Exposure-slot: Exposure-centric representations learning with Slot-in-Slot Attention for Region-aware Exposure Correction

Anonymous CVPR submission

Paper ID 10942

Abstract

Image exposure correction enhances images captured under diverse real-world conditions by addressing issues of under- and over-exposure, which can result in the loss of critical details and hinder content recognition. While significant advancements have been made, current methods often fail to achieve optimal feature learning for effective correction. To overcome these challenges, we propose Exposure-slot, a novel framework that integrates a prompt-based slot-in-slot attention mechanism to cluster exposed feature regions and learn exposure-centric features for each cluster. By extending the Slot Attention algorithm with a hierarchical structure, our approach progressively clusters features, enabling precise and region-aware correction. In particular, learnable prompts tailored to exposure characteristics of slots further enhance feature quality, adapting dynamically to varying conditions. Our method delivers superior performance on benchmark datasets, surpassing the current state-of-the-art with a PSNR improvement of over 1.85 dB on the SICE dataset and 0.4 dB on the LCDP dataset, thereby establishing a new benchmark for multi-exposure correction. The source code will be available upon publication.

1. Introduction

Image exposure correction enhances images captured under diverse real-world conditions. Under- or over-exposed images can appear excessively dark or bright, losing essential details and compromising accurate recognition. Despite advancements in imaging technology, robust automatic exposure correction remains a significant challenge.

Due to its critical importance, image exposure correction has been the focus of considerable research. Early efforts primarily treated under-exposure and over-exposure as separate issues [9, 35, 38]. More recent advancements, however, have introduced multi-exposure correction models capable of handling a wide range of exposure levels within a unified framework. For instance, MSEC [1] introduced an approach that considers both color and detail enhancement while performing multi-exposure correction in an end-to-end manner, and provided a dataset of images with different exposure errors for both training and evaluation. In addition, LCDPNet [33] provided another dataset of diverse scenes, covering both over-exposure and under-exposure, and incorporates retinex theory to account for local color distributions within images. Recent approaches increasingly focus on the separate processing of features based on their physical characteristics, such as exposure, frequency, contrast, color, and details. FECNet [12] uses Fourier transform to facilitate interactions between local spatial feature information and global frequency information, while ECLNet [13] uses a bilateral activation mechanism to process over- and under-exposed regions independently. In addition, DA [37] introduces a decoupling and aggregation scheme that separately enhances image details

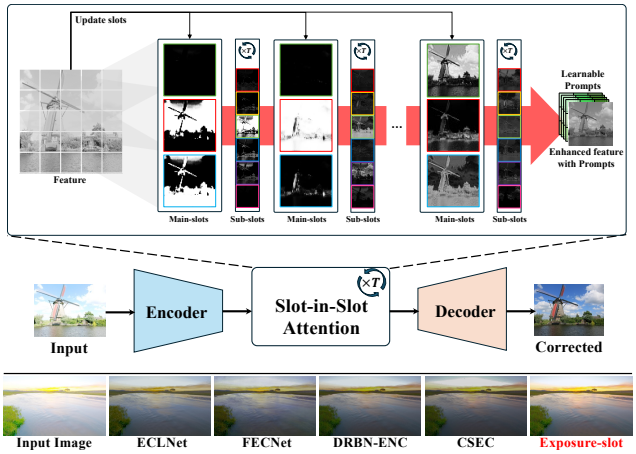

Figure 1. **(Top)** The Slot-in-Slot Attention mechanism hierarchically partitions exposure-aware regions using the attention maps of main and sub-slot attention, performing corrections with learnable prompts based on exposure levels. **(Bottom)** Comparison of the Exposure-slot with existing approaches. From left to right: ECLNet [13], FECNet [12], DRBN-ENC [11], CSEC [20] and our proposed approach, Exposure-slot.

and contrast.

However, relying on separating image features based on these physical properties may not ensure the optimal features for effective exposure correction. For example, at the bottom of Fig. 1, we visualize this limitation, showing that previous works [12–14, 20] often generate inappropriate artifacts in sky regions and struggle to accurately adjust sunset exposure levels.

To overcome these challenges, we propose a novel prompt-based slot-in-slot attention mechanism that clusters regions with similar exposure levels, enabling exposure-centric feature learning within each cluster. We utilize Slot Attention [22], designed to cluster features based on object representations, to identify distinct exposure regions within an image. In our approach, slots represent features grouped by exposure levels, and the corresponding attention maps act as exposure-aware region maps, guiding the grouping of features according to exposure characteristics. Therefore, these maps highlight regions based on exposure levels, enabling targeted exposure correction for each area.

To support effective correction, we introduce learnable prompt vectors that operate based on exposure-aware region maps, which are attention maps of each slot. These prompts adapt to the characteristics of each region defined by the attention map, helping the model learn targeted correction to improve under- or over-exposed areas. Specifically, we introduce Slot-Prompt Interaction Module (SPIM), which combines slot and prompt information using cross-attention. The cross-attention component of SPIM uses the prompts learned through Slot-in-Slot Attention as conditioning factors, enhancing the interaction between features and prompts.

Our approach builds upon standard slot attention with a hierarchical design, called Slot-in-Slot Attention, which processes slots in levels. This mechanism operates across multiple structure levels: the first level coarsely partitions regions based on distinct exposure characteristics, while subsequent levels progressively refine these regions with more partitions. Each slot captures exposure-specific regional information, distinguishing areas with varying exposure levels to enhance feature separation and improve the precision of local details.

The overall framework of the proposed approach is illustrated at the top of Fig. 1. The Slot-in-Slot Attention is applied to intermediate features between the encoder and decoder, generating attention maps for slots that are hierarchically partitioned according to exposure characteristics, as depicted in Fig. 1. The generated attention maps are then used as weight maps for learnable prompts, resulting in refined intermediate features that are subsequently fed into the decoder. We refer to our model as Exposure-Slot, as it is the first model to use Slot Attention for unsupervised feature partitioning based on exposure levels in exposure correction, and the first to apply region-aware prompts for feature enhancement. Our main contributions are as follows:

- Exposure-slot is the first approach to leverage Slot Attention mechanism for optimized exposure-specific feature partitioning.
- We introduce the slot-in-slot attention that enables sophisticated feature partitioning and learning.
- We apply exposure-aware prompts that enhance the exposure-centric characteristics of each image feature.
- Exposure-slot achieves state-of-the-art results on multi-exposure benchmark datasets [1, 5, 33], setting a new standard in this field.

2. Related Work

2.1. Exposure Correction

With the rise of deep neural networks (DNN), exposure correction has seen the emergence of a range of DNN-based methods leveraging diverse concepts. For under-exposed image enhancement, methods [34, 38, 39, 41, 42] inspired by retinex theory have been proposed. Retinex-based models like CMEC [25] have also incorporated attention mechanisms to address multi-exposure correction, while LCDP-Net [33] introduced a novel focus on local color distribution. To support multi-exposure correction tasks, dedicated datasets such as MSEC [1] and SICE [5] were developed for training and evaluation. MSEC further introduced a Laplacian pyramid architecture to handle varying exposure levels. ENC [11] contributed an exposure normalization module that refined feature maps by transforming diverse exposure features into an exposure-invariant feature space. CSNorm [40] also enhanced model generalization by selectively normalizing lightness-sensitive channels and ERL [14] proposed exposure relationship learning through regularization techniques for multi-exposure correction.

There are several recent studies that emphasize the feature separation based on their physical characteristics. ECLNet [13] used a bilateral activation mechanism to adjust processing across different exposure conditions, while FECNet [12] proposed a lightweight model that utilizes spatial-frequency interactions with a Fourier-based approach. DA [37] proposed a contrast- and detail-aware module that integrates with existing architectures. The latest advancement, CSEC [20], addresses color distribution shifts by defining darkened and brightened feature maps.

2.2. Slot Attention

Slot Attention [22] is a mechanism for clustering features corresponding to slots which is the main object of object-centric learning (OCL). Specifically, this mechanism iteratively employs dot-product attention [23] with attention coefficients to update slots, which function as queries within the attention process. Through multiple rounds of recurrent

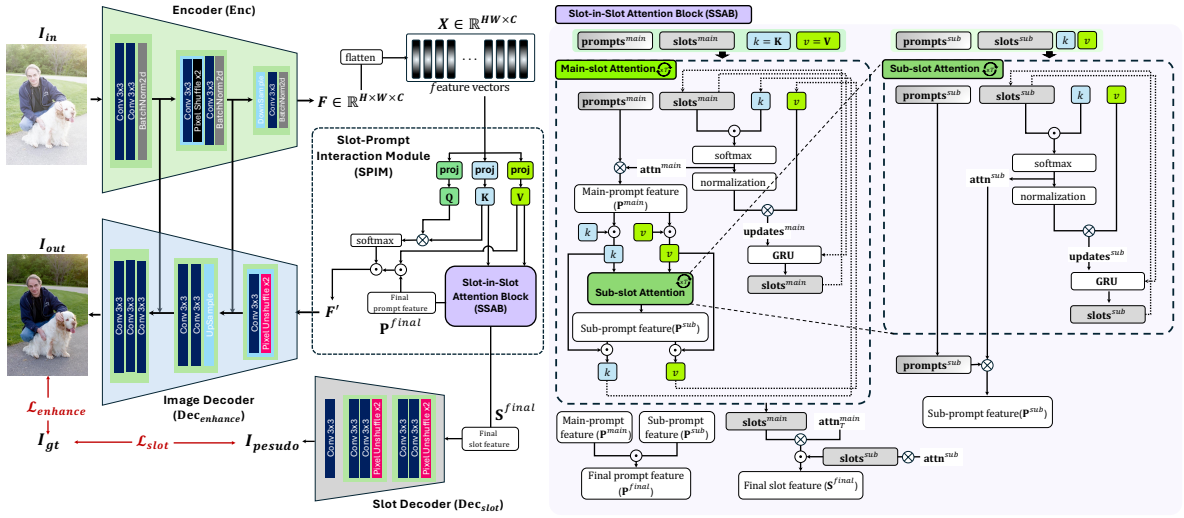

Figure 2. **Overview of Exposure-Slot**: Exposure-Slot operates within a U-shaped encoder-decoder network, with the Slot-Prompt Interaction Module (SPIM) bridging the encoder and decoder stages. In SPIM, Slot-in-Slot Attention Block (SSAB) adaptively partitions regions and generates prompt features to enhance the feature representation. In addition, a Slot Decoder ($\mathbf{Dec}_{slot}$) is employed during training to reconstruct slot features to the target. This ensures that slot attention maps capture exposure-centric information, helping prompts learn the relevant information for enhancement within each corresponding slot.

attention, Slot Attention achieves effective clustering and feature separation in OCL without the need for annotations.

Slot Attention has been effectively applied to image classification, as seen in SCOUTER [19, 31], which uses a Slot-Attention-based classifier for explainable recognition. In segmentation, Slot Attention has proven particularly valuable for video segmentation [3, 7, 17, 27]. For example, GSANet [18] utilizes Slot Attention to separate center objects from background elements across video frames.

In reconstruction and restoration tasks, Slot-VAE [36] integrates Slot Attention with a hierarchical Variational Autoencoder (VAE) framework, enhancing object-centric scene generation. Moreover, AID [16] employs Slot Attention to capture implicit representations of illuminant chromaticities, with each slot vector capturing the characteristics of a specific illuminant. This enables AID to generate accurate chromaticity and weight maps for each light source. In this paper, we leverage Slot Attention for partitioning in unsupervised way for the exposure correction. This allows our method to train each slot to capture feature-relevant exposure representations and effectively to cluster features implicitly for exposure correction without requiring annotations.

2.3. Prompt-Based Learning

The prompt-based learning method was first introduced by [4], proposing the concept of optimizing input prompts. This approach gained popularity in the field of natural language processing, inspiring numerous follow-up studies. CoOp [43] introduced a novel approach by treating prompt contexts as learnable parameters, moving beyond methods limited to fixed-format prompts, and demonstrating that learnable prompts can outperform handcrafted ones. On the basis of CoOp, several methods have been developed for dynamically generating suitable prompts [6, 10, 30]. For example, CODA [30] generates prompts tailored to specific inputs, while HyperPrompt [10] addresses multi-task learning through the use of a prompt generator.

In the computer vision field, visual prompts involve adding trainable parameters to modify inputs for efficient model adaptation. VPT [15] demonstrated substantial performance improvements over conventional fine-tuning methods by applying visual prompts to a fixed transformer backbone. Furthermore, [2] presented visual prompts compatible with input images, specifically optimized for CLIP. For low-level vision tasks, methods like PromptIR [26] and PromptRestorer [32] leverage prompts to encode degradation-specific information, guiding restoration networks to adapt to various types and intensities of degradation. In our method, we introduce exposure-specific prompts that guide feature separation and generate region-aware features specifically tailored to distinct exposure characteristics.

3. Proposed Method

3.1. Overall Flow

As illustrated in Fig. 2, the proposed Exposure-slot architecture is a U-shaped residual network composed of an encoder-decoder structure connected via skip connections

Algorithm 1 Slot-in-Slot Attention Block (SSAB). For clarity, we represent the algorithm using a 2-level hierarchical structure, consisting of main- and sub-slots.

Input: $\mathbf{K}, \mathbf{V}, \mathbf{slots}^{main}, \mathbf{slots}^{sub}, \mathbf{prompts}^{main}, \mathbf{prompts}^{sub}$

function SLOT_UPDATE($\mathbf{slots}, k, v$)
  $\mathbf{slots}_{prev} = \mathbf{slots}$
  $\mathbf{slots} = \text{LayerNorm}(\mathbf{slots})$
  $q = \mathbf{to\_q}(\mathbf{slots})$ ▷ $\mathbf{to\_q}$: linear projection layer
  $\mathbf{attn} = \text{Softmax}\left(\frac{1}{\sqrt{D}}\, k \cdot q^T\right)$ ▷ D : hyper parameter scales matrix multiplication
  $\mathbf{updates} = \text{WeightedMean}(\text{weights}=\mathbf{attn} + \epsilon, \text{values} = v)$
  $\mathbf{slots} = \text{GRU}(\text{state}=\mathbf{slots}_{prev}, \text{inputs}=\mathbf{updates})$
  return $\mathbf{slots}, \mathbf{attn}$
end function

$k = \mathbf{K}, v = \mathbf{V}$
for $t_{main} = 0 \ldots T$ **do** ▷ Main-slot Attention
  $\mathbf{slots}^{main}, \mathbf{attn}^{main} = \text{SLOT\_UPDATE}(\mathbf{slots}^{main}, k, v)$
  $\mathbf{P}^{main} = \mathbf{prompts}^{main} \times \mathbf{attn}^{main}$
  $k, v = k \cdot \mathbf{P}^{main}, v \cdot \mathbf{P}^{main}$ ▷ Update key & value for Sub-slot Attention
  for $t_{sub} = 0 \ldots T$ **do** ▷ Start: Sub-slot Attention
    $\mathbf{slots}^{sub}, \mathbf{attn}^{sub} = \text{SLOT\_UPDATE}(\mathbf{slots}^{sub}, k, v)$
  end for ▷ End: Sub-slot Attention
  $\mathbf{P}^{sub} = \mathbf{prompts}^{sub} \times \mathbf{attn}^{sub}$
  $k, v = k \cdot \mathbf{P}^{sub}, v \cdot \mathbf{P}^{sub}$ ▷ Update key & value for next iteration of Main-slot Attention
end for

$\mathbf{P}^{final} = \mathbf{P}^{main} \cdot \mathbf{P}^{sub}$
$\mathbf{S}^{final} = (\mathbf{slots}^{main} \times \mathbf{attn}^{main}) \cdot (\mathbf{slots}^{sub} \times \mathbf{attn}^{sub})$
return $\mathbf{P}^{final}, \mathbf{S}^{final}$

and Slot-Prompt Interaction Module (SPIM).

First, the encoder processes the poorly exposed input image I_{in} to produce a latent feature representation $F \in \mathbb{R}^{H \times W \times C}$ as $F = \mathbf{Enc}(I_{in})$. Then, SPIM, positioned between the encoder and decoder, refines the encoder output F using input-specific, region-aware prompts guided by the Slot-in-Slot Attention mechanism. The refined latent feature F' is subsequently passed to the image decoder ($\mathbf{Dec}_{enhance}$), which processes it to produce the output image I_{out} with well-enhanced exposure.

3.2. Slot-Prompt Interaction Module (SPIM)

SPIM is designed as a combination of the Slot-in-Slot Attention Block (SSAB) and a subsequent cross-attention process. SPIM initially projects F into query, key, and value representations, while SSAB receives the key and value to estimate the corresponding prompt features of F. Specifically, SSAB hierarchically applies slot attention to achieve coarse-to-fine region separation, which supports effective exposure correction. Then the derived prompt feature serves as conditioning elements in the cross-attention step involving the query, key, and value representations.

Specifically, as illustrated in Fig. 2, the encoded feature $F \in \mathbb{R}^{H \times W \times C}$ is first flattened into $X \in \mathbb{R}^{HW \times C}$. Subsequently, X is projected into query $\mathbf{Q} \in \mathbb{R}^{HW \times C}$, key $\mathbf{K} \in \mathbb{R}^{HW \times C}$, and value $\mathbf{V} \in \mathbb{R}^{HW \times C}$ through the linear projections $W_{\mathbf{Q}}$, $W_{\mathbf{K}}$, and $W_{\mathbf{V}}$ as follows:

$$\mathbf{Q} = XW_{\mathbf{Q}}^T, \quad \mathbf{K} = XW_{\mathbf{K}}^T, \quad \mathbf{V} = XW_{\mathbf{V}}^T. \tag{1}$$

Within our SSAB, $\mathbf{K}$ and $\mathbf{V}$ are used to generate the prompt feature $\mathbf{P}^{final}$, which serves as a condition for the subsequent cross-attention process to produce the refined feature map $F' \in \mathbb{R}^{H \times W \times C}$ as follows:

$$F' = reshape(softmax(\frac{\mathbf{Q}\mathbf{K}^T}{\sqrt{d}}) \cdot (\mathbf{V} \cdot \mathbf{P}^{final})), \tag{2}$$

where d denotes a learnable parameter that adaptively scales matrix multiplication.

3.2.1. Slot-in-Slot Attention Block (SSAB)

In the proposed Slot-in-Slot Attention Block within SPIM, we generate a refined feature map by integrating a hierarchically structured slot mechanism, which partitions regions with different exposure levels in a hierarchical and iterative manner, with learned prompts specifically tailored to distinct exposure characteristics.

For instance, in a 2-level slot-in-slot structure, the first level partitions regions with similar exposure characteristics using a relatively smaller number of slots. Building on

these results, the second level employs more slots to cluster the regions more finely. By iteratively repeating this process, the sub-slots gradually learn finer details of differently exposed regions. Notably, the hierarchical structure can extend beyond two levels, enabling the identification of exposure-centric feature maps with increasing precision. Next, we introduce exposure-specific prompts that enhance the image by targeting distinct regional exposure characteristics, multiplied by attention maps from the slot attention mechanism. Since the attention maps identify regions with different exposure levels, multiplying them with exposure-specific prompts generates region-aware guidance, enabling balanced exposure correction.

In Alg. 1, the pseudo-code for this 2-level slot-in-slot structure is provided, refines key (k) and value (v) through two nested loops: an outer loop and an inner loop. The outer loop updates the main-slots, which are then combined with prompts to refine k and v. In the inner loop, the refined k and v are used to update the sub-slots, which are further combined with prompts to refine k and v with fine details. This process is repeated until the outer loop is completed. Notably, to update the slots, we employ the SLOT_UPDATE function introduced process in [22]. The key difference is that each loop iteration uses the updated k and v as inputs.

Main-slot Attention In the outer loop of Alg. 1, we perform Main-slot Attention to update the main-slots ($\mathbf{slots}^{main}$), which represent the first level of the hierarchical slot structure. Notably, $\mathbf{slots}^{main} \in \mathbb{R}^{K^{main} \times D^{main}_{slot}}$, where K^{main}, and D^{main}_{slot} indicate the number of slots and the dimensionality of each slot, respectively, can be initialized with learnable parameters.

First, given $\mathbf{K}$ and $\mathbf{V}$ from the Encoder, assigned to k and v, respectively, we update the slots using the SLOT_UPDATE function as follows:

$$\mathbf{slots}^{main}, \mathbf{attn}^{main} = \mathbf{SLOT\_UPDATE}(\mathbf{slots}^{main}, k, v), \tag{3}$$

where $\mathbf{attn}^{main}$ represents the attended region of each $\mathbf{slots}^{main}$. Then, the attention map $\mathbf{attn}^{main}$ is used to produce prompt feature $\mathbf{P}^{main}$ as:

$$\mathbf{P}^{main} = \mathbf{prompts}^{main} \times \mathbf{attn}^{main}, \tag{4}$$

where $\mathbf{prompts}^{main} \in \mathbb{R}^{K^{main} \times D^{main}_{slot}}$ denotes learnable prompts tailored to each exposure characteristics. Thus, by multiplying the exposure-specific prompts with the attention map containing distinct exposure region information, our prompt feature $\mathbf{P}^{main}$ effectively encapsulates exposure-centric and region-aware information. Finally, prompt feature $\mathbf{P}^{main}$ is used to update the key and value as:

$$k, v = k \cdot \mathbf{P}^{main}, v \cdot \mathbf{P}^{main}, \tag{5}$$

and the resulting k and v are used as inputs for the subsequent inner loop process, referred to as Sub-slot Attention.

Sub-slot Attention Within the inner loop of Alg. 1, we perform Sub-slot Attention to update the sub-slots ($\mathbf{slots}^{sub}$), which represent the second level of hierarchical structure. Particularly, $\mathbf{slots}^{sub} \in \mathbb{R}^{K^{sub} \times D^{sub}_{slot}}$, where $\mathbf{slots}^{sub}$ are also initialized using learnable parameters. Here, K^{sub} denotes the number of sub-slots, D^{sub}_{slot} denotes their dimensionality, and D^{main}_{slot} is configured to ensure finer feature separation.

First, the updated k and v from the Main-slot Attention in Eq. 5 are used to update $\mathbf{slots}^{sub}$ via the SLOT_UPDATE function. Similarly to Eq. 3, Sub-slot Attention is updated as follows:

$$\mathbf{slots}^{sub}, \mathbf{attn}^{sub} = \mathbf{SLOT\_UPDATE}(\mathbf{slots}^{sub}, k, v), \tag{6}$$

where $\mathbf{attn}^{sub}$ represents the attended region of each $\mathbf{slots}^{sub}$, and this update process is repeated for T iterations. Subsequently, $\mathbf{attn}^{sub}$ is used to obtain sub-prompt feature $\mathbf{P}^{sub}$ from the Sub-slot Attention as:

$$\mathbf{P}^{sub} = \mathbf{prompts}^{sub} \times \mathbf{attn}^{sub}, \tag{7}$$

where $\mathbf{prompts}^{sub} \in \mathbb{R}^{K^{sub} \times D^{sub}_{slot}}$ is also learnable parameters tailored to each exposure characteristic in Sub-slot Attention. Finally, by multiplying $\mathbf{P}^{sub}$ with k and v, we transfer information from the Sub-slot Attention for the next iteration of the Main-slot Attention as:

$$k, v = k \cdot \mathbf{P}^{sub}, v \cdot \mathbf{P}^{sub}. \tag{8}$$

To summarize, the result of the Main-slot Attention is integrated into the Sub-slot Attention, and the output from the Sub-slot Attention is iteratively fed back into the Main-slot Attention loop. This complementary updating process is repeated T times. At the end of SSAB, the final outputs $\mathbf{P}^{final}$ and $\mathbf{S}^{final}$ are obtained as follows:

$$\begin{aligned} \mathbf{P}^{final} &= \mathbf{P}^{main} \cdot \mathbf{P}^{sub}, \\ \mathbf{S}^{final} &= (\mathbf{slots}^{main} \times \mathbf{attn}^{main}) \cdot (\mathbf{slots}^{sub} \times \mathbf{attn}^{sub}). \end{aligned} \tag{9}$$

Note that $\mathbf{P}^{final}$ is integrated with $\mathbf{Q}$, $\mathbf{K}$, and $\mathbf{V}$ as described in Eq. 2, and is used to decode the final output. Meanwhile $\mathbf{S}^{final}$ serves as an input to a separate decoder to facilitate slot training.

3.3. Decoder Process

The refined feature F' and slots S^{final} from SPIM are used as inputs for the final decoding procedures. Specifically, our Exposure-slot uses the enhanced feature F' to predict an enhanced image through the decoder, as follows:

$$I_{out} = \mathbf{Dec}_{enhance}(F'), \tag{10}$$

where $\mathbf{Dec}_{enhance}$ is image decoder and I_{out} is the final exposure-enhanced output.

Model	#Params (M)	SICE [5] Under	SICE [5] Over	SICE [5] Avg.	MSEC [1] Under	MSEC [1] Over	MSEC [1] Avg.	LCDP [33] Avg.
CLAHE [44]	-	12.69 / 0.5037	10.21 / 0.4878	11.45 / 0.4942	16.77 / 0.6211	14.45 / 0.5842	15.38 / 0.5990	16.33 / 0.6420
RetinexNet [38]	0.840	12.94 / 0.5171	12.87 / 0.5252	12.90 / 0.5212	12.13 / 0.6209	10.47 / 0.5953	11.14 / 0.6048	19.25 / 0.7041
ZeroDCE [8]	0.079	16.92 / 0.6330	7.11 / 0.4292	12.02 / 0.5311	14.55 / 0.5887	10.40 / 0.5142	12.06 / 0.5441	12.59 / 0.6530
RUAS [21]	0.002	16.63 / 0.5589	4.54 / 0.3196	10.59 / 0.4393	13.43 / 0.6807	6.39 / 0.4655	9.20 / 0.5515	13.76 / 0.6060
SCI [24]	0.001	17.86 / 0.6401	4.45 / 0.3629	12.49 / 0.5051	9.97 / 0.6681	5.84 / 0.5190	7.49 / 0.5786	11.87 / 0.5234
MSEC [1]	7.040	19.62 / 0.6512	17.59 / 0.6560	18.58 / 0.6536	20.52 / 0.8129	19.79 / 0.8156	20.08 / 0.8210	20.38 / 0.7800
LCDPNet [33]	0.960	17.45 / 0.5622	17.04 / 0.6463	17.25 / 0.6043	22.35 / 0.8650	22.17 / 0.8476	22.30 / 0.8552	23.24 / 0.8420
ECLNet [13]	0.018	22.05 / 0.6893	19.25 / 0.6872	20.65 / 0.6861	22.37 / 0.8566	22.70 / 0.8673	22.57 / 0.8631	22.44 / 0.8061
FECNet [12]	0.150	22.01 / 0.6737	19.91 / 0.6961	20.96 / 0.6849	22.96 / 0.8598	23.22 / 0.8748	23.12 / 0.8688	22.41 / 0.8402
DRBN-ENC [11]	0.580	21.89 / 0.7071	19.09 / 0.7229	20.49 / 0.7150	22.72 / 0.8590	22.11 / 0.8521	22.35 / 0.8530	22.09 / 0.8271
MSEC+DA [37]	7.040	20.94 / **0.7546**	17.49 / 0.6640	19.22 / 0.7093	21.53 / 0.8590	21.55 / 0.8750	21.54 / 0.8670	21.05 / 0.8119
ECLNet+ERL [14]	0.018	22.14 / 0.6908	19.47 / 0.6982	20.81 / 0.6945	22.90 / 0.8624	22.58 / 0.8676	22.70 / 0.8655	22.63 / 0.8096
PromptIR [26]	34.164	22.51 / 0.6955	19.29 / 0.6849	20.90 / 0.6902	15.80 / 0.7391	16.73 / 0.7852	16.36 / 0.7668	23.49 / 0.8513
CSEC [20]	1.364	20.79 / 0.7031	20.02 / 0.7093	20.41 / 0.7062	22.18 / 0.8502	22.69 / 0.8662	22.73 / 0.8638	23.63 / 0.8550
Exposure-slot	1.229	**23.85** / 0.7092	**21.77** / **0.7375**	**22.81** / **0.7239**	**23.09** / 0.8601	**23.24** / **0.8762**	**23.18** / **0.8697**	**24.03** / **0.8592**

Table 1. Quantitative results on SICE [5], MSEC [1], and LCDP [33] in terms of PSNR↑/SSIM↑. The best score is displayed in **Red**, the second in Blue. Additionally, the number of parameters (#Params) required for inference is also specified. Exposure-slot is lightweight compared to the previous SOTA approach [20] while consistently outperforming it across all datasets on average.

Meanwhile, $\mathbf{S}^{final}$ is decoded into a pseudo-corrected image using the slot reconstruction decoder:

$$I_{pseudo} = \mathbf{Dec}_{slot}(\mathbf{S}^{final}), \tag{11}$$

where $\mathbf{Dec}_{slot}$ represents the slot reconstruction decoder and I_{pseudo} is the resulting pseudo-corrected RGB image. This training strategy encourages each slot to focus on exposure-centric learning, similar to the set prediction approach used in [22]. It is important to note that the slot reconstruction decoder is used only during training and is not required during inference.

3.4. Loss functions

Exposure-slot is trained using two loss functions: image enhancement loss $\mathcal{L}_{enhance}$ and slot reconstruction loss $\mathcal{L}_{slot}$. The overall training objective is to minimize the combined loss, promoting both accurate image enhancement and effective slot-based reconstruction.

Image enhancement loss. The image enhancement loss $\mathcal{L}_{enhance}$ is defined using the L_1 distance between the exposure enhanced image I_{out} and the ground truth image I_{gt} as follows:

$$\mathcal{L}_{enhance} = ||I_{out} - I_{gt}||_1. \tag{12}$$

Slot Reconstruction loss. The slot reconstruction loss $\mathcal{L}_{slot}$ is similarly defined using the L_1 distance, but between the pseudo-corrected image I_{pseudo}, generated by the slot reconstruction decoder, and the ground truth image I_{gt}, as follows:

$$\mathcal{L}_{slot} = ||I_{pseudo} - I_{gt}||_1. \tag{13}$$

The final objective function $\mathcal{L}_{final}$ is defined as the sum of the two losses:

$$\mathcal{L}_{final} = \mathcal{L}_{enhance} + \mathcal{L}_{slot}. \tag{14}$$

4. Experiments

4.1. Experimental Setup

Implementation Details. We trained our model using the Adam optimizer with $\beta_1 = 0.9$ and $\beta_2 = 0.999$ using a patch size of 256×256 and a batch size of 16. The learning rate was set to 2×10^{-4}, and training was conducted for 500 epochs. Additionally, the parameters K^{main} and K^{sub} were set to 3 and 7, respectively. Our code will be made publicly available upon acceptance.

Datasets and Comparative methods. Our training and benchmarking settings align with established standards for existing exposure correction tasks [12, 13, 20, 33]. We train our network on three multi-exposure datasets: Single Image Contrast Enhancement (SICE) [5], Multiple Exposure (ME) [1], and LCDP [33].

We compare our Exposure-slot to existing state-of-the-art exposure correction methods, including LCPDNet [33], ERL [14], ENC [11], DA [37], ECLNet [13], FECNet [12], and CSEC [20]. In PromptIR [26], the number of prompts is set to match the number of exposure values in each dataset: 2 for SICE, 5 for MSEC, and 2 for LCDP. Evaluations are conducted using Peak Signal-to-Noise Ratio (PSNR) and Structural Similarity (SSIM) metrics.

4.2. Performance Evaluation

Table 1 presents the performance of our method on three representative multi-exposure datasets: SICE [5], MSEC [1], and LCDP [33]. On the SICE dataset, our approach achieves the highest performance overall, except for one SSIM value in the under-exposed condition, where it ranks second. Similarly, for the MSEC dataset, Exposure-slot consistently outperforms previous methods

CVPR #10942

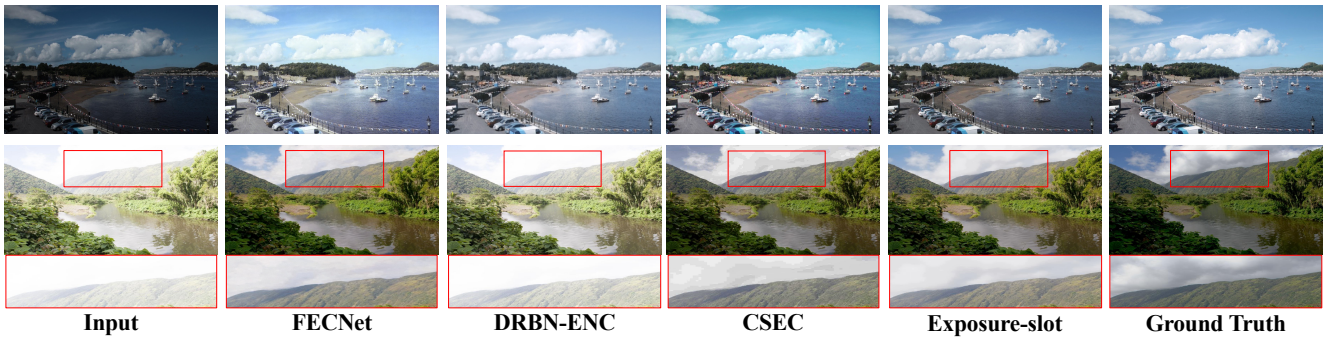

Figure 3. Qualitative comparison (FECNet [12], ENC [11], CSEC [20]) on the SICE [5] and MSEC [1] dataset. Examples of images enhanced from under-exposed condition **(Top)** and images enhanced from over-exposed condition **(Bottom)**.

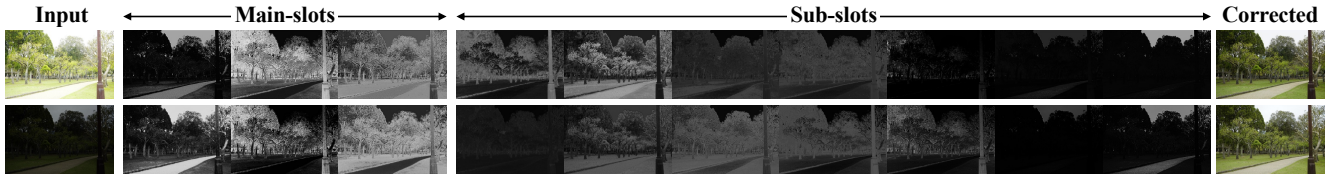

Figure 4. Visualization of slot attention maps (**attn**) of main- and sub-slot. It demonstrates how exposure-centric characteristics are effectively partitioned and refined through the slot attention mechanisms, transforming the input image into the corrected output.

Model	$\Delta E_{2000}\downarrow$ [29]	$\Delta E_{ab}\downarrow$ [28]
ZeroDCE [8]	23.78	29.31
RUAS [21]	29.59	37.05
SCI [24]	25.64	31.22
ECLNet [13]	9.15	11.58
FECNet [12]	8.85	11.19
DRBN-ENC [11]	8.68	11.02
PromptIR [26]	9.27	11.59
CSEC [20]	8.72	11.39
Exposure-slot	**6.94**	**8.89**

Table 2. Comparisons with color difference metrics, $\Delta E_{2000}\downarrow$ and $\Delta E_{ab}\downarrow$, on the SICE [5] dataset. The highest score is highlighted in **Red**, while the second-highest is marked in Blue.

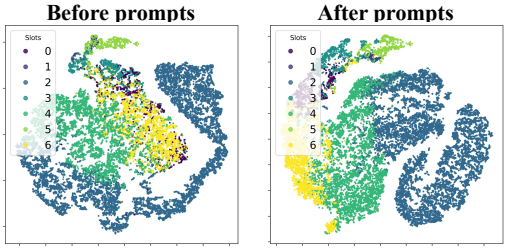

Figure 5. t-SNE results of features before and after prompts. Features from the same sub-slot are represented by the same color. With prompts, the features exhibit clearer separation.

in PSNR and SSIM, achieving top scores in all cases except for under-exposure in SSIM. On the LCDP dataset, our method also demonstrates superior performance, surpassing CSEC [20]. Compared to previous state-of-the-art methods [11, 12], our approach shows a notable gain of 1.85 dB in PSNR on the SICE dataset and 0.4 dB on the LCDP dataset compared to CSEC [20], highlighting its strong performance advantage.

Fig.3 shows qualitative comparisons on the SICE [5] and MSEC [1] datasets, demonstrating the performance of our model against other exposure correction methods. Under under-exposed conditions, Exposure-slot achieves superior color fidelity and detail recovery, closely matching the ground truth without excessive brightening or color distortion, unlike other models [12, 20]. In over-exposed areas, the zoomed-in views reveal that other models, particularly CSEC [20], produce artifacts and color inconsistencies due to over-correction. In contrast, Exposure-slot maintains stable color restoration and texture enhancement, preserving natural details and color harmony. These results highlight Exposure-slot's capability to handle diverse exposure conditions, achieving state-of-the-art performance in complex exposure correction tasks.

Additionally, to evaluate color correction performance, we conduct a comparison using ΔE_{2000} [29] and ΔE_{ab} [28] metrics in the LAB color space. Table 2 presents the results, demonstrating that our approach also excels in color correction. Exposure-slot demonstrates a performance improvement of 1.78 in ΔE_{2000} and 2.13 in ΔE_{ab} compared to previous state-of-the-art methods.

Fig. 4 visualizes the attention maps for main and sub-slots predicted by SSAB. These results demonstrate that each map effectively generates prompts region maps, and Exposure-slot leverages this partitioning information to achieve robust enhancement across different exposure conditions of the same scene. In Fig. 5, the t-SNE visualization of $V \cdot P^{final}$ in Eq. 2, demonstrates that our prompts improves clustering and effectively facilitate feature separation. Notably, the improved feature identification within each slot suggests that the model effectively captures

Figure 6. The visualization of the ablation study. From the top left, the images correspond to (e), (f), (g), and (h) in the Table 3, with the previous SOTA method, CSEC [20], included for comparison.

Model	$\mathbf{Dec}_{slot}$	**SSAB** (2-*level*)	**Prompts**	PSNR↑/SSIM↑
(a)				21.92/0.7091
(b)	✓			22.02/0.7154
(c)		✓		21.76/0.6998
(d)			✓	22.09/0.7062
(e)	✓	✓		22.10/0.7065
(f)	✓		✓	22.09/0.7210
(g)		✓	✓	21.95/0.7149
(h)	✓	✓	✓	**22.81/0.7239**

Table 3. Ablation studies on proposed modules.

exposure-specific or region-based information. For further visualizations of attention maps and additional details on t-SNE, please refer to the supplementary material.

4.3. Ablation Study

Effectiveness of the Proposed Modules. This section evaluates the effectiveness of SSAB, $\mathbf{Dec}_{slot}$, and prompts through ablation studies. Table 3 shows results for each ablation case on the SICE [5] dataset. Specifically, configurations without **SSAB** (2-*level*) refer to an SSAB with a single level consisting of 7 slots (*i.e.* main-slots only). Configurations without **Prompts** directly use the slot feature $\mathbf{S}^{final}$ for cross-attention in Eq. 2.

First, a comparison between cases with and without $\mathbf{Dec}_{slot}$ shows consistent performance improvements when $\mathbf{Dec}_{slot}$ is included. This result suggests that training $\mathbf{Dec}_{slot}$ helps the slots more accurately identify and represent regions with distinct exposure characteristics, leading to enhanced performance. When employing SSAB (Model (e) and (h)) also shows significant performance improvement when used together with $\mathbf{Dec}_{slot}$. Regarding prompts, although Model (d) shows a slight decrease in SSIM compared to Model (a), all other cases utilizing prompts demonstrate improved performance. Notably, our full model (h) achieves the highest performance, underscoring the importance of each component in enhancing model performance. In Fig. 6, we provide a part of ablations as visual results within state-of-the-art method CSEC [20].

Investigation of Model Configuration. Table 4 presents ablation studies on the number of main-slots (K^{main}), sub-slots (K^{sub}), and iterations (T) within SSAB to evaluate their impact on performance. The results include PSNR and SSIM metrics on the SICE [5] dataset, with runtime measurements taken on a single 844×1500 RGB image using an NVIDIA RTX 4090 GPU. In this analysis, we vary K^{main}, K^{sub}, and T by incrementing or decrementing each parameter by 1. Based on the PSNR values, the configuration of $K^{main} = 3$, $K^{sub} = 7$, and $T = 3$ yields the highest performance and is therefore selected for our method.

Model	K^{main}	K^{sub}	T	PSNR↑	SSIM↑	Time (S)
(a)	3	7	3	**22.81**	0.7236	0.06759
(b)	**2**	7	3	22.61	0.7245	0.06759
(c)	**4**	7	3	22.36	0.7201	0.06759
(d)	3	**6**	3	21.92	0.7038	0.06759
(e)	3	**8**	3	22.78	0.7241	0.06759
(f)	3	7	**2**	21.15	0.7058	0.06747
(g)	3	7	**4**	21.87	0.7222	0.06996

Table 4. Investigation of the number of slots (K^{main}, K^{sub}) and number of iterations (T) on the SICE [5] dataset.

Model	K^{main}	$K^{sub\text{-}1}$	$K^{sub\text{-}2}$	PSNR↑	SSIM↑	ΔE_{2000}↓ [29]	ΔE_{ab}↓ [28]
1-level	3	-	-	22.02	0.7131	7.12	9.41
2-level	3	7	-	22.81	0.7236	6.94	8.89
3-level	3	7	10	23.06	0.7306	6.84	8.69

Table 5. Ablation studies on n-level SSAB ($n = 1, 2, 3$)

Effectiveness of Structural Levels. In Sec.3, we use a 2-level SSAB structure as the default configuration. To further investigate the potential of SSAB, we perform experiments using both 1-level and 3-level SSAB structures on the SICE [5] dataset, as summarized in Table 5. In the 3-level configuration, we set $K^{sub\text{-}2} = 10$ for the number of second sub-slots, resulting in PSNR and SSIM improvements of 0.25 and 0.07 compared to 2-level SSAB, respectively. These results indicate that adding more SSAB levels can enhance performance. While adding additional levels improves performance, it also leads to an exponential increase in time complexity. Therefore, considering this trade-off, we adopt the 2-level SSAB as the default structure.

5. Conclusion

In this paper, we present Exposure-slot, a novel framework for exposure correction that integrates slot-in-slot attention with learnable prompts to achieve precise, exposure-centric feature learning. By hierarchically clustering and refining exposure regions, Exposure-slot facilitates accurate, region-aware adjustments and achieves state-of-the-art performance on multiple multi-exposure benchmarks. Our approach demonstrates significant improvements in both quantitative and qualitative metrics, highlighting the effectiveness of structured attention mechanisms for challenging exposure correction scenarios. Furthermore, this work advances exposure correction and paves the way for exploring slot-based architectures in other low-level vision tasks.

References

[1] Mahmoud Afifi, Konstantinos G Derpanis, Bjorn Ommer, and Michael S Brown. Learning multi-scale photo exposure correction. In *CVPR*, 2021. 1, 2, 6, 7

[2] Hyojin Bahng, Ali Jahanian, Swami Sankaranarayanan, and Phillip Isola. Exploring visual prompts for adapting large-scale models. *arXiv preprint arXiv:2203.17274*, 2022. 3

[3] Ondrej Biza, Sjoerd Van Steenkiste, Mehdi S. M. Sajjadi, Gamaleldin F. Elsayed, Aravindh Mahendran, and Thomas Kipf. Invariant slot attention: object discovery with slot-centric reference frames. In *ICML*, 2023. 3

[4] Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. Language models are few-shot learners. In *NeurIPS*, 2020. 3

[5] Jianrui Cai, Shuhang Gu, and Lei Zhang. Learning a deep single image contrast enhancer from multi-exposure images. *IEEE Transactions on Image Processing*, 27, 2018. 2, 6, 7, 8

[6] Mohammad Mahdi Derakhshani, Enrique Sanchez, Adrian Bulat, Victor G Turrisi da Costa, Cees GM Snoek, Georgios Tzimiropoulos, and Brais Martinez. Bayesian prompt learning for image-language model generalization. In *ICCV*, 2023. 3

[7] Ke Fan, Zechen Bai, Tianjun Xiao, Tong He, Max Horn, Yanwei Fu, Francesco Locatello, and Zheng Zhang. Adaptive slot attention: Object discovery with dynamic slot number. In *CVPR*, 2024. 3

[8] Chunle Guo, Chongyi Li, Jichang Guo, Chen Change Loy, Junhui Hou, Sam Kwong, and Runmin Cong. Zero-reference deep curve estimation for low-light image enhancement. In *CVPR*, 2020. 6, 7

[9] Xiaojie Guo, Yu Li, and Haibin Ling. Lime: Low-light image enhancement via illumination map estimation. *IEEE Transactions on image processing*, 2016. 1

[10] Yun He, Steven Zheng, Yi Tay, Jai Gupta, Yu Du, Vamsi Aribandi, Zhe Zhao, YaGuang Li, Zhao Chen, Donald Metzler, et al. Hyperprompt: Prompt-based task-conditioning of transformers. In *ICML*. PMLR, 2022. 3

[11] Jie Huang, Yajing Liu, Xueyang Fu, Man Zhou, Yang Wang, Feng Zhao, and Zhiwei Xiong. Exposure normalization and compensation for multiple-exposure correction. In *CVPR*, 2022. 1, 2, 6, 7

[12] Jie Huang, Yajing Liu, Feng Zhao, Keyu Yan, Jinghao Zhang, Yukun Huang, Man Zhou, and Zhiwei Xiong. Deep fourier-based exposure correction network with spatial-frequency interaction. In *ECCV*, 2022. 1, 2, 6, 7

[13] Jie Huang, Man Zhou, Yajing Liu, Mingde Yao, Feng Zhao, and Zhiwei Xiong. Exposure-consistency representation learning for exposure correction. In *ACMMM*, 2022. 1, 2, 6, 7

[14] Jie Huang, Feng Zhao, Man Zhou, Jie Xiao, Naishan Zheng, Kaiwen Zheng, and Zhiwei Xiong. Learning sample relationship for exposure correction. In *CVPR*, 2023. 2, 6

[15] Menglin Jia, Luming Tang, Bor-Chun Chen, Claire Cardie, Serge Belongie, Bharath Hariharan, and Ser-Nam Lim. Visual prompt tuning. In *ECCV*, 2022. 3

[16] Dongyoung Kim, Jinwoo Kim, Junsang Yu, and Seon Joo Kim. Attentive illumination decomposition model for multi-illuminant white balancing. In *CVPR*, 2024. 3

[17] Markus Krimmel, Jan Achterhold, and Joerg Stueckler. Attention normalization impacts cardinality generalization in slot attention. *Transactions on Machine Learning Research*, 2024. 3

[18] Minhyeok Lee, Suhwan Cho, Dogyoon Lee, Chaewon Park, Jungho Lee, and Sangyoun Lee. Guided slot attention for unsupervised video object segmentation. In *CVPR*, 2024. 3

[19] Liangzhi Li, Bowen Wang, Manisha Verma, Yuta Nakashima, Ryo Kawasaki, and Hajime Nagahara. Scouter: Slot attention-based classifier for explainable image recognition. In *CVPR*, 2021. 3

[20] Yiyu Li, Ke Xu, Gerhard Petrus Hancke, and Rynson WH Lau. Color shift estimation-and-correction for image enhancement. In *CVPR*, 2024. 1, 2, 6, 7, 8

[21] Risheng Liu, Long Ma, Jiaao Zhang, Xin Fan, and Zhongxuan Luo. Retinex-inspired unrolling with cooperative prior architecture search for low-light image enhancement. In *CVPR*, 2021. 6, 7

[22] Francesco Locatello, Dirk Weissenborn, Thomas Unterthiner, Aravindh Mahendran, Georg Heigold, Jakob Uszkoreit, Alexey Dosovitskiy, and Thomas Kipf. Object-centric learning with slot attention. *NeurIPS*, 2020. 2, 5, 6

[23] Minh-Thang Luong. Effective approaches to attention-based neural machine translation. *arXiv preprint arXiv:1508.04025*, 2015. 2

[24] Long Ma, Tengyu Ma, Risheng Liu, Xin Fan, and Zhongxuan Luo. Toward fast, flexible, and robust low-light image enhancement. In *CVPR*, 2022. 6, 7

[25] Ntumba Elie Nsampi, Zhongyun Hu, and Qing Wang. Learning exposure correction via consistency modeling. In *BMVC*, 2021. 2

[26] Vaishnav Potlapalli, Syed Waqas Zamir, Salman H Khan, and Fahad Shahbaz Khan. Promptir: Prompting for all-in-one image restoration. In *NeurIPS*, 2023. 3, 6, 7

[27] Rishav Pramanik, José-Fabian Villa-Vásquez, and Marco Pedersoli. Masked Multi-Query slot attention for unsupervised object discovery. In *IJCNN*, 2024. 3

[28] Gaurav Sharma and Raja Bala. *Digital color imaging handbook*. CRC press, 2017. 7, 8

[29] Gaurav Sharma, Wencheng Wu, and Edul N Dalal. The ciede2000 color-difference formula: Implementation notes, supplementary test data, and mathematical observations. *Color Research & Application*, 2005. 7, 8

[30] James Seale Smith, Leonid Karlinsky, Vyshnavi Gutta, Paola Cascante-Bonilla, Donghyun Kim, Assaf Arbelle, Rameswar Panda, Rogerio Feris, and Zsolt Kira. Coda-prompt: Continual decomposed attention-based prompting for rehearsal-free continual learning. In *CVPR*, 2023. 3

[31] Bowen Wang, Liangzhi Li, Yuta Nakashima, and Hajime Nagahara. Learning bottleneck concepts in image classification. In *CVPR*, 2023. 3

[32] Cong Wang, Jinshan Pan, Wei Wang, Jiangxin Dong, Mengzhu Wang, Yakun Ju, and Junyang Chen. Promptre-

storer: A prompting image restoration method with degradation perception. In *NeurIPS*, 2023. 3

[33] Haoyuan Wang, Ke Xu, and Rynson WH Lau. Local color distributions prior for image enhancement. In *ECCV*. Springer, 2022. 1, 2, 6

[34] Ruixing Wang, Qing Zhang, Chi-Wing Fu, Xiaoyong Shen, Wei-Shi Zheng, and Jiaya Jia. Underexposed photo enhancement using deep illumination estimation. In *CVPR*, 2019. 2

[35] Ruixing Wang, Qing Zhang, Chi-Wing Fu, Xiaoyong Shen, Wei-Shi Zheng, and Jiaya Jia. Underexposed photo enhancement using deep illumination estimation. In *CVPR*, 2019. 1

[36] Yanbo Wang, Letao Liu, and Justin Dauwels. Slot-vae: Object-centric scene generation with slot attention. In *ICML*, 2023. 3

[37] Yang Wang, Long Peng, Liang Li, Yang Cao, and Zheng-Jun Zha. Decoupling-and-aggregating for image exposure correction. In *CVPR*, 2023. 1, 2, 6

[38] Chen Wei, Wenjing Wang, Wenhan Yang, and Jiaying Liu. Deep retinex decomposition for low-light enhancement. *arXiv preprint arXiv:1808.04560*, 2018. 1, 2, 6

[39] Wenhui Wu, Jian Weng, Pingping Zhang, Xu Wang, Wenhan Yang, and Jianmin Jiang. Uretinex-net: Retinex-based deep unfolding network for low-light image enhancement. In *CVPR*, 2022. 2

[40] Mingde Yao, Jie Huang, Xin Jin, Ruikang Xu, Shenglong Zhou, Man Zhou, and Zhiwei Xiong. Generalized lightness adaptation with channel selective normalization. In *CVPR*, 2023. 2

[41] Yonghua Zhang, Jiawan Zhang, and Xiaojie Guo. Kindling the darkness: A practical low-light image enhancer. In *ACMMM*, 2019. 2

[42] Yonghua Zhang, Xiaojie Guo, Jiayi Ma, Wei Liu, and Jiawan Zhang. Beyond brightening low-light images. *International Journal of Computer Vision*, 129, 2021. 2

[43] Kaiyang Zhou, Jingkang Yang, Chen Change Loy, and Ziwei Liu. Learning to prompt for vision-language models. *International Journal of Computer Vision*, 2022. 3

[44] Karel Zuiderveld. Contrast limited adaptive histogram equalization. *Graphics gems*, 1994. 6