Deep Photo Enhancer: Unpaired Learning for Image Enhancement from Photographs with GANs

Yu-Sheng Chen Yu-Ching Wang Man-Hsin Kao Yung-Yu Chuang*

National Taiwan University

Abstract

This paper proposes an unpaired learning method for image enhancement. Given a set of photographs with the desired characteristics, the proposed method learns a photo enhancer which transforms an input image into an enhanced image with those characteristics. The method is based on the framework of two-way generative adversarial networks (GANs) with several improvements. First, we augment the U-Net with global features and show that it is more effective. The global U-Net acts as the generator in our GAN model. Second, we improve Wasserstein GAN (WGAN) with an adaptive weighting scheme. With this scheme, training converges faster and better, and is less sensitive to parameters than WGAN-GP. Finally, we propose to use individual batch normalization layers for generators in two-way GANs. It helps generators better adapt to their own input distributions. All together, they significantly improve the stability of GAN training for our application. Both quantitative and visual results show that the proposed method is effective for enhancing images.

1. Introduction

Photographs record valuable moments of our life. With the popularization of mobile phone cameras, users enjoy taking photographs even more. However, current cameras have limitations. They have to reconstruct a complete and high-quality image from a set incomplete and imperfect samples of the scene. The samples are often noisy, incomplete in color and limited in the resolution and the dynamic range. In addition, the camera sensor responds linearly to the incoming light while human perception performs more sophisticated non-linear mapping. Thus, users could be disappointed with photographs they take because the photographs do not match their expectations and visual experience. The problem is even aggravated for mobile cameras because of their small sensors and compact lenses.

Image enhancement methods attempt to address the issues with color rendition and image sharpness. There are interactive tools and semi-automatic methods for this purpose. Most interactive software provides elementary tools such as histogram equalization, sharpening, contrast adjustment and color mapping, and some advanced functions such as local and adaptive adjustments. The quality of the results however highly depends on skills and aesthetic judgement of the users. In addition, it often takes a significant amount of time to reach satisfactory retouching results. The semi-automatic methods facilitate the process by only requiring adjustments of a few parameters. However, the results could be very sensitive to parameters. In addition, these methods are often based on some heuristic rules about human perception such as enhancing details or stretching contrast. Thus, they could be brittle and lead to bad results.

This paper proposes a method for image enhancement by learning from photographs. The method only requires a set of "good" photographs as the input. They have the characteristics that the user would like to have for their photographs. They can be collected easily from websites or any stock of photographs. We treat the image enhancement problem as an image-to-image translation problem in which an input image is transformed into an enhanced image with the characteristics embedded in the set of training photographs. Thus, we tackle the problem with a two-way GAN whose structure is similar to CycleGAN [26]. However, GANs are notorious for their instability. For addressing the issue and obtaining high-quality results, we propose a few improvements along the way of constructing our two-way GAN. First, for the design of the generator, we augment the U-Net [20] with global features. The global features capture the notion of scene setting, global lighting condition or even subject types. They are helpful for determining what local operations should be performed. Second, we propose an adaptive weighting scheme for Wasserstein

*This work was supported by Ministry of Science and Technology (MOST) and MediaTek Inc. under grants MOST 105-2622-8-002-002 and MOST 104-2628-E-002-003-MY3.

GAN (WGAN) [1]. WGAN uses weight clipping for enforcing the Lipschitz constraint. It was later discovered a terrible way and some proposed to use the gradient penalty for enforcing the constraint [9]. However, we found that the approach is very sensitive to the weighting parameter of the penalty. Thus, we propose to use an adaptive weighting scheme to improve the convergence of WGAN training. Finally, most two-way GAN architectures use the same generator in both forward and backward passes. It makes sense since the generators in both paths perform similar mapping with the same input and output domains. However, we found that, although in the same domain, the inputs actually come from different sources, one from the input data and the other from the generated data. The discrepancy between distributions of input sources could have vicious effects on the performance of the generator. We propose to use individual batch normalization layers for the same type of generators. This way, the generator can better adapt to the input data distribution. With these improvements, our method can provide high-quality enhanced photographs with better color rendition and sharpness. The results often look more natural than previous methods. In addition, the proposed techniques, global U-Net, adaptive WGAN and individual batch normalization, can be useful for other applications.

2. Related work

Image enhancement has been studied for a long time. Many operations and filters have been proposed to enhance details, improve contrast and adjust colors. Wang *et al*. [22] proposed a method for enhancing details while preserving naturalness. Aubry *et al*. [2] proposed local Laplacian operator for enhancing details. Most of these operations are algorithmic and based on heuristic rules. Bychkovsky *et al*. [4] proposed a learning-based regression method for approximating photographers' adjustment skills. For this purpose, they collected a dataset containing images before and after adjustments by photographers.

The convolutional neural networks (CNNs) have become a major workhorse for a wide set of computer vision and image processing problems. They have also been applied to the image enhancement problem. Yan *et al*. [23] proposed the first deep-learning-based method for photo adjustment. Gharbi *et al*. [7] proposed a fast approximation for existing filters. Ignatov *et al*. [10] took a different approach by learning the mapping between a mobile phone camera and a DSLR camera. They collected the DPED dataset consisting of images of the same scene taken by different cameras. A GAN model was used for learning the mapping. Chen *et al*. [6] approximated existing filters using a fully convolutional network. It can only learn existing filters and cannot do beyond what they can do. All these methods are supervised and require paired images while ours is unpaired. The unpaired nature eases the process of collecting training data.

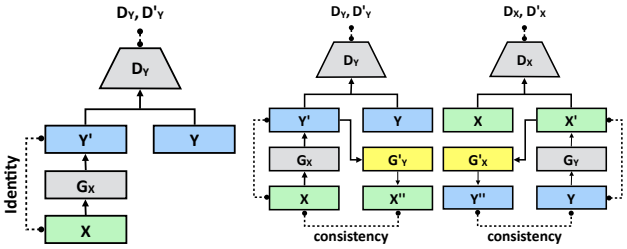
Our method is based on the generative adversarial networks (GANs) [8]. Although GANs have been proved powerful, they are notorious on training instability. Significant efforts have been made toward stable training of GANs. Wasserstein GAN uses the earth mover distance to measure the distance between the data distribution and the model distribution and significantly improves training stability [1]. Gulrajani *et al*. found that WGAN could still generate low-quality samples or fail to converge due to weight clipping [9]. Instead of weight clipping, they proposed to penalize the norm of the gradient of the discriminator with respect to the input. The resultant model is called WGAN-GP (WGAN with gradient penalty). It often generates higher-quality samples and converges faster than WGAN. There are also energy-based GAN variants, such as BEGAN [3] and EBGAN [25].

Isola *et al*. proposed a conditional adversarial network as a general-purpose solution to image-to-image translation problems [13], converting from one representation of a scene to another, such as from a semantic label map to a realistic image or from a day image to its night counterpart. Although generating amazing results, their method requires paired images for training. Two-way GANs were later proposed for addressing the problem by introducing cycle consistency. Famous two-way GANs include CycleGAN [26], DualGAN [24] and DISCOGAN [14]. We formulate image enhancement as an instance of the image-to-image translation problems and solve it with a two-way GAN.

3. Overview

Our goal is to obtain a photo enhancer Φ which takes an input image x and generates an output image $\Phi(x)$ as the enhanced version of x. It is however not easy to define enhancement clearly because human perception is complicated and subjective. Instead of formulating the problem using a set of heuristic rules such as "details should be enhanced" or "contrast should be stretched", we define enhancement by a set of examples Y. That is, we ask the user to provide a set of photographs with the characteristics he/she would like to have. The proposed method aims at discovering the common characteristics of the images in Y and deriving the enhancer so that the enhanced image $\Phi(x)$ shares these characteristics while still resembling the original image x in content.

Because of its nature with set-level supervision, the problem can be naturally formulated using the framework of GANs which learns the embedding of the input samples and generates output samples locating within the subspace spanned by training samples. A GAN model often consists of a discriminator D and a generator G. The framework has been used for addressing the image-to-image translation problem which transforms an input image from the source domain X to the output image in the target domain Y [13].



(a) 1-way GAN (b) 2-way GAN

Figure 1. The network architectures of 1-way and 2-way GANs.

In our application, the source domain X represents original images while the target domain Y contains images with the desired characteristics.

Figure 1(a) gives the architecture for 1-way GAN. Given an input $x \in X$, the generator G_X transforms x into $y' = G_X(x) \in Y$. The discriminator D_Y aims at distinguishing between the samples in the target domain $\{y\}$ and the generated samples $\{y' = G_X(x)\}$. To enforce cycle consistency for better results, several have proposed 2-way GANs such as CycleGAN [26] and DualGAN [24]. They require that $G'_Y(G_X(x)) = x$ where the generator G'_Y takes a G_X-generated sample and maps it back to the source domain X. In addition, 2-way GANs often contains a forward mapping ($X \to Y$) and a backward mapping ($Y \to X$). Figure 1(b) shows the architecture of 2-way GANs. In the forward pass, $x \xrightarrow{G_X} y' \xrightarrow{G'_Y} x''$ and we check the consistency between x and x''. In the backward pass, $y \xrightarrow{G_Y} x' \xrightarrow{G'_X} y''$ and we check the consistency between y and y''.

In the following sections, we will first present the design of our generator (Section 4). Next, we will describe the design of our 1-way GAN (Section 5) and the one for our 2-way GAN (Section 6).

4. Generator

For our application, the generator in the GAN framework plays an important role as it will act as the final photo enhancer Φ. This section proposes a generator and compares it with several options. Figure 2(a) shows the proposed generator. The size of input images is fixed at 512×512.

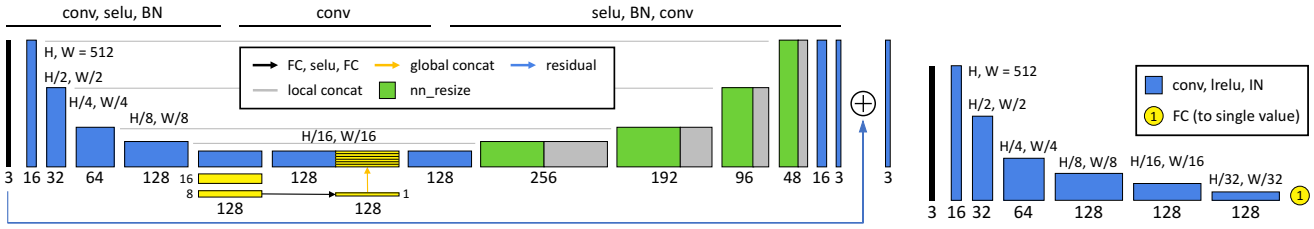
Our generator is based on the U-Net [20] which was originally proposed for biomedical image segmentation but later also showed strong performance on many tasks. U-Net however does not perform very well on our task. Our conjecture is that the U-Net does not include global features. Our vision system usually adjusts to the overall lighting conditions and scene settings. Similarly, cameras have scene settings and often apply different types of adjustments depending on the current setting. The global features could reveal high-level information such as the scene category, the subject type or the overall lighting condition which could be useful for individual pixels to determine their local adjustments. Thus, we add the global features into the U-Net.

In order to improve the model efficiency, the extraction of global features shares the same contracting part of the U-Net with the extraction of local features for the first five layers. Each contraction step consists of 5×5 filtering with stride 2 followed by SELU activation [15] and batch normalization [12]. Given the $32 \times 32 \times 128$ feature map of the 5th layer, for global features, the feature map is further reduced to $16 \times 16 \times 128$ and then $8 \times 8 \times 128$ by performing the aforementioned contraction step. The $8 \times 8 \times 128$ feature map is then reduced to $1 \times 1 \times 128$ by a fully-connected layer followed by a SELU activation layer and then another fully-connected layer. The extracted $1 \times 1 \times 128$ global features are then duplicated 32×32 copies and concatenated after the $32 \times 32 \times 128$ feature map for the low-level features, resulting a $32 \times 32 \times 256$ feature map which fuses both local and global features together. The expansive path of the U-Net is then performed on the fused feature map. Finally, the idea of residual learning is adopted because it has been shown effective for image processing tasks and helpful on convergence. That is, the generator only learns the difference between the input image and the label image.

Global features have been explored by other image processing tasks such as colorization [11]. However, their model requires an extra supervised network trained with explicit scene labels. For many applications, it is difficult to define labels explicitly. The novelty of our model is to use the U-Net itself to encode an implicit feature vector describing global features useful for the target application.

The dataset. We used the MIT-Adobe 5K dataset [4] for training and testing. The dataset contains $5,000$ images, each of which was retouched by five well-trained photographers using global and local adjustments. We selected the results of photographer C as labels since he was ranked the highest in the user study [4]. The dataset was split into three partitions: the first one consists of $2,250$ images and their retouched versions were used for training in the supervised setting in this section; for the unpaired training in Section 5 and Section 6, the retouched images of another $2,250$ images acted as the target domain while the $2,250$ images of the first partition were used for the source domain; the rest 500 images were used for testing in either setting.

The experiments. We evaluated several network architectures for the generator. (1) DPED [10]: since we only evaluate on generators, we took only the generator of their GAN architecture. (2) 8RESBLK [26, 17]: the generator has been used in CycleGAN [26] and UNIT [17]. (3) FCN [6]: a fully convolutional network for approximating filters. (4) CRN [5]: the architecture has been used to synthesize realistic images from semantic labels. (5) U-Net [20]. The residual learning is augmented to all of them. Because of the image size and the limitation on memory capacity, the



(a) generator (b) discriminator

Figure 2. The network architectures of the proposed generator (a) and the proposed discriminator (b).

	DPED	8RESBLK	FCN	CRN	U-Net	Ours
PSNR	25.50	31.46	31.52	33.52	31.06	**33.93**
SSIM	0.911	0.951	0.952	0.972	0.960	**0.976**

Table 1. The average accuracy of different network architectures on approximating fast local Laplacian filtering on the 500 testing images from the MIT-Adobe 5K dataset.

	DPED	8RESBLK	FCN	CRN	U-Net	Ours
PSNR	21.76	23.42	20.66	22.38	22.13	**23.80**
SSIM	0.871	0.875	0.849	0.877	0.879	**0.900**

Table 2. The average accuracy of different network architectures on predicting the retouched images by photographers on the 500 testing images from the MIT-Adobe 5K dataset.

number of features of the first layer is limited at 16. Otherwise, the overall architecture cannot be fitted within the memory. The loss function is to maximize PSNR:

$$\arg\min_{G_X} \mathop{\mathbb{E}}_{y,y'}\Big[\log_{10}(MSE(y,y'))\Big], \text{where} \tag{1}$$

$$MSE(x,y) = \frac{1}{mn}\sum_{i=0}^{m-1}\sum_{j=0}^{n-1}\|x(i,j)-y(i,j)\|^2. \tag{2}$$

Table 1 shows both the average PSNR and SSIM values for all compared architectures on approximating fast local Laplacian filtering for the 500 testing images from MIT-Adobe 5K dataset. By adding the global features, the proposed architecture provides nearly 3dB gain over its counterpart without global features, and outperforms all compared architectures. Our generator does an excellent job on approximating the fast local Laplacian filter with 33.93dB PSNR, better than FCN which is designed for such tasks. Table 2 reports the performance of these architectures on predicting the retouched images. This task is much more difficult as human retouching could be more complicated and less inconsistent than algorithmic filters. Again, the proposed global U-Net architecture outperforms others.

5. One-way GAN

This section presents our GAN architecture for unpaired training. Figure 2(b) illustrates the architecture of our discriminator. By employing the generator (Figure 2(a)) as G_X and the discriminator (Figure 2(b)) as D_Y in Figure 1(a), we have the architecture for our 1-way GAN. As mentioned in Section 4, for training GANs, to avoid the content correlation between the source and the target domains, we used $2,250$ images of the MIT-Adobe 5K dataset as the source domain while using the retouched images of another $2,250$ images as the target domain.

variant	PSNR	parameters
GAN	19.65	$\alpha = 10$, $D/G = 1$
LSGAN	20.27	$\alpha = 0.1$, $D/G = 1$
DRAGAN	20.20	$\alpha = 10$, $D/G = 10$
WGAN-GP	21.42	$\alpha = 1000$, $D/G = 50$, $\lambda = 10$
WGAN-GP	21.19	$\alpha = 1000$, $D/G = 50$, $\lambda = 1$
WGAN-GP	20.78	$\alpha = 1000$, $D/G = 50$, $\lambda = 0.1$
A-WGAN	22.17	$\alpha = 1000$, $D/G = 50$

Table 3. Comparisons of different GANs and parameters.

There are many variants of GAN formulations. We first experimented with several flavors of GANs, including GAN [8], LSGAN [18], DRAGAN [16] and WGAN-GP [9], with different parameter settings. Table 3 reports the results for some of them. All GANs require the parameter α for the weight of the identity loss $\mathop{\mathbb{E}}_{x,y'}\Big[MSE(x,y')\Big]$ which ensures the output is similar to the input. The parameter D/G represents the ratio between the numbers of discriminator and generator training passes. In our application, WGAN-GP performs better than GAN, LSGAN and DRAGAN. Table 3 only reports the best performance for methods other than WGAN-GP. However, the performance of WGAN-GP depends on an additional parameter λ which weights the gradient penalty.

WGAN relies on the Lipschitz constraint on the training objective: a differentiable function is 1-Lipschtiz if and only if it has gradients with norm at most 1 everywhere [9]. For satisfying the constraint, WGAN-GP directly constrains the gradient norm of the discriminator output with respect to its input by adding the following gradient penalty,

$$\mathop{\mathbb{E}}_{\hat{y}}\Big[\left(\|\nabla_{\hat{y}} D_Y(\hat{y})\|_2 - 1\right)^2\Big], \tag{3}$$

where $\hat{y}$ is point sampled along straight lines between points

	1-way GAN	2-way GAN	2-way GAN with iBN
WGAN-GP	21.42	21.96 (+0.54)	22.27 (+0.31)
A-WGAN	22.17	22.21 (+0.04)	22.37 (+0.16)

Table 4. The comparison of 1-way GAN, 2-way GAN and 2-way GAN with individual batch normalization (iBN). The numbers in parenthesis indicate the performance gains.

in the target distribution and the generator distribution. A parameter λ weights the penalty with the original discriminator loss. λ determines the tendency that gradients approaches 1. If λ is too small, the Lipschitz constraint cannot be guaranteed. On the other hand, if λ is too large, the convergence could be slow as the penalty could over-weight the discriminator loss. It makes the choice of λ important. Instead, we use the following gradient penalty,

$$\mathop{\mathbb{E}}_{\hat{y}}\Big[\max(0, \|\nabla_{\hat{y}} D_Y(\hat{y})\|_2 - 1)\Big], \tag{4}$$

which better reflects the Lipschitz that requires the gradients are less than or equal to 1 and only penalizes the part which is larger than 1. More importantly, we employ an adaptive weighting scheme for adjusting the weight λ, which chooses a proper weight so that the gradients locate within a desired interval, say $[1.001, 1.05]$. If the moving average of gradients within a sliding window (size=50) is larger than the upper bound, it means that the current weight λ is too small and the penalty is not strong enough to ensure the Lipschitz constraint. Thus, we increase λ by doubling the weight. On the other hand, if the moving average is smaller than the lower bound, we decay λ into a half so that it will not become too large.

Figure 3 compares WGAN-GP and the proposed A-WGAN (Adaptive WGAN) on the swissroll dataset. It is clear that the proposed adaptive weighting WGAN converges to the target distribution regardless of the initial setting of λ while the results of WGAN-GP significantly depend on λ. Table 3 also confirms that A-WGAN outperforms WGAN-GP and achieves the best performance among compared GANs. Note that the PSNR value is not very high even with A-WGAN. It is impossible to capture the exact mapping since the photographer can be subjective with different tastes and often inconsistent. The best that a method can do is to discover the common trend on image characteristics from the training data. However, even if we cannot accurately predict human labels, the learned operation can still improve image quality as shown in Section 7.

6. Two-way GAN

Section 5 has determined to use A-WGAN as our GAN formulation. For achieving better results, this section extends the GAN architecture into a 2-way GAN. Figure 1(b)

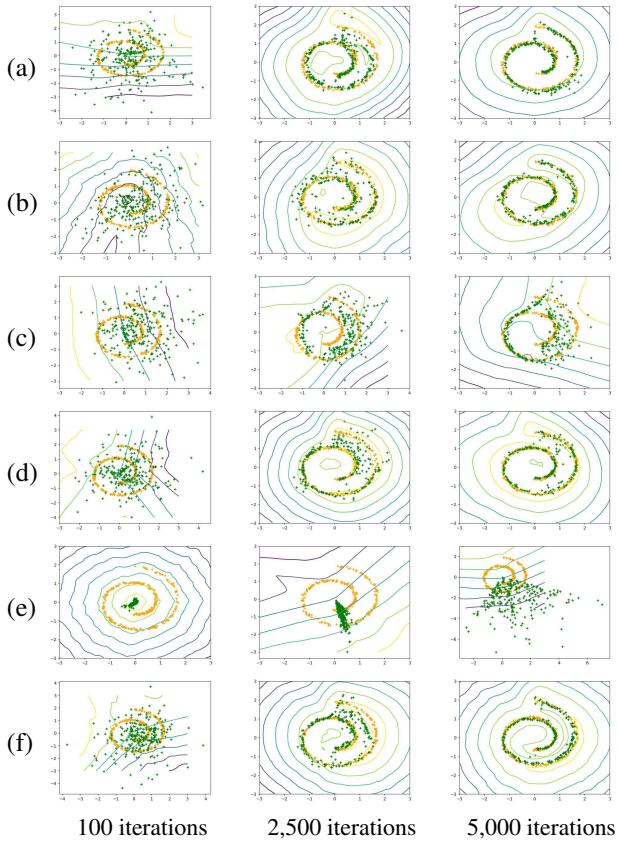


Figure 3. The experiment with swissroll. The orange points represent the target distribution. The green points are generated samples. (a) WGAN-GP ($\lambda = 0.1$). (b) A-WGAN ($\lambda = 0.1$). (c) WGAN-GP ($\lambda = 1$). (d) A-WGAN ($\lambda = 1$). (e) WGAN-GP ($\lambda = 10$). (f) A-WGAN ($\lambda = 10$).

shows the architecture for the 2-way GAN. Table 4 compares WGAN-GP and the proposed A-WGAN when extending to the 2-way GAN architectures. Both have gains from the use of the 2-way GAN architecture. However, the gain is more significant on WGAN-GP. It could be because there is less room for A-WGAN to improve.

Most 2-way GANs use the same generator for both G_X and G'_X because both map an input from the domain X to the domain Y, and similarly for G_Y and G'_Y. However, from Figure 1(b), the input of G_X is taken from the input samples X while G'_X takes the generated samples X' as its inputs. They could have different distribution characteristics. Thus, it is better that the generators G_X and G'_X adapt to their own inputs. Our solution is to use individual batch normalization (iBN) layers for G_X and G'_X. That is, our generators G_X and G'_X share all layers and parameters except for the batch normalization layers. Each generator has its own batch normalization so that it better adapts to the characteristics of its own inputs. Table 4 confirms that the use of individual batch normalization helps the generator adapt to input distributions. With iBN, both WGAN-GP

(a) input (b) ours (HDR) (c) ours (HDR) w/o iBN (d) ours (SL) (e) ours (UL)

(f) DPED (iPhone7) (g) CLHE (h) NPEA (i) FLLF (j) label

Figure 4. Comparisons of our models with DPED [10], CLHE [21], NPEA [22] and FLLF [2]. SL and UL respectively mean supervised and unpaired learning on the MIT-Adobe 5K dataset. The label is the retouched image by a photographer.

and A-WGAN were improved, by 0.31dB and 0.16dB respectively. As shown in Figure 4 and Figure 5, without iBN, the 2-way GAN could fail to capture the right distribution on more challenging cases.

To sum up, our objective consists of several losses. The first one is the identity mapping loss I which requires that the content of the transformed image y should be similar to the input image x. Because of the 2-way GAN, there is also a counterpart for mapping from y to x'.

$$I = \mathop{\mathbb{E}}_{x,y'}\Big[MSE(x,y')\Big] + \mathop{\mathbb{E}}_{y,x'}\Big[MSE(y,x')\Big]. \quad (5)$$

The second loss is the cycle consistency loss C defined as

$$C = \mathop{\mathbb{E}}_{x,x''}\Big[MSE(x,x'')\Big] + \mathop{\mathbb{E}}_{y,y''}\Big[MSE(y,y'')\Big]. \quad (6)$$

The adversarial losses for the discriminator A_D and the generator A_G are defined as

$$A_D = \mathop{\mathbb{E}}_{x}\Big[D_X(x)\Big] - \mathop{\mathbb{E}}_{x'}\Big[D_X(x')\Big] + \quad (7)$$

$$\mathop{\mathbb{E}}_{y}\Big[D_Y(y)\Big] - \mathop{\mathbb{E}}_{y'}\Big[D_Y(y')\Big], \quad (8)$$

$$A_G = \mathop{\mathbb{E}}_{x'}\Big[D_X(x')\Big] + \mathop{\mathbb{E}}_{y'}\Big[D_Y(y')\Big]. \quad (9)$$

When training the discriminator, the gradient penalty P is added where

$$P = \mathop{\mathbb{E}}_{\hat{x}}\Big[\max(0, \|\nabla_{\hat{x}} D_X(\hat{x})\|_2 - 1)\Big] + \quad (10)$$

$$\mathop{\mathbb{E}}_{\hat{y}}\Big[\max(0, \|\nabla_{\hat{y}} D_Y(\hat{y})\|_2 - 1)\Big]. \quad (11)$$

The term ensures 1-Lipschitz functions for Wasserstein distance. Thus, the discriminator is obtained by the following optimization,

$$\arg\max_{D}\Big[A_D - \tilde{\lambda} P\Big], \quad (12)$$

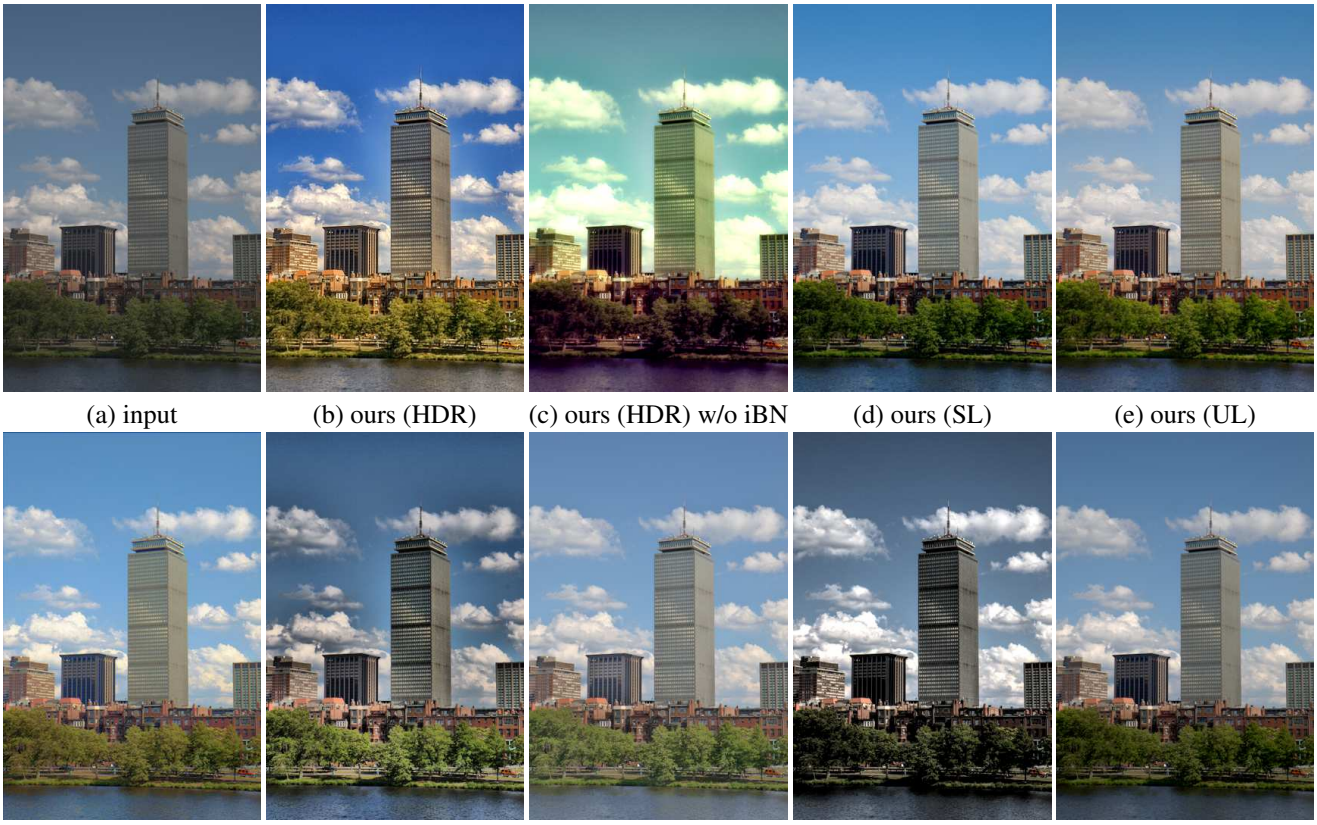
where the weight $\tilde{\lambda}$ is adaptively adjusted using A-WGAN. The generator is obtained by

$$\arg\min_{G}\Big[-A_G + \alpha I + 10\alpha C\Big], \quad (13)$$

where α weights between the adversarial loss and identity/consistency losses. Finally, we take the resultant generator G_X as the photo enhancer Φ.

7. Results

In addition to training on photographer labels of the MIT-Adobe 5K dataset, we have also collected an HDR dataset



(a) input (b) ours (HDR) (c) ours (HDR) w/o iBN (d) ours (SL) (e) ours (UL)

(f) DPED (iPhone7) (g) CLHE (h) NPEA (i) FLLF (j) label

Figure 5. Comparisons of our models with DPED [10], CLHE [21], NPEA [22] and FLLF [2].

for training. The images were selected from Flickr images tagged with HDR. They were ranked by interestness provided by Flickr and the top ones were further selected by users. Finally, 627 images were selected as the training set. The learned deep photo enhancer is very efficient and runs at 50fps with NVidia 1080 Ti. Although the model was trained on 512×512 inputs, we have extended it so that it can handle arbitrary resolutions[1].

Figure 4 compares our models with several methods including DPED [10] (Figure 4(f)), CLHE [21] (Figure 4(g)), NPEA [22] (Figure 4(h)) and FLLF [2] (Figure 4(i)). By learning from photographers using the MIT-Adobe 5K dataset, both our supervised method (Figure 4(d)) and unpaired learning method (Figure 4(e)) do a reasonable job on enhancing the input image (Figure 4(a)). Figure 4(j)) shows the photographer label of this image as a reference. The result of our method trained with the collected HDR dataset (Figure 4(b)) shows the best result among all compared methods. It looks sharp and natural with good color rendition. Using the same HDR dataset, without individual batch normalization (iBN), the color rendition is weird (Figure 4(c)). It shows the importance of the proposed iBN.

[1]The demo system is available at http://www.cmlab.csie.ntu.edu.tw/project/Deep-Photo-Enhancer/.

	CycleGAN	DPED	NPEA	CLHE	ours	total
CycleGAN	-	32	27	23	11	93
DPED	368	-	141	119	29	657
NPEA	373	259	-	142	50	824
CLHE	377	281	258	-	77	993
ours	389	371	350	323	-	1433

Table 5. The preference matrix from the user study.

Figure 5 show comparisons on another example. Figure 6 show more comparisons. Again, our HDR model consistently gives the best results. Note that the inputs of the first and the last rows of Figure 6 were taken by mobile phones.

User study. We have performed a user study with 20 participants and 20 images using pairwise comparisons on five methods. Table 5 reports the preference matrix. Our model trained on HDR images ranked the first and CLHE was the runner-up. When comparing our model with CLHE, 81% of users (323 among 400) preferred our results.

Limitations. Our model could amplify noise if the input is very dark and contains a significant amount of noise. In addition, since some HDR images for training are products of tone mapping, our model could suffer from halo artifacts inherited from tone mapping for some images.

Other applications. This paper proposes three improve-

input our (HDR) DPED (iPhone7, Nexus 5x) CLHE

Figure 6. Comparisons of our method with DPED and CLHE. The first and the last inputs were taken by iPhone7+ and Nexus respectively.

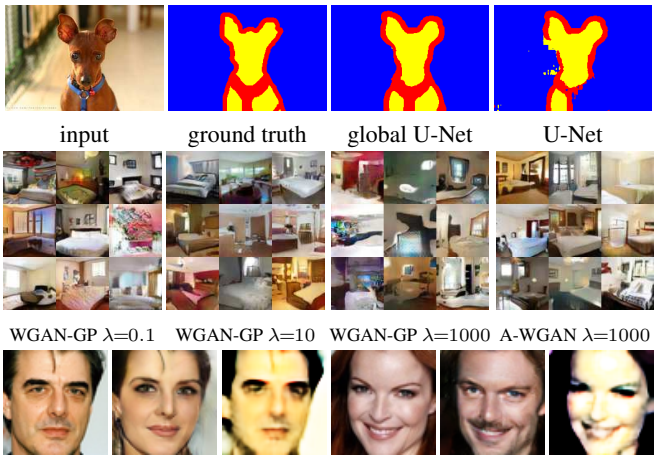


Figure 7. Other applications of global U-Net, A-WGAN and iBN.

ments: global U-Net, adaptive WGAN (A-WGAN) and individual batch normalization (iBN). They generally improve results; and for some applications, the improvement is sufficient for crossing the bar and leading to success. We have applied them to some other applications. For global U-Net, we applied it to trimap segmentation for pets using the Oxford-IIIT Pet dataset [19]. The accuracies of U-Net and global U-Net are 0.8759 and 0.8905 respectively. The first row of Figure 7 shows an example of pet trimap segmentation. The second row of Figure 7 shows the results of bedroom image synthesis. With different λ values, WGAN-GP could succeed or fail. The proposed A-WGAN is less dependent with λ and succeeded with all three λ values (only one shown here). Finally, we applied the 2-way GAN to gender change of face images. As shown in the last row of Figure 7, the 2-way GAN failed on the task but succeeded after employing the proposed iBN.

8. Conclusion

This paper presents Deep Photo Enhancer which learns image enhancement from a set of given photographs with user's desired characteristics. With its unpaired setting, the process of collecting training images is easy. By taking images provided by different users, the enhancer can be personalized to match individual user's preference. We have made several technical contributions while searching for the best architecture for the application. We enhance U-Net for image processing by augmenting global features. We improve the stability of WGAN by an adaptive weighting scheme. We also propose to use individual batch normalization layers for generators for improving 2-way GANs.

References

[1] M. Arjovsky, S. Chintala, and L. Bottou. Wasserstein GAN. In *arXiv preprint arXiv:1701.07875*, 2017. 2

[2] M. Aubry, S. Paris, S. W. Hasinoff, J. Kautz, and F. Durand. Fast local Laplacian filters: Theory and applications. *ACM Transactions on Graphics*, 33(5):167, 2014. 2, 6, 7

[3] D. Berthelot, T. Schumm, and L. Metz. BEGAN: Boundary equilibrium generative adversarial networks. *arXiv preprint arXiv:1703.10717*, 2017. 2

[4] V. Bychkovsky, S. Paris, E. Chan, and F. Durand. Learning photographic global tonal adjustment with a database of input/output image pairs. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2011. 2, 3

[5] Q. Chen and V. Koltun. Photographic image synthesis with cascaded refinement networks. In *Proceedings of IEEE International Conference on Computer Vision (ICCV)*, Oct 2017. 3

[6] Q. Chen, J. Xu, and V. Koltun. Fast image processing with fully-convolutional networks. In *Proceedings of IEEE International Conference on Computer Vision (ICCV)*, Oct 2017. 2, 3

[7] M. Gharbi, J. Chen, J. T. Barron, S. W. Hasinoff, and F. Durand. Deep bilateral learning for real-time image enhancement. *ACM Transactions on Graphics*, 36(4):118, 2017. 2

[8] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio. Generative adversarial nets. In *Advances in neural information processing systems (NIPS)*, pages 2672–2680, 2014. 2, 4

[9] I. Gulrajani, F. Ahmed, M. Arjovsky, V. Dumoulin, and A. Courville. Improved training of Wasserstein GANs. In *Advances in neural information processing systems (NIPS)*, 2017. 2, 4

[10] A. Ignatov, N. Kobyshev, K. Vanhoey, R. Timofte, and L. Van Gool. DSLR-quality photos on mobile devices with deep convolutional networks. In *Proceedings of IEEE International Conference on Computer Vision (ICCV)*, Oct 2017. 2, 3, 6, 7

[11] 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 Transactions on Graphics*, 35(4):110, 2016. 3

[12] S. Ioffe and C. Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. In *Proceedings of International Conference on Machine Learning (ICML)*, pages 448–456, 2015. 3

[13] P. Isola, J.-Y. Zhu, T. Zhou, and A. A. Efros. Image-to-image translation with conditional adversarial networks. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, July 2017. 2

[14] T. Kim, M. Cha, H. Kim, J. Lee, and J. Kim. Learning to discover cross-domain relations with generative adversarial networks. In *Proceedings of International Conference on Machine Learning (ICML)*, pages 1857–1865, Aug 2017. 2

[15] G. Klambauer, T. Unterthiner, A. Mayr, and S. Hochreiter. Self-normalizing neural networks. In *Advances in neural information processing systems (NIPS)*, 2017. 3

[16] N. Kodali, J. Abernethy, J. Hays, and Z. Kira. On convergence and stability of GANs. In *arXiv preprint arXiv:1705.07215*, 2017. 4

[17] M.-Y. Liu, T. Breuel, and J. Kautz. Unsupervised image-to-image translation networks. In *Advances in neural information processing systems (NIPS)*, 2017. 3

[18] X. Mao, Q. Li, H. Xie, R. Y. Lau, Z. Wang, and S. Paul Smolley. Least squares generative adversarial networks. In *Proceedings of IEEE International Conference on Computer Vision (ICCV)*, Oct 2017. 4

[19] O. M. Parkhi, A. Vedaldi, A. Zisserman, and C. V. Jawahar. Cats and dogs. In *Proceedings of CVPR IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2012. 8

[20] O. Ronneberger, P. Fischer, and T. Brox. U-net: Convolutional networks for biomedical image segmentation. In *Proceedings of International Conference on Medical Image Computing and Computer-Assisted Intervention (MICCAI)*, pages 234–241. Springer, 2015. 1, 3

[21] S. Wang, W. Cho, J. Jang, M. A. Abidi, and J. Paik. Contrast-dependent saturation adjustment for outdoor image enhancement. *JOSA A*, 34(1):2532–2542, 2017. 6, 7

[22] S. Wang, J. Zheng, H.-M. Hu, and B. Li. Naturalness preserved enhancement algorithm for non-uniform illumination images. *IEEE Transactions on Image Processing*, 22(9):3538–3548, 2013. 2, 6, 7

[23] Z. Yan, H. Zhang, B. Wang, S. Paris, and Y. Yu. Automatic photo adjustment using deep neural networks. *ACM Transactions on Graphics*, 35(2):11, 2016. 2

[24] Z. Yi, H. Zhang, P. Tan, and M. Gong. DualGAN: Unsupervised dual learning for image-to-image translation. In *Proceedings of IEEE International Conference on Computer Vision (ICCV)*, Oct 2017. 2, 3

[25] J. Zhao, M. Mathieu, and Y. LeCun. Energy-based generative adversarial network. In *Proceedings of International Conference on Learning Representations (ICLR)*, 2017. 2

[26] J.-Y. Zhu, T. Park, P. Isola, and A. A. Efros. Unpaired image-to-image translation using cycle-consistent adversarial networks. In *Proceedings of IEEE International Conference on Computer Vision (ICCV)*, 2017. 1, 2, 3