Optimization-Based Data Generation for Photo Enhancement

Mayu Omiya *
Waseda University
omiya_mayu@akane.waseda.jp

Yusuke Horiuchi *
Waseda University
y.horiuchi@suou.waseda.jp

Edgar Simo-Serra
Waseda University
ess@waseda.jp

Satoshi Iizuka
University of Tsukuba
iizuka@cs.tsukuba.ac.jp

Hiroshi Ishikawa
Waseda University
hfs@waseda.jp

Abstract

The preparation of large amounts of high-quality training data has always been the bottleneck for the performance of supervised learning methods. It is especially time-consuming for complicated tasks such as photo enhancement. A recent approach to ease data annotation creates realistic training data automatically with optimization. In this paper, we improve upon this approach by learning image-similarity which, in combination with a Covariance Matrix Adaptation optimization method, allows us to create higher quality training data for enhancing photos. We evaluate our approach on challenging real world photo-enhancement images by conducting a perceptual user study, which shows that its performance compares favorably with existing approaches.

1. Introduction

Deep learning has become the dominant technique in many computer vision tasks, with a special focus on supervised learning. However, current approaches require large amounts of high quality training data, which is not always readily available or easy to obtain. In this work, we propose an improvement to a recent approach, Black Box Model Optimization (BBMO) [20], which creates high-quality training data for photo enhancement tasks from readily available online images.

In [20], an automatic photo enhancement framework is proposed where a photo enhancement model is fixed and a set of parameters to the model defines a photo enhancement (Fig. 1.) A convolutional neural network (CNN) learns adaptive photo enhancement, outputing the enhancement parameters for given input photo. The photo enhancement model represents an off-the-shelf retouching software as a black box. Learning the parameters for the software, rather than the image transformation itself, has the advantage of making it easier to additionally edit the photo using the same software. To generate supervised training data for the CNN, the BBMO first obtains before-after image pairs of photo enhancement, and then estimate the enhancement parameters that realizes the change between the two, using optimization since the photo enhancement model is treated as a black box. The result is the training data that consists of pairs of an input image and a corresponding set of enhancement parameters that enhance the input.

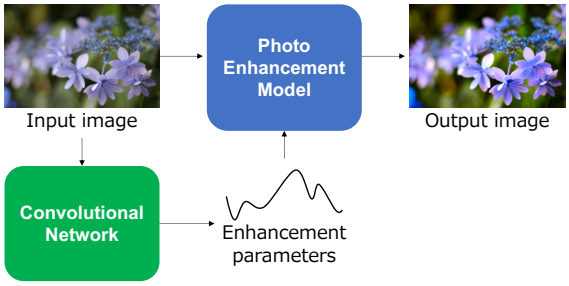


Figure 1. Overview of the automatic photo enhancement. The photo enhancement model represents an off-the-shelf photo retouch software. From an input image, a convolutional neural network estimates a set of enhancement parameters which can be used with the photo enhancement model to improve the image. In this work, we focus on how to automatically generate supervised training data, which consists of pairs of an image and a set of enhancement parameters, for this convolutional network. For that, it is necessary to estimate the parameters from input-output pairs of images.

While this approach shows good results, it has two drawbacks that hamper the performance: the optimization algorithm and the difficulty of determining when the optimiza-

* The authors assert equal contribution and joint first authorship.

tion process has found the parameters that sufficiently restore the image. In this work, we tackle both of these issues by using a Covariance Matrix Adaptation optimization approach and learning a photo similarity function to better determine whether the restored image is close enough to the original photo, which allows us to further prune the created data. Given that the quality of the training data significantly affects the results of the trained networks, our two improvements allow creating better training data, which in turn translates to higher performance of the photo enhancer.

Using our proposed improvements, we create a new training dataset for photo enhancement, and show how this data allows our trained automatic enhancer to outperform existing methods with fewer training images. We also perform a perceptual user study, which corroborates our results.

2. Related work

2.1. Annotated datasets

Deep learning, which requires a large amount of data, is a powerful method. As deep learning is used a lot, creating dataset for learning become more important. For tasks such as automatic coloring of black and white images using deep learning [10] and super resolution of images using deep learning [5, 6], if the data after conversion by neural network can be prepared, the input data for neural network can be easily prepared by using a simple algorithm. On the other hand, it is very difficult to prepare target data for tasks such as removal of reflections in images [1] and scene transformation of images [16]. These tasks require pairs of images taken so that moving objects in the image are completely the same, and it is necessary to completely control changes in camera position, fluids, human movements, etc. taking both of these pictures is very difficult.

Similarly, photo enhancement task needs a large amount of pair dataset of the input pre-enhance image and the output post-enhance images that people feel high-quality. For major tasks such as image classification, there are datasets with a large amount of images and annotations such as ImageNet [4], etc. However, when addressing individual problems as in this paper, it is rare that the dataset with sufficient amounts is prepared.

In the photo enhancement task, there is MIT-Adobe FiveK Dataset [2] . The FiveK dataset includes 5,000 images and images enhanced by five professionals for each image. The number of 5,000 is not always enough to use the deep learning method. Furthermore, expanding this dataset requires more images and a lot of expert annotations, making it difficult to create datasets.

2.2. Approach of generating datasets

There are methods using simulation for automatically generating dataset. Richter et al. [22] proposed automatically generating detailed semantic segmentation maps by using photorealistic video games. The simulation environment provided by the video game allows easily obtaining near-unlimited data, which all contains pixel-perfect semantic segmentation maps, and was used for training neural networks for autonomous vehicles. Park et al. [21] created a high-quality 3D model dataset, which then was used to create large amounts of synthetic renders. This synthesized data could then be used for training convolutional neural networks to perform semantic segmentation of object materials. However, in the task of photo enhancement, it is impossible to prepare paired images dataset using simulation, because the judgement of image quality is subjective and unclear.

There are also methods of data collection using the Internet. With the development of the Internet, many datasets are including images collected from the Internet. ImageNet [4], which is often used for image classification, has some images obtained by crawling from the Internet. DeepFashion [18], a dataset used in clothing classification, uses images obtained from shopping sites. Since making available the data collected on the Internet is considered to be a very effective means, our proposed method also uses the image data obtained from the photo sharing SNS called Flickr.

2.3. Photo Enhancement

Automatic photo enhancement has been studied for many years. Lischinski et al. [17] proposed an interactive system. In this system, users can easily enhance the image by specifying the area to be edited and the adjusted color in the image. Kang et al. [12] proposed a pipeline to suggest the enhancement parameters by analyzing user's past enhanced photos. Koyama et al. [14] created a software for users to make corrections, and proposed a system that learns each time a user enhancement a picture and learns a enhancement of a picture tailored to the user. These personalizations require some degree of self-correcting photos.

Efforts have also been made to make corrections that understand the content of the picture. Kaufman [13] has proposed a system that recognizes the position of the face and the sky, and determines the correction content. Yan et al. [28] proposed a system that performs semantic segmentation of images and decides the correction content from the judgment. These systems use deep learning for identification and have the weakness that they can not be corrected correctly if the identification fails.

Chen et al. [3] used a framework that used a generative adversarial network [7] to exchange the distribution of pre-enhanced and post-enhanced images. He used the GAN that using cycle-consistency loss [30]. This method do

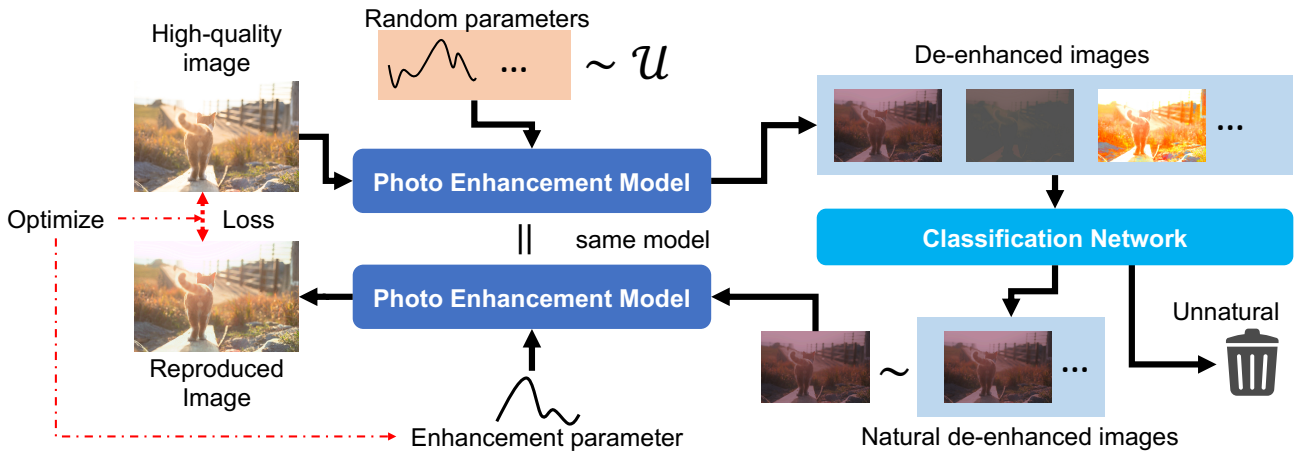


Figure 2. Overview of the data generation approach BBMO [20]. From high-quality images, de-enhanced images are generated using random parameters and a black-box photo enhancement model. Afterwards a classification network discards unrealistic de-enhanced images. Finally, optimization is used to find a set of parameters that, when used with the photo enhancement model, restore the de-enhanced image to the original high-quality image.

not require the pair-dataset pre-enhanced image and post-enhanced image. However, since the end-to-end deep learning image transformation framework based on UNet [23], pix2pix [11] etc. can not ensure what kind of processing is being performed inside, it may output inconvenient for the user.

A system with two elements is needed: understanding the content of the picture and proposing parameters that can be interpreted by the user. Omiya et al. [20] proposed a framework that treats the manual enhancement software itself as a Black-Box, and estimates the user's input from the image using a convolutional neural network. Efforts to estimate the user's input from the output image data to create a dataset for an interactive system have been carried out by Sangkloy et al. [24], Zhang et al. [29], Simo-Serra et al [25]. this is essentially different from creating user input in the Black-Box Model of Omiya et al. The work of Omiya et al. shows promising results, the optimization algorithm used, along with the simple criteria (based on RGB pixel differences) to determine when the data is suitable for training, limits its performance. In this work, we propose two improvements on this method and show that, with Covariance Matrix Adaptation optimization in combination with a learned image similarity function, we can significantly improve the data quality, which translates into higher photo enhancement performance.

3. Proposed Method

We propose two ways to improve the quality of the training data produced by BBMO [20]. The first is improving the optimization method used to generate training data from the high-quality images (§3.2). The quality of the training data crucially depends on the accuracy of optimization. Also, the optimization can be a highly non-convex problem, since we do not assume invertibility of the photo enhancement model. The second is better filtering the noisy data after the generation by introducing a data-driven image similarity function that is closer to human senses (§3.3).

3.1. Generating Training Data

We first review how the training data for the photo enhancement framework shown in Fig. 1 is generated by BBMO [20]. As explained above, in the framework a CNN learns to output the enhancement parameters for given input photo that best enhances it. The parameters are with respect to a fixed photo enhancement model, which represents an off-the-shelf photo retouch software. The photo enhancement model can be seen as a color conversion function that, given a set of enhancement parameters and an input image, outputs an enhanced image. Most professional retouch software has many different filters and operators which perform advanced non-local editing. Here, the software is treated as a black box and no hypothesis about it is made. In [20] and also in our experiment, we use a diverse set of functions from the open-source photo editing software Darktable[1]. Darktable is able to globally and locally enhance images, and we consider a particular subset of enhancement parameters—21 dimensions total—as our photo enhancement model, which we use to both generate training data and enhance photos. We use the same 21-dimensional enhancement parameters as BBMO [20], which are shown in Table 1.

For training the convolutional neural network, pairs of

[1]https://www.darktable.org/

(a) De-enhanced Image (b) Nelder-Mead (b) CMA-ES (d) Original Image

Figure 3. Comparison between the original high-quality images and the reconstructed images restored from de-enhanced images by different optimization methods.

Table 1. Initial values of the enhancement parameters with CMA-ES.

Parameter	Initial Value	Initial Std. Dev.
Exposure (Black, Color)	(0, 0)	(0.1, 0.75)
White Point	0	10
(Shadow, Highlight, Shadow Saturation, Highlight Saturation)	(50, -50, 100, 50)	(100, 87.5, 50, 50)
(Contrast, Lighting, Saturation)	(0, 0, 0.5)	(0.5, 0.625, 0.625)
Color Temperature (R, G, B)	(1, 1, 1)	(0.5, 0.5, 0.5)
Color Vibrance	25	50
Color Correction (Highlight XY, Shadow XY, Saturation)	(0, 0, 0, 0, 1)	(40, 40, 40, 40, 0.5)
Color Contrast (GM, BY)	(1, 1)	(0.2, 0.2)

a non-enhanced input image and the corresponding set of photo enhancement parameters that allows the photo enhancement model to enhance the input are required as training data. An overview of the process to generate the training data is shown in Fig. 2. The steps are as follows:

(1) First, random parameters are applied to the high-quality images to obtain de-enhanced images. Since the de-enhanced image has lower quality than the original high-quality image, it can be used as a substitute for a non-enhanced image.

(2) However, randomly de-enhanced images contain many unnatural images. They are filtered by a separate classification network. A de-enhanced image that is classified as natural is used as an image before enhancement.

The classification network consists of two VGG19 networks. The high quality image and the de-enhanced image is each convoluted by the two feature extractors of the two VGG19 with the last layer (fc8) replaced with a linear layer with 1024 hidden units. The 1024-dimensional feature maps from the two VGG19 are concatenated to form a 2048-dimensional vector and fed into two fully-connected layers with 512 and 1 unit respectively, which classify the de-enhanced image as natural or unnatural. There is a 50% dropout layer between the two fully-connected layers. All layers use ReLU activation function except the last layer which uses a Sigmoid function, and the model is trained with binary cross entropy.

(3) The enhancement parameters to convert the de-enhanced image back to the original high-quality image is estimated by optimizing the objective:

$$\arg\min_{\theta} \mid y^* - f(x, \theta) \mid^2 \quad , \tag{1}$$

where y^* is the original high-quality image, x is the de-enhanced image, $f(\cdot, \cdot)$ is the photo enhancement model, and θ is the set of enhancement parameters. The improvement to the optimization method is described in detail in §3.2.

(4) When the optimization is completed, whether or not the high-quality image has been successfully reproduced is checked. Specifically, the difference between the output image and the original high-quality image is evaluated by image similarity function. Only images with the similarity above threshold are used for the training data. In [20], the similarity is estimated by a simple MSE function. In this work, we improve the similarity function, which is explained in detail in §3.3.

3.2. Optimization using CMA-ES

In [20], the Nelder-Mead method [19] is used as the optimization method to obtain the enhancement parameters to convert the de-enhanced image back to the high-quality image. Although this method has the advantage of not requiring explicit gradient information, it has a disadvantage that it is easy to fall into a local solution. Therefore, here we adopt the Covariance Matrix Adaptation Evolution Strategy (CMA-ES) [9] method, which is robust against local solutions, as our optimization algorithm. CMA-ES is an improvement of conventional evolution strategy by using covariance matrix that has been observed to be capable of locating global optimal of many multimodal objective functions [8]. Like the Nelder-Mead method, it does not require explicit gradient information and can be used for black box optimization.

Table 2. Comparison of the difference between the high-quality images and the images restored from the de-enhanced images by each optimization method. The average values over 10 input high-quality images are shown, and the best results are shown in bold.

Method	L_1	MSE	SSIM
Nelder-Mead [19] (L_1)	0.0747	0.0110	0.8861
Nelder-Mead [19] (MSE)	0.0745	0.0105	0.8799
CMA-ES [9] (L_1)	0.0342	0.0031	0.9325
CMA-ES [9] (MSE)	**0.0317**	**0.0025**	**0.9328**

The CMA-ES optimization approach is based on using an evolutionary strategy based on sampling new candidate solution according to a multivariate normal distribution. At each iteration, the mean and covariance matrix of this distribution is updated according to the underlying objective function. In contrast to commonly used approaches, very few assumptions are made on the underlying objective function, which allows us to optimize enhancement parameters with with a black-box off-the-shelf photo enhancement model without having to worry about the nature of its implementation.

In Table 2, we provide an empirical comparison between the Nelder-Mead method and the CMA-ES method. As in the generation process explained above, high-quality images were first de-enhanced, and then each optimization method was used to estimate the enhancement parameters that convert the de-enhanced images back to the original images. As the objective function of these method, we tried two loss functions, the RGB L_1 loss and the RGB mean square error. How close the results were to the original images, in the average values of RGB L_1 loss, RGB mean square error (MSE) and SSIM [27] over 10 images are shown in the table. SSIM is used for image quality evaluation and values closer to 1 indicate better image quality. In both evaluation indices, CMA-ES method shows less difference from the original high-quality image than Nelder-Mead method. In this experiment, we use a population size of 10 and tolerance of 0.001 as CMA-ES method hyperparameters. Given that the different enhancement parameters being optimized have different ranges, we initialize them with the values and standard deviations shown in Table 1.

Qualitative examples are also shown in Fig. 3. With Nelder-Mead, the color of the background of the top image and the overall color of the bottom image is significantly different from the original high-quality image, while the CMA-ES results look very close to the original high-quality images as compared with the Nelder-Mead method.

3.3. Image Similarity Function

It is important to be able to detect when the optimization fails and the generated data is not suitable for train-

ing. If this poor data is not detected and thus not excluded from the training data, it will lead to lower quality photo enhancement. We use a data-driven approach to learn an image similarity function that can be used to determine if the optimization process has converged and the resulting data is suitable for training photo enhancement parameter estimation network.

There are several cases where optimization fails. First, if the de-enhanced image is too far from the high-quality image, optimization fails. In other cases, the objective function may not capture the crucial difference between two images well enough. For example, the color of the most noticeable object may be different, although most area of the two images are well reproduced. In this case, although the average difference in pixel value of the entire image is small, the appearance and photo impression differs greatly. Similar problem can often occur when the image is dull overall and the color saturation is low. In order to cope with such cases, we devised an similarity function closer to the human sense rather than mere pixel value difference.

In [20], thresholding mean square error between the high-quality image and the reconstructed image is used in the RGB color space, resulting in data not suitable for training. We instead consider a weighted combination of the mean square errors in RGB, Lab, and HSV color space, in addition to the one minus the SSIM, so that the total becomes 0 when the images are identical. Instead of relying on a purely heuristically determined similarity function, we opted for a data-driven approach, and obtained 15 pairs each of similar and dissimilar de-enhanced and high-quality images. This small set of 30 images is then used to determine an appropriate image similarity function. We first performed feature selection on the different values we compute with random forest regression. In particular, we train a regression model and then eliminate the all but the top k features. This was repeated for all possible number of features and the best performing number of features was used. In the end, the RGB, Lab, and Saturation pixel values, in combination with one minus the SSIM value were used. Figure 4 shows an example where the RGB MSE cannot detect the failure although the high-quality image is not reproduced well, while our similarity function detects it.

Also, we compared discrimination of the optimization results when using the RGB MSE and our image similarity function. Table 3 shows the results of thresholding for 60 sets of images annotated as to whether high-quality images could be reproduced. Since our aim is to remove data that failed to optimize, the image similarity function with higher specificity has better performance.

4. Experimental Results

We utilized the generated training data to train the CNN in the automatic photo enhancement framework [20] shown in Fig. 1. We obtained 3,224 training images using our method. Of these, 3,162 were used for training and the remaining 62 were used for validation while training the CNN. As with [20], we used a VGG19 model [26] pre-trained on ImageNet [15] fine-tuned for our task with SGD by replacing the last layer (fc8) to one that outputs the values of the 21 enhancement parameters.

(a) Original Image (b) Reproduced Image

Figure 4. An example of failed optimization. Although the original high-quality image (a) and the reproduced image (b) are very different, the RGB MSE cannot detect it, whereas our Image Similarity Function can.

Table 3. Comparison of discrimination of the optimization results when using RGB MSE and our image similarity function. The results of thresholding for 60 sets of images annotated as to whether high-quality images could be reproduced are shown.

	Accuracy	Specificity
RGB MSE	0.52	0.19
Proposed Image Similarity	**0.70**	**0.81**

We compared the automatic photo enhancer trained with our training data with

i) the same enhancer trained with the data generated by the Black Box Model Optimization (BBMO) [20],

ii) Deep Photo Enhancer (DPE) [3], and

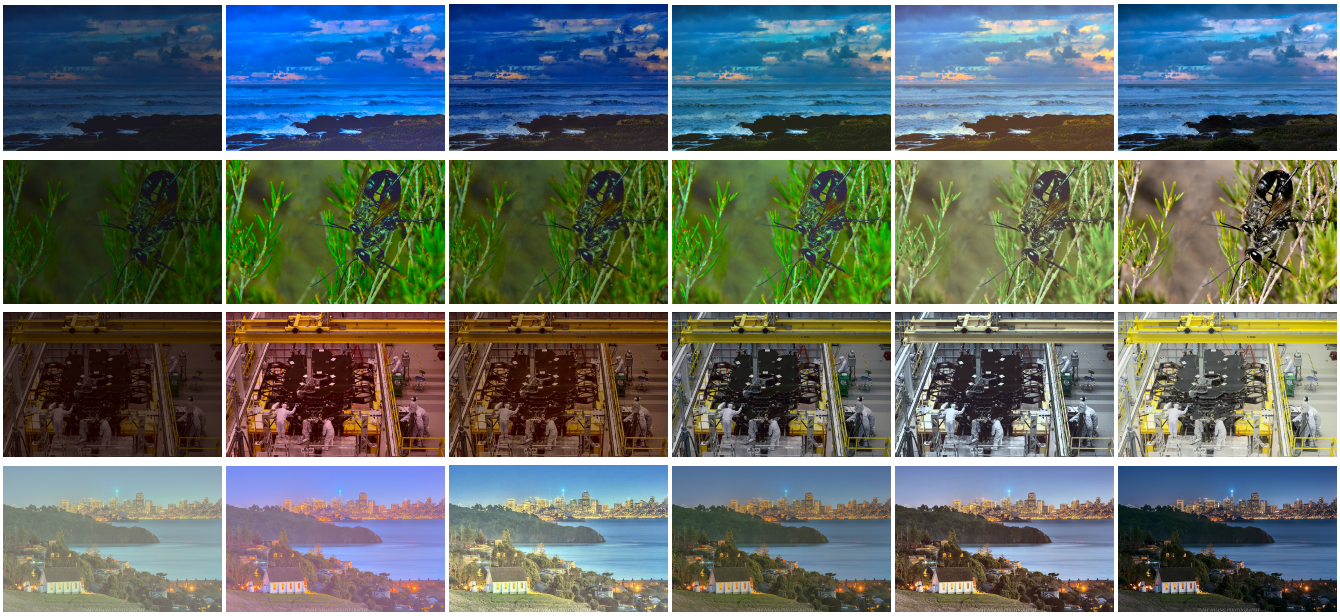
ii) Adobe Photoshop.[2].

We obtained 14,049 images using the original BBMO, of which 13,750 were used for training the CNN in the same framework and 299 are used for validation. DPE was trained using 5,000 images, of which 4,500 were used for training and 500 for validation.

4.1. Qualitative Results

Example results using de-enhanced photos as inputs are shown in Fig. 5. On the top row, the results by the proposed method looks better than the results by other methods. In the middle row, the input image is generally made brighter and of higher contrast by all methods; but the result by the proposed method has relatively natural saturation among them. We also show the results of enhancing real

[2]https://www.adobe.com/uk/products/photoshop.html



(a) Input Image (b) BBMO [20] (c) DPE [3] (d) Photoshop (e) Ours (f) Original Image

Figure 5. Comparison of automatic enhancement results with existing approaches on images from the test dataset. (a) Input image. (b)-(e) Outputs by different approaches. (f) Ground truth original high quality image.

(a) Input Image (b) BBMO [20] (c) DPE [3] (d) Photoshop (e) Ours

Figure 6. Comparison of automatic enhancement results with existing approaches, using real photographs, i.e., raw photos taken by a camera and no color editing.

Table 4. Results of the perceptual user study.

Training Images	Ours 3,224	Input -	BBMO [20] 14,049	DPE [3] 4,500
vs. Ours	-	0.11	0.32	0.43
vs. Input	**0.89**	-	0.64	0.77
vs. BBMO	0.68	0.36	-	**0.71**
vs. DPE	**0.58**	0.23	0.29	-

photographs in Fig. 6. Even though the data looks significantly different from the de-enhanced photos, our approach shows good results for a variety of images.

4.2. Perceptual user study

We evaluated these automatic enhancement results with a perceptual user study. We pooled the input image, the output by the enhancer trained by our data set, the output by the enhancer trained by the data set generated in [20], and the output by Deep Photo Enhancer [3]. The input images consisted of 30 de-enhanced photos and 20 real raw photos. We showed 25 participants 50 sets of two images randomly selected from the pool and asked them to choose the one that looks better. The results are shown in Table 4. For each comparison, the score is the percentage of the image selected by the participants as better. Despite the small number of training images, the output of the enhancer trained by our data set had better score than those by BBMO[20]. Our results had a slightly lower score than Deep Photo Enhancer [3].

We are mildly puzzled by the result because in [20] BBMO outperforms DPE. Although the conditions of the studies are different, this needs further investigation. However, since the setting is identical between our method and BBMO in our experiment, we at least confirmed that learning can be performed effectively even with a smaller set of training data by improving its quality. From this, it can be said that our improvements of the optimization method and the image similarity function are effective.

5. Conclusion

In this work, we have proposed an approach to generate high quality data for training a photo enhancement model, that in combination with using more advanced optimization techniques, significantly improve the enhancement results. As our approach is based on predicting enhancement parameters to be used with an off-the-shelf photo enhancement software, it is amenable to human interpretation and further corrections.

6. Acknowledgement

This work was partially supported by JST ACT-I (Iizuka, Grant Number: JPMJPR16U3), JST PRESTO (Simo-Serra, Grant Number: JPMJPR1756), and JST CREST (Ishikawa, Iizuka, and Simo-Serra, Grant Number: JPMJCR14D1).

References

[1] N. Arvanitopoulos, R. Achanta, and S. Ssstrunk. Single image reflection suppression. In *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1752–1760, July 2017. 2

[2] 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*, pages 97–104, June 2011. 2

[3] Y. Chen, Y. Wang, M. Kao, and Y. Chuang. Deep photo enhancer: Unpaired learning for image enhancement from photographs with gans. In *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6306–6314, June 2018. 2, 6, 7, 8

[4] J. Deng, W. Dong, R. Socher, L. Li, and and. Imagenet: A large-scale hierarchical image database. In *2009 IEEE Conference on Computer Vision and Pattern Recognition*, pages 248–255, June 2009. 2

[5] C. Dong, C. C. Loy, K. He, and X. Tang. Learning a deep convolutional network for image super-resolution. In D. Fleet, T. Pajdla, B. Schiele, and T. Tuytelaars, editors, *Computer Vision – ECCV 2014*, pages 184–199, Cham, 2014. Springer International Publishing. 2

[6] C. Dong, C. C. Loy, K. He, and X. Tang. Image super-resolution using deep convolutional networks. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 38(2):295–307, Feb 2016. 2

[7] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio. Generative adversarial nets. In Z. Ghahramani, M. Welling, C. Cortes, N. D. Lawrence, and K. Q. Weinberger, editors, *Advances in Neural Information Processing Systems 27*, pages 2672–2680. Curran Associates, Inc., 2014. 2

[8] N. Hansen and S. Kern. Evaluating the cma evolution strategy on multimodal test functions. In X. Yao, E. K. Burke, J. A. Lozano, J. Smith, J. J. Merelo-Guervós, J. A. Bullinaria, J. E. Rowe, P. Tiňo, A. Kabán, and H.-P. Schwefel, editors, *Parallel Problem Solving from Nature - PPSN VIII*, pages 282–291, Berlin, Heidelberg, 2004. Springer Berlin Heidelberg. 5

[9] N. Hansen and A. Ostermeier. Adapting arbitrary normal mutation distributions in evolution strategies: the covariance matrix adaptation. In *Proceedings of IEEE International Conference on Evolutionary Computation*, pages 312–317, May 1996. 5

[10] S. Iizuka, E. Simo-Serra, and H. Ishikawa. Let there be color!: Joint end-to-end learning of global and local image priors for automatic image colorization with simultaneous classification. *ACM Trans. Graph.*, 35(4):110:1–110:11, July 2016. 2

[11] P. Isola, J. Zhu, T. Zhou, and A. A. Efros. Image-to-image translation with conditional adversarial networks. In *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 5967–5976, July 2017. 3

[12] S. B. Kang, A. Kapoor, and D. Lischinski. Personalization of image enhancement. In *2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages 1799–1806, June 2010. 2

[13] L. Kaufman, D. Lischinski, and M. Werman. Content-aware automatic photo enhancement. In *Computer Graphics Forum*, volume 31, pages 2528–2540. Wiley Online Library, 2012. 2

[14] Y. Koyama, D. Sakamoto, and T. Igarashi. Selph: Progressive learning and support of manual photo color enhancement. In *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems*, CHI '16, pages 2520–2532, New York, NY, USA, 2016. ACM. 2

[15] A. Krizhevsky, I. Sutskever, and G. E. Hinton. Imagenet classification with deep convolutional neural networks. In *Proceedings of the 25th International Conference on Neural Information Processing Systems - Volume 1*, NIPS'12, pages 1097–1105, USA, 2012. Curran Associates Inc. 6

[16] P.-Y. Laffont, Z. Ren, X. Tao, C. Qian, and J. Hays. Transient attributes for high-level understanding and editing of outdoor scenes. *ACM Trans. Graph.*, 33(4):149:1–149:11, July 2014. 2

[17] D. Lischinski, Z. Farbman, M. Uyttendaele, and R. Szeliski. Interactive local adjustment of tonal values. *ACM Trans. Graph.*, 25(3):646–653, July 2006. 2

[18] Z. Liu, P. Luo, S. Qiu, X. Wang, and X. Tang. Deepfashion: Powering robust clothes recognition and retrieval with rich annotations. In *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1096–1104, June 2016. 2

[19] J. A. Nelder and R. Mead. A Simplex Method for Function Minimization. *The Computer Journal*, 7(4):308–313, 01 1965. 5

[20] M. Omiya, E. Simo-Serra, S. Iizuka, and H. Ishikawa. Learning photo enhancement by black-box model optimization data generation. In *SIGGRAPH Asia 2018 Technical Briefs*, SA '18, pages 7:1–7:4, New York, NY, USA, 2018. ACM. 1, 3, 5, 6, 7, 8

[21] K. Park, K. Rematas, A. Farhadi, and S. M. Seitz. Photoshape: Photorealistic materials for large-scale shape collections. *ACM Trans. Graph.*, 37(6):192:1–192:12, Dec. 2018. 2

[22] S. R. Richter, V. Vineet, S. Roth, and V. Koltun. Playing for data: Ground truth from computer games. In B. Leibe, J. Matas, N. Sebe, and M. Welling, editors, *Computer Vision – ECCV 2016*, pages 102–118, Cham, 2016. Springer International Publishing. 2

[23] O. Ronneberger, P. Fischer, and T. Brox. U-net: Convolutional networks for biomedical image segmentation. In N. Navab, J. Hornegger, W. M. Wells, and A. F. Frangi, editors, *Medical Image Computing and Computer-Assisted Intervention – MICCAI 2015*, pages 234–241, Cham, 2015. Springer International Publishing. 3

[24] P. Sangkloy, J. Lu, C. Fang, F. Yu, and J. Hays. Scribbler: Controlling deep image synthesis with sketch and color. In *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 6836–6845, July 2017. 3

[25] E. Simo-Serra, S. Iizuka, and H. Ishikawa. Real-time data-driven interactive rough sketch inking. *ACM Trans. Graph.*, 37(4):98:1–98:14, July 2018. 3

[26] K. Simonyan and A. Zisserman. Very deep convolutional networks for large-scale image recognition. In *International Conference on Learning Representations (ICLR)*, 2015. 6

[27] 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*, 13(4):600–612, April 2004. 5

[28] Z. Yan, H. Zhang, B. Wang, S. Paris, and Y. Yu. Automatic photo adjustment using deep neural networks. *ACM Trans. Graph.*, 35(2):11:1–11:15, Feb. 2016. 2

[29] R. Zhang, J.-Y. Zhu, P. Isola, X. Geng, A. S. Lin, T. Yu, and A. A. Efros. Real-time user-guided image colorization with learned deep priors. *ACM Trans. Graph.*, 36(4):119:1–119:11, July 2017. 3

[30] J. Zhu, T. Park, P. Isola, and A. A. Efros. Unpaired image-to-image translation using cycle-consistent adversarial networks. In *2017 IEEE International Conference on Computer Vision (ICCV)*, pages 2242–2251, Oct 2017. 2