Optimal LED Spectral Multiplexing for NIR2RGB Translation

Lei Liu[1†], Yuze Chen[1†], Junchi Yan[1*], Yinqiang Zheng[2*]
[1]Department of CSE & MoE Key Lab of Artificial Intelligence, Shanghai Jiao Tong University
[2]The University of Tokyo
{loklz,cyz2096,yanjunchi}@sjtu.edu.cn yqzheng@ai.u-tokyo.ac.jp
https://github.com/cccyz/NIR2RGB

Abstract

The industry practice for night video surveillance is to use auxiliary near-infrared (NIR) LEDs, usually centered at 850nm or 940nm, for scene illumination. NIR LEDs are used to save power consumption while hiding the surveillance coverage area from naked human eyes. The captured images are almost monochromatic, and visual color and texture tend to disappear, which hinders human and machine perception. A few existing studies have tried to convert such NIR images to RGB images through deep learning, which can not provide satisfying results, nor generalize well beyond the training dataset. In this paper, we aim to break the fundamental restrictions on reliable NIR-to-RGB (NIR2RGB) translation by examining the imaging mechanism of single-chip silicon-based RGB cameras under NIR illuminations, and propose to retrieve the optimal LED multiplexing via deep learning. Experimental results show that this translation task can be significantly improved by properly multiplexing NIR LEDs close to the visible spectral range than using 850nm and 940nm LEDs.

1. Introduction

A visual surveillance system should ensure continuous and stable capture of high-quality images all day long. However, images of a scene will appear quite different due to the change of ambient illumination. During the daytime, the camera works well under sufficient daylight, while the result is not satisfactory when the visible light is little. In order to improve the imaging quality, one general idea is to utilize additional illumination to enhance images, akin to the case for ordinary cameras that add a flash unit to increase lighting or lengthen the exposure time. However, the long exposure can cause motion blur [4], and the usage of white light will easily reveal the surveillance coverage, which is unwanted in many application cases.

Since human eye is not sensitive to light with a wavelength above 720nm, near-infrared (NIR) LEDs have been widely used for night-time surveillance. NIR LEDs take advantage of the sensitivity of the camera's silicon sensor around the NIR band, making it possible to obtain visual information in the absence of visible light. Specifically, 850nm and 940nm narrow-band NIR LEDs are commonly used in surveillance device. But in most cases, the acquired images with NIR LEDs will still lose much color and texture information even if issues regarding brightness and noise do not exist. The reasons can be in two folds: 1) Camera spectral sensitivity (CSS) of three RGB channels almost overlap around both 850nm and 940nm, so it is hard to record 'color' (RGB three-channel variance). 2) Reflectance spectra of many materials become indistinguishable beyond 850nm. It makes one-to-one mapping difficult.

There are recent works [11, 14, 22, 23] trying to recover RGB directly from NIR images, but the quality of recovered RGB images is limited. The fundamental restriction lies in the fact that the mapping between NIR and RGB becomes ambiguous when using existing 850nm and 940nm LEDs for illumination. Therefore, a natural question is how to find a combination of LEDs such that the NIR2RGB translation task will be more well-posed. In this paper, we propose to retrieve the optimal LED multiplexing to reasonably maximize the distinguishability of different materials in the NIR band, and finally to achieve stable NIR2RGB restoration.

Based on the principle of camera imaging, we establish two criteria for finding the optimal LED spectral multiplexing: First, based on typical spectral curves, we set the goal of maximizing the number of distinguished colors, and the optimal LED spectral multiplexing should maximize the overall three-channel color variations. Second, more directly, the optimal LED combination should correspond to the smallest NIR2RGB image reconstruction error. Through deep learning, we directly minimize the reconstruction loss of NIR2RGB, and get the LED spectral

*The first two authors contribute equally. The last two are the correspondence authors. This work was in part supported by Shanghai Municipal Science and Technology Major Project (2021SHZDZX0102), and the JSPS KAKENHI Grant Number 20H05951.

multiplexing that might be physically realized by lightening a set of LEDs. The main highlights of this work are:

1) Rather than simply developing another network for NIR2RGB translation, we bring a novel perspective to the community on how to robustify this task by engineering on the illumination multiplexing of existing NIR LEDs.

2) We propose two optimization schemes for retrieving the optimal LED spectral multiplexing: i) maximizing the number of distinguishable colors based on the variance of typical reflectance spectra; ii) minimizing NIR2RGB translation error directly. To our best knowledge, this is the first work on optimal LED spectral selection for NIR2RGB translation. As such, the translation is achieved in a more lightweight way to achieve effective nighttime imaging compared with the majority line of works [5, 13, 25–27, 29] on developing complex enhancement models.

3) We have collected and released a Hyperspectral Images (HSIs) dataset named IDH (Indoor-Darklight-Hyperspectral images) to supplement the existing HSI datasets in terms of quality and quantity. To the best of our knowledge, IDH is the first wide-range HSI dataset that simulates night surveillance imaging. The experimental results show that high-quality RGB image generation can be effectively achieved with our method.

2. Related Work

Low-light Image Enhancement. Many methods have been proposed to enhance low-light images in the visible range. Histogram equalization [1] tries to broaden gray scale distribution without denoising the image. Retinex [19] can be used for low-light image enhancement through layer separation and composition. Recently, researchers are committed to applying Deep Learning to image enhancement. Models like Low-light Net [12], Multi-Scale Retinex Net [21] and Single Image Contrast Enhancement [3] all have good experimental results, when the environment still has weak illumination. Nevertheless, they tend to fail in darker environment, *e.g.* wild field with sky light only, where the captured RGB signals are almost indistinguishable from noise. We target at this challenging scenario but introduce NIR LED illuminations to avoid the SNR issue.

Colorization. Current colorization methods are mainly developed for RGB restoration from the grayscale images. [6] implements gray-to-rgb restoration based on the optimization of Euclidean distance between prediction and ground truth (GT). [27] shows better results in saturation and color richness. In their methods, color prediction module performs multi-mode modeling through utilizing Deep Learning to increase choices of color predictions in each pixel. In addition, methods of [8, 10, 26] also obtain excellent colorization results. All these work including ours are translating colorless images to RGB ones. However, colorization from gray scale images only needs to restore chrominance information, since luminance is included in the input already. In contrast, the NIR2RGB task is more challenging, since it has to deal with chrominance and luminance recovery from an input with domain gap.

NIR2RGB. NIR light is invisible to human eyes, yet can be captured by silicon-based sensors. So it is appropriate for low-light imaging. To recover RGB images, [11] proposes a deep convolutional neural network (CNN) for NIR2RGB translation. The idea is to train a direct transfer network between RGB and NIR images without any guidance in the recall phase. [22] puts forward NIR images colorization method based on CNN and GANs. The model learns three channels independently, and thus the convergence can be faster. However, their results are insufficient in both contrast and luminance. The fundamental obstacle lies in the ambiguity of the mapping, when dealing with NIR images captured under the widely used 850nm and 940nm LEDs.

Very recently, [24] tries to enhance weak RGB singals with the assistance of a bright image captured under deep red flash illumination. Although the selection of 680nm deep red light is rooted in the characteristic of human eye sensitivity, a key factor that we rely on as well, this work is fundamentally different from ours. Firstly, they assume weak RGB signals for chrominance, yet we use NIR information only. Secondly, they assume that the IR-cut filter is equipped and thus can not receive any light longer than 700nm. We work in the NIR range beyond 700nm, for which the human eye sensitivity is weaker, and try to translate the NIR image into RGB.

3. Methodology

As discussed above, existing solutions have difficulty in achieving stability and effectiveness for NIR2RGB translation. To solve this problem, besides optimizing model's translation capability as done in previous works [9] and further developed in our work, another key is to find an optimal LED spectral multiplexing (LSM) to get NIR images which refers to the major technical novelty of this paper.

In the following part, Sec. 3.1 introduces our proposals for selecting optimal LSM. Sec. 3.2 presents the module for NIR2RGB translation based on U-Net [20] and GANs. In addition, more details are provided in Sec. 3.3. The whole framework of the proposed method is shown in Fig. 1.

3.1. Optimal LSM Selection Module

We have tried to directly input a fixed combination of LEDs for RGB translation. Different inputs lead to very different results, and better results have a tendency to choose certain LEDs. Therefore, we believe that it is necessary to design the selection modules for the optimal LSM searching, which are based on different theories.

RGB Variance Maximization (RVM). Even if the image taken in the NIR band has three-channel, it is similar

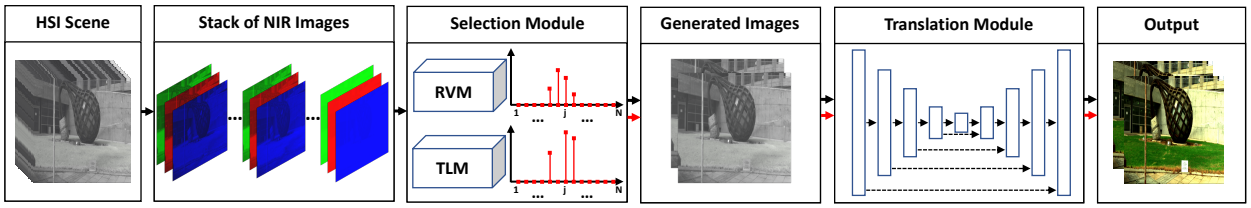

Figure 1. Our approach combines optimal LSM selection and RGB translation into a unified CNN-based framework. The optimal LSM of RVM (RGB Variance Maximization) and TLM (Target Loss Minimization) enter our translation module in parallel, which is abbreviated as one pipe. The black and red arrow shows the training and testing stage, respectively.

to that of a single-channel gray scale image to human eyes. The reason is that the RGB response values in the NIR band are almost the same among the channels. For the reflectance spectra of many objects, the corresponding RGB intensities are very close, so the captured image looks grayish. If our selection module can make pixels of specific material and color carry more information, it will definitely help the effect and robustness of subsequent NIR2RGB translation. We believe that such information can be shaped by a sufficiently large intensity variance of the RGB channels.

With this idea, we need to determine the source of the colors as the typical spectral curves (TSC). We compare two schemes: 1) using the standard ColorCheck [15] to generate TSC; 2) clustering all spectra in the training set to obtain TSC. Considering that the LSM obtained from a certain dataset is susceptible to the distribution of the dataset, while the standard ColorCheck with typical colors is more stable, we choose to use ColorCheck to generate TSC.

The three-channel's response I of light intensity is obtained via photoelectric conversion. This process of acquiring I can be formulated as:

$$I_i = \int_{NIR} (T_{i,w} \cdot L_w \cdot C_w) dw + N_i, i \in N, \tag{1}$$

where L_w and C_w represent NIR LED spectrum (NLS) and camera spectral sensitivity (CSS), respectively. $T_{i,w}$ represents the spectral curve of the i-th color in ColorCheck. N_i refers to the overall noise of the system, including the dark current noise of the camera and Gaussian white noise, and these two noise can be largely eliminated by averaging several independent measurements.

Note that TSC and CSS are both fixed, then what determines I is the LED with a different spectrum L_w. Hence, for each LSM, there is a response I of N typical colors. We enumerate every possible LSM in a brute-force way, get the corresponding I by Eq. 1, and calculate the variance of the RGB three-channel intensity of each color. Direct summation of variance from different colors inevitably leads to the loss of information, and makes it harder for the model to restore more colors. Thus, we set a threshold k to count the number of colors whose variance reaches the threshold in each LSM. Pick the LSMs with the largest number and take their mean as the final result:

$$MEAN(I_{var}), I_{var} = \begin{cases} lsm & , var \geq k \\ 0 & , others \end{cases} \tag{2}$$

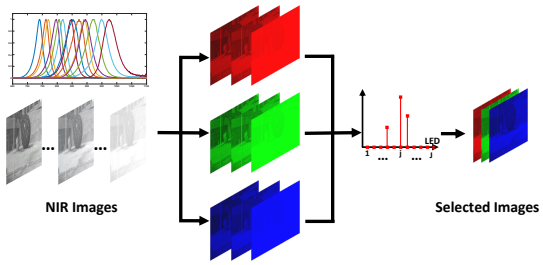

Figure 2. The process of optimal LSM (LED spectral multiplexing) selection based on TLM: 1) extract a set of NIR images from HSIs with their respective LSM; 2) split the NIR images into three channels; 3) use the same selection module on each channel's image set to select images; 4) synthesize the three channel's selected images into NIR images.

Target Loss Minimization (TLM). In addition to RVM selection based on intuition, we also design another selection module based on target loss, which can be integrated with the subsequent translation module. The comparison and integration of these two schemes are shown in Sec. 4.

In order to select the optimal LSM, NIR images for each HSIs are synthesized under all LSM at the first of the corresponding dataset. Let $C_j (j = 1, ..., J)$ represent the j-th LSM. Then, the synthesized NIR image via the j-th LSM and the t-th HSI of the training dataset can be given by:

$$Y_{j,t} = C_j X_t. \tag{3}$$

For each scene in the dataset, by stacking all the NIR images with every LSM, the selection network gets the input:

$$y_t = stack(Y_{1,t}, ...Y_{j,t}, ..., Y_{J,t}). \tag{4}$$

According to the imaging principle, synthesizing images can be seen as adding the corresponding intensity of RGB channels from NIR images. In the selecting procedure, we design a module based on CNN for the optimal LSM selection. Notice that the weights in the layer of selecting should be positive, as negative NIR in real life is meaningless.

After the stack for the NIR images with all the LSM from HSIs, the optimal LSM selection is equivalent to the NIR images selection in y_t, as shown in Fig. 2. The NIR image channels are separated into three channel branches, which are inputs of our selecting module V. The size of V is $J \times 1 \times O_i$, where $O_i = 1$ is the number of the output in i-th channel. Therefore, the output of V can be described as:

$$Y_t = stack(V * y_t(R), V * y_t(G), V * y_t(B)), \quad (5)$$

where $y_t(R), y_t(G)$ and $y_t(B)$ denote all the channels in y_t.

The weights in V can be determined by minimizing the mean squared error under the positive sparse constraint between the selected NIR image Y_t and the corresponding optimal multiplexing NIR images:

$$\mathcal{L}_s(V) = \frac{1}{T}\sum_{t=1}^{T} \|Y_t(V) - \hat{Y}_t\|^2, \quad s.t. V \geq 0, \quad (6)$$

where Y_t is the t-th output, and $\hat{Y}_t$ is the t-th corresponding optimal LSM. V is the weights of the selection module here.

3.2. RGB Translation Module

Strictly speaking, RGB here refers to the RGB image in the visible light band. The main purpose of the translation between the two image types is to learn the nonlinear mapping from NIR to RGB space. For this purpose, we build the translation model based on conditional GANs with [9].

For the generator G in GANs, the input and output are the same in resolution and structure, which means that G should not only extract the feature of the input NIR image, but also have to recover it to the RGB image with the same structure and resolution. For the requirement of this symmetrical architecture, we design a U-Net with 16 layers as the base structure. In the network G, the inputs pass through 8 layers of down sampling. After the middle of the network, the process is reversed to up sampling. Besides, there is a skip connection between layer i and layer 16-i, which concatenates every channel between these 2 layers and provides more low-level information to help translation.

For the discriminator network D, it is used to give the probability of whether the output of network G can be distinguished from the ground truth or not. We create an $N \times N$ patch on the output image which is called $PatchGAN$. It judges with L1-loss on every $N \times N$ patch in the image and tries to classify if the patch is real or not. After the patch goes through the image via convolution, an average result of responses is obtained as the output.

3.3. Learning Strategy

Selection and translation modules are two main parts of our model. During the training process, the result of RVM is input to the RGB translation module as one optimal LSM choice. As for TLM, a large set of LSMs and HSIs are given, whereby multiple NIR images can be synthesized from the HSI sets with different LSMs. Then, put the NIR image set into the network to search for the optimal LSM and its corresponding NIR images for RGB translation. The optimization of the discriminator promotes the images generated by the generator to be closer to the recovered RGB images, and finally we can use the well-trained generator to realize NIR2RGB translation. When testing, the input NIR image is obtained under the selected LSM, and its corresponding RGB image will be obtained by feeding NIR one into the generator of translation module, which has already been trained in the former process.

	ICVL	TokyoTech	IDH (ours)
Scale (Scenes × wavelengths)	201×480	16×59	112×(36+3)*
Shooting Environment	outdoor	indoor	indoor
Real Shot RGB†	✓	×	✓
White Balance‡	×	×	✓

Table 1. Dataset comparison. *: 36 refers to the number of wavelengths used to synthesize NIR images (650nm-1000nm), and 3 refers to the RGB images of each scene obtained in the visible light band with our 15S5C camera .†: Both ICVL and IDH have real shot RGB images, while one has to use white light LED spectrum to generate RGB images in TokyoTech. ‡: Only IDH adjusts the white balance, which can reduce the color shift.

The parameters of the RGB translation module are denoted by α. For the generator G in GANs, its objective has two parts: 1) L1 distance between the output of G and the ground truth; 2) MSE of the discriminator D's output with the correct judgment. Thus the objective $\mathcal{L}_t$ is written as:

$$\mathcal{L}_t(\alpha) = \frac{1}{T}\sum_{t=1}^{T} \|D(G_t(Y_t, \alpha)) - 1\|^2 + \lambda L_1(G_t(Y_t, \alpha), Z_t), \quad (7)$$

where G_t is the t-th output, Y_t is the corresponding selected NIR image from the LSM selection module. Z_t is the corresponding ground truth. λ is a predefined parameter.

The joint training of the entire network is by minimizing:

$$\mathcal{L} = \mathcal{L}_s(V) + \tau \mathcal{L}_t(\alpha), \quad (8)$$

where τ is a predefined hyperparameter. Note that in Eq. 6, $\hat{Y}_t$ that corresponds to the selected optimal LSM has no need to be labeled in the joint training process, hence $\|Y_t(V) - \hat{Y}_t\|^2$ can be ignored and we replace it with the largest value in V as the corresponding LSM. With this selection, the NIR image can be synthesized and be input to the translation module to obtain the RGB image.

Since the value in V should be non-negative, all the weights in the convolution layer are initialized as positive by uniform distribution, and all the negative numbers calculated in the backward propagation later will be set to zero.

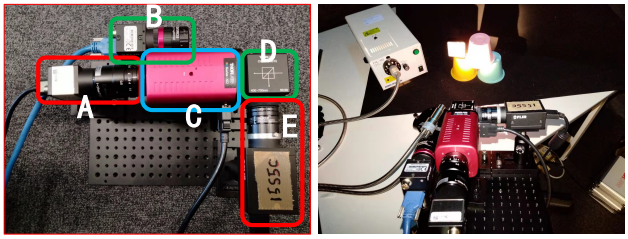

Figure 3. Our equipment (left) and scene (right) of preparing IDH, which is composed of 5 components marked with colored boxes. A: IDS UI-3860CP Grayscale Camera; B: FLIR BFLY-13S2C Color Camera; C: THORLABS Kurios-XE2 Tunable Filter (range from 650nm to 1000nm with 10nm interval); D: Beam Splitter; E: FLIR GS-U3-15S5C Color Camera.

4. Experiments

4.1. Setup and Protocols

Datasets. Three groups of spectral data are used for evaluation: 1) HSI. HSIs are stacked by the reflectance spectrum of the scene at different wavelengths. The wavelength range covers from 420nm to 1000nm. According to the light sensitivity of the human eye [24], 420nm-700nm is used for the synthesis of visible light band (RGB) images, and 700nm-1000nm is used for NIR image synthesis. The main sources of HSIs we use are: ICVL [2], TokyoTech [17] and Indoor-Darklight-Hyperspectral images (IDH). Details of these three HSIs are shown below. 2) NIR LED Spectrum (NLS). We measure the spectrum of 14 narrow-band LEDs, whose energy mainly concentrated between 700nm to 1000nm. Our optimal LSM is the combination of these LEDs. Besides, in the visible light band, we also test a white light LED (Panasonic-PremiumX) for RGB image restoration. 3) CSS. We measure the response curves of three cameras after removing the IR-cut filter: FLIR GS3-U3-15S5C, FLIR BFLY-U3-13S2C and EO 2113C. The CSSs are different as the silicon modules in these cameras are different.

Both ICVL and TokyoTech are public hyperspectral datasets. HSIs in ICVL are taken in sufficient light by using a Specim PS Kappa DX4 hyperspectral camera and a rotary stage for spatial scanning, and most of them are captured outdoors. HSIs in TokyoTech are captured indoor by using a monochrome camera and two VariSpec tunable filters.

It requires reflectance spectrum in the NIR band, while most open source datasets have only a narrow wavelength range in the visible light band. ICVL and TokyoTech are very few datasets that can meet our needs, but the ICVL do not include indoor scenes, and the scale of TokyoTech is not large enough. Thus, we have photographed IDH to expand the scale/diversity of the broad-range hyperspectral image dataset. See Tab. 1 for a more detailed comparison among these datasets. Fig. 3 shows the equipment and scene where images are collected for IDH. We use a fiber light source with a halogen lamp that emits both visible and NIR light. The UI-3860CP grayscale camera, together with the Kurios-XE2 tunable filter, is used to record spectral images from 650nm to 1000nm, at 10nm interval. The 15S5C camera is used to record the RGB image. These two cameras are accurately aligned through geometric calibration. The 13S2C camera is used to capture RGB images for test.

Data Processing. 1) The wavelength intervals of datasets are different. The intervals of NLS and HSIs in ICVL are 0.76nm and 1.25nm respectively, while the intervals of CSS and HSIs in TokyoTech and IDH are 10nm due to the sampling accuracy of tunable filter. We set all intervals to 10nm for alignment. 2) Since the LEDs we use are of narrow-band, the illumination intensity is mainly distributed near the crest (about 60nm) and is very low at other wavelengths. Intensity lower than the detection threshold turns out to be negative due to inaccuracies in dark compensation. We set them to zero directly, as the value of NLS will affect the weight of the corresponding LED in our model, while negative response makes no sense to the actual situation. 3) When synthesizing NIR and RGB images, the white balance is adjusted to prevent colors from shifting too much, and all RGB images (except for the real shots in IDH) are synthesized by the same white LED mentioned above, so that the overall tone of the image is more stable.

Metrics. The quality of the restored image is the criterion for evaluating the performance of both models and LEDs. Peak Signal to Noise Ratio (PSNR), SSIM (Structural Similarity), and Root-Mean-Square Error (RMSE) are utilized to quantify the difference between restored images and ground truth. Delta-E is used to evaluate the color quality of restoration. Specifically, PSNR is the most common and widely used objective measurement method for comparing within RGB images. SSIM shows the similarity of brightness, contrast, and spatial structure. When the value turns out to be 1, it means these two images are the same. RMSE represents the square root of the differences between predicted values and observed values. And the Delta-E indicates the average chromatic aberration between ground-truth and the restored image, the lower the score, the harder for human to distinguish the two colors.

Baselines. This part, we design multiple experiments for comparison, and results are listed in Tab. 2. Note that all the methods involved can be used for NIR2RGB translation.

In the first group, several colorization methods based on deep learning are chosen for comparison. In addition to [27] mentioned in Sec. 2, the work [5] trains a network with the consistent reflectance of paired low/normal light images and smoothness of illumination. MBLLEN [13] achieves a good performance in the real night-time scene, since this network can handle various factors simultaneously including brightness, contrast, artifacts, and noise. The results of all these methods are the mean of multiple experiments. In [7], two low-light image enhancement techniques are

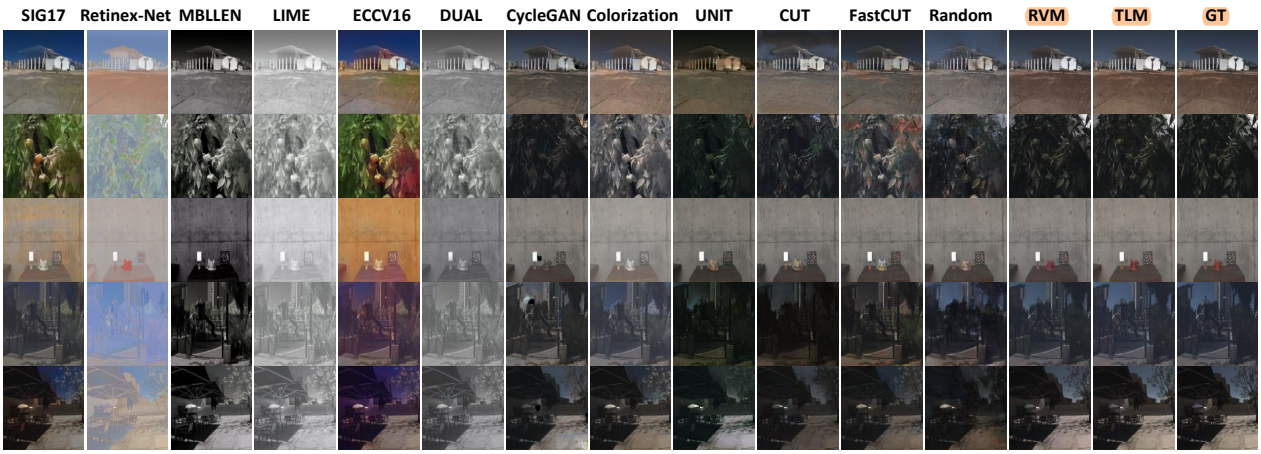

Figure 4. Visual quality comparison on ICVL in typical scenes.

Methods	PSNR (dB) ↑	SSIM ↑	RMSE ↓	Delta-E ↓
SIG17 [27]	12.52	0.52	10.55	10.49
Retinex-Net [5]	8.99	0.23	10.35	13.06
MB-v1 [13]†	11.11	0.56	10.32	11.05
MB-v2 [13]†	12.26	0.46	10.55	11.00
LIME [7]	8.51	0.33	10.45	13.83
ECCV16 [26]	12.96	0.52	10.54	10.54
DUAL [25]	11.86	0.39	10.52	11.95
CycleGAN [28]	22.31	0.76	8.42	8.18
Colorization [9]	16.95	0.55	9.85	14.97
UNIT [16]	21.58	0.70	9.05	11.14
CUT [18]	20.50	0.62	8.80	10.03
FastCUT [18]	19.47	0.59	9.35	11.19
850nm‡	22.01	0.62	8.54	8.68
940nm‡	21.85	0.65	8.46	8.61
Random*	22.94	0.68	8.30	8.60
RVM (Ours)	**24.21**	**0.80**	**8.01**	**7.33**
TLM (Ours)	**24.53**	**0.80**	**7.74**	**7.28**

Table 2. Restoration of different methods under the CSS of FLIR GS3-U3-15S5C. The first two groups contain various methods in Sec. 4.1, all the model have been trained on ICVL, and the input is the same with our model. The last two groups contain the results based on our translation module with industrial LEDs and our LSM selection. †: For the compared method MBLLEN (MB), two models v1 and v2 are trained using low-light images with Poisson noise and images without additional noise respectively. ‡: The restoration process is done by our translation module. *: Remove the selection module of our model and replace it with a randomly generated combination to put into translation module.

proposed via illumination map estimation. Both methods are based on Retinex modeling, aiming to estimate the illumination map by preserving the prominent structure of the image while removing redundant texture details. In [26], an approach is devised to produce vibrant and realistic colorizations, with promising results in both gray-scale images and NIR images. And [25] uses automatic exposure correction to produce high quality results for low-light images.

The second group also contains several classical methods for image-to-image translation and colorization. CycleGAN [28] is a method for style translation between two domains, which achieves the migration between source and target without establishing one-to-one mapping between NIR and RGB images. Colorization [9] aims to colorize a grayscale image, that is, changing black-white images into color ones. UNIT [16] makes a shared-latent space assumption and proposes an unsupervised image-to-image translation framework based on Coupled GANs. CUT and FastCUT [18] propose to directly establish a corresponding relationship between two domains based on contrastive learning to maximize their mutual information. For fair comparison, we feed the synthesized images under the optimal LSM with the TLM criteria for all methods. When evaluating, all methods have been retrained on our data. The input image is synthesized using the CSS of FLIR GS3-U3-15S5C with the original hyperspectral data from ICVL. As the LSM of TLM is already known, the input data can be synthesized individually, and the input of these methods is the same as TLM in our model. In addition, Colorization have to take one more preprocess step, that is to translate the synthesised NIR images and RGB images into Lab space to get the real input and ground-truth respectively.

We compare our optimal LSM with the NIR LEDs as commonly used in surveillance. We take out the 850nm and 940nm LEDs as the input of our translation module separately. Finally, to verify whether the combination of translation module and selection module has better results, we remove selection module and generate a random combination for translation, and this step can also be regarded as a comparison with the efficient image translation model.

4.2. Main Results

Tab. 2 lists comparison of baselines and our method on metrics, Fig. 4 and Fig. 5 are the corresponding visual displays. Results show that our method has advantage both in terms of reconstruction metrics and image quality. Tab. 4 is

Dataset		ICVL				TokyoTech				IDH			
Metrics		PSNR	SSIM	RMSE	Delta-E	PSNR	SSIM	RMSE	Delta-E	PSNR	SSIM	RMSE	Delta-E
15S5C	RVM	24.2054	0.8005	8.0120	7.3325	16.5380	0.6247	9.1241	13.4263	26.0707	0.8006	6.1820	4.5289
	TLM	24.5263	0.7938	7.8438	7.2782	16.3309	0.6408	9.2550	13.6712	25.0017	0.7871	6.4829	4.9443
13S2C	RVM	25.5941	0.8343	7.6359	6.4647	17.8095	0.6946	9.1302	12.0473	25.3464	0.8063	6.1364	4.8523
	TLM	24.9453	0.8064	7.7102	6.8793	17.0359	0.6885	9.4096	12.9665	25.9891	0.8164	6.2660	4.6960
2113C	RVM	24.0810	0.7884	7.8578	7.4591	16.2189	0.6245	9.2569	13.6378	-	-	-	-
	TLM	24.2325	0.7884	7.8578	7.2572	16.2743	0.6547	9.3188	13.6128	-	-	-	-

Table 3. Results of translation from NIR to RGB in different conditions by the proposed method.

LED		739	760	768	796	804	818	845	852	872	888	894	923	948	973
15S5C	RVM	0.2465	0.3221	0.2869	0.0573	0.0221	0.0110	0.0072	0.0080	0.0066	0.0058	0.0058	0.0057	0.0053	0.0050
	TLM	0.1538	0.4615	0.3846	0	0	0	0	0	0	0	0	0	0	0
13S2C	RVM	0.2196	0.2945	0.2528	0.0820	0.0377	0.0217	0.0127	0.0141	0.0098	0.0089	0.0089	0.0095	0.0099	0.0095
	TLM	0.0833	0.4861	0.4306	0	0	0	0	0	0	0	0	0	0	0
2113C	RVM	0.1235	0.7059	0.1706	0	0	0	0	0	0	0	0	0	0	0
	TLM	0.1266	0.4051	0.3544	0.1139	0	0	0	0	0	0	0	0	0	0

Table 4. The optimal LSM ratio of ICVL in selection module with different CSS. Values are all normalized and 0 denotes less than 10^{-4}.

the optimal LSM in selection module after training. Tab. 3 and Fig. 6 show the results of our method based on different camera and dataset, and the results show that our method implements NIR2RGB efficiently and with good quality.

Does Selection Module Work? Fig. 5 shows the comparison between the commonly used LEDs in the industry and our optimal LSM, both of them are individually trained. Obviously, the output of selection module directly affect the recovery performance of our model and our selection module do find a better LSM. According to Tab. 2, the results of TLM are slightly better than those of RVM. Note that TLM is related to camera type and dataset while RVM only camera, hence we can draw a conclusion that TLM gives better results with specific dataset and camera, but RVM is more robust when light condition changes.

Performance of Translation Module. Tab. 3 compares RVM and TLM under the same conditions. We qualify the performance of RGB translation within the datasets. A better RGB recovery result corresponds to a better LSM as shown in Tab. 4. After the joint training process, our method successfully selects the optimal LSM based on the three different CSSs, and the results shows that the LSMs of both methods in one CSS are slightly different.

4.3. Further Study and Analysis

Our model achieves good results under **different cameras**, **color distributions**, and **illumination conditions**.

Specifically, we use different CSS of three cameras to synthesize NIR and RGB images. The results in Tab. 3 and Fig. 6 show that the output of our selection module is changing with the CSS, while the final recovery images are satisfactory in terms of metrics and intuitive perception of the human eye. It means that our model has strong tolerance and can be applied to various cameras. Moreover, according to Tab. 1, the three datasets have a large discrepancy in color distribution and illumination conditions. Scenes in ICVL are under abundant light, and its corresponding outputs are excellent. Similarly, in our IDH, even with the lacks of visible light, our model can still restore the NIR images very well. This not only verifies that our model is effective for image restoration of the night-time surveillance system, but also shows its robustness against color distribution and illumination conditions change.

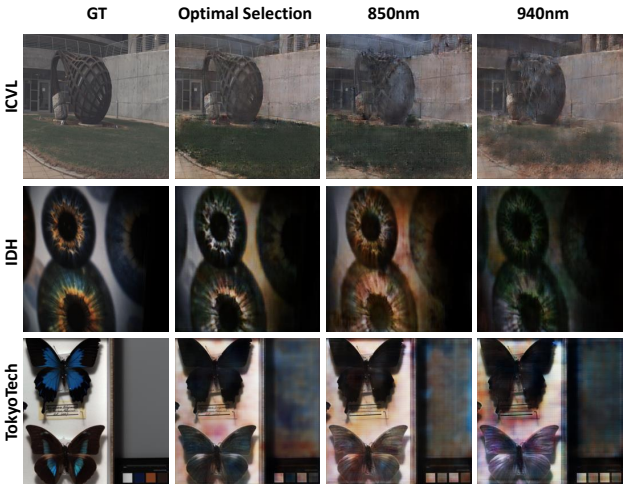

Figure 5. Visual comparison with widely used NIR LEDs.

We further give several examples on the RGB translation module to see its performance. In addition, the effect of the optimal LSM selection is also discussed.

NIR2RGB Translation. Our method mainly focuses on the search of the nonlinear mapping between two types of images, which have intrinsic relationship in the physical world. We employ the input NIR image to guide the RGB information translation, which is modeled by stacking the input NIR images. Fig. 4 and Tab. 2 show that our designed network structure for nonlinear mapping performs better than other low-light enhancement, colorization and domain transfer methods in Sec. 4.1.

LSM Selection. To evaluate the effectiveness of the selection module, we remove it and then put a fixed random

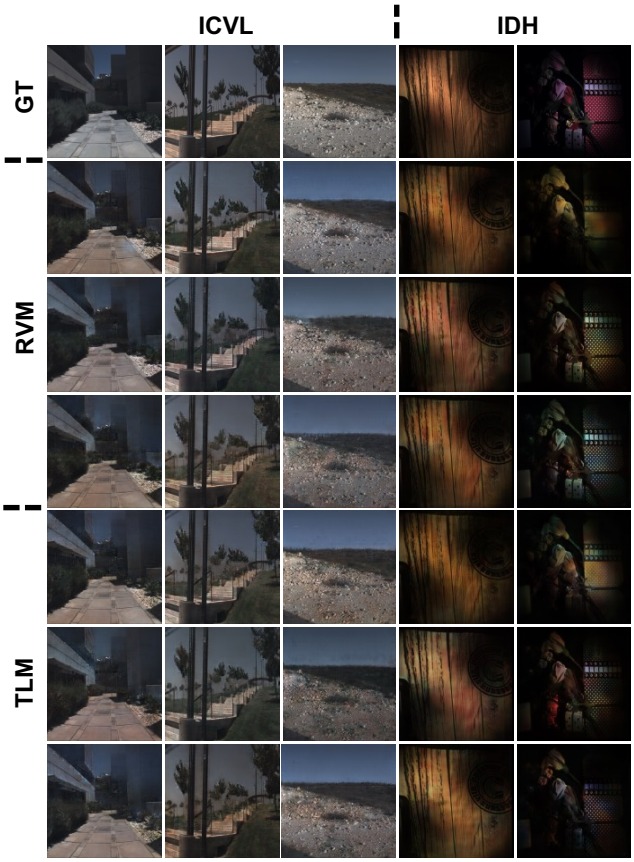

Figure 6. Visual comparison of three cameras in 2 multiplexing scheme on five typical scenes on the studied datasets ICVL and IDH. The first row is the ground-truth, the rest rows are divided into two parts: RVM and TLM, in each part, from top to bottom is the result of 15S5C, 13S2C and 2113C respectively.

multiplexing of LEDs into the translation model, so that the model becomes a pure image-to-image translation model. The results of the last three lines in Tab. 2 show that the addition of selection modules makes the results much better. It demonstrates that the NIR2RGB conversion can be improved by selecting the appropriate LEDs. Besides, as shown in Fig. 5, the optimal LSM outperforms the most commonly used LEDs in industry.

Tab. 3 further shows that in the scenarios of experimental settings, RVM and TLM tend to choose LEDs which are close to the visible light band. The value of CSS is nearly the same in three channels when the wavelength exceeds 800nm, which means that the use of LED in this range can cause the lack of information for mapping into the RGB space. The reduction of information is related to the performance of color restoration, and TLM tends to choose the LSM which can generate more distinguishable message for restoration due to the decrease of loss. Meanwhile, the results of RVM is close to those of TLM, which proves that our color variance maximization scheme in Sec. 3.1 do work. Specifically, the results of RVM are related to the

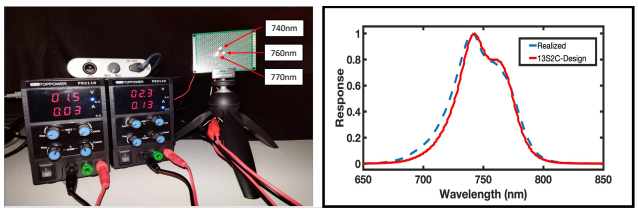

Figure 7. Using narrow-band LEDs to realize the spectrum corresponding to the LSM. Left is the instrument for adjusting the LED power to fit the target spectrum. Right is the fitting result.

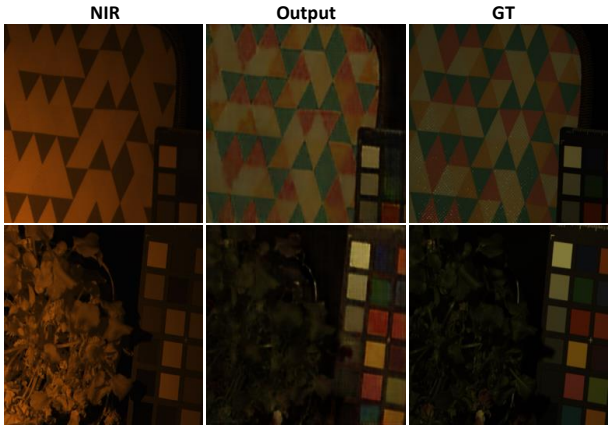

Figure 8. The restoration result when applying the optimal LSM of TLM on camera 13S2C and dataset IDH in real world.

camera type only as TSC is a fixed standard, the change of the dataset will not affect the output of the module. As for TLM, the LED model is stable when changing dataset, only the weights change slightly. That is, our selection module can be well applied to night-time surveillance under a variety of illumination conditions, as the optimal LSM in selection module is stable enough when camera is fixed.

On-device Verification. We choose the LSM based on TLM selection module and camera 13S2C to verify its applicability on device. The fitting result in Fig. 7 shows that the realized LED spectrum is basically the same as the LSM. We capture several NIR images with realized LEDs as the illumination source, and put them into our model. Fig. 8 indicates that our model works well in real scenes.

5. Conclusion

We have explored the fundamental hurdle towards stable NIR-to-RGB translation. The industry practice for night video surveillance inspires us to retrieve better spectral multiplexing of narrow-band NIR LEDs, which is realized by a novel selection module, so as to maximize the accuracy and stability of the translation task. Two strategies are devised for multiplexing optimization, whose performance has been verified using existing spectral datasets and a newly captured one. We also have noticed that the quality of different datasets is uneven, which may lead to inconsistent performance in different scenarios. A thorough investigation on this is left as our future work.

References

[1] Mohammad Abdullah-Al-Wadud, Md. Hasanul Kabir, M. Ali Akber Dewan, and Oksam Chae. A dynamic histogram equalization for image contrast enhancement. *IEEE Transactions on Consumer Electronics*, 53(2):593–600, 2007. 2

[2] Boaz Arad and Ohad Ben-Shahar. Sparse recovery of hyperspectral signal from natural rgb images. In *ECCV*, 2016. 5

[3] Jianrui Cai, Shuhang Gu, and Lei Zhang. Learning a deep single image contrast enhancer from multi-exposure images. *IEEE Transactions on Image Processing*, 27(4):2049–2062, 2018. 2

[4] Chen Chen, Qifeng Chen, Jia Xu, and Vladlen Koltun. Learning to see in the dark. In *CVPR*, 2018. 1

[5] Wei Chen, Wang Wenjing, Yang Wenhan, and Liu Jiaying. Deep retinex decomposition for low-light enhancement. In *British Machine Vision Conference*, 2018. 2, 5, 6

[6] Zezhou Cheng, Qingxiong Yang, and Bin Sheng. Deep colorization. In *ICCV*, 2015. 2

[7] Xiaojie Guo, Yu Li, and Haibin Ling. Lime: Low-light image enhancement via illumination map estimation. *IEEE Transactions on image processing*, 26(2):982–993, 2016. 5, 6

[8] Satoshi Iizuka, Edgar Simo-Serra, and Hiroshi Ishikawa. Let there be color! joint end-to-end learning of global and local image priors for automatic image colorization with simultaneous classification. *ACM Transactions on Graphics*, 35(4):1–11, 2016. 2

[9] Phillip Isola, Junyan Zhu, Tinghui Zhou, and Alexei A Efros. Image-to-image translation with conditional adversarial networks. In *CVPR*, 2017. 2, 4, 6

[10] Gustav Larsson, Michael Maire, and Gregory Shakhnarovich. Learning representations for automatic colorization. In *ECCV*, 2016. 2

[11] Matthias Limmer and Hendrik PA Lensch. Infrared colorization using deep convolutional neural networks. In *IEEE International Conference on Machine Learning and Applications*, pages 61–68. IEEE, 2016. 1, 2

[12] Kin Gwn Lore, Adedotun Akintayo, and Soumik Sarkar. Llnet: A deep autoencoder approach to natural low-light image enhancement. *Pattern Recognition*, 61:650–662, 2017. 2

[13] Feifan Lv, Feng Lu, Jianhua Wu, and Chongsoon Lim. Mbllen: Low-light image/video enhancement using cnns. In *BMVC*, page 220, 2018. 2, 5, 6

[14] Feifan Lv, Yinqiang Zheng, Yicheng Li, and Feng Lu. An integrated enhancement solution for 24-hour colorful imaging. In *Proceedings of the AAAI conference on artificial intelligence*, 2020. 1

[15] Calvin S McCamy, Harold Marcus, James G Davidson, et al. A color-rendition chart. *J. App. Photog. Eng*, 2(3):95–99, 1976. 3

[16] Liu Mingyu, Breuel Thomas, and Kautz Jan. Unsupervised image-to-image translation networks. In *International Conference on Neural Information Processing Systems*, 2017. 6

[17] Yusuke Monno, Hayato Teranaka, Kazunori Yoshizaki, Masayuki Tanaka, and Masatoshi Okutomi. Single-sensor rgb-nir imaging: High-quality system design and prototype implementation. *IEEE Sensors Journal*, 19(2):497–507, 2018. 5

[18] Taesung Park, Alexei A Efros, Richard Zhang, and Junyan Zhu. Contrastive learning for unpaired image-to-image translation. In *ECCV*, pages 319–345. Springer, 2020. 6

[19] Ana Belén Petro, Catalina Sbert, and Jean-Michel Morel. Multiscale retinex. *Image Processing On Line*, pages 71–88, 2014. 2

[20] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *International Conference on Medical image computing and computer-assisted intervention*, pages 234–241. Springer, 2015. 2

[21] Liang Shen, Zihan Yue, Fan Feng, Quan Chen, Shihao Liu, and Jie Ma. Msr-net: Low-light image enhancement using deep convolutional network. *arXiv preprint arXiv:1711.02488*, 2017. 2

[22] Patricia L Suárez, Angel D Sappa, and Boris X Vintimilla. Infrared image colorization based on a triplet dcgan architecture. In *CVPR*, 2017. 1, 2

[23] Guangming Wu, Yinqiang Zheng, Zhiling Guo, Zekun Cai, Xiaodan Shi, Xin Ding, Yifei Huang, Yimin Guo, and Ryosuke Shibasaki. Learn to recover visible color for video surveillance in a day. In *ECCV*, 2020. 1

[24] Jinhui Xiong, Jian Wang, Wolfgang Heidrich, and Shree Nayar. Seeing in extra darkness using a deep-red flash. In *CVPR*, 2021. 2, 5

[25] Qing Zhang, Yongwei Nie, and Weishi Zheng. Dual illumination estimation for robust exposure correction. In *CGF*, 2019. 2, 6

[26] Richard Zhang, Phillip Isola, and Alexei A Efros. Colorful image colorization. In *ECCV*, 2016. 2, 6

[27] Richard Zhang, Junyan Zhu, Phillip Isola, Xinyang Geng, Angela S Lin, Tianhe Yu, and Alexei A Efros. Real-time user-guided image colorization with learned deep priors. *arXiv preprint arXiv:1705.02999*, 2017. 2, 5, 6

[28] Junyan Zhu, Taesung Park, Phillip Isola, and Alexei A Efros. Unpaired image-to-image translation using cycle-consistent adversarial networks. In *ICCV*, 2017. 6

[29] Barret Zoph, Vijay Vasudevan, Jonathon Shlens, and Quoc V Le. Learning Transferable Architectures for Scalable Image Recognition. In *CVPR*, 2018. 2