Universal Filter-Based Lightweight Image Enhancement Model with Unpaired Learning Mode

Aleksei Samarin, Artem Nazarenko,
Aleksei Toropov, Egor Kotenko,
Alina Dzestelova, Elena Mikhailova,
Valentin Malykh
ITMO University
St. Petersburg, Russia
avsamarin@itmo.ru, aanazarenko@itmo.ru,
toropov.ag@hotmail.com, kotenkoed@gmail.com,
aldzestelova@gmail.com, e.mikhailova@itmo.ru,
valentin.malykh@phystech.edu

Alexander Savelev, Alexander Motyko
St. Petersburg Electrotechnical University “LETI”
St. Petersburg, Russia
algsavelev@gmail.com, aamotyko@etu.ru

***Abstract*—Image enhancement is crucial in digital image processing to improve visual quality across various applications. Recent advancements in deep learning and computer vision have significantly advanced automatic color correction. While heavyweight solutions excel in quality, they demand substantial computational resources, whereas emerging lightweight models promise efficient operation on mobile devices. This study introduces a lightweight neural network model suitable for mobile devices for image color gamut correction. Our model demonstrates performance comparable with heavyweight models. We propose an approach that integrates unsupervised learning methods with multimodal visual-language priors. To our knowledge, this is the first study to use multimodal architectures as a discriminator for automatic image color correction. Also, we proposed a method for evaluating the quality of Image Enhancement models based on unpaired data using binary questions answering.**

I. Introduction

Image enhancement is one of the oldest tasks in computer vision, playing a crucial role in digital image processing. The primary goal is to increase the visual and overall quality of images to improve human perception and to enable higher-quality image processing for various image processing tasks. The task of preliminary image enhancement finds wide application across different domains, such as medicine (preprocessing of medical snapshots), history (enhancing the quality of historical photographs), and general photography. Poor-quality images result from variations in shooting conditions, camera and scene parameters, and the type and position of lighting. Images may suffer from unbalanced illuminance distribution, insufficient or excessive contrast, long exposure times, and other parameters that directly impact the perceived quality of images.

In recent years, the advancement of deep learning and computer vision has led to the emergence of many intriguing and inspiring works in the field of automatic color correction of images. Most solutions operate in an End2End manner, where the original image is input into a neural network and the enhanced image is expected as output. These approaches are heavyweight [1]–[5] and require significant computational resources while providing high-quality image enhancement. Recently, heavy solutions have increasingly employed attention mechanisms [4]–[7], which allow models to focus on the most significant parts of an image that require enhancement. This helps the network better capture the context and produce higher-quality image reconstructions.

The second direction of research comprises lightweight models [8]–[13], designed for use on edge devices, providing fast prediction and reduced computational resource requirements. Despite the progress in lightweight models for color correction, this area remains under-explored and represents a promising direction for further research. One of the current research directions in lightweight models involves using hybrid neural network approaches, which combine the simultaneous use of a neural network feature extractor to predict parameters that are subsequently applied using pre-defined transformations. A notable work in this area is by Tatanov et al. [8], which employs multiple separate generators to apply pre-determined filters.

Most of the approaches discussed use supervised learning paradigms on paired datasets (images before and after retouching). A significant limitation of this approach is the need to prepare a large, representative dataset. Unsupervised approaches are less commonly used due to the complexity of training, but they allow for the use of smaller amounts of data and can learn interesting patterns and distribution features.

Moreover, there is currently rapid development in the field of natural language processing (NLP) and multimodal approaches. A key study in this area is the CLIP model [14], which jointly trains image and text encoders to predict the correct image-text pair. The CLIP model has been applied to numerous other tasks at the intersection of computer vision and natural language processing. CLIP enables multimodal operations and uses textual prompts to obtain more context-rich features and guidance for solving various tasks. CLIP demonstrates an integrated understanding of visual and textual data. Despite the achievements of the scientific community, the potential of visual-language models is not yet fully realized, and relatively few studies have been conducted in this direction.

In this work, we propose a lightweight combined model for image color correction that matches the quality of modern state-of-the-art approaches. The aforementioned ideas inspired our research; unlike other unsupervised approaches, we applied a combination of unsupervised learning along with the rich visual language prior provided by the CLIP model. In our method, CLIP is used as a discriminator to achieve a higher-quality lightweight generator based on a modified architecture proposed in LFIEM [8]. We utilize predefined pairs of positive and negative textual prompts to obtain adversarial signals from the CLIP discriminator. CLIP excels in distinguishing between good and poor-quality images. To our knowledge, this is the first work where CLIP is used as a discriminator for automatic image color correction.

Contributions. Thus, our paper has the following contributions:

1) We propose a lightweight neural network model for image color correction that can be used on mobile devices. Our model demonstrates comparable performance on the MIT Adobe FiveK [15] and FilmSet [16] datasets in terms of PSNR and SSIM metrics. As the base architecture, we use the LFIEM model [8] but eliminate the use of multiple generator models for various filter combinations. Our model employs all considered differentiable filters and output parameters for them.
2) We combine unsupervised learning methods with the use of the visual-language prior provided by the CLIP model [14]. Using CLIP as a discriminator allows us to achieve higher-quality results, as shown in the Ablation Study section. We utilize predefined pairs of prompts corresponding to image quality. To our knowledge, this is the first work to apply CLIP as a discriminator for the task of automatic image color correction.
3) We proposed a method for evaluating the quality of Image Enhancement models on unpaired data using binary questions to VQA models. This method eliminates the need for paired datasets when assessing model performance.

II. Related Work

In recent years, the advancements in deep learning have given rise to numerous intriguing and inspiring studies in the domain of color correction for image enhancement aimed at improving visual perception. These contemporary studies can be broadly classified into several categories based on the methodologies they employ.

All proposed works can be generally classified into two main groups based on computational resource requirements: heavyweight and lightweight approaches. Heavyweight approaches are most commonly encountered in the literature [1]–[5] and require significant computational resources. However, they deliver high-quality image enhancement.

These works often use complex neural network architectures with custom blocks. An end-to-end approach is frequently employed, where the enhanced image is the direct output of the neural network, contributing to the approach's resource intensity. It is also noteworthy that mechanisms of attention are increasingly prevalent in many recent works [4]–[7], allowing models to focus on the most critical parts of the image that require enhancement. Moreover, the use of attention maps can allow the network to better capture context and produce higher-quality image reconstruction.

Models in the second group are lightweight [8]–[13]. Most of the models presented in this group can be ported for use on edge devices. These models are characterized by high prediction speed and lower computational resource requirements.

Approaches utilizing Look-up Tables (LUT) can be highlighted as a distinct area of recent research [17]–[20]. In these works, precomputed transformation tables are applied to enhance images. Substantial research endeavors are being pursued in this domain, driven by the potential of LUT-based models to notably streamline and accelerate the image enhancement process. Several of these relatively recent models contribute to the expansion of the range of lightweight architectures.

The domain of lightweight models for color correction remains relatively underexplored and represents a promising direction for further research. Many lightweight architectures are intricately designed and employ hybrid approaches, where the output image is generated by applying predefined differentiable transformations to the input image. One notable work in this domain is Tatanov et al. [8]. In this paper, the authors use multiple separate generators to apply predefined filters.

Most of the discussed methodologies adhere to the paradigm of supervised learning on datasets with paired annotations (images before and after retouching) [1], [12], [21], [22]. An inherent limitation of this approach is the necessity for preparing a substantial and representative dataset. Unsupervised approaches, owing to the intricacies of training, are less frequently used; nevertheless, they offer the advantage of applying to smaller datasets.

There is a surge of interest in the field of Natural Language Processing (NLP) and rapid development of multimodal approaches. One of the main works in this domain is the CLIP model [14], where image and text encoders are jointly trained to predict correct image-text pairs. The CLIP model has been applied to a multitude of tasks at the intersection of Computer Vision and NLP. For instance, in the work named StyleCLIP [23], the authors used the model to construct a loss function, while in studies such as [24], [25], researchers use CLIP for evaluating the quality of generative images.

These ideas served as the foundation for our research, in which we developed a model that rivals state-of-the-art methods in color correction quality. Our approach uses the CLIP model as a discriminator to attain a higher-quality lightweight generator. To the best of our knowledge, our work represents the first instance of utilizing CLIP as a discriminator for addressing the task of automatic color correction in images.

III. Methodology

The overall design of our proposed approach is illustrated in Figure 1. We use the classical generative-adversarial training scheme [26] for our model. As the generator, we use a custom-modified LFIEM model [8], and as the discriminator, we use the CLIP model [14] with predefined prompts. The multimodal embeddings of CLIP possess richer contextual information. Using CLIP as a discriminator can enable the generator model to produce higher-quality parameters for the filters by leveraging the valuable signal from the discriminator. The generator and discriminator will be discussed in more detail in the subsequent sections.

A. Generator

The overall concept of the generator architecture corresponds to the LFIEM model proposed in [8]. The generator is also two-staged. The generator architecture is detailed in the upper part of Figure 1.

In the first stage, a lightweight convolutional feature extractor with three convolutional layers is used, each with a stride of 2, Batch Normalization [27], and LeakyReLU activation [28]. The number of feature maps is equal to 16, 32, and 128 for the first, second, and third convolutional layers, respectively. This is followed by two fully connected layers with the ReLU activation function. Depending on the applied filter, the output either uses one of the activation functions: sigmoid for the [0,1] range or the hyperbolic tangent function for the [-1,1] range, or no activation function is applied at all.

The second stage involves applying classical filters with the predicted parameters. However, we have made some modifications to this architecture: instead of using multiple parameter generators for different filter combinations, we use a single parameter generator, whose output is the parameters for all filters considered in the LFIEM paper. This approach allows us to avoid the need to iterate through various filter combinations to obtain the highest quality model. In cases where a filter is not needed, our model can simply output neutral or zeroed parameters that do not alter the original image.

Let's examine the formulas for the filters in more detail. We will introduce some notations. Let I_o be the original image, and I_{so} be its resized copy. We denote the convolutional parameter generator for the filters as h. Let I_e be the enhanced version of the original image. Then I_{so} is fed into the generator h, the output of which is a parameter vector $p_i, i = [1, n]$ for the predefined n differentiable filters, which model changes in the digital image by adjusting white balance, exposure, and other color transformations. Each filter outputs a modified version of the original image I_o. Subsequently, a summator is applied, where the outputs of the filters are summed together with clipping at the maximum value of 1 to obtain the enhanced version of the original image I_e. The aforementioned manipulations can be represented by the following general formula:

$$p_{1..n} = h(I_{so}); I_e = min(max(I_o + \sum_{i=1}^{n} f_i(I_o, p_i), 0), 1),$$

where the clipping of the result to the range $[0, 1]$ is shown using the min and max functions, as we are working in the RGB space. Thus, the generator is called once to obtain the parameters for all filters simultaneously. Moreover, the approach remains lightweight, as only the value of the output parameter in the final layer is changed.

It is also important to consider the different transformations (filters) used in the generator model. All these transformations were mentioned in the LFIEM paper [8], and we have also used them in the present study.

Let us introduce some notations again. Let I_{in} and I_{out} be the input and output images, respectively. c denotes the color channel (red, green, blue), and (x, y) are the pixel coordinates on the image. p, q, r, s, t, u are trainable parameters. We assume that we are working in the RGB space, therefore the channel values will be normalized to the $[0, 1]$ range.

In the current work, automatic color ***saturation*** correction was applied to the image. This filter is applied to the image on a per-pixel basis and is defined by the following formula:

$$\Delta[x, y] = \begin{cases} (m - I_{in}[x, y]) \cdot (1 - \frac{1}{1-p}), & \text{if } p > 0 \\ -(m - I_{in}[x, y]) \cdot p, & \text{otherwise;} \end{cases}$$

$$I_{out}[x, y] = I_{in}[x, y] + \Delta[x, y],$$

where $p \in [-1, 1]$ is the trainable parameter and m is the per-channel mean value of the pixels in the original image.

Additionally, automatic image ***contrast*** correction was considered, where $r \in [-1, 1]$ is the trainable parameter. The image contrast correction filter can be described by the following formula:

$$I_{out}[x, y] = \begin{cases} (I_{in}[x, y] - 0.5) \cdot \frac{1}{1-r}, & \text{if } r > 0 \\ (I_{in}[x, y] - 0.5) \cdot (1 - r), & \text{otherwise.} \end{cases}$$

The transformation for automatic ***white balance*** correction is presented in the formula below. The trainable parameter $s_c \in [0, 1]$ is multiplied by each pixel in the channel c of the input image (the white balance correction filter requires three trainable parameters for the original image in the RGB color space).

$$I^c_{out}[x, y] = I_{in}[x, y] \cdot s_c.$$

The image transformation performing automatic ***exposure*** correction is written as follows:

$$I_{out}[x, y] = I_i n[x, y] \cdot 2^t,$$

where t is the trainable parameter with real value.

The trainable ***linear image*** transformation [29] is an additional important mapping. It can be described with the following formula:

$$I_{out}[x, y] = P \cdot I_{in}[x, y] + b,$$

where $P \in R^{3\times3}$ stands for the trainable affine mapping matrix, $b \in R^{3\times3}$ is the trainable vector in RGB space.

The ***channel-wise*** image ***color*** transformation that was described in [30] was also used. The transformation is a triplet

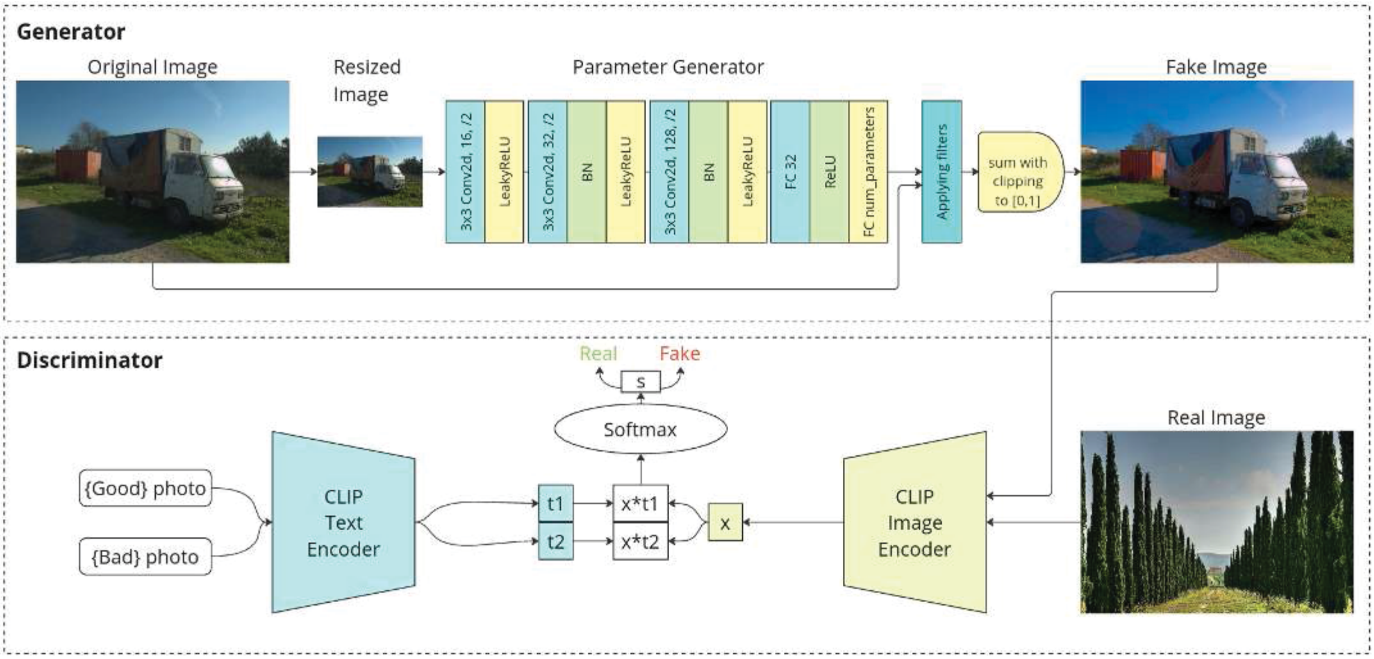

Fig. 1. An overview of the proposed method. Our model is trained in an unsupervised manner. The generator is a hybrid model that predicts parameters for differentiable classical filters. The discriminator is the multimodal CLIP model, utilizing predefined prompts such as "Good/Bad Photo".

of functions that are applied to the red, green, and blue color channels respectively. Each function is a linear combination of the elements $f_1, f_2, ...f_n$ of a n-dimensional basis, the coefficients for which are calculated from the output of our parameter generator. Therefore, a channel value for each pixel of the input image is evaluated by the formula:

$$I^c_{out}[x, y] = I^c_{in}[x, y] + \sum_{i=1}^{n} u_{ic} \cdot f_i(I^c_{in}[x, y]),$$

where $f_1, f_2, ...f_n$ – functional basis mentioned above, and u_c – trainable parameters (one parameter for each channel).

Because of its proven effectiveness [30], we considered only the piece-wise basis and used the following set of functions:

$$f_i(x) = \max(0, 1 - |(n-1) \cdot x - i + 1|), i \in \{1, 2, ...n\},$$

where x is the value of the current pixel of the input image.

B. CLIP Discriminator

The discriminator distinguishes between real and generated images. Aesthetic or enhanced images serve as the real samples, while the output of our model serves as the generated samples. In our work, we use the CLIP model with predefined prompts as the discriminator. The generative adversarial approach [26] is typically used to solve the image-to-image translation problem, where the input image is translated from a source domain X to a target domain Y. In our setup, the source domain X consists of the original images, while Y contains images that are of higher quality in terms of aesthetics and visual perception.

The paper [25] demonstrated that using a pair of antonyms as prompts for CLIP-like models is more effective, as it helps to avoid the issue of language ambiguity, specifically the ambiguous interpretation of prompts. For instance, "a rich image" could mean both a highly detailed image and an image depicting wealth. We selected "Good photo" and "Bad photo" as the baseline prompts.

C. Loss Function

We train our lightweight generator in an unsupervised manner. For training, we utilized the standard adversarial loss [26]. The adversarial loss L_{adv} describes the competition between the generator and the discriminator:

$$L_{adv} = \log D(x) + \log(1 - D(G(I_o))),$$

where D is the discriminator, G is the generator, x represents real data (aesthetic or enhanced images), and I_o represents the input image. $D(\cdot)$ represents the probability that the image is "real." During adversarial training, the discriminator aims to maximize L_{adv}, while the generator tries to minimize it.

Additionally, we aim for the parameter generator to be invariant to weak augmentations and to produce the same values for similar images. To achieve this effect, we utilized the loss function with Consistency Regularization, as proposed in [8], and represented by the following formula:

$$L_{cr} = \sum_{i=1}^{n} \|h_i(I_{so}) - h_i(T(I_{so}))\|^2,$$

where I_{so} is the resized copy of the original image, h is the parameter generator, and $T(\cdot)$ denotes a weak augmentation of the image, and $\| \cdot \|$ denotes the L^2 norm of the vector. We used RandomCrop as the weak augmentation. Therefore, consistency regularization aims to minimize the gap between the parameters of the original image and its random crop.

Finally, our model is trained on a combination of loss functions:

$$L = \omega_1 L_{cr} + \omega_2 L_{adv},$$

where ω_1, ω_2 are the weight coefficients.

IV. Experimental Setup

A. Dataset

In this section, we discuss three different datasets that we used for both training and evaluation: the MIT Adobe FiveK [15] dataset for training and its subset RANDOM250 [2], [8], [31] for validation; the FilmSet [16] dataset for training and validation; and we also used the LSDIR [32] dataset of aesthetic images, we describe it in more detail below.

The MIT Adobe FiveK [15] dataset is a well-established benchmark in the field of Image Enhancement and contains 5000 pairs of images before and after processing by five different experts. We follow previous methods [2], [8], [10], [33], [34] and use the annotations by the expert with the code name C as the baseline. We use the RANDOM250 subset [2], [8], [31] of the MIT Adobe FiveK dataset for model evaluation, while the remaining 4750 pairs of images are used for training.

The second dataset we used for training and validation is FilmSet [16]. This extensive dataset contains 5285 images for each of three different film genres: Cinema, Classical Negative (Class-Neg), and Velvia. We address the task of Film Enhancement using our generator with this dataset. For training, we used 4657 images and 638 images for testing, as suggested in the original paper.

We also attempted to train the lightweight generator using aesthetic images as the target. Our main idea is to teach our generator the properties of ”aesthetic quality”. For this purpose, we selected the LSDIR dataset [32], which contains more than 80,000 curated images from Flickr. We chose 72,000 images for training and 8,000 images for testing. As the source images, we use a shuffled set of 4,300 training images from MIT Adobe FiveK and 4,207 images from FilmSet [16]. Additionally, we set aside 450 images from each dataset for validation and testing. Hereafter, we will refer to this dataset as the ”comboset”.

It is worth noting that although most of the aforementioned datasets have annotations in the form of ”image – its enhanced version”, we use these sets only for training in an unsupervised unpaired mode. We prepare the datasets for training as follows: first, we resize the images to a resolution of 800 on the shortest side while maintaining the original aspect ratio using bicubic interpolation [35]. It should be noted that such interpolation up to certain sizes is the most commonly used among methods similar to ours. Next, we get a random crop of size 512×512 from the images. Another crop of the same size is taken from the same image for consistency regularization. During testing, we do not use random crops, we pad the images to a size of 512×512. We use the RGB color space for training and validating the models, thus, we normalize the images to the $[0, 1]$ range.

B. Evaluation Metrics

1. Paired Data Evaluation

To evaluate the performance of our model on paired datasets, we used two metrics: the peak-signal-to-noise ratio (PSNR) and the structural similarity index measure (SSIM) [36]. Higher values of PSNR and SSIM indicate improved model performance. While SSIM is a more suitable metric for comparing local artifacts, PSNR represents the standard mean squared error in pixel-by-pixel comparison of two images [37].

2. Unpaired Data Evaluation

Paired data are not always available, and existing dataset annotations for Image Enhancement are often subjective, influenced by varying perceptions of aesthetics among different experts. To ensure a comprehensive evaluation of our approach, we employed a custom assessment method on unpaired data using Visual Question Answering (VQA) models. We used multiple models to assess the quality of enhanced images through a series of binary questions. This approach enables a quantitative assessment of the subjective aspects of image quality. We employed an experimentally derived set of 10 questions, presented in Table I.

TABLE I. Questions for unpaired model evaluation

Questions
Does the image have good contrast?
Are the dark and light areas of the image well-defined?
Is the white balance of the image correct?
Do the colors in the image look natural?
Is the image properly exposed?
Is the color palette of the image pleasing to the eye?
Are the colors in the image well-balanced?
Are the colors in the image vibrant and lively?
Do the colors in the image complement each other?
Are the colors in the image harmonious?

We appended our questions with the phrase ”Answer Yes or No.” at the end to establish a response pattern. This way, we obtain 10 binary answers (”Yes” or ”No”), which we then transform into 1s and 0s, respectively. The target for our evaluation is a vector of all ones, representing the highest quality image. The higher the metric value, the better the quality of the image produced by the model. The resulting values are then averaged to provide a single score in the range $[0, 1]$, reflecting the overall quality of the enhanced image:

$$Q = \frac{1}{n} \sum_{i=1}^{n} q_i,$$

where n is the number of questions, and q_i is the *i*-th question.

For a more robust evaluation, we use multiple VQA models. Their answers are also averaged using the following formula:

$$IQA_{VQA} = \frac{1}{m} \sum_{j=1}^{m} \lambda_j Q_j,$$

where m is the number of VQA models, Q_j is the evaluation of the *j*-th model, and λ_j is the weight coefficient (default $\lambda_j = 1$).

C. Implementation Details

All our experiments were conducted on a system with 1 x NVIDIA GeForce RTX 3060 and an Intel Core i5-10400 CPU with 2.90GHz. We developed the model using the Torch framework [38]. Our models were trained using the Adam optimizer [39] with the following parameters: $\beta_1 = 0$, $\beta_2 = 0.9$, and a batch size of 40. The learning rate for our lightweight generator was initialized at $1e-3$, while the learning rate for the CLIP discriminator was set at $1e-4$. Both learning rates were adjusted with a decay factor of 0.95 every 1,200 iterations. All experiments were terminated after 10K iterations, and the best checkpoint was selected based on quality metrics.

We employed several techniques to stabilize GAN training [40]. We used Label Smoothing with labels of 0.2 for fake samples and 0.8 for real samples instead of 0 and 1, respectively. Additionally, since we used a pre-trained CLIP model (based on ViT-B/32), we decided to update the discriminator weights less frequently. We performed an optimizer step for the discriminator once every 10 steps.

V. Results

In Table II, we present the comparison results of our best configurations with existing state-of-the-art approaches. In this section, we reviewed for the MIT Adobe FiveK dataset [15] nine heavyweight methods (CE+PRNL [29], Pix2Pix [1], Distort-and-Recover [2], DPED [3], 8RESBLK [41], CRN [42], HDRNet [33], MAXIM [4], MIRNET-v2 [5]) and eight lightweight methods (U-Net [9], DeepUPE [10], DeepLPF [11], DPE [12], SULPCE [13], 3D-LUT [19], SepLUT [17], LFIEM [8]).

In Table III, we present the quantitative comparison results of the models on the FilmSet dataset [16] for three different film styles: Cinema, ClassNeg, and Velvia. We considered ten different models: HDRNet [33], DPE [12], UPE [10], DeepLPF [11], 3D-LUT [19], STAR-DCE [43], LPTN [44], SepLUT [17], FilmNet [16], and CLIP-LUT [20].

All the values in Table II and Table III are adopted from the respective papers. It is worth noting that while our proposed solution is not the best in terms of PSNR and SSIM metrics, it demonstrates a decent performance that is comparable to state-of-the-art approaches. Additionally, a significant advantage of our solution is its lightweight nature.

We also evaluated the performance of our model in comparison with other unsupervised approaches using our proposed metric based on VQA models. The comparison results are presented in Table IV. We used four VQA models trained on the VQAv2 dataset [45]: BLIP [46], BLIP2 [47], ALBEF [48], and PNP [49].

We also present a visual comparison of the models' output in Fig. 2. Additional visual results are presented in Fig. 3.

As one can see, our proposed solution based on the combined dataset with aesthetic images performs better when tested on the MIT Adobe FiveK dataset. This is because the FilmSet dataset is not a classical dataset for the Image Enhancement task; it is intended for the Style Transfer task and includes three different target styles. To effectively learn these styles, it is necessary to use *only* target images from this dataset without shifting the distribution towards images with different styles.

TABLE II. Comparisons of different methods with our best model on MIT Adobe FiveK dataset (RGB color space) – the first result is bold and the second is underlined

Method	# params	PSNR↑	SSIM↑	Train-Test Split
Heavyweight				
CE+PRNL	>30M	24.19	0.915	4750-250
Pix2Pix	54M	-	0.857	4750-250
Distort&Recover	153M	-	0.905	4750-250
DPED	-	21.76	0.871	2250-500
8RESBLK	-	23.42	0.875	2250-500
CRN	-	22.38	0.877	2250-500
HDRNet	-	21.96	0.866	4500-500
MAXIM	14.1M	**26.15**	**0.945**	4500-500
MIRNET-v2	5.9M	23.97	0.931	4500-500
Lightweight				
U-Net	1.3M	22.24	0.850	4500-500
UPE	1.0M	23.04	0.893	4500-500
DeepLPF	800K	24.48	0.887	4500-500
DPE	2.2M	23.89	0.906	4750-250
SULPCE	>1M	23.93	0.920	4000-1000
3D-LUT	<600K	24.59	0.846	4500-500
SepLUT	-	25.02	0.873	4500-498
LFIEM	101K	24.77	0.911	4750-250
Ours, FiveK	**47k**	25.92	0.939	4750-250
Ours, comboset	**47k**	25.98	0.925	4750-250

TABLE III. Comparisons of different methods on the FilmSet dataset – the first result is bold and the second is underlined

Method	Cinema		ClassNeg		Velvia	
Metric	PSNR↑	SSIM↑	PSNR↑	SSIM↑	PSNR↑	SSIM↑
HDRNet	35.18	0.990	35.41	0.988	34.37	0.975
DPE	3.980	0.358	3.790	0.320	3.480	0.313
UPE	22.81	0.946	22.50	0.936	22.23	0.893
DeepLPF	36.34	0.985	33.40	0.978	34.06	0.956
3D-LUT	35.49	0.990	33.82	0.989	34.07	0.976
STAR-DCE	28.12	0.949	25.54	0.945	34.06	0.956
LPTN	36.55	0.985	34.22	0.972	33.19	0.948
SepLUT	35.82	0.986	34.10	0.982	32.88	0.964
FilmNet	**40.07**	0.993	38.89	0.992	37.60	0.981
CLIP-LUT	39.85	**0.994**	**39.05**	**0.994**	37.68	**0.982**
Ours, FilmSet	38.11	0.993	38.08	0.991	**37.69**	0.981
Ours, comboset	36.51	0.980	34.76	0.971	34.19	0.943

TABLE IV. Comparative study of unpaired models

Method	IQA_{VQA}
DPE	0.89
Pix2Pix	0.78
EnhanceGAN	0.81
Ours, comboset	**0.94**

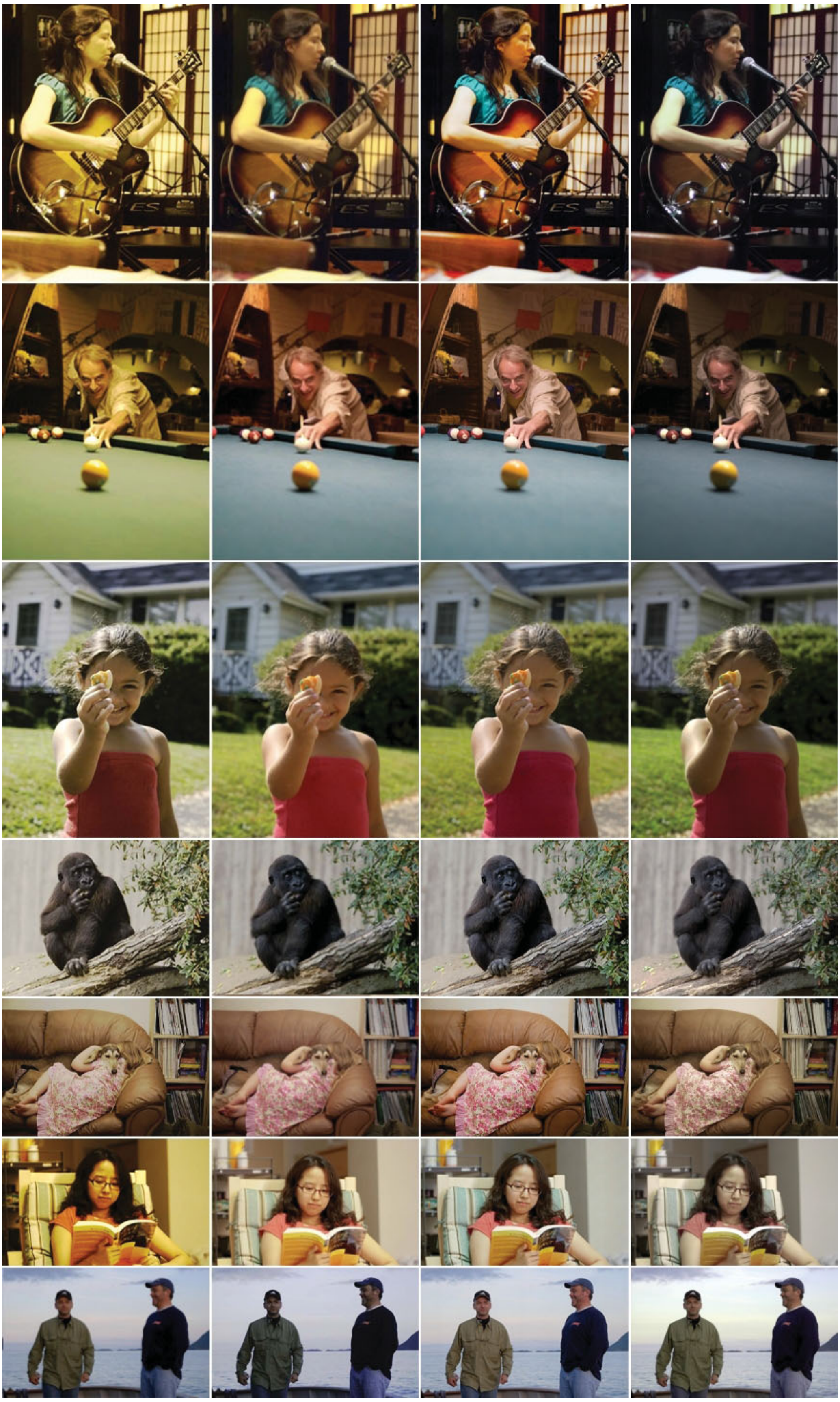

Fig. 2. Visual comparison of output for Exposure, LFIEM [23], MAXIM [24], and Our model (comboset) by columns

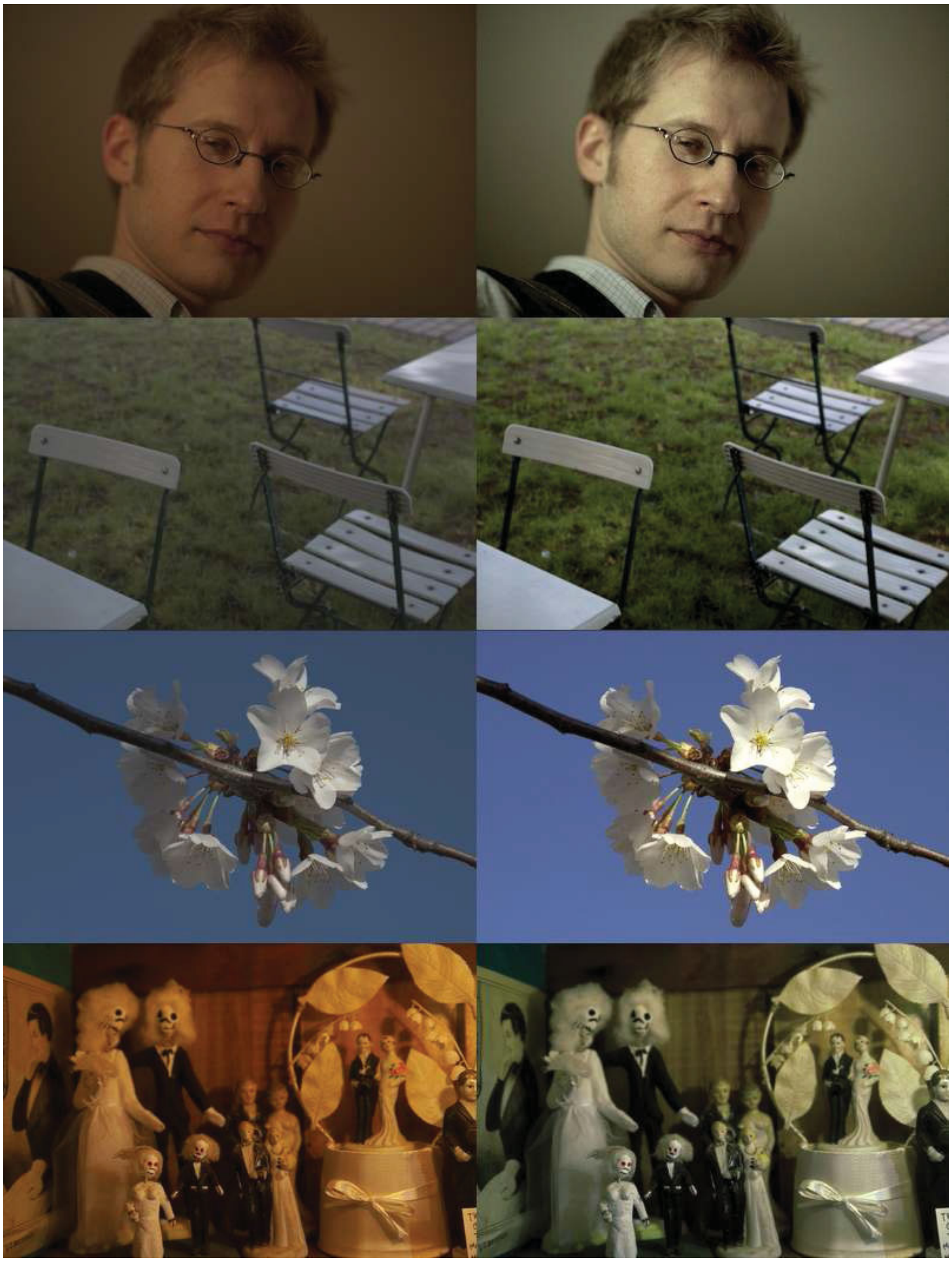

Fig. 3. Source images and outputs of Our model (comboset) on RANDOM250

A. Ablation Study

In this section, we briefly review several ablation experiments.

Our use of Consistency Regularization is motivated by the fact that training only with adversarial loss leads to unstable training and suboptimal results. In our case, Consistency Regularization serves the role of a reconstruction loss.

Additionally, we used different prompt pairs for training the discriminator. In Table V, we present a comparative study of various prompt pairs on the RANDOM250 subset of the MIT Adobe FiveK dataset (4750-250 split). The aim of this ablation experiment is to determine which prompt pairs are most effective in guiding the discriminator. We evaluate the effectiveness of different prompt pairs and show that it is most effective to use a random pair of prompts from the presented list as input to the CLIP discriminator at each optimizer step.

In another ablation experiment, we aimed to demonstrate that training with the CLIP discriminator is effective and

TABLE V. COMPARATIVE STUDY OF PAIRED PROMPTS ON RANDOM250 (MIT ADOBE FIVEK)

Prompt pair	PSNR↑	SSIM↑
good/bad photo	25.63	0.924
high/low contrast photo	25.54	0.920
bright/dark photo	24.87	0.918
high/low quality photo	24.92	0.920
clean/noisy photo	25.11	0.919
sharp/blurry photo	25.42	0.926
high/low saturated photo	25.51	0.929
correctly exposed/overexposed photo	25.34	0.923
correctly exposed/underexposed photo	25.28	0.921
highly/loss detailed photo	25.25	0.915
random prompt pair	**25.92**	**0.939**

comparable to classical paired training in terms of quality metrics. The comparison results are presented in Table VI. For this comparison, we used the MIT Adobe FiveK dataset and the same loss function configuration as presented in the LFIEM paper [8]. For the loss function in paired training, we chose a linear combination of L_1 and L_{SSIM}, where L_1 is the L^1 norm between the generated image and the ground truth image (annotated by expert C), and L_{SSIM} aims to improve the structural similarity index measure between the enhanced image and the target image. The results demonstrate that using the CLIP discriminator in an unpaired training setup is more effective than classical paired training.

TABLE VI. COMPARISON OF TRAINING WITH THE CLIP DISCRIMINATOR AGAINST CLASSICAL PAIRED TRAINING

Method	PSNR↑	SSIM↑
Unpaired Training with CLIP Discriminator	25.92	0.939
Classical Paired Training	24.51	0.924

VI. LIMITATIONS

Although our solution demonstrated good results and we showed that using CLIP with predefined prompts as a discriminator in a GAN scheme is effective, the ideas presented in this paper can be further developed. For instance, instead of using specific textual prompts, one could use learnable prompts.

VII. CONCLUSION

In this study, we presented a lightweight neural network model for image color correction optimized for mobile devices (47 thousand trainable parameters). We conducted a comprehensive comparison of our approach against competing state-of-the-art solutions, and our model demonstrated comparable performance on the MIT Adobe FiveK and FilmSet datasets, evaluated using PSNR and SSIM metrics. We integrated an unsupervised approach with the multimodal CLIP model serving as a discriminator.

Our contributions include simplifying the neural network architecture based on the LFIEM model. Specifically, instead of multiple parameter generators, we now employ a single generator that outputs parameters for various differentiable filters. Moreover, using CLIP as a discriminator enabled significant improvements in image correction quality. The application of predefined textual prompts to CLIP effectively distinguishes desirable and undesirable image characteristics, thereby enhancing the overall quality of our lightweight image correction model. We conducted a brief ablation study to validate our ideas.

To conduct a more comprehensive evaluation of model performance, we proposed a method for assessing the quality of Image Enhancement models based on unpaired data using binary questions to VQA models. Our model demonstrates superior results according to the metric we introduced.

Considering the advancements in computational resources and the widespread adoption of mobile devices, developing efficient solutions for image enhancement will play a pivotal role in advancing digital image processing technologies. We hope that our research contributes to the development of image processing in computer vision. The future work could focus on reducing encoder weights by using convolution decomposition and improving quality in unsupervised learning.

REFERENCES

[1] P. Isola, J.-Y. Zhu, T. Zhou, and A. A. Efros, "Image-to-image translation with conditional adversarial networks," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 1125–1134.

[2] J. Park, J.-Y. Lee, D. Yoo, and I. So Kweon, "Distort-and-recover: Color enhancement using deep reinforcement learning," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 5928–5936.

[3] A. Ignatov, N. Kobyshev, R. Timofte, K. Vanhoey, and L. Van Gool, "Dslr-quality photos on mobile devices with deep convolutional networks," 10 2017, pp. 3297–3305.

[4] Z. Tu, H. Talebi, H. Zhang, F. Yang, P. Milanfar, A. Bovik, and Y. Li, "Maxim: Multi-axis mlp for image processing," *arXiv preprint arXiv:2201.02973*, 2022.

[5] S. W. Zamir, A. Arora, S. Khan, M. Hayat, F. S. Khan, M.-H. Yang, and L. Shao, "Learning enriched features for fast image restoration and enhancement," *IEEE transactions on pattern analysis and machine intelligence*, vol. 45, no. 2, pp. 1934–1948, 2022.

[6] W. Ouyang, Y. Dong, X. Kang, P. Ren, X. Xu, and X. Xie, "Rsfnet: A white-box image retouching approach using region-specific color filters," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2023, pp. 12 160–12 169.

[7] Y. Cai, H. Bian, J. Lin, H. Wang, R. Timofte, and Y. Zhang, "Retinexformer: One-stage retinex-based transformer for low-light image enhancement," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2023, pp. 12 504–12 513.

[8] O. Tatanov and A. Samarin, "Lfiem: Lightweight filter-based image enhancement model," *2020 25th International Conference on Pattern Recognition (ICPR)*, pp. 873–878, 2021.

[9] O. Ronneberger, P. Fischer, and T. Brox, "U-net: Convolutional networks for biomedical image segmentation," 05 2015.

[10] R. Wang, Q. Zhang, C.-W. Fu, X. Shen, W.-S. Zheng, and J. Jia, "Underexposed photo enhancement using deep illumination estimation," 06 2019, pp. 6842–6850.

[11] S. Moran, P. Marza, S. McDonagh, S. Parisot, and G. Slabaugh, "Deeplpf: Deep local parametric filters for image enhancement," 03 2020.

[12] Y.-S. Chen, Y.-C. Wang, M.-H. Kao, and Y.-Y. Chuang, "Deep photo enhancer: Unpaired learning for image enhancement from photographs with gans," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 6306–6314.

[13] Y. Chai, R. Giryes, and L. Wolf, "Supervised and unsupervised learning of parameterized color enhancement," 03 2020, pp. 981–989.

[14] A. Radford, J. W. Kim, C. Hallacy, A. Ramesh, G. Goh, S. Agarwal, G. Sastry, A. Askell, P. Mishkin, J. Clark *et al.*, "Learning transferable visual models from natural language supervision," in *International conference on machine learning*. PMLR, 2021, pp. 8748–8763.

[15] V. Bychkovsky, S. Paris, E. Chan, and F. Durand, "Learning photographic global tonal adjustment with a database of input/output image pairs," in *CVPR 2011*. IEEE, 2011, pp. 97–104.

[16] Z. Li, X. Chen, S. Wang, and C.-M. Pun, "A large-scale film style dataset for learning multi-frequency driven film enhancement," in *Proceedings of the Thirty-Second International Joint Conference on Artificial Intelligence, IJCAI*, vol. 23, 2023, pp. 1160–1168.

[17] C. Yang, M. Jin, Y. Xu, R. Zhang, Y. Chen, and H. Liu, "Seplut: Separable image-adaptive lookup tables for real-time image enhancement," in *European Conference on Computer Vision*. Springer, 2022, pp. 201–217.

[18] C. Yang, M. Jin, X. Jia, Y. Xu, and Y. Chen, "Adaint: Learning adaptive intervals for 3d lookup tables on real-time image enhancement," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 17 522–17 531.

[19] H. Zeng, J. Cai, L. Li, Z. Cao, and L. Zhang, "Learning image-adaptive 3d lookup tables for high performance photo enhancement in real-time," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 44, no. 4, pp. 2058–2073, 2020.

[20] Z. Li, Q. Ke, and W. Chen, "Clip guided image-perceptive prompt learning for image enhancement," *arXiv preprint arXiv:2311.03943*, 2023.

[21] Y. Deng, C. C. Loy, and X. Tang, "Aesthetic-driven image enhancement by adversarial learning," 10 2018, pp. 870–878.

[22] Z. Liang, C. Li, S. Zhou, R. Feng, and C. C. Loy, "Iterative prompt learning for unsupervised backlit image enhancement," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2023, pp. 8094–8103.

[23] O. Patashnik, Z. Wu, E. Shechtman, D. Cohen-Or, and D. Lischinski, "Styleclip: Text-driven manipulation of stylegan imagery," in *Proceedings of the IEEE/CVF international conference on computer vision*, 2021, pp. 2085–2094.

[24] A. Jain, B. Mildenhall, J. T. Barron, P. Abbeel, and B. Poole, "Zero-shot text-guided object generation with dream fields," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2022, pp. 867–876.

[25] J. Wang, K. C. Chan, and C. C. Loy, "Exploring clip for assessing the look and feel of images," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 37, no. 2, 2023, pp. 2555–2563.

[26] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, "Generative adversarial nets," in *Advances in neural information processing systems*, 2014, pp. 2672–2680.

[27] S. Ioffe and C. Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," in *International conference on machine learning*. pmlr, 2015, pp. 448–456.

[28] A. L. Maas, A. Y. Hannun, A. Y. Ng *et al.*, "Rectifier nonlinearities improve neural network acoustic models," in *Proc. icml*, vol. 30, no. 1. Atlanta, GA, 2013, p. 3.

[29] C. Shan, Z. Zhang, and Z. Chen, "A coarse-to-fine framework for learned color enhancement with non-local attention," in *2019 IEEE International Conference on Image Processing (ICIP)*, Sep. 2019, pp. 949–953.

[30] S. Bianco, C. Cusano, F. Piccoli, and R. Schettini, "Learning parametric functions for color image enhancement," in *International Workshop on Computational Color Imaging*. Springer, 2019, pp. 209–220.

[31] Z. Yan, H. Zhang, B. Wang, S. Paris, and Y. Yu, "Automatic photo adjustment using deep neural networks," *ACM Transactions on Graphics (TOG)*, vol. 35, no. 2, p. 11, 2016.

[32] Y. Li, K. Zhang, J. Liang, J. Cao, C. Liu, R. Gong, Y. Zhang, H. Tang, Y. Liu, D. Demandolx *et al.*, "Lsdir: A large scale dataset for image restoration," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023, pp. 1775–1787.

[33] M. Gharbi, J. Chen, J. Barron, S. Hasinoff, and F. Durand, "Deep bilateral learning for real-time image enhancement," *ACM Transactions on Graphics*, vol. 36, 07 2017.

[34] Y. Hu, H. He, C. Xu, B. Wang, and S. Lin, "Exposure: A white-box photo post-processing framework," *ACM Transactions on Graphics (TOG)*, vol. 37, no. 2, p. 26, 2018.

[35] Y. Zhu, Y. Dai, K. Han, J. Wang, and J. Hu, "An efficient bicubic interpolation implementation for real-time image processing using hybrid computing," *Journal of Real-Time Image Processing*, vol. 19, pp. 1–13, 09 2022.

[36] Z. Wang, A. C. Bovik, H. R. Sheikh, E. P. Simoncelli *et al.*, "Image quality assessment: from error visibility to structural similarity," *IEEE transactions on image processing*, vol. 13, no. 4, pp. 600–612, 2004.

[37] A. Hore and D. Ziou, "Image quality metrics: Psnr vs. ssim," in *2010 20th international conference on pattern recognition*. IEEE, 2010, pp. 2366–2369.

[38] A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z. Lin, N. Gimelshein, L. Antiga *et al.*, "Pytorch: An imperative style, high-performance deep learning library," *Advances in neural information processing systems*, vol. 32, 2019.

[39] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.

[40] T. Salimans, I. Goodfellow, W. Zaremba, V. Cheung, A. Radford, and X. Chen, "Improved techniques for training gans," *Advances in neural information processing systems*, vol. 29, 2016.

[41] J. Chen, A. Adams, N. Wadhwa, and S. Hasinoff, "Bilateral guided upsampling," *ACM Transactions on Graphics*, vol. 35, pp. 1–8, 11 2016.

[42] Q. Chen and V. Koltun, "Photographic image synthesis with cascaded refinement networks," 10 2017, pp. 1520–1529.

[43] Z. Zhang, Y. Jiang, J. Jiang, X. Wang, P. Luo, and J. Gu, "Star: A structure-aware lightweight transformer for real-time image enhancement," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2021, pp. 4106–4115.

[44] J. Liang, H. Zeng, and L. Zhang, "High-resolution photorealistic image translation in real-time: A laplacian pyramid translation network," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021, pp. 9392–9400.

[45] Y. Goyal, T. Khot, D. Summers-Stay, D. Batra, and D. Parikh, "Making the V in VQA matter: Elevating the role of image understanding in Visual Question Answering," in *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017.

[46] J. Li, D. Li, C. Xiong, and S. Hoi, "Blip: Bootstrapping language-image pre-training for unified vision-language understanding and generation," in *International conference on machine learning*. PMLR, 2022, pp. 12 888–12 900.

[47] J. Li, D. Li, S. Savarese, and S. Hoi, "Blip-2: Bootstrapping language-image pre-training with frozen image encoders and large language models," in *International conference on machine learning*. PMLR, 2023, pp. 19 730–19 742.

[48] J. Li, R. Selvaraju, A. Gotmare, S. Joty, C. Xiong, and S. C. H. Hoi, "Align before fuse: Vision and language representation learning with momentum distillation," *Advances in neural information processing systems*, vol. 34, pp. 9694–9705, 2021.

[49] A. M. H. Tiong, J. Li, B. Li, S. Savarese, and S. C. Hoi, "Plug-and-play vqa: Zero-shot vqa by conjoining large pretrained models with zero training," *arXiv preprint arXiv:2210.08773*, 2022.