ADAPTIVE ACTOR-CRITIC BILATERAL FILTER

Bo-Hao Chen, Hsiang-Yin Cheng

Yuan Ze University
Taoyuan, Taiwan

Jia-Li Yin *

Fuzhou University
Fuzhou, China

ABSTRACT

Recent research on edge-preserving image smoothing has suggested that bilateral filtering is vulnerable to maliciously perturbed filtering input. However, while most prior works analyze the adaptation of the range kernel in one-step manner, in this paper we take a more constructive view towards multi-step framework with the goal of unveiling the vulnerability of bilateral filtering. To this end, we adaptively model the width setting of range kernel as a multi-agent reinforcement learning problem and learn an adaptive actor-critic bilateral filter from local image context during successive bilateral filtering operations. By evaluating on eight benchmark datasets, we show that the performance of our filter outperforms that of state-of-the-art bilateral-filtering methods in terms of both salient structures preservation and insignificant textures and perturbation elimination.

Index Terms— Bilateral filtering, multi-agent reinforcement learning, advantage actor critic, image smoothing.

1. INTRODUCTION

Edge-preserving image smoothing has good performance on a wide variety of computer vision tasks, ranging from image texture removal [1], structure extraction [2] and sharpness enhancement [3] to HDR tone mapping [4]. Recent advances in interpolation [4, 5], approximation [2, 6, 7], convolution [8, 9, 3], and histogram [1, 10, 11] demonstrate that effect of bilateral filtering (BF) [12] can be approximated by such lightweight computational frameworks and mechanisms that reduce computational burden caused by brute-force implementation of BF. Improved bilateral filters show high performance in performing the edge-preserving image smoothing task [1]. Following this success, BF-based image filters are starting to be employed in edge-preserving image smoothing related applications, such as image enhancement [3], as well as in image deraining and dehazing [13].

Researchers [14, 15] have also found that bilateral filtering is vulnerable to small perturbations. As illustrated in Fig. 1, BF-based image filters can produce incorrect smoothing results when filtering inputs are corrupted with small perturbations that are generally not perceptible by human observers. Attackers can thus craft perturbed inputs that cause BF-based image filters to misbehave for applications that use edge-preserving image smoothing as a preprocessing step. Therefore, small perturbation attacks can easily affect sophisticated computer vision tasks at scale as we may not tell the attacks happened.

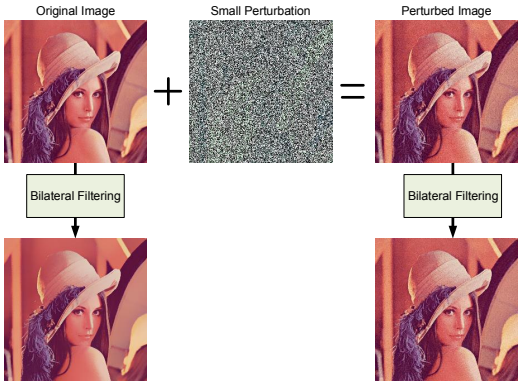


Fig. 1. Illustration of the small perturbation attack on edge-preserving image smoothing. The bilateral filtering operation initially smoothes the Lenna image correctly, but when small Gaussian random perturbation via a zero-mean distribution with a standard deviation $\sigma \in 10$, almost imperceptible to humans, is injected into the image, the smoothing result of the bilateral filter is inevitably invalid.

In this paper, we study bilateral filtering under threat of small perturbation attacks, which can be seen as a step towards enabling edge-preserving image smoothing to behave for various computer vision tasks in everyday front-end development. Specifically, a range kernel is used along with a spatial kernel in bilateral filtering. The input to the range kernel ensures the preservation of sharp edges that often possess a large intensity difference between the pixel of interest and its neighbor. The recent advances and breakthroughs in bilateral filtering mainly lie in adaptive mechanism of range kernel [1], which allows the width setting to change from pixel to pixel. However, such adaptive mechanisms simply seek an

*Corresponding author. This paper was supported in part by the Ministry of Science and Technology, Taiwan, under Grant MOST 108-2221-E-155-034-MY3; in part by the Fujian Provincial Youth Education and Scientific Research Project under Grant JAT200055; in part by the Fujian Natural Science Project under Grant No. 2021J05129.

adaptive function that transforms pixels to another representations in one-step manner. Those widths of the range kernel lack awareness of the small perturbation and can lead to sub-optimal performance in the edge-preserving image smoothing.

Unlike adaptive bilateral filters, we seek the help of both reinforcement learning (RL) that iteratively learns how to maximise reward by operating bilateral filtering from state to state and adaptive width-setting scheme that changes width setting from pixel to pixel through local image context. Hence, we propose a novel adaptive actor-critic bilateral filter (A2CBF) that leverages multi-agent RL algorithm [16] to adaptively expose the width of range kernel to the local image context in successive bilateral filtering operations. Our A2CBF is trained to learn width-setting policies that maximise the improvement of edge-preserving image smoothing through multiple episodes under small perturbation attacks, as shown in bottom part of Fig. 2. Specifically, we cast the problem of bilateral filtering into a Markov Decision Process (MDP), where action in each step is defined as a local width adjustment operation. Next, we propose an adaptive width-setting scheme with a multi-agent RL framework, PixelRL [16], to solve such MDP problem pixel by pixel over multiple episodes. We evaluate the accuracy of our A2CBF on eight benchmark datasets for the task of edge-preserving image smoothing under small perturbation attacks, where it is essentially excluded from the small perturbation and substantially outperforms state-of-the-art BF-based image filters.

To summarize, the contributions of this paper include (1) we provide a multi-step framework to perform edge-preserving image smoothing under threat of small perturbation attacks; (2) we show that, for the first time, attacking bilateral filtering is possible in edge-preserving image smoothing task; (3) we build state-to-state models to explore a policy for each pixel to get maximum reward in various states; and (4) we demonstrate that Gaussian random perturbation can still be an effective attack if an adaptive function is devised to simply transform pixels in one-step manner.

2. METHOD

2.1. System Overview

Given a i-th pixel of filtering input with Gaussian random perturbation at t-th time step, denoted as a state s_i^t, our target is to develop the optimal width-setting policies $\{\pi_1, ..., \pi_i, ..., \pi_n\} \in \boldsymbol{\pi}$ for the range kernel of bilateral filtering that can maximize the expected reward r_i^t at the i-th pixel by

$$\pi_i^* = \arg\max_{\pi_i} \mathbb{E}_{\pi_i}(\sum_{t=0}^{m} \gamma^t r_i^t), \tag{1}$$

where n and m are total numbers of pixels and steps, respectively; and γ^t is the t-th power of discount factor. In the case of edge-preserving image smoothing, we measure reward r_i^t

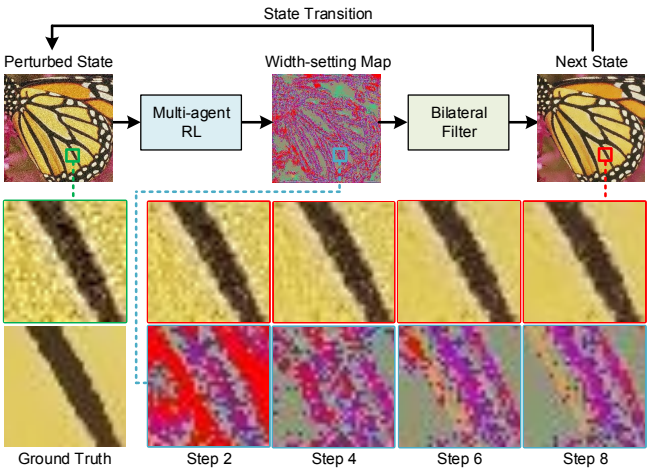


Fig. 2. We propose an adaptive actor-critic bilateral filter for adaptively exposing the width of range kernel to the local image context to achieve edge-preserving image smoothing under small perturbation attacks; we iteratively optimize our multi-agent RL model by maximising improvements that correspond to internal edge-preserving at different orientations and disparity-decreasing between nearby pixels on the non-edge area, respectively. Results from left to right, top to bottom: perturbed filtering input and ground truth, smoothed images and the corresponding width-setting maps at each step, respectively.

on i-th pixel by comparing squared error between two consecutive states with ground-truth smoothed pixel $\hat{s}_i$ as

$$r_i^t = \left(\hat{s}_i - s_i^t\right)^2 - \left(\hat{s}_i - s_i^{t+1}\right)^2. \tag{2}$$

Such definition encourages larger improvement from state to state, and it also reduces the variances of squared error across the episodes.

However, this multi-agent system is composed of n agents which is computationally impractical for large image sizes. To tackle the problem, we employ the PixelRL framework rather than training n agents for each individual pixel. Following the previous work [16], we share the parameters of all n agents through a fully convolutional network since it can be computationally parallelized on a GPU. Specifically, our multi-agent RL network takes entire pixels of state s^t as input, and then generates entire policies $\pi(a^t|s^t;\theta_a)$ and expected total rewards $V(s^t;\theta_c)$ from current state s^t that indicates how good each pixel in the state is, respectively. As illustrated in Fig. 3, we construct the backbone branch by using three dilated convolution layers with different dilated rates in which each of them is associated with a ReLU activation function. Moreover, the parallel actor branch θ_a and critic branch θ_c share the same structure with two dilated convolution layers followed by ReLU functions except the last activation functions.

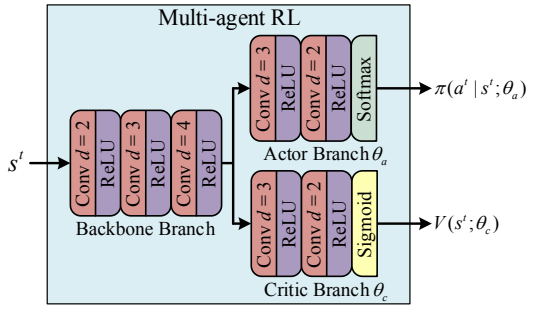
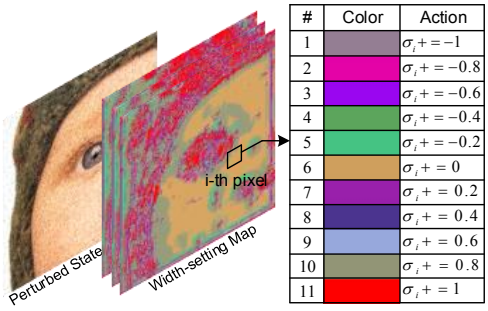


Fig. 3. Details of multi-agent RL framework. We feed entire pixels of perturbed state s^t into the backbone branch, and then the parallel actor branch θ_a and critic branch θ_c generate entire policies $\pi(a^t|s^t;\theta_a)$ and expected total rewards $V(s^t;\theta_c)$, respectively.



#	Color	Action
1		$\sigma_i += -1$
2		$\sigma_i += -0.8$
3		$\sigma_i += -0.6$
4		$\sigma_i += -0.4$
5		$\sigma_i += -0.2$
6		$\sigma_i += 0$
7		$\sigma_i += 0.2$
8		$\sigma_i += 0.4$
9		$\sigma_i += 0.6$
10		$\sigma_i += 0.8$
11		$\sigma_i += 1$

Fig. 4. Visualization of the adaptive width-setting scheme used in the range kernel of BF. For each pixel in the scheme, the width of range kernel σ_i needs to be locally adapted to both preserve strong edges and smooth the perturbation as steps progress, where distribution of σ_i is colored according to different actions for better visualization.

2.2. Adaptive Width-Setting Scheme

To be able to change the width setting of range kernel from pixel to pixel, each pixel of the state has an agent in our multi-agent RL network, and its policy is $\pi_i(a_i^t|s_i^t)$, where a_i^t denotes an action for each pixel in predefined action set $\mathcal{A}$. On top of the multi-agent RL framework, we design an adaptive width-setting scheme to make local width adjustment operations. Particularly, for one episode consisting of multiple states $\{s^0, ..., s^t, ..., s^m\}$, the agent of each pixel starts from s^0 and explores continuously towards s^m by

$$s_i^{t+1} = f\left(s_i^t; \sigma_i^t\right), \tag{3}$$

where $f(\cdot)$ denotes the ordinary bilateral filtering operation [12], and σ^t denotes the t-th adaptive width-setting scheme for the range kernel. At each time step t, the i-th agent will choose either increment or decrement by some constants as an action to adjust width setting of range kernel as follows:

$$\sigma_i^{t+1} = \sigma_i^t + a_i^t. \tag{4}$$

Table 1. Quantitative comparison on different variants of our constant (const) setting for width adjustment. We evaluate these variants on the average PSNR/SSIM values.

Const	Set5	Set14	BSDS100	Urban100	Manga109
0.1	27.86/0.77	25.73/0.71	25.75/0.71	23.86/0.74	24.42/0.78
0.2	**29.81/0.86**	**27.43/0.79**	**27.32/0.78**	**24.99/0.79**	**25.55/0.84**
0.3	29.26/0.84	27.03/0.77	26.99/0.76	24.68/0.78	25.20/0.82

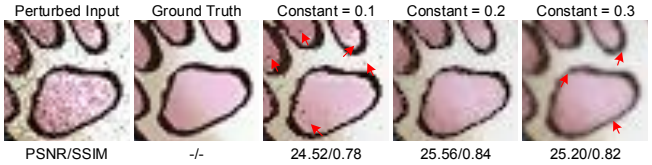


Fig. 5. Qualitative comparison on different variants of our constant setting for width adjustment. Red arrows indicate most obvious differences.

These actions are visualized in Fig. 4, where we use different color to represent different adjustment constants for each pixel in our predefined action set $\mathcal{A}$. By using this adaptive width-setting scheme, our multi-agent RL network should learn to adjust the width of range kernel with higher rewards.

3. EXPERIMENTS

3.1. Implementation Details

We have built our A2CBF with Tensorflow framework and conducted training process on the DIV2K dataset [17], where each image is cropped into a size of 256×256 and then added Gaussian random perturbation with standard deviation 0.1 and mean 0.0 as an input of our model. Our A2CBF was trained in 200 epochs using a batch size of 10 on a NVIDIA Titan RTX GPU, with the learning rate and discount factor γ setting as $10e-5$ and 0.95, respectively. In bilateral filtering operation, the width of spatial kernel is fixed as 30 according to [7]. The width of range kernel is initalized as 10 and adjusted by the adaptive width-setting scheme as steps progress. We evaluated our A2CBF on eight frequently used benchmark datasets: Set5 [18], Set14 [19], BSDS100 [20], Usban100 [21], Manga109 [22], Adobe FiveK [23], DTD [24], and Flickr1024 [25]. In addition, we use two measures to evaluate our A2CBF: PSNR and SSIM.

3.2. Ablation Study

We verify the effectiveness of different constants for width increment or decrement in our action set. Specifically, we design three variants of our constant setting ranging from 0.1 to 0.3 for comparison. Table 1 gives the results of this ab-

Table 2. Quantitative comparison results via the average PSNR/SSIM values on Set5, Set14, BSDS100, Usban100, Manga109, Adobe FiveK, DTD, and Flickr1024.

Filter	Set5	Set14	BSDS100	Urban100	Manga109	Adobe FiveK	DTD	Flickr1024
BF	20.99/0.27	20.91/0.39	20.94/0.40	20.89/0.55	21.20/0.52	21.13/0.31	20.98/0.43	21.10/0.47
FKBF	20.76/0.26	20.71/0.38	20.79/0.40	20.76/0.55	21.03/0.52	20.96/0.30	20.89/0.43	20.96/0.47
GPABF	20.66/0.26	20.61/0.38	20.65/0.39	20.65/0.54	20.96/0.52	20.75/0.30	20.78/0.42	20.81/0.46
OFABF	20.80/0.26	20.75/0.38	20.82/0.40	20.79/0.55	21.07/0.52	20.99/0.30	20.90/0.43	20.99/0.47
GABF	21.63/0.31	20.91/0.39	21.49/0.43	20.78/0.55	20.92/0.59	22.08/0.40	21.33/0.45	21.45/0.50
FABF	23.78/0.38	22.01/0.57	23.09/0.48	23.20/0.47	21.98/0.54	23.63/0.28	22.91/0.50	22.40/0.52
DBLBF	20.24/0.24	20.23/0.37	20.22/0.38	20.35/0.53	20.56/0.50	19.91/0.51	19.79/0.40	20.41/0.45
TPBF	25.03/0.47	24.22/0.54	24.50/0.57	23.39/0.66	23.65/0.66	25.36/0.36	24.57/0.58	24.09/0.61
A2CBF	**29.81/0.86**	**27.43/0.79**	**27.32/0.78**	**24.99/0.79**	**25.55/0.84**	**29.40/0.82**	**27.68/0.78**	**26.13/0.78**

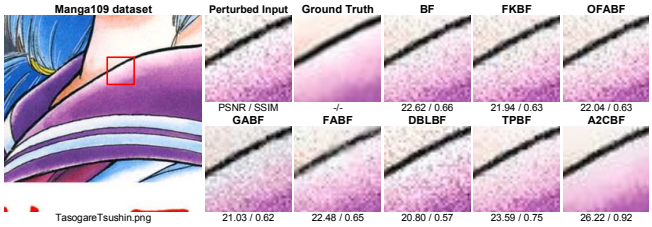


Fig. 6. Qualitative comparison on a testing image from Manga109. The red bounding box is the zoom-in view which indicates most obvious differences.

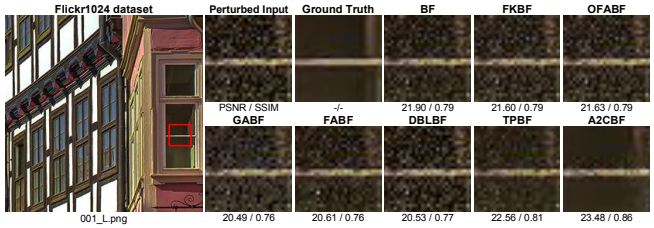


Fig. 7. Qualitative comparison on testing images from Flickr1024. The red bounding box is the zoom-in view which indicates most obvious differences.

lation study. As we can see, our A2CBF can attain the best performance when setting the constant as 0.2 among these configurations. To further illustrate the qualitative effect of constant settings, results of our A2CBF trained with different constants as our action set are demonstrated in Fig. 5. It is clear that the setting of constant to 0.2 preserves strong edges while smoothing most of perturbations.

3.3. Comparison with State-of-The-Arts

We compare our A2CBF with eight representative BF-based image filters in the literature: BF [12], FKBF [6], GPABF [2], OFABF [7], GABF [14], FABF [1], DBLBF [3], and TPBF [15]. Following the previous work [7], we set widths of spatial and range kernels in each compared BF-based filter as 30 and 10 for fair comparison, respectively. First, two qualitative examples from Manga109 and Flickr1024 are shown in Figs. 6-7. We can see that the difficulty of removing coarse textures increases as the perturbed pixels show up for most BF-based image filters. In contrast, it can be observed that images smoothed by our A2CBF are visually better compared to these filters. We conjecture the high performance of our A2CBF is due to the adaptive width-exploring to local image context in multi-step manner that maximises the improvement of edge-preserving image smoothing under small perturbation attacks. Meanwhile, the quantitative comparisons are shown in Table 2, where bold numbers denote the highest scores. Our A2CBF outperforms the other eight BF-based image filters consistently on all datasets. Particularly, we improve +4.04dB and +0.46 compared with TPBF (the second best) on PSNR and SSIM values in Adobe FiveK, respectively. These comparisons demonstrate that our A2CBF smooths out most coarse textures and perturbations with the adaptive width-setting scheme.

4. CONCLUSION

In this paper, we explored how attackers can craft small perturbation attacks against bilateral filtering. To improve the bilateral filtering, we proposed to learn an adaptive actor-critic bilateral filter via reinforcement learning for edge-preserving image smoothing under small perturbation attacks. By leveraging multi-agent RL algorithm and adaptively designed width-setting scheme, our A2CBF achieves more effective edge-preserving image smoothing in multi-step manner than state-of-the-art BF-based image filters. This is further confirmed by qualitative and quantitative evaluations on various datasets. By visualising the adaptive width-setting scheme made by our A2CBF, we explained why our A2CBF can make better edge-preserving image smoothing than others. This is the first work to apply adaptive width-setting scheme with RL on edge-preserving image smoothing under small perturbation attacks, and it can enable various computer vision works to robustly work in everyday front-end development.

5. REFERENCES

[1] R. G. Gavaskar and K. N. Chaudhury, "Fast adaptive bilateral filtering," *IEEE Trans. Image Process.*, vol. 28, no. 2, pp. 779–790, 2019.

[2] K. N. Chaudhury and S. D. Dabhade, "Fast and provably accurate bilateral filtering," *IEEE Trans. Image Process.*, vol. 25, no. 6, pp. 2519–2528, 2016.

[3] Michaël Gharbi, Jiawen Chen, Jonathan T Barron, Samuel W Hasinoff, and Frédo Durand, "Deep bilateral learning for real-time image enhancement," *ACM Trans. Graph.*, vol. 36, no. 4, pp. 118, 2017.

[4] Frédo Durand and Julie Dorsey, "Fast bilateral filtering for the display of high-dynamic-range images," *ACM Trans. Graph.*, vol. 21, no. 3, pp. 257–266, July 2002.

[5] Q. Yang, K. Tan, and N. Ahuja, "Real-time o(1) bilateral filtering," in *Proc. Conf. Comput. Vis. Pattern Recognit.*, 2009, pp. 557–564.

[6] S. Ghosh and K. N. Chaudhury, "On fast bilateral filtering using fourier kernels," *IEEE Signal Process. Lett.*, vol. 23, no. 5, pp. 570–573, 2016.

[7] S. Ghosh, P. Nair, and K. N. Chaudhury, "Optimized fourier bilateral filtering," *IEEE Signal Process. Lett.*, vol. 25, no. 10, pp. 1555–1559, 2018.

[8] Qifeng Chen, Jia Xu, and Vladlen Koltun, "Fast image processing with fully-convolutional networks," in *Proc. Int. Conf. Comput. Vis.*, 2017, pp. 2516–2525.

[9] L. Dai, M. Yuan, and X. Zhang, "Speeding up the bilateral filter: A joint acceleration way," *IEEE Trans. Image Process.*, vol. 25, no. 6, pp. 2657–2672, 2016.

[10] J. Van de Weijer and R. Van den Boomgaard, "Local mode filtering," in *Proc. Conf. Comput. Vis. Pattern Recognit.*, 2001, vol. 2, pp. II–II.

[11] F. Porikli, "Constant time o(1) bilateral filtering," in *Proc. Conf. Comput. Vis. Pattern Recognit.*, 2008, pp. 1–8.

[12] C. Tomasi and R. Manduchi, "Bilateral filtering for gray and color images," in *Proc. Int. Conf. Comput. Vis.*, 1998, pp. 839–846.

[13] Yinda Zhang and Thomas Funkhouser, "Deep depth completion of a single rgb-d image," in *Proc. Conf. Comput. Vis. Pattern Recognit.*, June 2018.

[14] B. H. Chen, Y. S. Tseng, and J. L. Yin, "Gaussian-adaptive bilateral filter," *IEEE Signal Process. Lett.*, vol. 27, pp. 1670–1674, 2020.

[15] B. H. Chen, H. Y. Cheng, Y. S. Tseng, and J. L. Yin, "Two-pass bilateral smooth filtering for remote sensing imagery," *IEEE Geosci. Remote. Sens. Lett.*, pp. 1–5, 2021.

[16] R. Furuta, N. Inoue, and T. Yamasaki, "Pixelrl: Fully convolutional network with reinforcement learning for image processing," *IEEE Trans. Multimedia*, vol. 22, no. 7, pp. 1704–1719, 2020.

[17] Eirikur Agustsson and Radu Timofte, "Ntire 2017 challenge on single image super-resolution: Dataset and study," in *Proc. Conf. Comput. Vis. Pattern Recognit.*, July 2017.

[18] Marco Bevilacqua, Aline Roumy, Christine Guillemot, and Marie line Alberi Morel, "Low-complexity single-image super-resolution based on nonnegative neighbor embedding," in *Proc. British Mach. Vis. Conf.*, 2012, pp. 135.1–135.10.

[19] Roman Zeyde, Michael Elad, and Matan Protter, "On single image scale-up using sparse-representations," in *Proc. Int. Conf. Curves and Surfaces*, 2012, pp. 711–730.

[20] D. Martin, C. Fowlkes, D. Tal, and J. Malik, "A database of human segmented natural images and its application to evaluating segmentation algorithms and measuring ecological statistics," in *Proc. Int. Conf. Comput. Vis.*, 2001, vol. 2, pp. 416–423 vol.2.

[21] J. Huang, A. Singh, and N. Ahuja, "Single image super-resolution from transformed self-exemplars," in *Proc. Conf. Comput. Vis. Pattern Recognit.*, 2015, pp. 5197–5206.

[22] Yusuke Matsui, Kota Ito, Yuji Aramaki, Azuma Fujimoto, Toru Ogawa, Toshihiko Yamasaki, and Kiyoharu Aizawa, "Sketch-based manga retrieval using manga109 dataset," *Multimed. Tools. Appl.*, vol. 76, no. 20, pp. 21811–21838, 2017.

[23] V. Bychkovsky, S. Paris, E. Chan, and F. Durand, "Learning photographic global tonal adjustment with a database of input / output image pairs," in *Proc. Conf. Comput. Vis. Pattern Recognit.*, 2011, pp. 97–104.

[24] M. Cimpoi, S. Maji, I. Kokkinos, S. Mohamed, and A. Vedaldi, "Describing textures in the wild," in *Proc. Conf. Comput. Vis. Pattern Recognit.*, 2014, pp. 3606–3613.

[25] Y. Wang, L. Wang, J. Yang, W. An, and Y. Guo, "Flickr1024: A large-scale dataset for stereo image super-resolution," in *Proc. Int. Conf. Comput. Vis. Workshop*, 2019, pp. 3852–3857.