CURVE: CLIP-UTILIZED REINFORCEMENT LEARNING FOR VISUAL IMAGE ENHANCEMENT VIA SIMPLE IMAGE PROCESSING

Supplementary Material

Yuka Ogino, Takahiro Toizumi, and Atsushi Ito

NEC Corporation

This supplementary material provides additional details and experimental results for our CURVE method. We first explain details of our tone curve module and reinforcement learning framework that were omitted from the main paper. Then, we demonstrate visual comparisons for both multi-exposure and low-light datasets..

5. SUPPLEMENTARY DETAILS

5.1. Bézier-Curve Tone Adjustment (Sec. 2.1)

Tone adjustment maps the intensity of an input image to output image values, thereby changing the image contrast. Our method uses a cubic Bézier curve to define this mapping function. For example, "A cubic Bézier curve is a parametric curve whose shape can be modified by adjusting the positions of its control points. For a parameter q ($0 \leq q \leq 1$), the two-dimensional curve coordinates $c_P(q) = [x(q), y(q)]$ of a cubic Bézier curve can be expressed using four control points p_i as follows:

$$c_P(q) = (1-q)^3 p_0 + 3q(1-q)^2 p_1 + 3q^2(1-q)p_2 + q^3 p_3 \quad (11)$$

Since p_0 and p_3 represent the start and end points of the curve, we set $p_0 = [0, 0]$ and $p_3 = [1, 1]$ to fix the tone curve's endpoints at the origin and unit point. This yields Eqs. 1 and 2 in the main paper. Our tone adjustment module changes the values of p_1 and p_2 to modify the shape of the curve, thereby altering the mapping function. Fig. 6 shows examples of applying different mapping functions to an image. As illustrated, the contrast changes according to the curve shape.

For implementation, our tone adjustment module approximates this Bézier curve using piecewise linear segments. Fig. 5 illustrates the curve when $L = 8$ (seven linear segments, eight sample points). As shown in the lower part of Fig. 5, each point of the piecewise linear approximation corresponds to $(c_{P_i}(q_j), c_{P_o}(q_j))$ (as in Eq. 3 of the main paper).

We design the action parameters for reinforcement learning using r_i and θ_i as illustrated in the upper part of Fig. 5 (corresponding to Eq. 4 and 5 in the main paper). The purpose of this design is to make the curve shape approach an identity mapping when these values are close to zero.

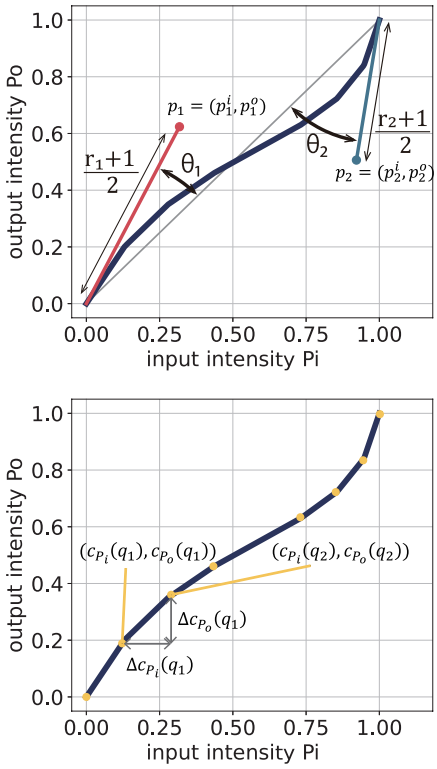


Fig. 5. Definition of our Bézier curve tone adjustment. The upper figure shows parameterization using r_i and θ_i for control points p_1 and p_2. The lower part illustrates the relationship between input and output intensity points $cP_i(q_j)$ and $cP_o(q_j)$ (as defined in Eq. 3 of the main paper).

5.2. Reinforcement Learning Framework (Sec.2.2)

We adopted the Soft Actor-Critic (SAC) algorithm as our reinforcement learning (RL) approach, following prior research. Since we made no significant modifications to SAC itself, we kept the algorithm's description brief in the main paper. This section provides additional details of our RL framework for clarity. In particular, we explain how the CLIP module and the tone adjustment are integrated into the training process, and we describe the inference (testing) framework as well.

RL aims to maximize the expected cumulative reward obtained during an episode consisting of repeated actions and state transitions. Various algorithms have been proposed for

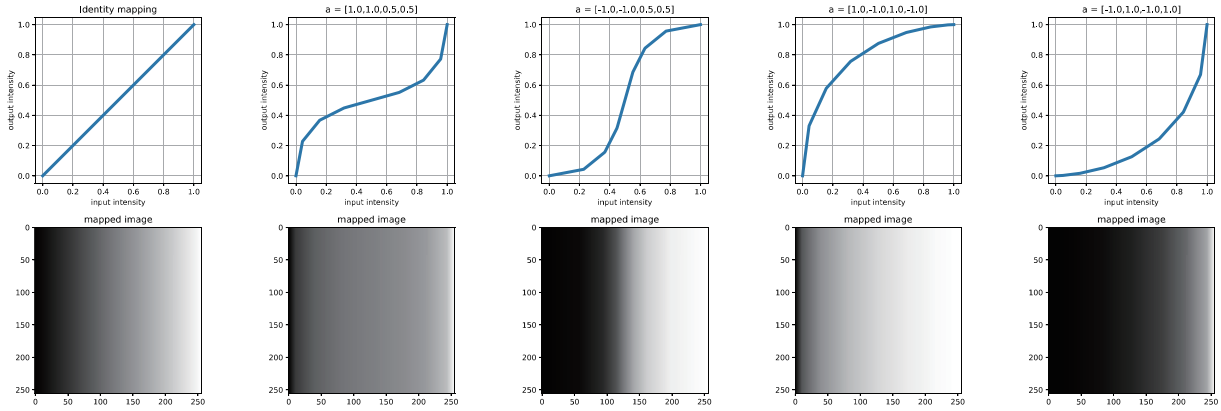


Fig. 6. Visualization of different tone curves and their effects on a gradient image. The top row shows various tone mapping functions. The leftmost is the identity mapping. The bottom row shows the corresponding output images when these mappings are applied to a gradient image (leftmost bottom).

this purpose. Soft Actor-Critic (SAC) is an off-policy method that maximizes the expected cumulative reward and policy entropy (diversity of action selection). During training, SAC simultaneously updates two networks: the policy network π_ϕ and the Q-function network Q_θ (only π_ϕ is used during inference). The Q-function network estimates the expected sum of future discounted rewards and policy entropy (the so-called soft Q-value) for a given state $\mathbf{s}_t$ and action $\mathbf{a}_t$ (Fig. 1(b) bottom in the main paper). The policy network takes a state as input and outputs parameters (μ, σ) that define a Gaussian distribution over actions. During training, actions are sampled from this distribution for exploration, while during inference (testing), the mean μ is used for deterministic action selection (Fig. 1(b) top in the main paper).

We integrate CLIP and the tone-adjustment module into the SAC framework shown in Fig. 7(a). Since SAC is an off-policy method, we utilize past exploration data obtained during training. We store tuples of $[\mathbf{s}_t, \mathbf{a}_t, r_t, \mathbf{s}_{t+1}, \mathbf{a}_{t+1}]$ and sample batches from this experience replay buffer when updating each network. CLIP is used as a fixed encoder to evaluate perceptual quality, and no gradients are propagated through the CLIP model.

Fig. 7(b) illustrates the flow for calculating the loss L_t used for reward design (Sec. 2.2.2 in the main paper). From positive/negative text T, we obtain feature vector f_T using CLIP's text encoder, and from image $\mathbf{X}$, we also obtain image feature vector $f_\mathbf{X}$ from the CLIP's image encoder. The cosine similarity $m(\cdot, \cdot)$ between the features is expressed as:

$$m(f_T, f_\mathbf{X}) = \frac{f_T^T f_\mathbf{X}}{||f_T|| ||f_\mathbf{X}||}. \tag{12}$$

This similarity is used in a softmax-based loss (see Eq. (6) in the main paper) that encourages the enhanced image feature $f_\mathbf{X}$ to be closer to the 'good image' text feature f_{T_p} than to the 'bad image' prompt f_{T_n}.

Finally, Fig. 8 shows the flow during inference (testing). Our method iteratively optimizes the mapping function and the intensity mapping. Our goal is to obtain the final high-resolution image $\mathbf{X}_T$ with all mapping processes applied, but applying the mapping process to every image $\mathbf{X}_t$ would be time-consuming. It can create state $\mathbf{s}_{t+1}$ by applying the mapping function to the downsized image state $\mathbf{s}_t$ without generating intermediate high-resolution image $\mathbf{X}_t$. Meanwhile, even when we have obtained $\mathbf{a}_t$ for all t, it is theoretically hard to derive a composite mapping function through an episode (from $t = 0$ to $t = T$). Therefore, we take advantage of the discrete value and limited range of pixel values by applying our mappings to a lookup table (LUT) that covers all possible input values. By sequentially applying actions $\mathbf{a}_t$ to this LUT $\mathbf{l}$, we create a composite LUT $\mathbf{l}_T$ from the initial LUT $\mathbf{l}_0$, which is then applied to the original image $\mathbf{X}_0$ to obtain the final enhanced result (Sec. 2.2.3 in the main paper).

6. EXTENDED QUALITATIVE COMPARISONS

6.1. Multi-Exposure Image Dataset

We show the comparison results on the SICE Part 2 dataset in Fig. 9 and Fig. 10. While our proposed method consistently maintains equivalent brightness levels across images with different exposures, Zero-DCE tends to further brighten already over-exposed images.

6.2. Low-Light Image Dataset

We show the comparison results on the LOLv2 dataset in Fig. 12 and Fig. 13. Compared to other low-light image enhancement methods, our proposed approach achieves a brightness balance closer to the ground truth.

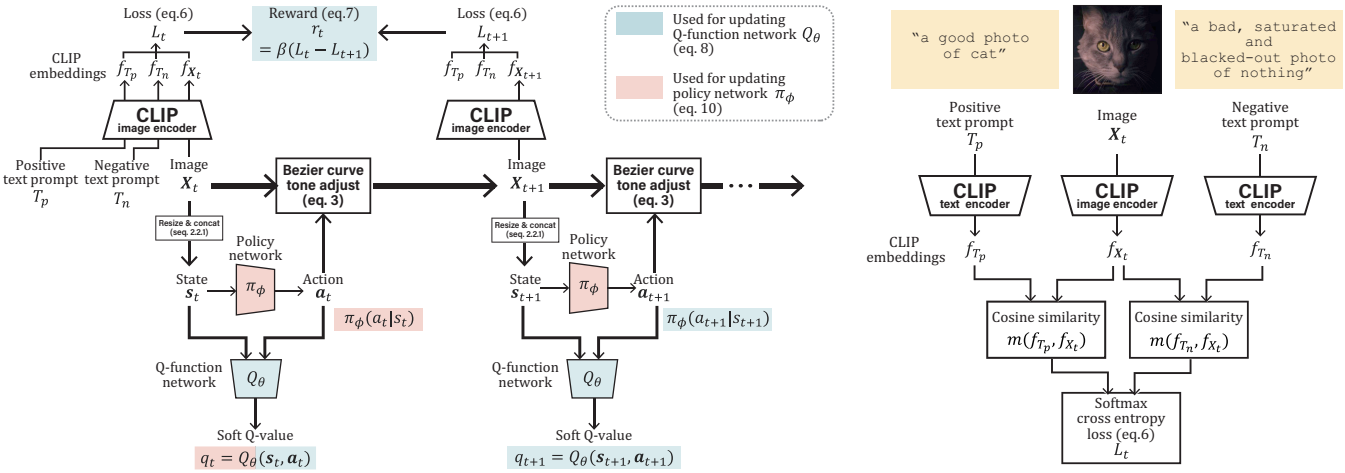


Fig. 7. Training framework of our CURVE method. (a) The exploration flow of SAC. The obtained state, action and reward $[\mathbf{s}_t, \mathbf{a}_t, r_t, \mathbf{s}_{t+1}, \mathbf{a}_{t+1}]$ are stored in the replay buffer and used to update the policy network π_ϕ and Q network Q_θ. The highlighted components (red and blue) are used for updating each network. (b) The CLIP-based reward calculation process.

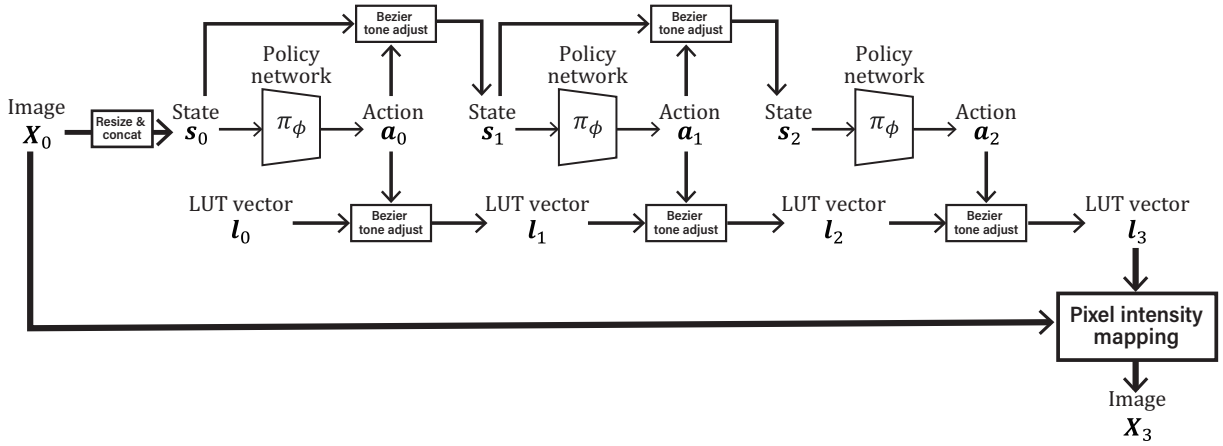
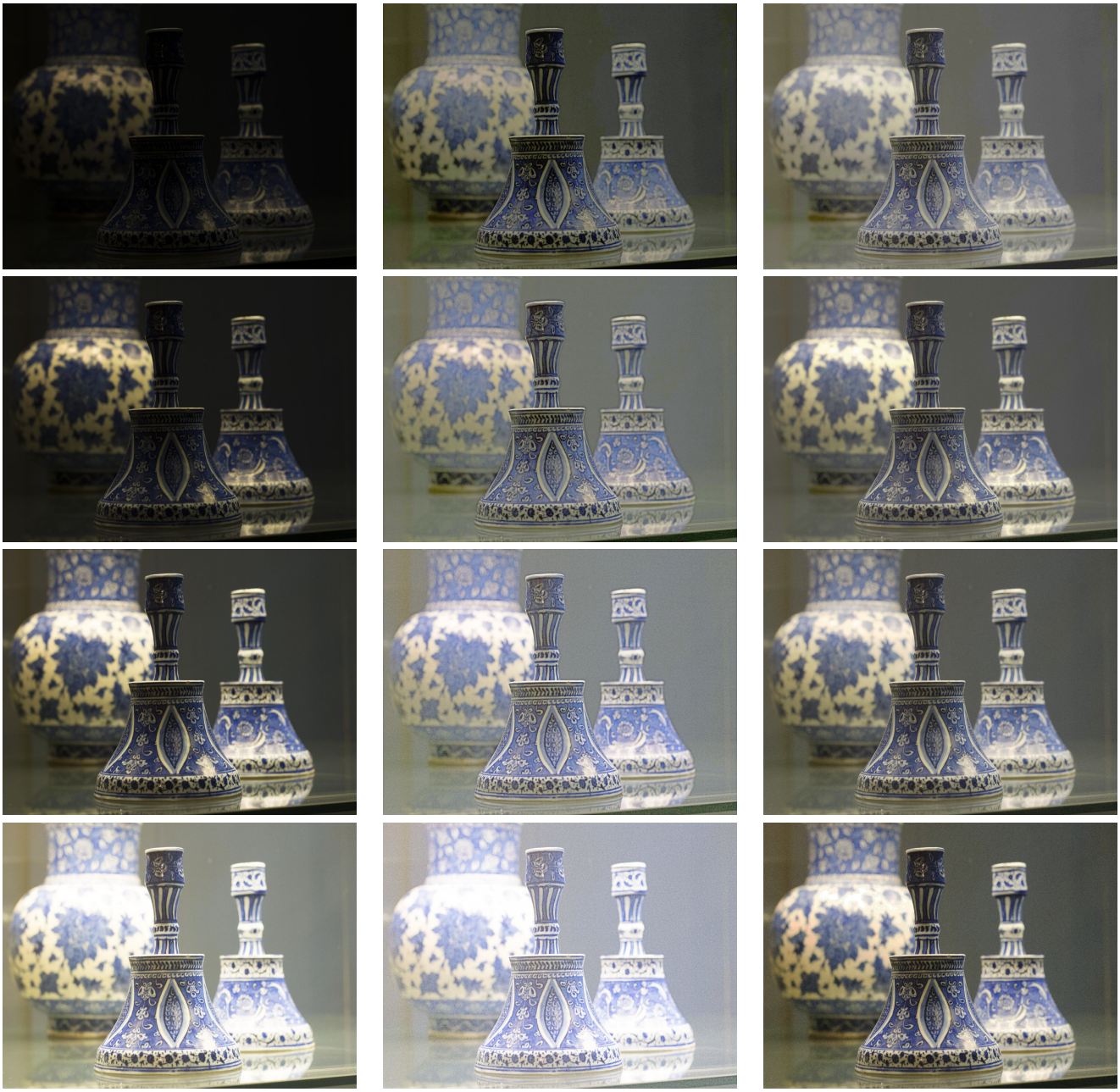


Fig. 8. Inference (testing) framework of our CURVE method



Original(input) Zero-DCE CURVE(ours)

Fig. 9. Enhancement results on multi-exposure images from the SICE Part 2 dataset. Left: Input multi-exposure images. Middle: Results of Zero-DCE. Right: Results of our proposed CURVE.

Original(input) Zero-DCE CURVE(ours)

Fig. 10. Enhancement results on multi-exposure images from the SICE Part 2 dataset. Left: Input multi-exposure images. Middle: Results of Zero-DCE. Right: Results of our proposed CURVE.

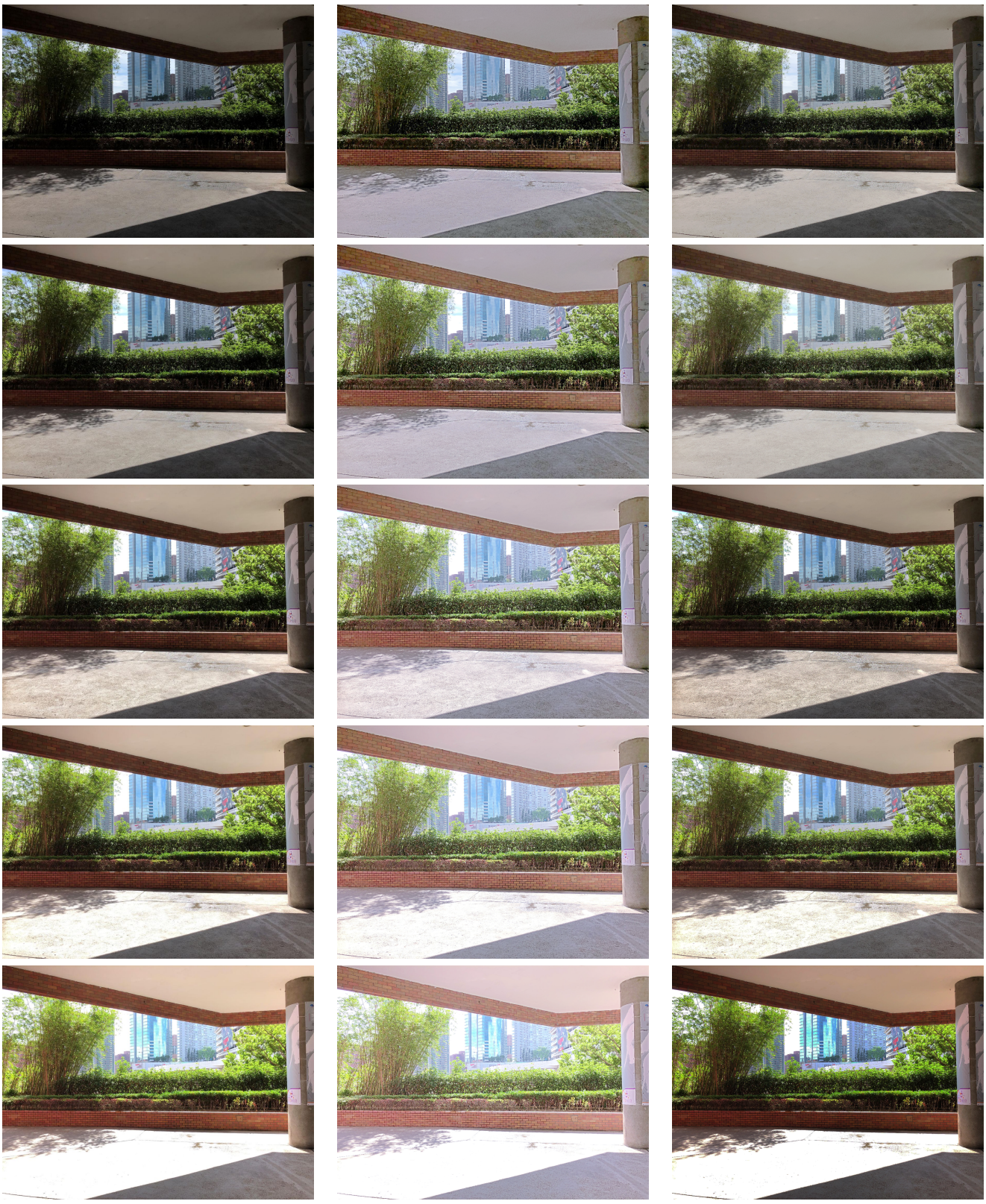


Fig. 11. Enhancement results on multi-exposure images from the SICE Part 2 dataset. Left: Input multi-exposure images. Middle: Results of Zero-DCE. Right: Results of our proposed CURVE.

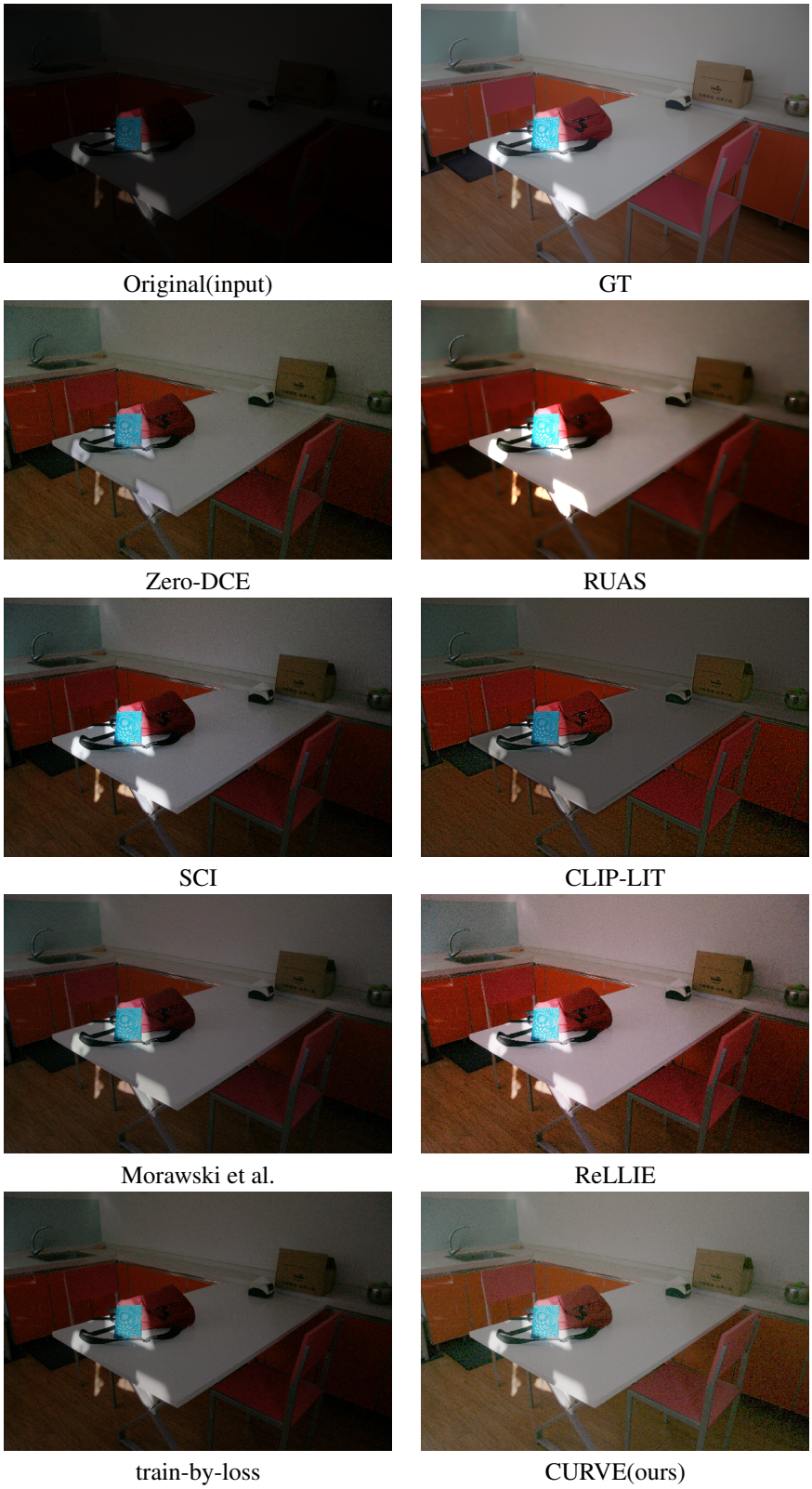


Fig. 12. Enhancement results of our experiments on low-light images from the LoLv2Real dataset. The top row shows the input low-light image and ground truth (GT). Rows 2-5 show the results of six conventional zero-reference LLIE methods, an ablation study (train-by-loss), and our proposed CURVE.

Fig. 13. Enhancement results of our experiments on low-light images from the LoLv2Real dataset. The top row shows the input low-light image and ground truth (GT). Rows 2-5 show the results of six conventional zero-reference LLIE methods, an ablation study (train-by-loss), and our proposed CURVE.