# Generalized Imaging Augmentation via Linear Optimization of Neurons: Supplementary Document

Daoyu Li[1†]
li.daoyu@bit.edu.cn
Lu Li[1†]
lilu@bit.edu.cn
Bin Li[2]
Binli@bupt.edu.cn
Liheng Bian[1‡]
bian@bit.edu.cn

[1] MIIT Key Laboratory of Complex-field Intelligent Sensing, Beijing Institute of Technology
Beijing, China

[2] School of Information and Communication Engineering, Beijing University of Posts and Telecommunications
Beijing, China

## 1 Implementation details

The detailed structure of the reported generalized augmentation network is shown in Fig. S1. The Adam solver was utilized in the experiments. In stage 1, all the transformation coefficients $\lambda_i$ of features were set as 1. We trained the decoder with the fixed VGG encoder. In stage 2, we optimized the coefficients $\lambda_i$ with the fixed encoder and decoder. The total iterations of stage 1 and stage 2 were set as 1000 (learning rate=$1 \times 10^{-4}$) and 15000 (learning rate=$1 \times 10^{-5}$), respectively. In practice, we balanced the three components in the color regularization $p_c$ via the assignment of weights to each (Eq. (S1)).

$$\begin{aligned} p_c(z) &= exp(-(\omega_s s(z) + \omega_c c(z) + \omega_\mu \mu(z))), \\ \omega_c &= 1, \\ \omega_s &= 10^{log_{10}(c(z)/s(z))+\alpha}, \\ \omega_\mu &= 10^{log_{10}(c(z)/\mu(z))-\beta}. \end{aligned} \tag{S1}$$

For lowlight imaging, $\alpha$=0, $\beta$=0; For underwater imaging, $\alpha = 0$, $\beta = 2$; For lensless imaging, $\alpha = 2$, $\beta = 2$.

## 2 Ablation study

This section presents a series of experiments that aim to illustrate the impact of color ($p_c$) and texture ($p_e$) loss on the overall performance. The pre-reconstructed images, as depicted in


† Equal contribution. ‡ Corresponding author.

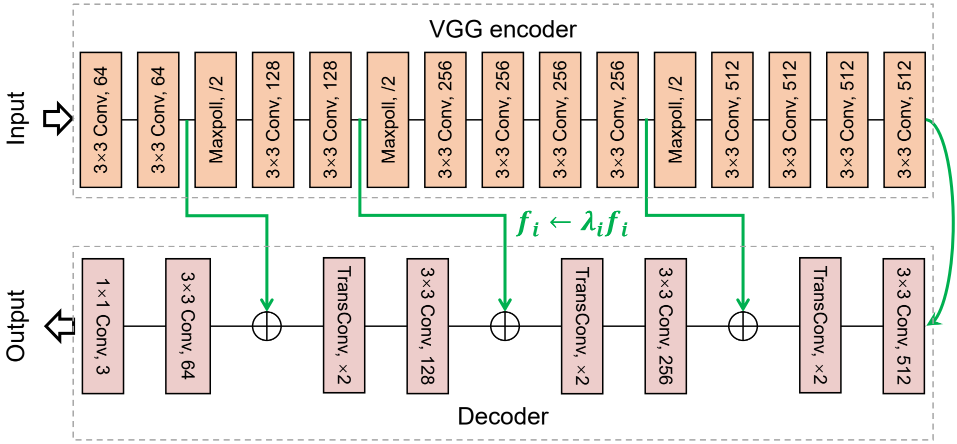


Figure S1: Detailed structure of the network used for augmenting lowlight and underwater imaging techniques. The 3×3 Conv layer in the VGG encoder consists of a Conv2d (kernel size = 3, stride = 1, padding=1) and a ReLU. The 3×3 Conv layer in the Decoder consists of a Conv2d (kernel size = 3, stride = 1, padding=1) and a Leaky ReLU (slope = 0.2). The TransConv layer is the ConvTranspose2d (kernel size=2, stride=2) in Pytorch. The operator $\oplus$ denotes the channel-wise concatenation.

Fig. S2, were obtained following the methodology outlined in the main text. The color loss $p_c$ can enhance the chroma, contrast, and saturation of output images. However, the LION augmentation only using the color loss may yield unrealistic colors (Fig. S2 #1). Instead, the texture loss $p_e$ plays a crucial role in recovering sharp textures and enhancing local edge contrast. However, relying solely on texture loss-based LION augmentation may cause color distortion (Fig. S2 #1) or result in over-enhancement (Fig. S2 #2). Experiments demonstrate that incorporating both color and texture loss can achieve superior reconstruction results with vivid color and sharp textures.

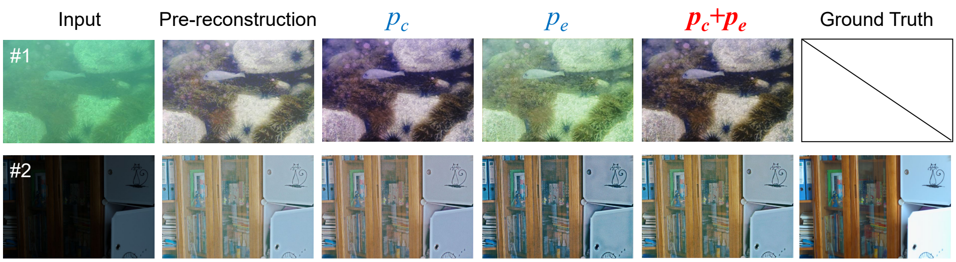


Figure S2: Reconstruction results using different losses ($p_c$, $p_e$, and $p_c + p_e$).