims.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

   Answer: [Yes]

   Justification: We have included the URL to the data and code of this paper in the abstract.

6. **Experimental setting/details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer) necessary to understand the results?

   Answer: [Yes]

   Justification: The Experimental Setup describes the public and self-collected datasets, scene-level train/validation/test organization, the 70 held-out test scenes, the use of the training split to train the PDCNN teacher and compile LUTs, and the validation split for selecting implementation hyperparameters such as the quantization interval.

7. **Experiment statistical significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [Yes]

Justification: The paper reports PSNR and SSIM for analyzer, S_0, and DoLP reconstruction, and angular MAE for AoP, which are sufficient to demonstrate the statistical significance of the experiments.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: The paper reports deployment runtime, deployed storage, and energy cost on RK3588S edge platform with four Arm Cortex-A76 Cores at the specified input resolution.

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

   Justification: The research in the paper conform with the NeurIPS Code of Ethics in every respect.

10. **Broader impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [N/A]

    Justification: There is no societal impact of this work.

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pre-trained language models, image generators, or scraped datasets)?

    Answer: [N/A]

    Justification: This paper poses no risks for misuse.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [Yes]

    Justification: The paper cites the public datasets, baselines, and prior methods used in the experiments. The authors use these assets for academic evaluation and follow the corresponding usage terms to the best of their knowledge.

13. **New assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [N/A]

    Justification: The paper uses a self-collected dataset for evaluation but does not release it as a new public asset at submission time. Therefore, no new released asset requires separate documentation.

14. **Crowdsourcing and research with human subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [N/A]

    Justification: The paper does not involve crowdsourcing experiments or research with human subjects.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [N/A]

    Justification: The paper does not involve human subjects, crowdsourcing participants, or personal data collection, so IRB approval is not applicable.

16. **Declaration of LLM usage**

    Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does *not* impact the core methodology, scientific rigor, or originality of the research, declaration is not required.

    Answer: [N/A]

    Justification: The core method does not use LLMs as an algorithmic component. Any use of language tools, if any, is limited to writing, editing, or formatting and does not affect the scientific method, experiments, or originality of the research.