

# Noise Reduction on Bracketed Images for High Dynamic Range Imaging

Seong-O Shim

Department of Computer & Network Engineering  
College of Computer Science and Engineering  
University of Jeddah  
Jeddah, Saudi Arabia

**Abstract**—The quality of a high dynamic range (HDR) image produced from bracketed images taken at different camera exposure times can be degraded by noise contained in bracketed images. In this paper, we propose a noise reduction method on bracketed images based on exposure time ratio. First, for each pixel pair of a same scene point lying on two different images, the ratio of their intensity values is compared with the ratio of exposure times of the images on which the pixels are lying. If the compared ratios are close, these two pixels are included in noise-free pixels set. The complement of noise-free pixels set is defined as noisy pixels set. Then, the intensity value of each pixel in noisy pixels set is corrected by its expected value computed from noise-free pixel of the same scene point lying on another image. Experimental results show that all the noisy intensity values can be correctly restored from noise-free pixels except the saturated pixels. Denoising results by PSNR show that the proposed method outperforms the other recent denoising methods such as based-on pixel density filter (BPDF), noise adaptive fuzzy switching median filter (NAFSMF), and adaptive Riesz mean filter (ARmF).

**Keywords**—Image denoising; high dynamic range imaging; noise detection; noise removal; image restoration

## I. INTRODUCTION

The human visual system (HVS) can perceive higher dynamic range (i.e., the ratio between the maximum and minimum intensity values in an image) than most of the cameras. This causes the gap between the visual quality of image captured from standard camera and the quality of actual scene perceived by human eye. One way to reduce this gap is using exposure bracketing technique [1]-[3] which is the process to achieve high dynamic range (HDR) image from merging multiple low dynamic range (LDR) images, which is called bracketed images, acquired by gradually increasing the camera exposure settings.

A typical problem in constructing HDR image is that if there are moving objects while capturing a series of bracketed images, they appear blended in merged HDR image which is called as ghosting artefacts. Thus, a lot of research has been conducted to remove ghosting artefacts in HDR imaging [4]-[9]. Like the standard images, the images acquired in bracketing process suffer from noise that usually occurs during acquisition. And this noise can degrade the quality of the final constructed HDR image. Therefore, the denoising of bracketed

images is one of the most important processes to obtain high quality HDR image. Since each bracketed image can be denoised using existing image denoising methods [10]-[14], more attention of research in HDR imaging has been focused on removing ghosting artefacts than image denoising. In this article, we propose a method to exploit information during exposure bracketing process which can be used to reduce noise in bracketed images.

Most of the exposure bracketing techniques use three images: slightly underexposed image, correctly exposed image and slightly overexposed image. Thus, we used three bracketed images in this work. It is found that the exposure time ratio between two differently exposed images is constant and the value is equivalent to the ratio between intensity values of the two corresponding pixels [15]. Therefore, if the ratio of the two corresponding pixels' intensity values is close to the exposure time ratio, these two pixels can be assumed to be noise-free. In this way, all the pixels in the bracketed images are checked whether they are noise-free, and the classified noise-free pixels are grouped into the noise-free pixels set. The noisy pixels set is made by taking complement of the noise-free pixels set. Then, the intensity values of the pixels in noisy pixels set are corrected by their expected values computed from noise-free pixels of the same scene point lying on another image. Experimental results on several datasets illustrate that the proposed method can correctly restore intensity value of any noisy pixel when there exists corresponding noisy-free pixel of the same scene point in another image. However, noisy intensity values of the saturated pixels that have maximum intensity values cannot be correctly restored because once an intensity value reaches a maximum value, it does not increase as the sensor exposure of a pixel increases.

The rest of this paper is organized as follows. Section II presents the proposed algorithm of based-on gamma-corrected exposure time ratio filter (BGEF). In Section III, we present experimental analysis on different datasets. Conclusions are given in Section IV.

## II. PROPOSED ALGORITHM

Through the article, we let  $L = [l_{ij}]_{r \times c}$  be an image where  $l_{ij}$  is the measured pixel intensity value at location  $(i, j)$ ,  $r$  is the number of pixels in a row of  $L$ ,  $c$  is the number of pixels in a column of  $L$ . The total quantity of light accumulated at

This work was funded by Deanship of Scientific Research (DSR), University of Jeddah, under grant No. UJ-02-034-DR.

camera sensor location  $(i, j)$  for  $\Delta t$  units of time produces a sensor exposure  $e_{ij} \cdot \Delta t$ , where  $e_{ij}$  is sensor irradiation at  $(i, j)$ .

The measured pixel intensity value  $l_{ij}$  at location  $(i, j)$  is proportional to the sensor exposure  $e_{ij} \cdot \Delta t$  scaled by the camera response function  $f$  as  $f(e_{ij} \cdot \Delta t)$ . Among various methods to estimate  $f$  [16], [17], gamma curve approximation is widely used by many camera manufacturers [18]. Assuming the camera response function to be a gamma curve, the pixel intensity at position  $(i, j)$  can be expressed as

$$l_{ij} = f(e_{ij} \cdot \Delta t) \approx (e_{ij} \cdot \Delta t)^\gamma \quad (1)$$

For each pixel location  $(i, j)$ , the gamma-corrected exposure time ratio  $\alpha_s^t(i, j)$  between differently exposed images  $L_s$  and  $L_t$  is defined in [15] as

$$\alpha_s^t(i, j) = \frac{l_{ij}^t}{l_{ij}^s} \approx \left( \frac{e_{ij} \cdot \Delta t_t}{e_{ij} \cdot \Delta t_s} \right)^\gamma = \left( \frac{\Delta t_t}{\Delta t_s} \right)^\gamma, \quad (2)$$

where  $l_{ij}^s$  and  $l_{ij}^t$  are pixel intensity values at  $(i, j)$  in image  $L_s$  and  $L_t$  respectively.

It can be observed in (2) that the gamma-corrected exposure time ratio  $\alpha_s^t$  is constant for all pixel location  $(i, j)$ . From (2), we can estimate the pixel intensity value  $\hat{l}_{ij}^t$  at  $(i, j)$  in image  $L_t$  from the observed intensity value  $l_{ij}^s$  in image  $L_s$  as  $\hat{l}_{ij}^t = \alpha_s^t \cdot l_{ij}^s$ . Thus, if the estimated pixel intensity value  $\hat{l}_{ij}^t$  is different from the observed value  $l_{ij}^t$ , one or both of  $l_{ij}^s$  and  $l_{ij}^t$  can be thought to be corrupted by noise.

#### A. Selection of Noisy Pixels

We obtain three images  $L_1, L_2, L_3$  of the same scene by gradually increasing the camera exposure time ( $\Delta t_1 < \Delta t_2 < \Delta t_3$ ) and compute  $\alpha_1^2, \alpha_2^3$  and  $\alpha_1^3$  from (2). For each pixel location  $(i, j)$  in each image frame, we compute the estimated pixel intensity value from another image frame as

$$\begin{aligned} \hat{l}_{ij}^1 &= \alpha_3^1 \cdot l_{ij}^3 \\ \hat{l}_{ij}^2 &= \alpha_1^2 \cdot l_{ij}^1 \\ \hat{l}_{ij}^3 &= \alpha_2^3 \cdot l_{ij}^2 \end{aligned} \quad (3)$$

Let  $p_{ij}^k$  be pixel at location  $(i, j)$  in image  $L_k$ ,  $P_n$  is noisy pixels set,  $P_f$  is noise-free pixels set and  $P_u$  is collection of all the pixels.  $P_n$  and  $P_f$  are initialized as empty set. Gamma-corrected exposure time ratios  $\alpha_1^2, \alpha_2^3$  and  $\alpha_1^3$  cannot be corrupted values since they are computed from pre-determined camera parameters (i.e., exposure time and gamma value) as shown in (2). However, pixel intensity values  $l_{ij}^1, l_{ij}^2$  and  $l_{ij}^3$  can be corrupted by various types of noises. Thus, for each pixel  $p_{ij}^k$ , we guess from (3) whether the pixel belongs to  $P_n$  or  $P_f$  as

$$|l_{ij}^2 - \hat{l}_{ij}^2| < \epsilon \cdot l_{ij}^2 \rightarrow P_f = P_f \cup \{p_{ij}^1, p_{ij}^2\},$$

$$|l_{ij}^3 - \hat{l}_{ij}^3| < \epsilon \cdot l_{ij}^3 \rightarrow P_f = P_f \cup \{p_{ij}^2, p_{ij}^3\},$$

$$|l_{ij}^1 - \hat{l}_{ij}^1| < \epsilon \cdot l_{ij}^1 \rightarrow P_f = P_f \cup \{p_{ij}^3, p_{ij}^1\},$$

$$P_n = P_u - P_f, \quad (4)$$

where  $\epsilon$  is predefined tolerance and it is set to 0.1 in this work. When the difference between the observed pixel intensity value and the estimated pixel intensity value is small, both of the two pixels (i.e., the pixel whose intensity value is estimated and the pixel from which the other pixel's intensity value is estimated) can be assumed to be noise-free pixels. Finally, the noisy pixels set is constructed by subtracting noise-free pixels from the collection of all pixels.

#### B. Correction of Noisy Pixels

The intensity values of pixels classified as noisy in previous section can be corrected from the intensity values of noise-free pixels in another image lying on the same location. Equation (4) indicates that each pixel location  $(i, j)$ , there are three cases: all three pixels are noise-free, two pixels are noise-free and the remaining one pixel is noisy, all three pixels are noisy. The case that two pixels are noisy and the remaining one pixel is noise-free cannot occur in (4). In this method, the noisy pixel in the second case can be corrected. For each pixel location  $(i, j)$ , if only one value among  $l_{ij}^1, l_{ij}^2$  and  $l_{ij}^3$  is noisy, the noisy intensity value can be corrected as

$$(p_{ij}^1 \in P_n) \wedge (p_{ij}^2 \in P_f) \wedge (p_{ij}^3 \in P_f) \rightarrow l_{ij}^1 = \hat{l}_{ij}^1,$$

$$(p_{ij}^1 \in P_f) \wedge (p_{ij}^2 \in P_n) \wedge (p_{ij}^3 \in P_f) \rightarrow l_{ij}^2 = \hat{l}_{ij}^2,$$

$$(p_{ij}^1 \in P_f) \wedge (p_{ij}^2 \in P_f) \wedge (p_{ij}^3 \in P_n) \rightarrow l_{ij}^3 = \hat{l}_{ij}^3 \quad (5)$$

In (5), the noisy pixel intensity value is corrected by its estimated value in (3) since the pixel, from which the value is estimated, is noise-free. However, if all three pixels lying on same location are noisy, the noisy intensity values cannot be corrected in this method.

### III. EXPERIMENTAL RESULTS

The proposed method was applied on three sample datasets in [6] – ‘Yard’, ‘Shop’ and ‘Cars’ scenes. Each dataset has three images of a scene captured by doubling the camera exposure time. The sample datasets are shown in Fig. 1.

Out of several types of image noise that occurs during image acquisition and transmission, the impulse noise is one of common types of noise which includes salt-and-pepper noise (SPN) [19]-[21] and random-valued impulse noise [22], [23]. SPN is caused by sudden disturbances in the signal and the pixels affected by SPN hold a maximum or a minimum gray value. In experiments, we imposed SPN noises on the bracketed images with various ratios.

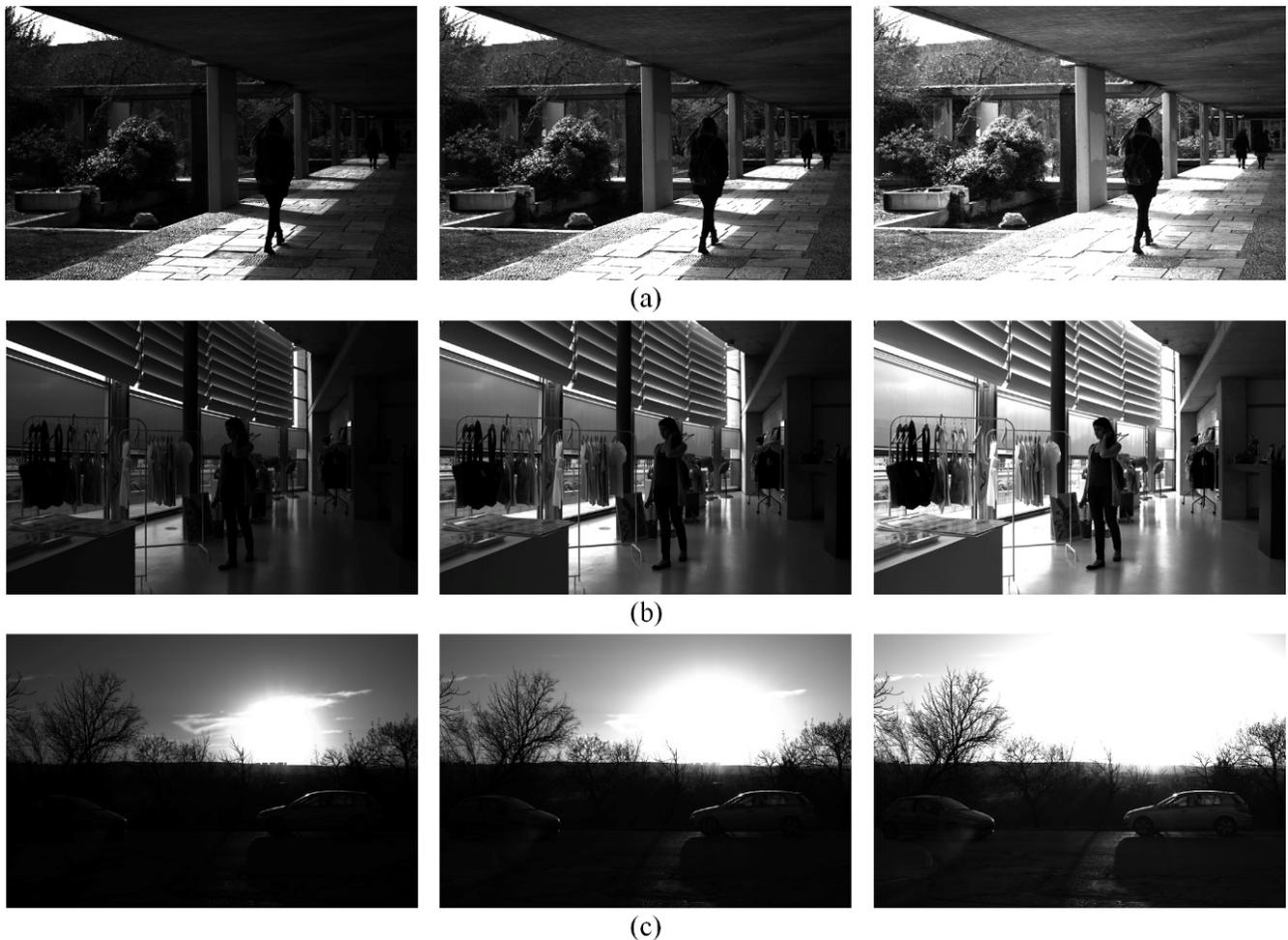


Fig. 1. Three Sample Scenes: (a) Bracketed Images of 'Yard' Scene, (b) Bracketed Images of 'Shop' Scene, (c) Bracketed Images of 'Cars' Scene.

Fig. 2 shows the denoising results of the middle image by the proposed method for 'Yard' scene where the first column shows the corrupted middle image by SPN, the second column shows the denoising results in case only the middle image is corrupted by SPN and the third column shows the denoising results in case all the three images are corrupted by SPN. Denoising for only one image is corrupted shows good results and denoising quality does not decrease much as the noise level increases. The floor parts and the top-left corner of the scene are saturated (full white) in all three images in Fig. 1(a). In these saturated regions, the proposed filter does not work well since the method rely on intensity ratio between images and the intensity ratio on saturated region does not reflect correct exposure time ratio. However, for all three images are corrupted, the denoising quality decreases as the noise level increases. For a certain location  $(i, j)$ , the proposed method can correct the noisy intensity value when only one pixel is noisy

out of three pixels as in (5). It means, when only one image is corrupted by noise, theoretically 100% of noisy pixels except saturated pixels can be corrected by corresponding two other noise-free pixels lying on the same location in other two images. However, when all the three images are corrupted by  $\alpha$  % noise level, the probability of a noisy pixel can be corrected is equivalent to the probability that the corresponding two other pixels are noise-free which is  $(1 - \alpha/100)^2 \times 100$  %; thus, only some portion of noisy pixels can be corrected. The same experiments were conducted on 'Shop' and 'Cars' scenes and their denoising results are shown in Fig. 3 and Fig. 4, respectively. In both experiments, we can see the expected results as in Fig. 2. In 'Cars' scene in Fig. 1, the saturated parts around sun in the sky occupy large areas in the image. Thus, the denoising failure around sun is notable in this scene.





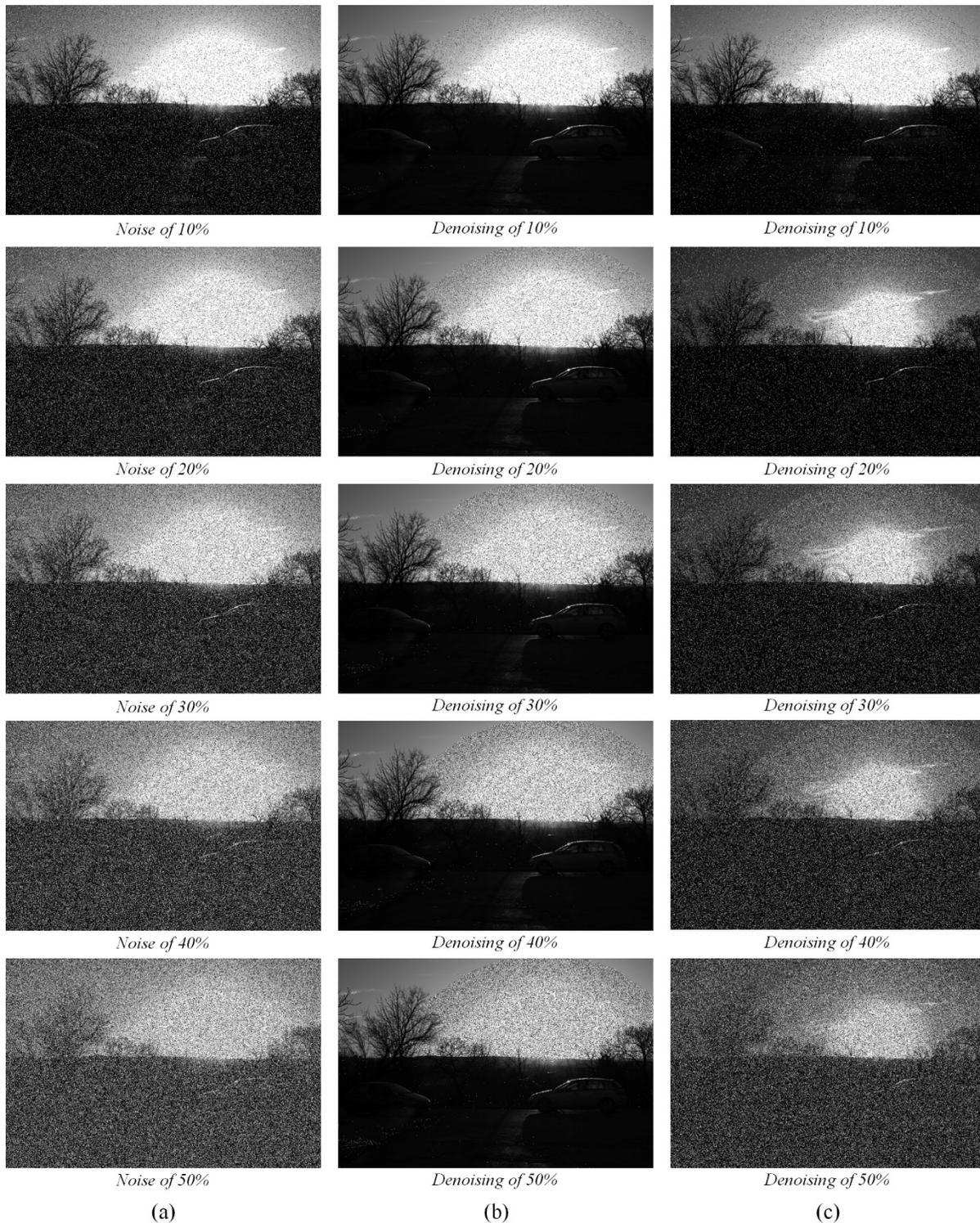


Fig. 4. Denoising Results of the Middle Image by the Proposed Method for 'Cars' Scene: (a) Corrupted Images with Different SPN Ratios, (b) Denoising Results in Case Only the Middle Image is Corrupted by SPN, (c) Denoising Results in Case All the Three Images are Corrupted by SPN.

For quantitative assessment of the effectiveness of image denoising, we use the common error metric peak signal-to-noise ratio (PSNR) which is defined as

$$PSNR(U, V) = 10 \log_{10} \left( \frac{peakval^2}{MSE(U, V)} \right), \quad (6)$$

where  $peakval$  is the maximum intensity value in image datatype and  $MSE$  stands for mean square error defined as

$$MSE(U, V) = \frac{1}{rc} \sum_{i=1}^r \sum_{j=1}^c (u_{ij} - v_{ij})^2, \quad (7)$$

where  $U = [u_{ij}]$  is a ground truth and  $V = [v_{ij}]$  is an evaluated (denoised) image. The higher value of PSNR indicates a better image quality.

In Table I, PSNR results of the proposed BGEF method on middle images are compared with recent SPN filtering techniques – based-on pixel density filter (BPDF) [24], noise adaptive fuzzy switching median filter (NAFSMF) [25], adaptive Riesz mean filter (ARmF) [26]. PSNR results show that BGEF performs better than the other filtering methods in all three scenes. Since BGEF reduces noise in bracketed

images using exposure time ratio computed from a priori knowledge, it can be considered as preprocessing technique before applying existing denoising filters. Thus, we also tried to apply BPDF, NAFSMF and ARmF after applying the proposed BGEF, each of which is termed as BGEF-BPDF, BGEF-NAFSMF and BGEF-ARmF respectively. We can see these combined filters have further improved the results when only one image is corrupted. However, in some cases when all three images are corrupted, the combined filters showed worse performance than single BGEF filter as the noise level increases.

TABLE I. PSNR RESULTS OF THE METHODS FOR THREE SCENES ('YARD', 'SHOP' AND 'CARS') WITH DIFFERENT SPN RATIOS

Scene	Filter	10%	20%	30%	40%	50%	60%	70%	80%	90%
Yard	BPDF	14.7948	11.4381	6.2123	5.6429	5.1229	4.6501	4.1760	3.7424	3.2312
(1 image	NAFSMF	15.0718	12.0522	7.1495	6.0000	4.7246	3.6800	3.0632	2.6904	2.3894
corrupted)	ARmF	15.0571	11.9665	6.8119	6.1399	5.4926	4.8384	4.1750	3.5102	2.8162
Yard	<b>BGEF</b>	<b>21.8402</b>	<b>18.7996</b>	<b>16.1532</b>	<b>13.9080</b>	<b>13.4803</b>	<b>13.0882</b>	<b>12.7175</b>	<b>12.4245</b>	<b>12.1324</b>
(1 image	BGEF-BPDF	29.6673	25.5969	22.2504	19.0001	18.2520	17.6507	16.7814	16.2212	15.5347
corrupted)	BGEF-NAFSMF	30.4498	27.2424	24.6925	22.2148	21.7592	21.3514	20.9294	20.4732	20.1004
	BGEF-ARmF	30.9520	27.7618	25.2913	22.9088	22.4732	22.0864	21.6867	21.2919	20.9479
Yard	<b>BGEF</b>	<b>18.9944</b>	<b>14.4602</b>	<b>8.4810</b>	<b>6.7813</b>	<b>6.1835</b>	<b>5.6645</b>	<b>5.2017</b>	<b>4.8173</b>	<b>4.4624</b>
(3 images	BGEF-BPDF	21.5826	15.9649	8.3388	6.4079	5.6515	4.9834	4.3708	3.8375	3.2594
corrupted)	BGEF-NAFSMF	21.7394	16.2873	9.3411	7.4919	6.0903	4.4046	3.3017	2.7642	2.4070
	BGEF-ARmF	21.8109	16.3091	9.0471	7.0728	6.1896	5.3225	4.4799	3.6633	2.8587
Shop	BPDF	14.7087	11.3417	9.3041	7.8520	6.7130	5.7847	4.9741	4.2544	3.5631
(1 image	NAFSMF	15.0598	12.0510	10.2880	9.0314	7.9707	6.5051	4.4296	3.1085	2.5226
corrupted)	ARmF	15.0192	11.9364	10.0583	8.6448	7.4472	6.3683	5.3174	4.2662	3.1910
Shop	<b>BGEF</b>	<b>24.9141</b>	<b>21.9940</b>	<b>20.1777</b>	<b>18.9365</b>	<b>17.9786</b>	<b>17.1791</b>	<b>16.5227</b>	<b>15.9288</b>	<b>15.4317</b>
(1 image	BGEF-BPDF	31.4590	27.0027	24.3549	22.8316	21.7053	20.8822	20.1632	19.4482	18.8491
corrupted)	BGEF-NAFSMF	35.7117	32.6009	30.8208	29.4552	28.3801	27.2541	25.9238	24.8079	23.9785
	BGEF-ARmF	35.9673	32.8380	31.0408	29.6687	28.5375	27.5648	26.5999	25.7053	24.7991
Shop	<b>BGEF</b>	<b>20.2299</b>	<b>15.2058</b>	<b>12.1941</b>	<b>10.1033</b>	<b>8.5269</b>	<b>7.2946</b>	<b>6.3060</b>	<b>5.5085</b>	<b>4.8720</b>
(3 images	BGEF-BPDF	21.8178	15.9985	12.5090	10.0352	8.1471	6.6743	5.4721	4.4840	3.6301
corrupted)	BGEF-NAFSMF	22.1633	16.4435	13.1993	10.9770	9.3180	7.9352	5.7493	3.4285	2.5703
	BGEF-ARmF	22.1693	16.4216	13.1239	10.8032	8.9757	7.4191	5.9865	4.6163	3.2977
Cars	BPDF	14.8187	11.3948	9.2533	7.8058	6.7074	5.7750	5.0027	4.2856	3.2390
(1 image	NAFSMF	15.3301	12.3603	10.5488	9.2893	8.2503	6.7610	4.7317	3.4027	2.5738
corrupted)	ARmF	15.2982	12.2514	10.3253	8.9084	7.7372	6.6436	5.6071	4.5534	2.9151
Cars	<b>BGEF</b>	<b>19.6201</b>	<b>16.6098</b>	<b>14.7801</b>	<b>13.6059</b>	<b>12.6231</b>	<b>11.8479</b>	<b>11.1817</b>	<b>10.5849</b>	<b>9.8555</b>
(1 image	BGEF-BPDF	25.2672	20.6474	18.0166	16.6609	15.6628	14.8775	14.3970	13.8015	13.2473
corrupted)	BGEF-NAFSMF	29.8784	26.8546	25.1235	23.8845	22.8178	21.5618	19.9973	18.8440	17.9175
	BGEF-ARmF	30.0175	26.9541	25.1497	23.8159	22.6637	21.6471	20.7404	19.8288	18.4150
Cars	<b>BGEF</b>	<b>17.8410</b>	<b>13.6806</b>	<b>11.0502</b>	<b>9.2383</b>	<b>7.8255</b>	<b>6.6908</b>	<b>5.7810</b>	<b>5.0148</b>	<b>4.1730</b>
(3 images	BGEF-BPDF	20.7992	15.3088	11.9637	9.7250	7.9892	6.5736	5.4645	4.4877	3.3172
corrupted)	BGEF-NAFSMF	21.9909	16.5503	13.3413	11.1763	9.5627	8.1487	6.0204	3.7153	2.6201
	BGEF-ARmF	22.0124	16.5366	13.2700	11.0044	9.2214	7.6584	6.2648	4.8930	3.0291

#### IV. CONCLUSIONS

A noise reduction method based on exposure time ratio on three bracketed images has been proposed. For each location  $(i, j)$ , the ratio of intensity values of the corresponding pixels between two images is equivalent to gamma-corrected exposure time ratio which can be computed from a priori knowledge of camera parameters. Thus, when a ratio of two corresponding pixels' intensity values is different from gamma-corrected exposure time ratio, one or both of them are classified as noisy pixels. Then, the classified noisy pixels are corrected from noise-free pixels on the same location lying in another image using gamma-corrected exposure time ratio as a cue. Experimental results show that if only one image out of three is corrupted by noise, most of the noisy pixels' intensity values except saturated pixels can be correctly restored by the proposed method regardless of noise level imposed on the image. However, if all the three images are corrupted by noise, the denoising performance decreases as the noise level increases. Experimental results on several datasets using PSNR show that the denoising performance of the proposed method is better than the recent denoising filters BPDF, NAFSMF and ARmF. When the proposed method was used as a pre-processing filter and the existing noise specific denoising filters were applied on the processed images, the denoising performance has further improved.

#### REFERENCES

- [1] S. Im, H. G. Jeon, and I. S. Kweon, "Robust Depth Estimation Using Auto-Exposure Bracketing," *IEEE Trans. Image Process.*, vol. 28, no. 5, pp. 2451–2464, 2018.
- [2] A. Leong-Hoi, P. C. Montgomery, B. Serio, P. Twardowski, and W. Uhring, "High-dynamic-range microscope imaging based on exposure bracketing in full-field optical coherence tomography," *Opt. Lett.*, vol. 41, no. 7, pp. 1313–1316, 2016.
- [3] T. Liu, H. Liu, Y. Wu, B. Yin, and Z. Wei, "Exposure Bracketing Techniques for Camera Document Image Enhancement," *Appl. Sci.*, vol. 9, no. 21, pp. 4529, 2019.
- [4] Q. Yan et al., "Deep hdr imaging via a non-local network," *IEEE Trans. Image Process.*, vol. 29, pp. 4308–4322, 2020.
- [5] P. Sen et al., "Robust patch-based HDR reconstruction of dynamic scenes," *ACM Trans. Graph.*, vol. 31, no. 6, pp. 203–1, 2012.
- [6] O. T. Tursun, A. O. Akyüz, A. Erdem, and E. Erdem, "The state of the art in HDR deghosting: a survey and evaluation," *Comput. Graph. Forum*, vol. 34, pp. 683–707, 2015.
- [7] S. O. Shim and I. R. Khan, "Removal of ghosting artefacts in HDRI using intensity scaling cue," *SIGGRAPH Asia 2017 Technical Briefs*, pp. 1–4, 2017.
- [8] H. Guo, B. Sheng, P. Li, and C. P. Chen, "Multiview High Dynamic Range Image Synthesis Using Fuzzy Broad Learning System," *IEEE Trans. Cybern.*, pp. 1–13, 2019.
- [9] Q. Yan, Y. Zhu, and Y. Zhang, "Robust artifact-free high dynamic range imaging of dynamic scenes," *Multimed. Tools. Appl.* vol. 78, no. 9, pp. 11487–11505, 2019.
- [10] M. Haider, M. U. Riaz, I. Touqir, and A. M. Siddiqui, "Denoising in wavelet domain using probabilistic graphical models," *Int. J. Adv. Comput. Sci. Appl.*, vol. 7, no. 11, pp. 317–321, 2016.
- [11] X. Yang, Y. Xu, Y. Quan, and H. Ji, "Image Denoising via Sequential Ensemble Learning," *IEEE Trans. Image Process.* vol. 29, pp. 5038–5049, 2020.
- [12] A. M. Elmogy, E. Mahmoud, and F. A. Turki, "Image noise detection and removal based on enhanced gridLOF algorithm," *Int. J. Adv. Comput. Sci. Appl.*, vol. 8, no. 12, pp. 454–462, 2017.
- [13] C. Tian, Y. Xu, and W. Zuo, "Image denoising using deep CNN with batch renormalization," *Neural Netw.*, vol. 121, pp. 461–473, 2020.
- [14] Y. Hou et al., "NLH: A blind pixel-level non-local method for real-world image denoising," *IEEE Trans. Image Process.*, vol. 29, pp. 5121–5135, 2020.
- [15] S. O. Shim, "Estimation of gamma-corrected exposure time ratio in multi-exposure images for removal of moving objects," *Appl. Opt.*, vol. 59, no. 13, pp. 4076–4080, 2020.
- [16] H. Farid, "Blind Inverse Gamma Correction," *IEEE Trans. Image Process.*, vol. 10, no. 10, pp. 1428–1433, 2001.
- [17] M. D. Grossberg and S. K. Nayar, "Determining the Camera Response from Images: What is Knowable?" *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 25, no. 11, pp. 1455–1467, 2003.
- [18] M. D. Grossberg and S. K. Nayar, "What is the Space of Camera Response Functions," *IEEE Conf. Computer Vision and Pattern Recognition, USA*, vol. 2, pp. II-602, 2003.
- [19] L. Chun, S. Bishen, L. Shaohui, T. Kun, and M. Yingrui, "Iterative Removing Salt and Pepper Noise based on Neighbourhood Information," *Int. J. Adv. Comput. Sci. Appl.*, vol. 9(1), pp.261–265, 2018.
- [20] M. S. Christo, K. Vasanth, and R. Varatharajan, "A decision based asymmetrically trimmed modified winsorized median filter for the removal of salt and pepper noise in images and videos," *Multimed. Tools. Appl.*, vol. 79, no. 1, pp. 415–432, 2020.
- [21] Y. Xing, J. Xu, J. Tan, D. Li, and W. Zha, "Deep CNN for removal of salt and pepper noise," *IET Image Process.*, vol. 13, no. 9, pp.1550–1560, 2019.
- [22] A. Roy, L. Manam, and R. H. Laskar, "Region adaptive fuzzy filter: an approach for removal of random-valued impulse noise," *IEEE Trans. Ind. Electron.*, vol. 65, no. 9, pp.7268–7278, 2018.
- [23] J. Chen, G. Zhang, S. Xu, and H. Yu, "A blind CNN denoising model for random-valued impulse noise," *IEEE Access*, vol. 7, pp.124647–124661, 2019.
- [24] U. Erkan and L. Gökrem, "A new method based on pixel density in salt and pepper noise removal," *Turkish J. Elect. Eng. Comput. Sci.*, vol. 26, no. 1, pp. 162–171, 2018.
- [25] K. K. V. Toh and N. A. M. Isa, "Noise adaptive fuzzy switching median filter for salt-and-pepper noise reduction," *IEEE Signal Process. Lett.*, vol. 17, no. 3, pp. 281–284, 2010.
- [26] S. Enginoğlu, U. Erkan, and S. Memiş, "Pixel similarity-based adaptive Riesz mean filter for salt-and-pepper noise removal," *Multimed. Tools. Appl.*, vol. 78, no. 24, pp. 35401–35418, 2019.