國立政治大學資訊科學系

碩士論文

基於知識萃取之內容解構影像去反射

Image Reflection Removal based on Knowledge-distilling Content Disentanglement

指導教授：彭彥璁 助理教授

研究生：鄭楷翰 撰

中華民國 一百一十 年 八 月

August 2021

DOI:10.6814/NCCU202101688

摘要

當我們通過玻璃等透明介質拍攝照片時，可能會出現不可避免的反射，模糊了我們想要捕捉的場景。我們提出了一種基於知識蒸餾的方式來將影像內容進行透射層及反射層的分解，進一步解決影像反射的問題。透過實驗證明，該模型具有一定的清除反射之能力。

關鍵詞：影像處理、影像去反射

DOI:10.6814/NCCU202101688

ABSTRACT

When we shoot pictures through transparent media, such as glass, reflection can undesirably occur, obscuring the scene we intended to capture. Therefore, removing reflection is practical in image restoration. However, a reflective scene mixed with that behind the glass is challenging to be separated, considered significantly ill-posed. This letter addresses the single image reflection removal (SIRR) problem by proposing a knowledge-distilling-based content disentangling model that can effectively decompose the transmission and reflection layers. The experiments on benchmark SIRR datasets demonstrate that our method performs favorably against state-of-the-art SIRR methods.

Keywords : Image Processing、Image reflection removal

DOI:10.6814/NCCU202101688

TABLE OF CONTENTS

DOI:10.6814/NCCU202101688

DOI:10.6814/NCCU202101688

LIST OF FIGURES

DOI:10.6814/NCCU202101688

LIST OF TABLES

DOI:10.6814/NCCU202101688

1. INTRODUCTION

1.1. Motivation and Challenges

With the popularity of smartphones, people can use the built-in camera to take pictures anywhere, anytime. It is common to shoot photos through a material with reflectivity and transparency, such as windows, glass, etc., often capturing a scene with a reflection. However, the reflection is usually undesirable and disturbing to viewers, preventing them from seeing the actual scene. It could also negatively affect the performance of downstream computer vision tasks, such as object detection and recognition. Hence, image reflection removal has become essential and gained much attention in recent years [1]–[8]. There have been many attempts toward single image reflection removal (SIRR). Conventionally, one could cast it to an optimization problem based on some observed priors to separate the transmission layer from the reflection layer in a single image [9]–[11]. However, these handcrafted priors are often not applicable to different reflection scenes with various shooting conditions. Another type of work utilizes multiple images taken at the same scene to find the correlation of the transmission and reflection layers across these images [12], [13]. Due to the great success of deep learning in low-level image processing tasks, deep convolution neural networks have been widely applied to SIRR [2], [4]–[8], [14]–[17]. These deep-learning-based models adopt various architectures and techniques to address SIRR, including generative adversarial networks [2], [5], [8], [16], cascaded models [4], [8], [14], [18], supplementary information (such as depth maps and edges) [6], [7], [14], [17], etc.

This paper adopts a different approach to separate the transmission and reflection layers from an image with reflection. We propose to utilize knowledge distillation techniques to disentangle contents for the transmitted and reflected scenes. Our design has two

DOI:10.6814/NCCU202101688

networks, a Reflection Teacher network, and a Content Disentangling Student network. Unlike the conventional teacher-student learning paradigm, where the Student network is a lightweight version of the Teacher network for network compression, we regard the Teacher as a representation extractor for the reflection layer of the input image. The Teacher functions to enforce the Student to disentangle the content for both reflection and transmission layers. Moreover, we devise content disentangling loss, representation mimicking loss, and fidelity loss to supervise the entire knowledge transfer process between the Teacher and Student.

1.2. Contributions

Our main contributions of the thesis are summarized as follows:

1. We propose knowledge-distilling-based networks to disentangle transmission and reflection layers for SIRR, incorporating Content-aware Layers that adopt multi-kernel strip pooling [19] in the Student network to mimic transmission and reflection features.
2. We collect a Natural Reflection Dataset (NRD) to facilitate SIRR research, which contains 136 real image triplets with various scenes, each of which has an image with reflection, corresponding transmission image, and reflection image. We captured these images with a Samsung Galaxy Note10 mounted on a tripod and 50 × 50 cm, 3-5mm thick glass with a holder to shot w/ or w/o the glass to create a reflection. The dataset can be download via the link: https://reurl.cc/xgG2VE.

DOI:10.6814/NCCU202101688

1.3. Thesis Structure

This thesis contains the following four sections:

1. Chapter 1 briefly introduce our research background, motivation and discusses its challenge.
2. Chapter 2 reviews related works and the proposed distilling approach.
3. Chapter 3 introduces our proposed method, model architecture, and dataset collection.
4. Chapter 4 briefly introduces the evaluation metric and then presents the experimental results and ablation study.
5. Chapter 5 concludes this thesis.

DOI:10.6814/NCCU202101688

2. RELATED WORKS

This section briefly reviews the relevant state-of-the-art methods on the single image reflection removal problem(SIRR) and the development of the knowledge distillation method.

2.1. Single Image Reflection Removal

To better understand and analyze SIRR, one can model an image I with reflection as $I = I_T \oplus I_R$, where I_T is the scene transmission image, and I_R is the reflection image [1]. The input image I can be decomposed into two layers: the transmission and reflection layers, as shown in Figure 1. Removing reflection from an image means separating the reflection layer from the input image and then dropping it. It involves two main difficulties: identifying the reflection layer and revealing the scene transmission.

Figure 1 Illustration of Single Image Reflection Removal. The first row represents the transmission layer; the second row represents image I, the last row represent the reflection layer.

Deep-neural-network-based methods have shown their potential to SIRR [2], [4]–[8], [14], [16]–[18] in these years. Fan et al. [14] proposed a two-stage model that first

DOI:10.6814/NCCU202101688

computes the gradient edge map (i.e., supervision) based on the input image and then uses it to remove the reflection layer in the second stage, the network architecture, as shown in Figure 2.

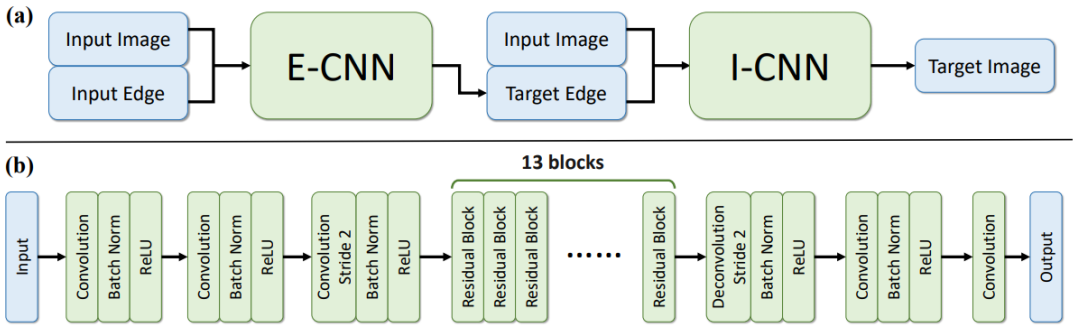

Figure 2 The CEIL network architecture.

Lee et al. [16] proposed to use adversarial learning to predict both the transmitted and reflected scenes from the input image. However, it often suffers from the unstable visual quality and training difficulty of generative adversarial networks (GANs). Zhang et al. [5] adopted an exclusion loss that effectively enforces the separation of transmission and reflection at the pixel level. Figure 3 shows the edge of the transmission image and reflection image in a gradient domain.

Figure 3 An illustration of the different edges in the gradient domain.

Sun et al. [7] exploited depth images captured by infrared sensors to remove image

DOI:10.6814/NCCU202101688

reflection. Li et al. [8] proposed a two-stage model, which produces an initial background image based on the balance between background preservation and elimination. The second uses a generative adversarial network to reconstruct the background's background gradients. Wei et al. [2] utilized alignment-invariant loss based on high-level features extracted using a pre-trained VGG model and adversarial loss to measure similarity even for unaligned training image pairs. Yang et al. [4] presented bi-directional cascaded networks that alternately generate the transmission and reflection results. Li et al. [18] adopted a cascaded model with LSTM to iteratively refine the results. Kim et al. [6] proposed to use RGBD/RGB images to synthesize more realistic reflection images for training. Chang et al. [17] introduced an auxiliary extension model consisting of edge guidance, reflection classifier, and recurrent decomposition networks to separate an image into reflection and transmission layers.

2.2. Knowledge Distillation

Nowadays, much research aims to increase the model's speed and accuracy, making knowledge distillation one of the most popular techniques. Hinton et al. [20] introduced a knowledge distillation concept to deep learning, proposing a teacher-student learning scheme using network mimicking to achieve knowledge transfer between the Teacher and the Student networks, where the Teacher is large but slow whereas the Student is small but fast. The Student network distills the Teacher network's knowledge by approximating the soft output of the Teacher. Romero et al. [21] proposed a feature mimicking method to fit the Student network's representations to the Teacher. Feature mimicking knowledge distillation has been applied to several computer vision tasks, such as image segmentation [22] and object detection [23]. Unlike these works, the proposed model aims to disentangle image content into the transmission and reflection layers to remove reflection from the input image and attain a clean image.

DOI:10.6814/NCCU202101688

DOI:10.6814/NCCU202101688

3. METHODOLOGY

3.1 Network Architecture

The proposed model contains a Reflection Teacher Network T_R and a Content Disentangling Student network S. Figure 4 shows the overall model architecture. In general, having a Reflection Teacher would be equivalent to a Transmission Teacher; however, since the reflection layer generally has fewer features, easier to be learned by a teacher network. We choose to exploit a Reflection Teacher network to distill features from the reflection image I_R, assuming we have the input image I and its transmission I_T (i.e. $I_R = I - I_T$) The Student network decomposes the input image into the transmission and reflection layers by gradually separating the reflection features extracted by the Reflection Teacher. In other words, our model disentangles the content of both layers by mimicking reflection features extracted from the Teacher network. The common setting used for our Teacher and Student Networks is that all the convolutional layers use 3 × 3 kernels with the stride of 1, followed by a ReLU activation function, except the last layer uses a Tanh.

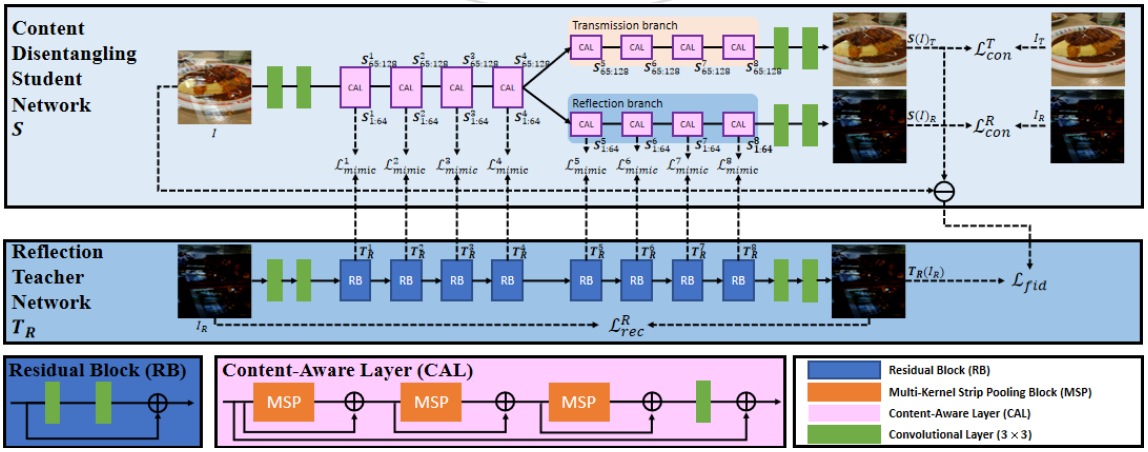

Figure 4 Overview of the Our network architecture.

DOI:10.6814/NCCU202101688

Reflection Teacher Network (T_R)

It starts with a pair of convolutional layers, then eight Residual Blocks (RB), and ends with another pair of convolutional layers. An RB consists of two convolutional layers with a residual connection between the input and output. We utilize the intermediate features extracted from each RB to assist the Student network in learning the transmission and reflection components' distribution. All the convolutional layers have 64 channels except for the input and output layers.

Content Disentangling Student Network (S)

Takes an image with reflection as the input and has two split outputs: one for the transmission image and the other for the reflection image denoted as $\mathrm{S(I)}_T$ and $\mathrm{S(I)}_R$ respectively. It starts with two convolutional layers and ends with two convolutional layers for each output. Based on our observation that a reflected scene is usually non-uniform and locally appears, the network aims to learn locally regional information for the reflected scene from the Reflection Teacher and delaminate from the transmitted scene. Motivated by [19], where Multikernel Strip pooling (MSP) was proposed to achieve region-based attention for disentangling different feature patterns, we construct the Student network's backbone with multiple MSP-based Content-aware Layers (CAL). A CAL that consists of three MSP blocks with residual connections can attend to reflection-affected or transmission-dominated regions with different degrees. In the network, there are twelve CALs in total. The first four CALs preliminarily decompose the reflection and transmission parts. Then, we divide it into the transmission and reflection branches for the following layers, each of which has another four CALs to disentangle different contents progressively and effectively. Figure 5 shows a content disentangling example, where transmission and reflection layers are gradually

DOI:10.6814/NCCU202101688

separated.

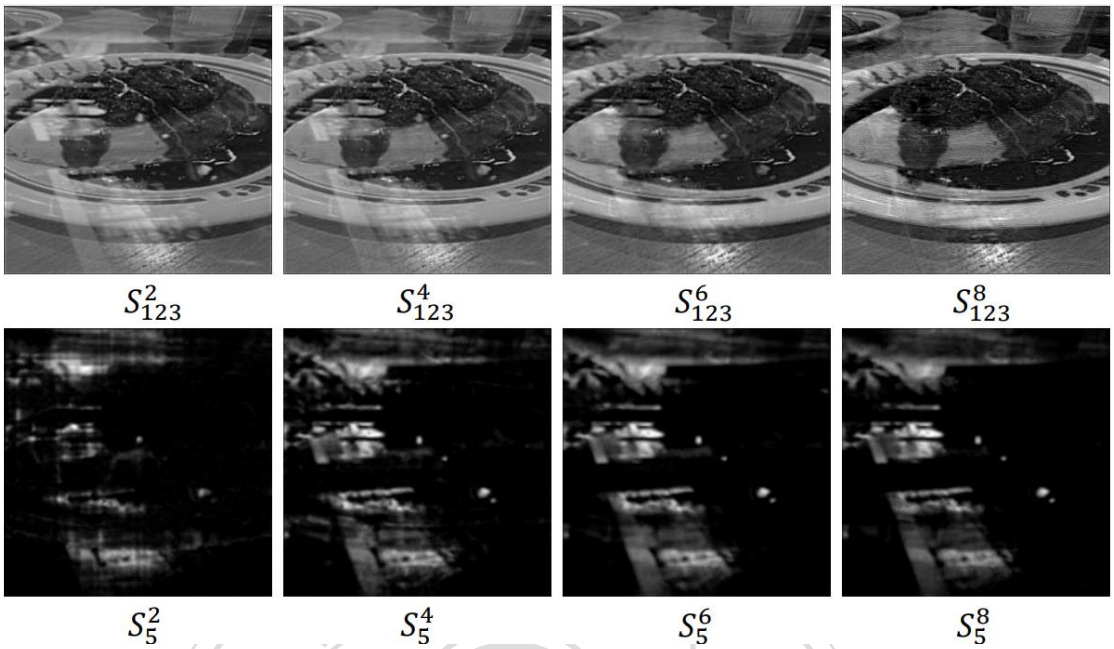

Figure 5 A content disentangling example using the Student Network S. We sample output transmission and reflection features (The 123rd and 5th channels) from the 2nd, 4th, 6th, and 8th layers to demonstrate disentangled features.

3.2 Loss functions

For Teacher — Reconstruction Loss:

To extract representative features from the Teacher network, we adopt an $L1$ loss to enforce the similarity between the input image and output image. For the Reflection Teacher, the reconstruction loss is

$$\mathrm{L}_{rec}^{R} = |I_R - T_R(I_R)|_1.$$

For Student — Content Disentangling Loss:

To disentangle transmission and reflection contents, we devise a content disentangling loss as

$$\mathrm{L}_{\mathrm{con}} = \mathrm{L}_{\mathrm{con}}^{\mathrm{T}} + \mathrm{L}_{\mathrm{con}}^{\mathrm{R}}$$

DOI:10.6814/NCCU202101688

, where $L_{con}^{T} = |I_T - S(I)_T|_1$ and $L_{con}^{R} = |I_R - S(I)_R|_1$. $S(I)_T$ and $S(I)_R$ represent the output images generated by the Student network with the input image I (i.e., an image with reflection).

For Student --- Representation Mimicking Loss:

We enforce the Student network to mimic the Teacher networks' intermediate features for content disentanglement to achieve knowledge transfer. Each of the first four CALs has 128 channels, the first 64 of which correspond to the 64 channels of one RB from the Reflection Teacher. Following are four CALs in the transmission branch and four CALs in the reflection branch, as shown in Figure 4. All the CALs have 64 channels in these branches, but only CALs in the reflection branch correspond to RBs in the Reflection Teacher. The mimicking loss is denoted as

$$L_{mim}^{R} = \sum_{n=1}^{8} L_{mimic}^{n} \sum_{n=1}^{8} |T_R{}^{n}(I_R) - S_{1:64}^{n}(I)|_1,$$

where $T_R{}^{n}(I_R)$ denotes the extracted features of the n^{th} RBs of T_R. $S_{c_1:c_2}^{n}(I)$ denotes the features from the c_1^{th} to c_2^{th} channel of the n^{th} CAL.

For both Teacher and Student — Fidelity Loss:

To reinforce the relationship between the input image I, transmission image I_T, and reflection image I_R, we introduce the fidelity loss based on $I = I_T + I_R$ as $L_{fid}^{R} = |I - S(I)_T - T_R(I_R)|_1$.

Total loss function:

At last, combining the Reconstruction Loss, Content Disentangling Loss, Representation Mimicking Loss, and Fidelity Loss, the total loss is written as:

$$L = \lambda_1 L_{rec}^{R} + \lambda_2 L_{con} + \lambda_3 L_{mim}^{R} + \lambda_4 L_{fid}^{R},$$

DOI:10.6814/NCCU202101688

where $\lambda_1 = \lambda_2 = 1\ and\ \lambda_3 = \lambda_4 = 0.3$. Minimizing it, we can train the proposed knowledge-distilling-based content-disentangling model for SIRR effectively.

DOI:10.6814/NCCU202101688

4. Dataset

4.1. Synthetic data

The training of the model requires a large of training data, but it is difficult to obtain a large amount of training dataset in the real world for the single image reflection removal task since it requires a lot of time and workforce and is limited to the impact of weather and environment.

For most research, PASCAL VOC dataset[24] and Flickr dataset are used to generate synthetic data. For the single image reflection removal task, two images are randomly taken as the transmission layer and the reflection layer. To imitate the reflection situation in the real world, usually applying a Gaussian smoothing kernel with a random kernel size to blur the reflected scene of the blend image, the following Figure 6 shows the result of the synthesis.

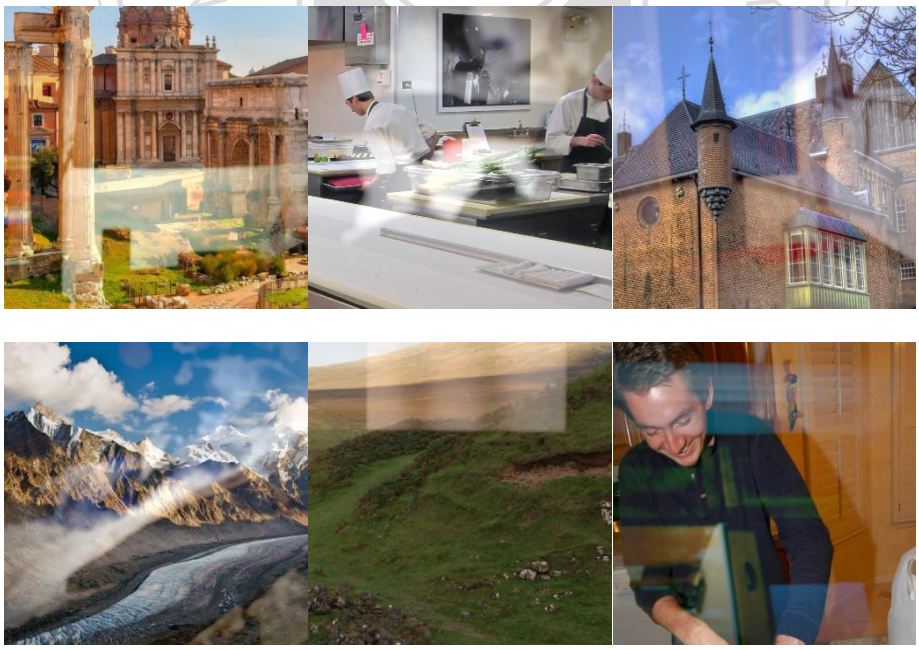

Figure 6 The example of synthetic data.

DOI:10.6814/NCCU202101688

As mentioned earlier, the following Table 1 shows synthetic data used in the current state-of-the-art papers.

DOI:10.6814/NCCU202101688

Model	Dataset	Released
ERRNET	Synthetic data from PASCAL VOC dataset	O
Physical	Synthetic data	X
Zhang	Synthetic data from Flickr dataset	O
IBCLN	Synthetic data from PASCAL VOC dataset	O
BDN	Synthetic data from PASCAL VOC dataset	X
Ours	Synthetic data from Flickr dataset	X

Table 1 Comparison of training datasets used by different methods.

4.2. Real-world data

The current well-known real datasets are introduced as follows:

- SIRR dataset

  As shown in Figure 7, the famous benchmark dataset with 454 pairs and a great diversity of mixture images. Each pair has a transmission image and its reflection image. These pairs can be roughly classified into three categories: wild scene, postcard, and solid.

DOI:10.6814/NCCU202101688

Figure 7 The sample of SIRR dataset: The left column are reflection images, and the right column are transmission images.

- UC Berkeley dataset

  As shown in Figure 8, the dataset has 110 real image pairs: the reflection layer and its corresponding ground-truth transmission layer. One often uses 20 of them as training data and the rest as testing data.

DOI:10.6814/NCCU202101688

Figure 8 The sample of UC Berkeley dataset: The left column is reflection images, and the right one is transmission images.

- CEILNet dataset

  This dataset provides 45 real-world images.

DOI:10.6814/NCCU202101688

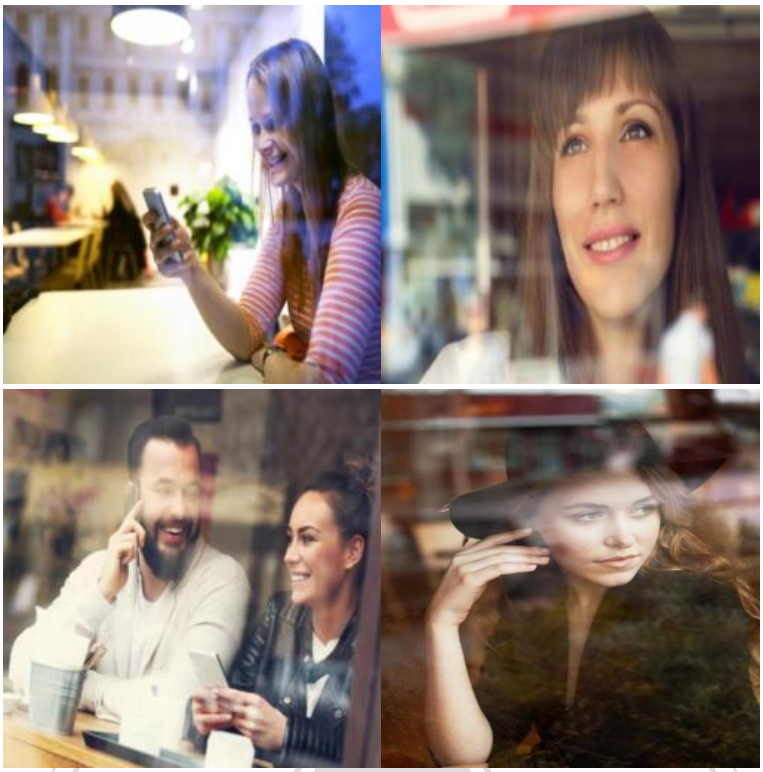

Figure 9 The sample of CEILNet dataset.

4.3. Our data collection

Due to insufficient real-world image datasets, we decided to collect real-world image datasets to evaluate our model. We use the equipment as follows to capture reflection images, shown in Figure 10.

Figure 10 An illustration of our equipment and device.

- Equipment
  - ■ Samsung Note 10+

    For all the images, we always use Samsung Note 10+ to take pictures.
  - ■ Tripod

DOI:10.6814/NCCU202101688

A tripod is used to prevent the phone from shaking and avoid data misalignment.

- Mirror

  We use the mirror and glass to capture the background side.

- Glass

  We use the mirror to capture the reflection layer.

- Fixed plate

  To ensure that the mirror does not shake, we use a fixed plate to stabilize the glass.

- Bluetooth receiver

  When we use a mobile device to shoot images, we must use a Bluetooth receiver to take photos as much as possible to avoid direct contact with the equipment to avoid shaking.

- Location

  To increase the diversity of the dataset, we select several different types of locations, including Gymnasium & General Bldg. of Colleges, Da Yong Bldg, Administrative Bldg, Bookstore, Co-op. All the places are located at National Chengchi University, Taipei, Taiwan.

- Process

  1. Place the fixing plate on the chair.
  2. Set up the tripod and fix the phone on it.
  3. Use the Bluetooth receiver to take reflection layer image *I*.
  4. Place the mirror behind the glass fixing plate and make sure the glass does not shake.
  5. Use the Bluetooth receiver to take the background image *B*.
  6. Remove the glass and mirror, and take the transmission layer image *T*.

DOI:10.6814/NCCU202101688

The data collection sequence is shown in Figure 11.

Figure 11 left: image with reflection *I*, middle: reflection image *R*, right: transmission image *T*.

- The Natural Reflection Dataset (NRD) dataset contains 136 real image triplets with various scenes, each of which has an image with reflection *I*, corresponding transmission image *T*, and reflection image *R*. (*Note that we do not use the reflection image *R* in this work)
- Challenges

  The lack of real-world datasets for a single image reflection removal motivates us to collect more data. To increase the diversity of the dataset, we must change different scenes and shooting locations many times. Each set of data must be in the same situation to shoot. Therefore, it is a big challenge for us to carry heavy equipment and electronic devices to move. We must be cautious to observe whether the device shakes and moving objects (such as people, weaving leaves) when shooting. We hope our collected dataset to facilitate research in this field further.

DOI:10.6814/NCCU202101688

5. EXPERIMENTAL RESULTS

5.1. Dataset and environment detail

Our training dataset, chosen as in [5], contains 200 real image pairs from ICBLN [18], 13, 697 synthetic pairs from Flickr, and 398 real pairs from ERRNet [2]. The test datasets have 590 images in total, including SIRR (Postcard, Object, Wild) [1], and our Natural Reflection Dataset (NRD). We implemented our model with PyTorch library on an Intel(R) Xeon(R) Gold 6154 CPU @ 3.00GHz and NVIDIA Tesla V100 GPU. Our parameter size for the Student network is 18M, and the model size is 73MB. We train our model for 120 epochs with a batch size of 8 using Adam optimizer with parameters $\beta 1 = 0.9$ and $\beta 2 = 0.999$ and a fixed learning rate of 10^{-4}. We compare our proposed model against five state-of-the-art SIRR methods, including Zhang et al. [5], BDN [4], ERRNet [2], Physical [6], and IBCLN [18].

5.2. Evaluation metrics

- **Peak Signal-to-Noise Ratio as an Image Quality (PSNR)**

  PSNR is an expression for the ratio between the maximum possible value (power) of a signal and the power of distorting noise that affects the quality of its representation.

  The mathematical representation of the PSNR is as follows:

$$\text{PSNR} = 20log_{10}(\frac{MAX_f}{\sqrt{MSE}})$$

where the MSE (Mean Squared Error) is:

DOI:10.6814/NCCU202101688

$$\mathrm{MSE} = \frac{1}{mn}\sum_{0}^{m-1}\sum_{0}^{n-1}\|f(i,j) - g(i,j)\|^2$$

- **The structural similarity index measure (SSIM)**

  SSIM is a method for predicting the perceived quality of digital television, cinematic pictures, and other digital images and videos. SSIM is used for measuring the similarity between two images.

  SSIM is a perception-based model that considers image degradation as perceived change in structural information while also incorporating important perceptual phenomena, including luminance masking and contrast masking terms.

  The mathematical representation of the SSIM is as follows:

$$\mathrm{SSIM}(\mathrm{x},\mathrm{y}) = \frac{(2\mu_x\mu_y + c_1)(2\sigma_{xy} + c_2)}{(\mu_x^2 + \mu_y^2 + c_1)(\sigma_x^2 + \sigma_y^2 + c_2)}$$

- **Learned Perceptual Image Patch Similarity (LPIPS) [25]**

  Today's most widely used perceptual metrics, such as PSNR and SSIM, are simple, shallow functions and fail to account for many nuances of human perception. The LPIPS metric evaluates the distance between image patches to calculate similarity. Higher means further/more different. A lower value means the compared images are more similar perceptually.

5.3. Quantitative comparison

We conduct experiments to evaluate SIRR performance with three major metrics: PSNR, SSIM, and Learned Perceptual Image Patch Similarity (LPIPS [25]). PSNR

DOI:10.6814/NCCU202101688

represents pixel similarity, SSIM measures luminance, contrast, and structure similarity, and LPIPS exploits deep features to evaluate a perceptual similarity. Table 2 shows the results, where our method works best in PSNR, SSIM, and LPIPS on average. It indicates that our knowledge-distilling model can disentangle the transmission layer from the reflection layer in the input image to keep better structural fidelity and learn perceptual features from clean images to present more consistent visual and perceptual similarity.

Dataset	Index	Methods					
		Zhang	BDN	ERRNET	Physical	IBCLN	Ours
SIRR-Postcard (199)	SSIM↑	0.68	0.84	0.87	0.88	0.85	**0.89**
	PSNR↑	16.24	20.92	22.29	22.93	21.97	**23.29**
	LPIPS↓	0.1218	0.1077	0.1177	0.0871	0.1157	**0.0855**
SIRR-Object (200)	SSIM↑	0.79	0.85	**0.86**	0.85	**0.86**	**0.86**
	PSNR↑	23.30	22.79	24.41	23.70	**24.46**	23.92
	LPIPS↓	0.0673	0.0653	0.0591	0.0652	0.0582	**0.0550**
SIRR-Wild (55)	SSIM↑	0.84	0.84	0.86	0.87	0.87	**0.91**
	PSNR↑	22.73	22.14	25.25	**25.58**	25.26	25.12
	LPIPS↓	0.1503	0.2711	0.1228	0.1168	0.1259	**0.0964**
NRD (136)	SSIM↑	0.79	0.79	0.80	0.81	0.80	**0.83**
	PSNR↑	21.41	19.53	22.40	**23.03**	22.31	23.01
	LPIPS↓	0.1381	0.1359	0.1186	0.1224	0.1246	**0.1186**
Average (590)	SSIM↑	0.76	0.83	0.85	0.85	0.84	**0.87**
	PSNR↑	20.43	21.35	23.31	23.46	23.20	**23.61**
	LPIPS↓	0.1097	0.1151	0.0985	0.0905	0.0992	**0.0838**

DOI:10.6814/NCCU202101688

Table 2 Objective quality comparisons. The best score is in bold, and the second-best is underlined. In the parentheses is the number of images in a dataset.

5.4. Ablation study

Table 3 shows an ablation study on different loss combinations (numbered from C1-C5). Comparing C1 to C2-5, having Teacher(s) works better than without it. C2 and C3 show a performance boost by adding the fidelity loss. C3 and C4 indicate training with the Reflection Teacher is better than the Transmission Teacher. Contrasting C3 to C5 shows having two teachers would not be more beneficial to only the Reflection Teacher.

#	T_R	T_T	Loss Functions	SSIM↑	PSNR↑	LPIPS↓
C1			$\mathcal{L}_{con}$(no teacher)	0.84	22.89	0.1233
C2	✓		$\mathcal{L}_{con} + \mathcal{L}_{rec}^{R} + \mathcal{L}_{min}^{R}$	0.86	23.16	0.0900
C3	✓		$\mathcal{L}_{con} + \mathcal{L}_{rec}^{R} + \mathcal{L}_{min}^{R} + \mathcal{L}_{fid}^{R}$	**0.87**	**23.61**	**0.0838**
C4		✓	$\mathcal{L}_{con} + \mathcal{L}_{rec}^{T} + \mathcal{L}_{min}^{T} + \mathcal{L}_{fid}^{T}$	0.85	22.81	0.1050
C5	✓	✓	$\mathcal{L}_{con} + \mathcal{L}_{rec}^{TR} + \mathcal{L}_{min}^{TR} + \mathcal{L}_{fid}^{TR}$	**0.87**	23.33	0.0847

Table 3 The ablation study table.

5.5. Qualitative results

To visually compare SIRR results obtained using our and SOTA methods, Figure 12-Figure 15 demonstrate our model performs favorably against the other methods in transmission restoration. The first two images are synthesized, and the last two

DOI:10.6814/NCCU202101688

images are real images.

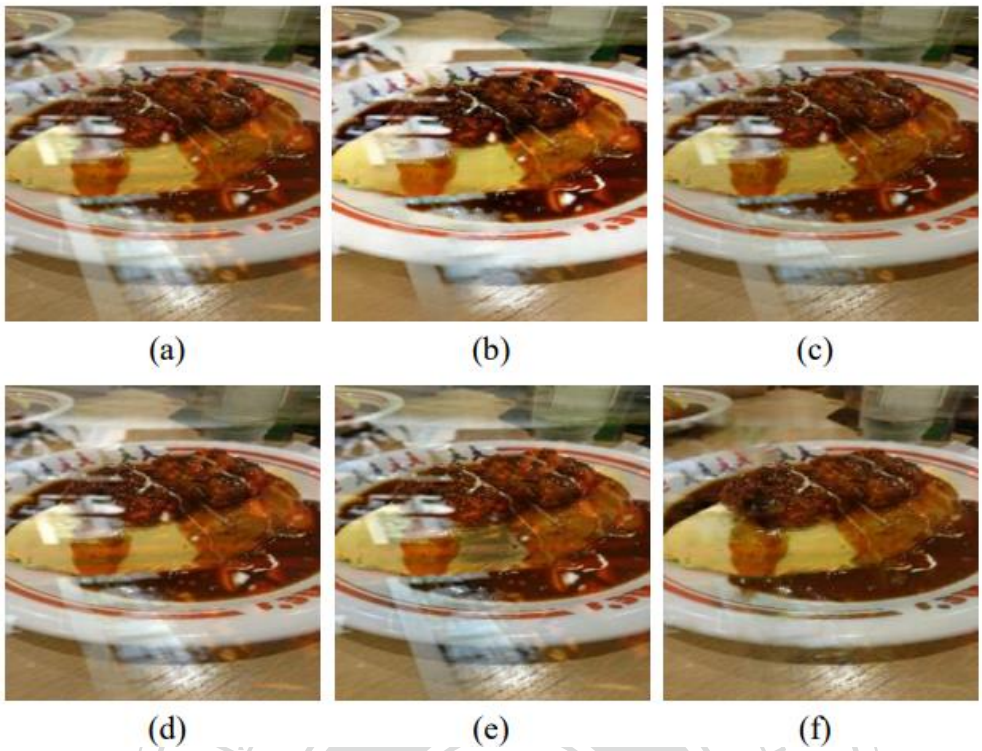

Figure 12 A visual comparison of SIRR results. (a) Input. The results generated by (b) BDN [4], (c) ERRNet [2], (d) Physical [6], (e) IBCLN [18], and (f) ours.

DOI:10.6814/NCCU202101688

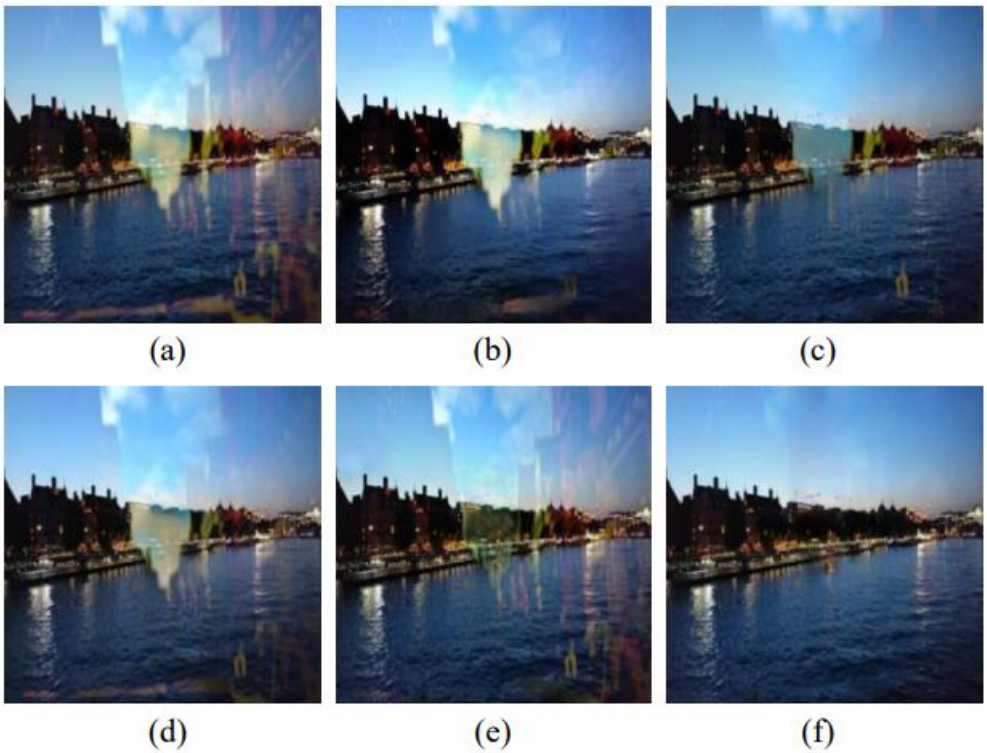

Figure 13 (a) Input. More SIRR results generated by (b) BDN [4], (c) ERRNet [2], (d) Physical [6], (e) IBCLN [18], and (f) ours.

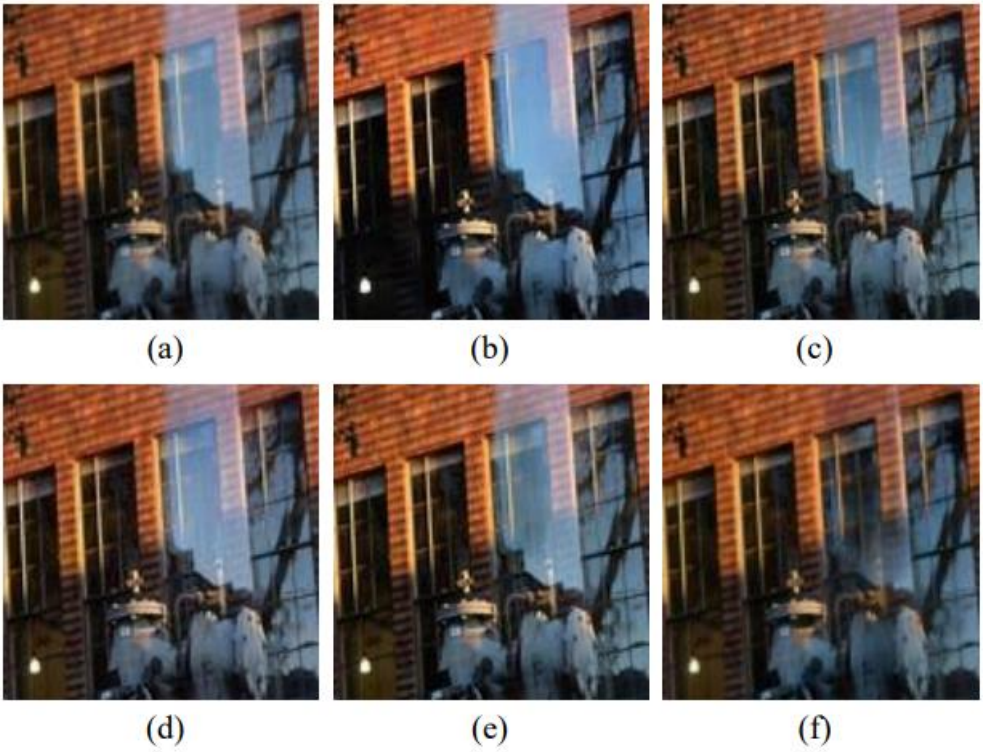

Figure 14 (a) Input. More SIRR results produced by (b) BDN [4], (c) ERRNet [2], (d) Physical [6], (e) IBCLN [18], and (f) ours.

DOI:10.6814/NCCU202101688

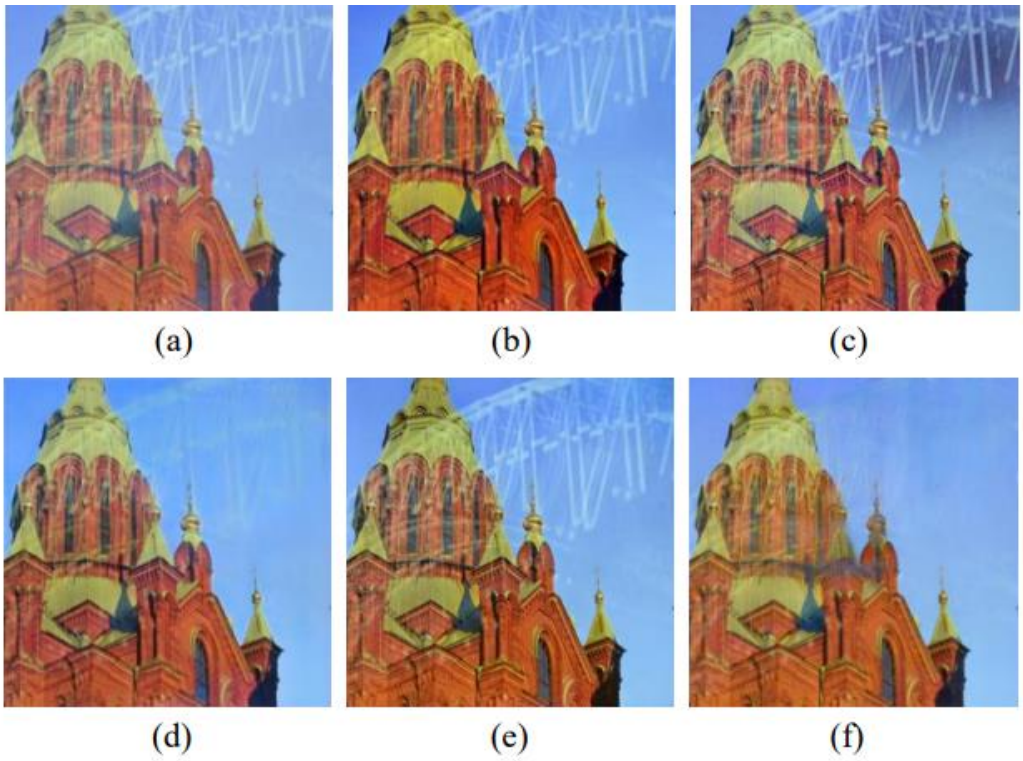

Figure 15 (a) Input. More SIRR results produced by (b) BDN [4], (c) ERRNet [2], (d) Physical [6], (e) IBCLN [18], and (f) ours.

DOI:10.6814/NCCU202101688

6. CONCLUSIONS

In this work, we proposed to use knowledge distillation and feature mimicking to disentangle image content and achieved better performance in single image reflection removal (SIRR). The experiments on benchmark SIRR datasets indicated that our model could produce SIRR results with better perceptual quality and structural fidelity. In addition, we contribute the Natural Reflection Dataset (NRD) to the SIRR field and expect knowledge-distilling-based content disentanglement to inspire more related work.

DOI:10.6814/NCCU202101688

REFERENCES

[1] Renjie Wan, Boxin Shi, Ling-Yu Duan, Ah-Hwee Tan, and Alex C Kot, "Benchmarking single-image reflection removal algorithms," in Int. Conf. Comput. Vis., 2017.

[2] Kaixuan Wei, Jiaolong Yang, Ying Fu, David Wipf, and Hua Huang, "Single image reflection removal exploiting misaligned training data and network enhancements," in IEEE Conf. Comput. Vis. Pattern Recog. , 2019.

[3] Patrick Wieschollek, Orazio Gallo, Jinwei Gu, and Jan Kautz, "Separating reflection and transmission images in the wild," in Eur. Conf. Comput. Vis., 2018, pp. 89–104.

[4] Jie Yang, Dong Gong, Lingqiao Liu, and Qinfeng Shi, "Seeing deeply and bidirectionally: A deep learning approach for single image reflection removal," in Eur. Conf. Comput. Vis., 2018.

[5] Xuaner Zhang, Ren Ng, and Qifeng Chen, "Single image reflection separation with perceptual losses," in IEEE Conf. Comput. Vis. Pattern Recog., 2018.

[6] Soomin Kim, Yuchi Huo, and Sung-Eui Yoon, "Single image reflection removal with physically-based training images," in IEEE Conf. Comput. Vis. Pattern Recog., 2020.

[7] Jun Sun, Yakun Chang, Cheolkon Jung, and Jiawei Feng, "Multi-modal reflection removal using convolutional neural networks," IEEE Sign. Process. Letters, 2019.

[8] Tingtian Li and Daniel P. K. Lun, "Single-image reflection removal via a two-stage background recovery process," IEEE Sign. Process. Letters , 2019.

[9] Nikolaos Arvanitopoulos, Radhakrishna Achanta, and Sabine Susstrunk, "Single image reflection suppression," in IEEE Conf. Comput. Vis. Pattern Recog., 2017.

[10] Yu Li and Michael S Brown, "Single image layer separation using relative smoothness," in IEEE Conf. Comput. Vis. Pattern Recog., 2014.

DOI:10.6814/NCCU202101688

[11] Renjie Wan, Boxin Shi, Tan Ah Hwee, and Alex C Kot, "Depth of field guided reflection removal," in IEEE Int. Conf. Image Process. IEEE, 2016.

[12] Xiaojie Guo, Xiaochun Cao, and Yi Ma, "Robust separation of reflection from multiple images," in IEEE Conf. Comput. Vis. Pattern Recog. , 2014.

[13] Richard Szeliski, Shai Avidan, and Padmanabhan Anandan, "Layer extraction from multiple images containing reflections and transparency," in IEEE Conf. Comput. Vis. Pattern Recog., 2000. [14] Qingnan Fan, Jiaolong Yang, Gang Hua, Baoquan Chen, and David Wipf, "A generic deep architecture for single image reflection removal and image smoothing," in Proceedings of the IEEE International Conference on Computer Vision, 2017, pp. 3238–3247.

[15] Renjie Wan, Boxin Shi, Ling-Yu Duan, Ah-Hwee Tan, and Alex C. Kot, "Crrn: Multi-scale guided concurrent reflection removal network," in IEEE Conf. Comput. Vis. Pattern Recog., 2018.

[16] Donghoon Lee, Ming-Hsuan Yang, and Songhwai Oh, "Generative single image reflection separation," arXiv preprint arXiv:1801.04102 , 2018.

[17] Ya-Chu Chang, Chia-Ni Lu, Chia-Chi Cheng, and Wei-Chen Chiu, "Single image reflection removal with edge guidance, reflection classifier, and recurrent decomposition," in IEEE Winter Conference on Applications of Computer Vision (WACV), 2021.

[18] Chao Li, Yixiao Yang, Kun He, Stephen Lin, and John E Hopcroft, "Single image reflection removal through cascaded refinement," in IEEE Conf. Comput. Vis. Pattern Recog., 2020.

[19] Fu-Jen Tsai, Yan-Tsung Peng, Yen-Yu Lin, Chung-Chi Tsai, and Chia Wen Lin, "Banet: Blur-aware attention networks for dynamic scene deblurring," arXiv preprint arXiv:2101.07518, 2021.

[20] Geoffrey Hinton, Oriol Vinyals, and Jeffrey Dean, "Distilling the knowledge in a

DOI:10.6814/NCCU202101688

neural network," in NIPS Deep Learning and Representation Learning Workshop, 2015.

[21] Adriana Romero, Nicolas Ballas, Samira Ebrahimi Kahou, Antoine Chassang, Carlo Gatta, and Y. Bengio, "Fitnets: Hints for thin deep nets," in Int. Conf. Learn. Represent., 2015.

[22] Yifan Liu, Ke Chen, Chris Liu, Zengchang Qin, Zhenbo Luo, and Jingdong Wang, "Structured knowledge distillation for semantic segmentation," in IEEE Conf. Comput. Vis. Pattern Recog., 2019. [23] Tao Wang, Li Yuan, Xiaopeng Zhang, and Jiashi Feng, "Distilling object detectors with fine-grained feature imitation," in IEEE Conf. Comput. Vis. Pattern Recog., 2019.

[24] Everingham, M., Eslami, S. M. A., Van Gool, L., Williams, C. K. I., Winn, J. and Zisserman, A. *International Journal of Computer Vision, 111(1), 98-136, 2015*

[25] Richard Zhang, Phillip Isola, Alexei A Efros, Eli Shechtman, and Oliver Wang, "The unreasonable effectiveness of deep features as a perceptual metric," in IEEE Conf. Comput. Vis. Pattern Recog., 2018.

DOI:10.6814/NCCU202101688