Estimation of Noise Level in Complex Textured Images and Monte Carlo-Rendered Images

I-Gil Kim*
KT Institute of Convergence Technology
Seoul - KOREA
[e-mail: igil.kim@kt.com]
*Corresponding author: I-Gil Kim

Received August 10, 2015; revised October 19, 2015; revised November 10, 2015; accepted November 13, 2015; published January 31, 2016

Abstract

The several noise level estimation algorithms that have been developed for use in image processing and computer graphics generally exhibit good performance. However, there are certain special types of noisy images that such algorithms are not suitable for. It is particularly still a challenge to use the algorithms to estimate the noise levels of complex textured photographic images because of the inhomogeneity of the original scenes. Similarly, it is difficult to apply most conventional noise level estimation algorithms to images rendered by the Monte Carlo (MC) method owing to the spatial variation of the noise in such images. This paper proposes a novel noise level estimation method based on histogram modification, and which can be used for more accurate estimation of the noise levels in both complex textured images and MC-rendered images. The proposed method has good performance, is simple to implement, and can be efficiently used in various image-based and graphic applications ranging from smartphone camera noise removal to game background rendition.

Keywords: Noise level estimation, Monte Carlo rendering, inhomogeneous noise, histogram modification

http://dx.doi.org/10.3837/tiis.2016.01.022 **ISSN : 1976-7277**

1. Introduction

Noise level estimation is an important aspect of image processing. This is because of the inevitability of noise generation during the acquisition of image data and the processing and transmission of the image. Various denoising-based noise estimation algorithms have thus been developed. With respect to the adopted denoising approach, such algorithms can be classified chronologically into four groups, namely, filter-based, block-based, patch-based, and transform-based algorithms. In filter-based algorithms [1][2][3][4], low-pass filtering is used to blur the noisy image to suppress its structure. A difference image is then generated by subtracting the filtered image from the original one, and the difference is used to estimate the noise level. In the block-based algorithms, the image is partitioned into a sequence of blocks, and the noise level is estimated by calculating the weighted noise level, namely, the average of the noise levels of the most homogeneous blocks [5][6][7]. Incidentally, a hybrid algorithm that combines the methods of the two preceding algorithms has also been proposed [8][9]. In the patch-based algorithms [10][11][12], principle component analysis (PCA) is used to divide the noisy image into strongly textured and weakly textured patches. The main idea is to use the weakly textured patches in the noisy image to estimate the noise level. In the transform-based algorithms [13][14][15], singular value decomposition (SVD) is used to separate the noise from the image. The noise level can then be easily estimated in the SVD domain using the artificial noise addition technique.

However, it is difficult to use these algorithms to accurately determine the noise level in complex textured images. This is because such images do not contain sufficient homogeneous regions from which the noise can be easily extracted. Moreover, the algorithms cannot be easily used to estimate the noise level in computer graphics. Generally, Monte Carlo (MC) rendering can be better used to generate photorealistic images compared to other rendering techniques. However, the low sampling rate of MC rendering causes noise contamination of the image. Moreover, most of the powerful denoising algorithms used for image processing can only handle spatially invariant noise, which is noise that does not globally change across the image. This makes them unsuitable for MC-rendered images, which often contain local and spatially varying noise. Nevertheless, an algorithm for estimating spatially invariant noise has been successfully used to denoise MC-rendered images [16]. This involves, firstly, accurate determination of the noise level standard deviation in every local region of the noisy image. This requires segmentation and localization of the noise because of its spatial variation [17]. Subsequently, multilevel spatially invariant denoising is performed on the entire image using a small set of noise parameters. Finally, the noise-free image is produced by combining the results for the different segments. The method is a post-process filtering method wherein a sample image is first generated by MC rendering, and then denoised.

This paper proposes a noise level estimation method based on histogram modification in a correlated color space. The proposed method is robust and applicable to difficult noise estimation environments such as in complex textured images or MC-rendered images. The method may be used to reduce noise-induced streak artifacts by filtering prior to 3D reconstruction [18][19][20], and also to improve the quality of 3D game backgrounds [21].

The rest of this paper is organized as follows. Section 2 offers a detailed description of the proposed noise level estimation algorithm. Section 3 presents the results of a computer simulation of the application of the proposed method, and compares the results with those of certain representative noise estimation methods currently in use. Finally, the conclusions of the study are presented in Section 4.

2. Proposed Noise Level Estimation Algorithm

This section will first address the drawbacks of current methods for estimating the noise level in digital images. How the proposed method is used to overcome these drawbacks will then be discussed.

2.1 Categorization of Images

Image noise is generally regarded as zero mean additive white Gaussian noise with an unknown standard deviation. The standard deviation is considered as a good representation of the noise level, and most conventional methods were designed to estimate it. Usually, the standard deviation of the noise in an image is estimated by considering the homogeneous regions of the image. This is because any fluctuation in the image intensity in such regions is mainly caused by noise. The various conventional methods used to estimate the noise standard deviation generally involve the following steps:

Step 1: Detection of the homogeneous regions in the noisy image. (As implied earlier, this is because the fluctuations in the flat area pixels are supposed to be exclusively due to noise.)
Step 2: Computation of the local standard deviation in the detected homogeneous regions.
Step 3: Repetition of steps 1 and 2 until the entire image has been processed.
Step 4: Estimation of the noise level (or standard deviation of the noise) by averaging the computed local standard deviations.

This procedure has two major drawbacks. Firstly, if the homogeneous (or flat) regions in the noisy image are too small, their detection would be difficult, and an insufficient number of detected homogeneous regions may result in overestimation or underestimation of the noise level. Secondly, the process of detecting the homogeneous regions requires a high number of computations. Incidentally, the detection of the non-homogeneous regions in an image and using such to estimate the noise level is highly unreliable. These drawbacks originate from the fact that most conventional noise estimation methods presuppose the existence of many homogeneous regions in the image. Towards solving this problem, we categorized noisy images into the three following types:

1. Homogeneous images: contain many homogeneous (flat) regions.
2. Complex textured images: the original noise-free images have very complex structures, the local fluctuations of which make it difficult to detect the homogeneous regions.
3. MC-rendered images: contain spatially varying noise due to the low sampling rate during the generation of the image.

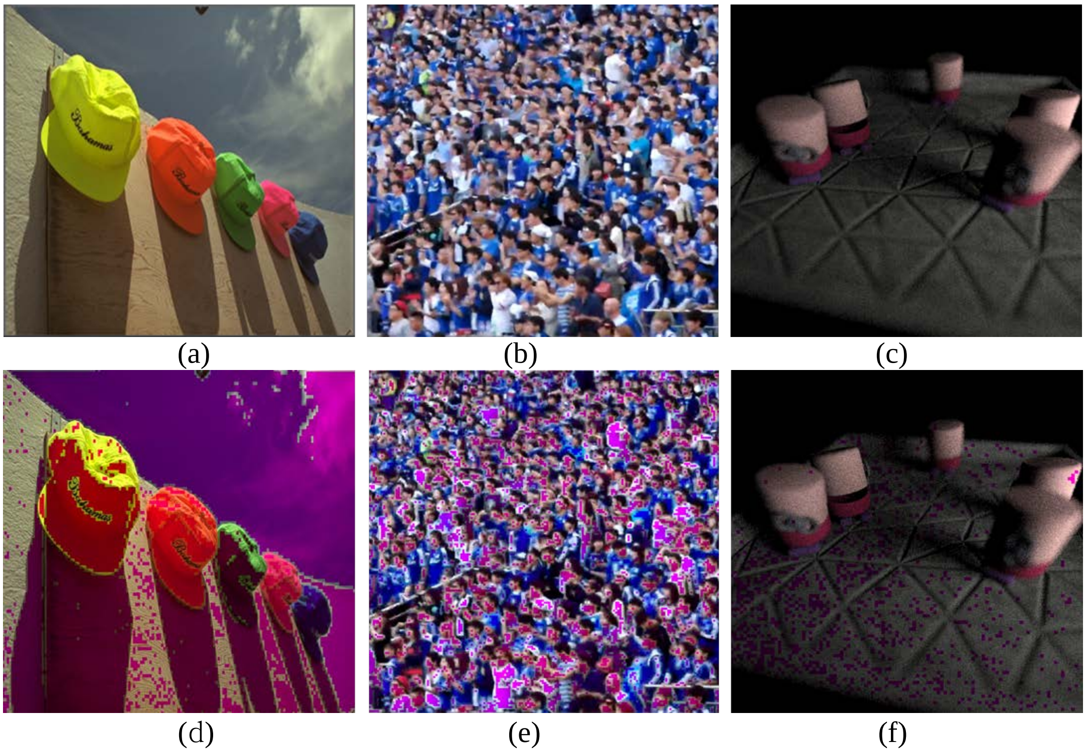


Fig. 1. Three types of images: (a) homogeneous image, (b) complex textured image, and (c) MC-rendered image. Also shown are the homogeneous regions of the (d) homogeneous image, (e) complex textured image, and (f) MC-rendered image.

Fig. 1 shows examples of the three different types of images based on the homogeneity and spatial variation of the noise. There are homogeneous regions in **Figs. 1**(a), (b), and (c), but those in **Figs. 1**(b) and (c) are not easy to detect by conventional methods. **Fig. 1**(c) particularly contains spatially variant noise. **Figs. 1**(d), (e), and (f) show the homogenous regions detected by a block-based algorithm; the violet blocks in each image indicate the detected regions. As can be seen from **Figs. 1**(e) and (f), the complex textured and MC-rendered images do not contain sufficient homogeneous regions for estimation of their noise level. As noted earlier, traditional high-performance noise level estimation algorithms generally used for homogeneous-type noisy images are not suitable for accurate determination of the noise level in these two special types of images.

2.2 Proposed noise level estimation algorithm

This subsection describes how the proposed noise estimation method is used to overcome the drawbacks of conventional noise estimation methods. The proposed method utilizes histogram compression of the image intensity to exploit the correlation among the R, G, and B components of the noisy colored image regardless of the number of its homogeneous regions. The best means of accurately determining the noise level in an image is to make the image homogeneous without affecting its noise deviation. In other words, if the variance in the original noise-free image can be suppressed without changing the noise deviation in the noisy image, it would be possible to more accurately estimate the noise level from the suppressed noisy image.

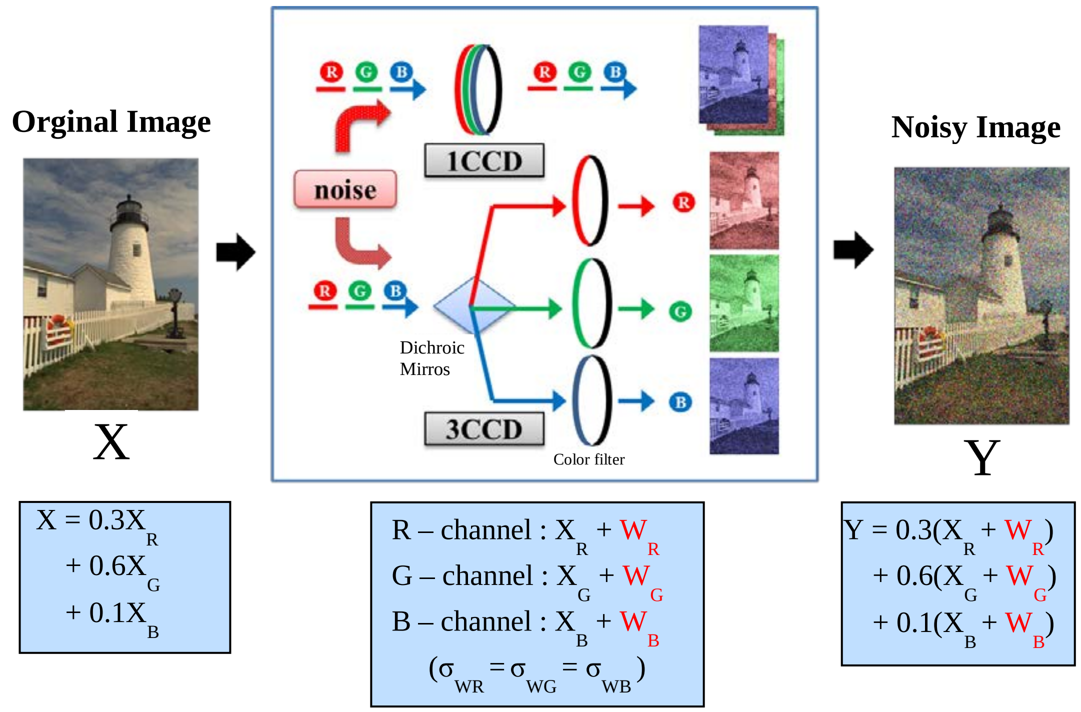


Fig. 2. Acquisition of colored digital image

Fig. 2 shows the structure of the current digital image acquisition system. Most digital images today are colored and each pixel of the image has red (R), green (G), and blue (B) components, and an RGB color model is mainly used for their processing for denoising [22]. Each RGB component of a colored image may be corrupted by noise due to faulty image sensors, environmental interference, and faulty communication. The noise tends to have a Gaussian distribution in a digital image acquisition system and can thus be modeled by a white Gaussian noise with the same standard deviation σ_W [23]. To reduce the correlation among the RGB components, a conversion from the RGB color space to a YUV color space is implemented using the following:

$$Y_i\ (Luminance) = 0.3R_i + 0.6G_i + 0.1B_i$$
$$U_i\ (Chrominance1) = B_i - Y_i$$
$$V_i\ (Chrominance2) = R_i - Y_i$$

However, if the noise components are uncorrelated in the RGB space, they will be correlated in the YUV space. The noise components and RGB components are independent of each other and are thus uncorrelated in the RGB space. However, a linear operation such as histogram modification in the YUV space may affect the uncorrected RGB components and the correlation of the noise components. Histogram compression, which is one of several histogram modification methods, is useful for suppressing the deviation of the RGB components without affecting the deviation of the noise components.

In the fields of image processing and computer graphics, noise is generally modeled with a Gaussian distribution, and the noisy image Y is typically expressed as

$$Y = X + W \tag{1}$$

where X represents the original noise-free image and W is the additive white Gaussian noise, which has a standard deviation σ that is constant for all the pixels of the image. As can be seen from **Fig. 3**, the main difference between a complex textured image and an MC-rendered image is that the noise in the latter is spatially variant.

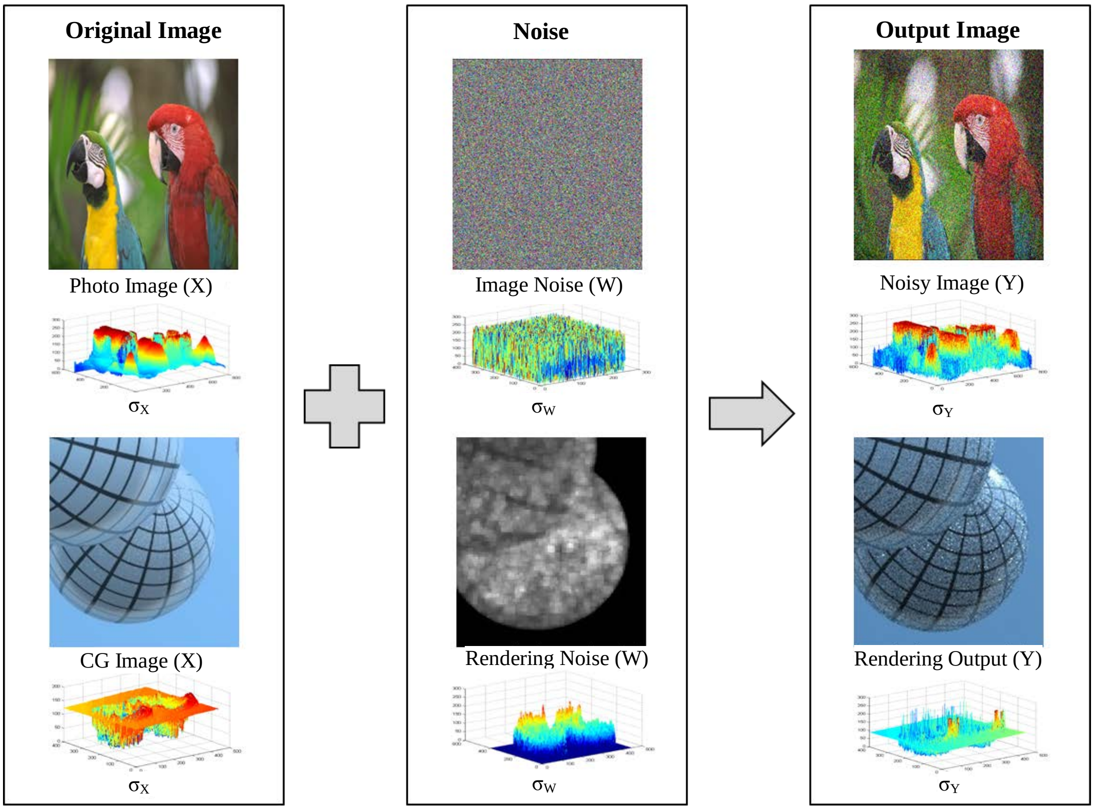


Fig. 3. Noises in complex textured and MC-rendered images

A noisy image Y has red (R), green (G), and blue (B) components in the RGB color space, and there is some correlation among these RGB components in other color spaces such as the YUV space:

$$\begin{aligned} Y &= \alpha Y_R + \beta Y_G + \gamma Y_B \\ &= \alpha (X_R + W_R) + \beta(X_G + W_G) + \gamma(X_B + W_B) \end{aligned} \tag{2}$$

where the variable subscripts indicate the RGB components, and the coefficients α, β, and γ are the correlation parameters. Because X and W are independent of each other, the variations of Y and X are related as follows:

$$\sigma_Y^{\ 2} = \sigma_X^{\ 2} + \sigma_W^{\ 2} \tag{3}$$

Modification of the histogram of the noisy image Y using (1) gives

$$Z = aY - b \tag{4}$$

$$= Y - ((1 - a)Y + b) = Y - T$$
$$= \alpha Y_R + \beta Y_G + \gamma Y_B - (\alpha + \beta + \gamma)T \quad (5)$$

where Z is the histogram-modified noisy image, and $T = (1 - a)Y + b$. If $\alpha + \beta + \gamma = 1$, (5) can be rewritten as

$$Z = \alpha(Y_R - T) + \beta(Y_G - T) + \gamma(Y_B - T), \text{ if } \alpha + \beta + \gamma = 1 \quad (6)$$

From (5) and (6), the R component of Z, Z_R, can be expressed as follows:

$$\begin{aligned} Z_R &= Y_R - T = X_R + W_R - (1 - a)Y - b \\ &= X_R + W_R - (1 - a)(X + W) - b \\ &= [X_R - (1 - a)X - b] + [W_R - (1 - a)(\alpha W_R + \beta W_G + \gamma W_B)] \end{aligned}$$

If a and b in the histogram modification equation (4) have values of 0.01 and 128, respectively, the term $(1 - a)$ can be approximated to 1. Compression of the histogram thus gives

$$Z_R \approx (X_R - X - 128) + ((1 - \alpha)W_R - \beta W_G - \gamma W_B) = P + Q \quad (7)$$

where $P = X_R - X - 128$, and $Q = (1 - \alpha)W_R - \beta W_G - \gamma W_B$. Hence, Z_R can be divided into RGB component term P and noise term Q. From (4), using $a = 0.01$, the variance relationship between Z and Y is

$$\sigma_Z^2 = 0.01\sigma_Y^2 \rightarrow \sigma_Z^2 << \sigma_Y^2 \quad (8)$$

Equation (8) indicates that the variation of Y is much higher than that of Z, and the deviations of the R components of Y and Z vary similarly. From (2), (3), and (7), we have

$$\begin{aligned} \sigma_{ZR}^2 << \sigma_{YR}^2 &\leftrightarrow \sigma_P^2 + \sigma_Q^2 << \sigma_{XR}^2 + \sigma_W^2 \\ &\leftrightarrow \sigma_P^2 + (1 - \alpha)^2\sigma_{WR}^2 + \beta^2\sigma_{WG}^2 + \gamma^2\sigma_{WB}^2 << \sigma_{XR}^2 + \sigma_W^2 \end{aligned} \quad (9)$$

Because the standard deviations of a white Gaussian noise are the same ($\sigma_W = \sigma_{WR} = \sigma_{WG} = \sigma_{WB}$), (9) becomes

$$\sigma_P^2 + ((1 - \alpha)^2 + \beta^2 + \gamma^2)\sigma_W^2 << \sigma_{XR}^2 + \sigma_W^2$$

If $(1 - \alpha)^2 + \beta^2 + \gamma^2 = 1$, the noise term σ_W^2 can be eliminated by approximation. Hence,

$$\sigma_P^2 << \sigma_{XR}^2 \text{ and } \sigma_Q \approx \sigma_W \quad (10)$$

The variance relationship (8) thus becomes

$$\sigma_{ZR}^2 = \sigma_P^2 + \sigma_Q^2 \approx \sigma_P^2 + \sigma_W^2 \quad (11)$$

Equations (10) and (11) indicate that the standard deviation of the R component is suppressed by histogram compression in the converted color space, whereas the standard deviation is not suppressed. To summarize, after the histogram compression of a noisy image, if the following two conditions of the correlation parameters of the RGB components are satisfied, the deviation of the original noise-free image can be suppressed without affecting the deviation of the contaminated white Gaussian noise:

$$\alpha + \beta + \gamma = 1\ ,\ (0 < \alpha, \beta, \gamma < 1)$$
$$(1 - \alpha)^2 + \beta^2 + \gamma^2 = 1$$

The application of the proposed noise level estimation method involves the following steps:

Step 1: Histogram compression of the intensity of the noisy image.
Step 2: Detection of the homogeneous regions in the image after performance of step 1. The homogeneous blocks can be detected by local standard deviations that are less than a given threshold value.
Step 3: Estimation of the noise level based on the detected homogeneous regions. The estimation is done by averaging the local standard deviations of the detected homogeneous blocks, or identifying the most frequent local standard deviation by histogram approximation.

3. Experimental Results and Analysis

3.1 Graphical Analysis

The proposed method can be better understood by more clearly visualizing the effect of histogram compression using a graphical analysis based on the correlation among σ_X, σ_Y, and σ_W in (3). The correlation is illustrated in **Fig. 4**, where the value of σ_Y for each pixel of the noisy image is given by the distance from the origin, as determined by the Pythagoras relationship between σ_X and σ_Y. Each blue point defines the relative local standard deviations σ_X, σ_Y, and σ_W of the 3×3 blocks of the R components. Even if σ_X and σ_W are unknown in a given noisy image, the local standard deviation σ_X can be determined before the noise level σ_W is added to the original noise-free image. Closeness of the blue points to the vertical axis indicates homogeneity of the local blocks of the points, including in the homogeneous regions in the noisy image. The homogeneous blocks in **Fig. 4** are many because the noisy image is homogeneous, for which reason there are also many blue points near the vertical axis.

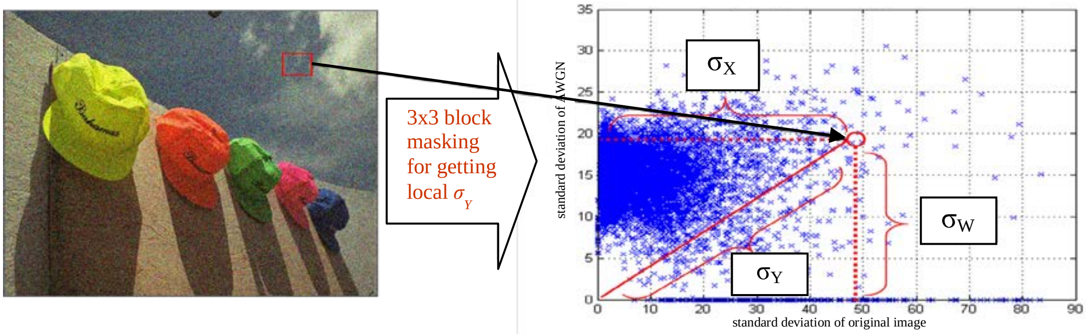


Fig. 4. Correlation among σ_X, σ_X, and σ_W. Each blue pixel in the right panel defines the relative local standard deviation of a 3×3 block.

However, the results of this graphical analysis may differ for the three types of images mentioned in the previous section. In the cases of complex textured and MC-rendered images, most of the blue points may be farther from the vertical axis. **Fig. 5** shows the results of the graphical analysis for the three different types of images, as well as the effects of the histogram compression on the R components. The red boxes indicate where the local blocks of the blue points are homogeneous. In the cases of the complex textured and MC-rendered images, there are very few homogeneous blocks, and it is not easy to detect the homogeneous regions by conventional noise estimation methods. However, after histogram compression, σ_X can be suppressed without affecting the distribution of σ_W in all three types of images. Moreover, the compression increases the number of points in the red box for the complex textured and MC-rendered images. In other words, histogram compression increases the homogeneous regions in a noisy image.

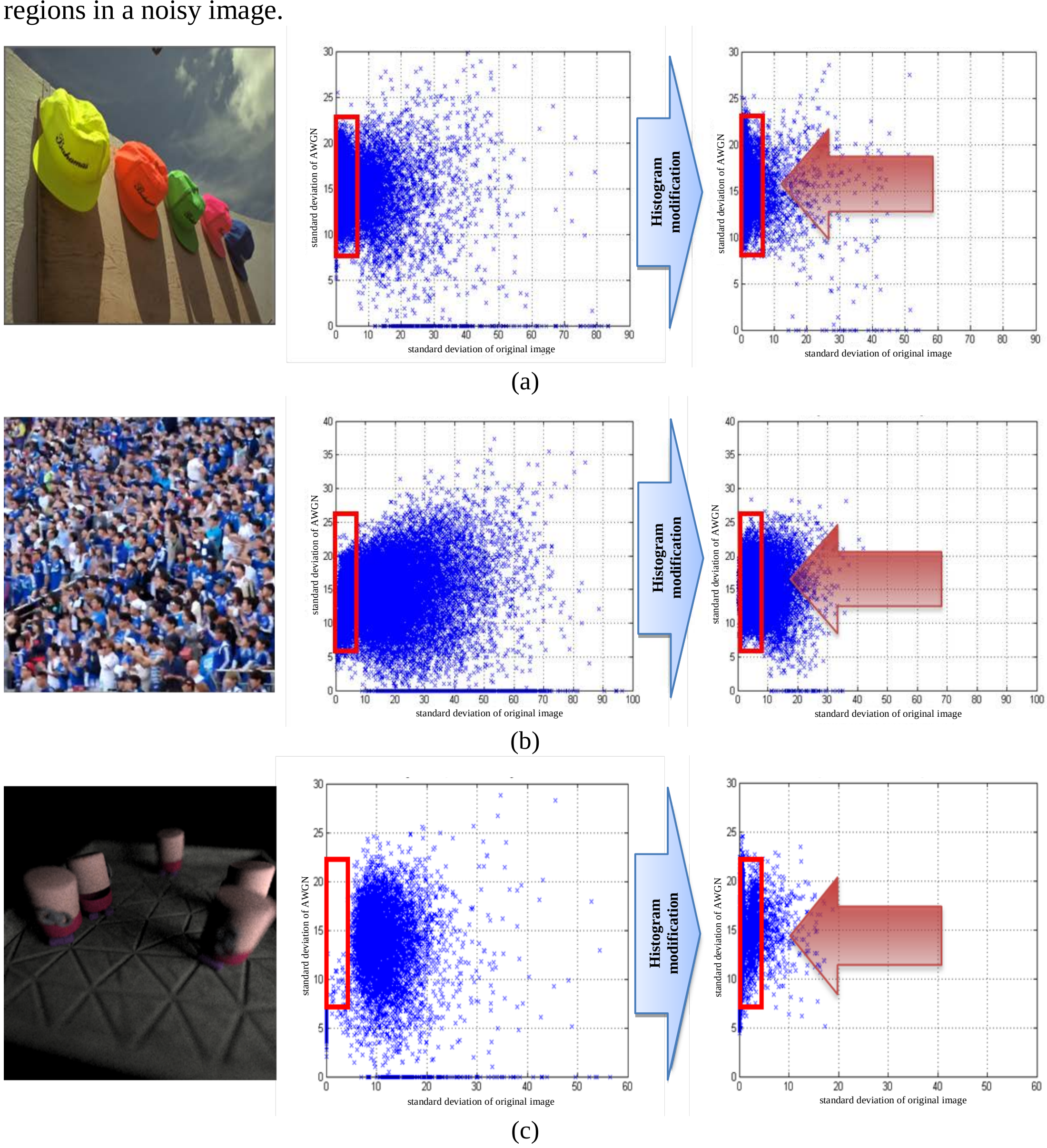


Fig. 5. Graphical analysis of the results of histogram compression: (a) homogeneous image, (b) complex textured image, (c) MC-rendered image

Another means of illustrating the effect of histogram compression is to compare the local standard deviations before and after the compression of the original noise-free image *X*, the noisy image Y, and the image W with added white Gaussian noise, respectively. The Comparisons are done in **Fig. 6**. In **Figs. 6**(a) and (b), it is apparent that the local standard deviation of the original noise-free image σ_X can be suppressed by histogram compression. However, the compression does not affect the local standard deviations of the white Gaussian noise. In the case of the MC-rendered image in **Fig. 6**(c), histogram modification appears to strengthen the localization of the scene details. This enables easy detection of spatially variant noise using local noise level estimation algorithms.

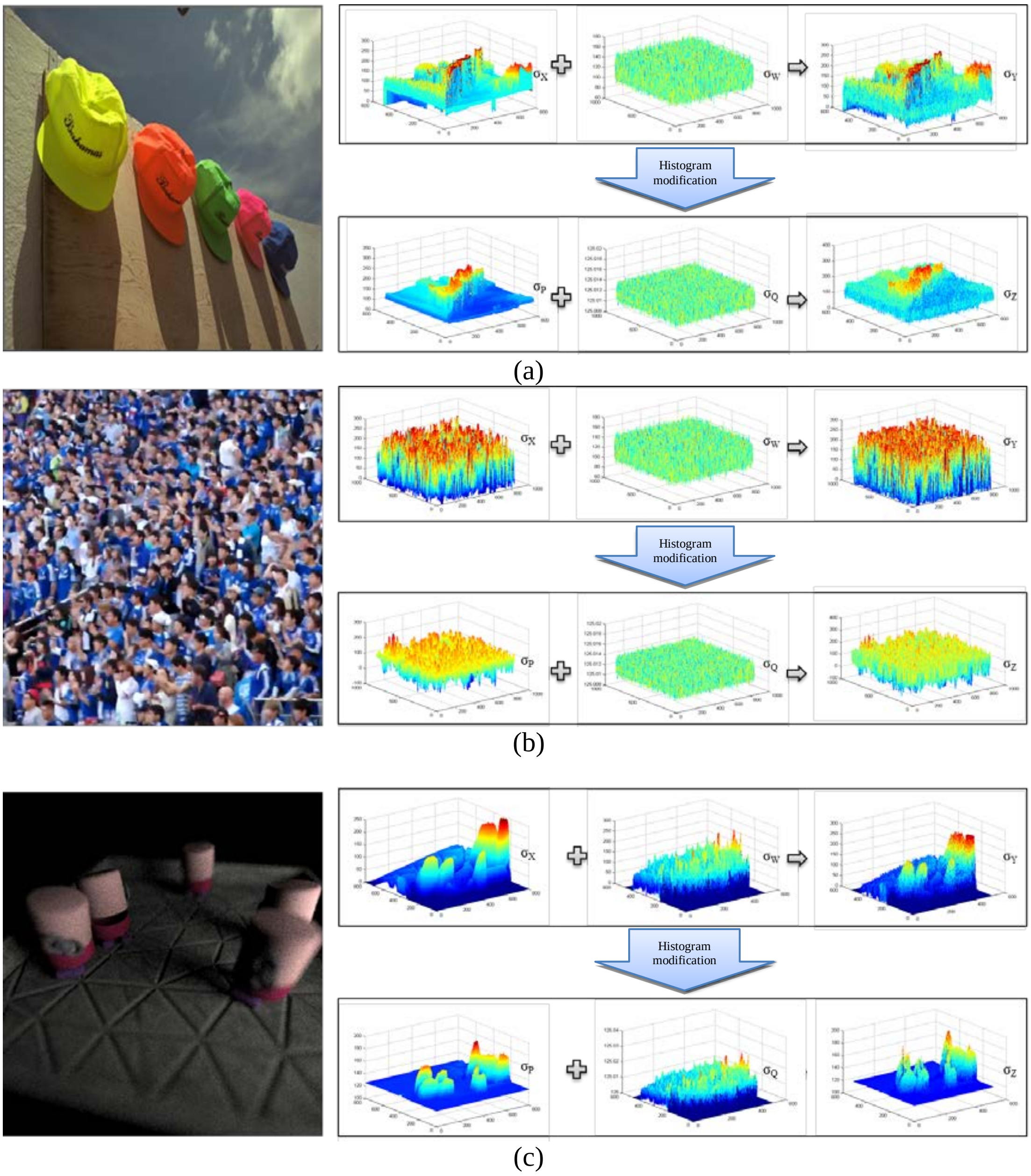
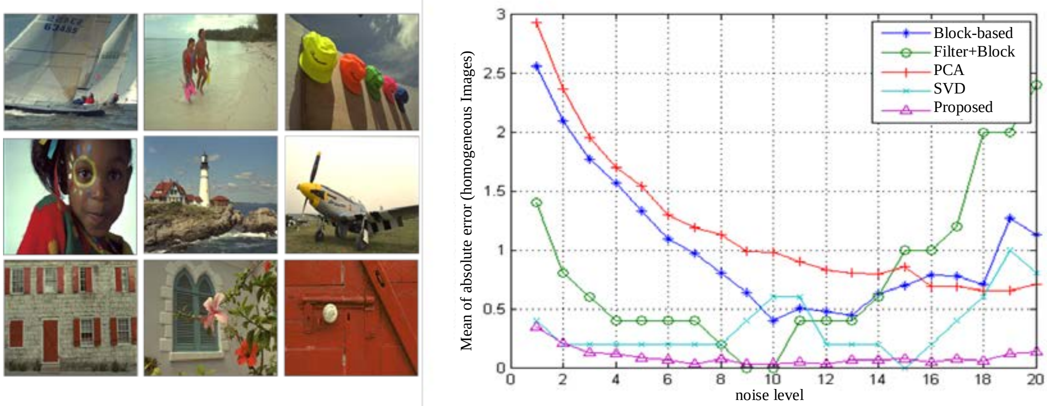


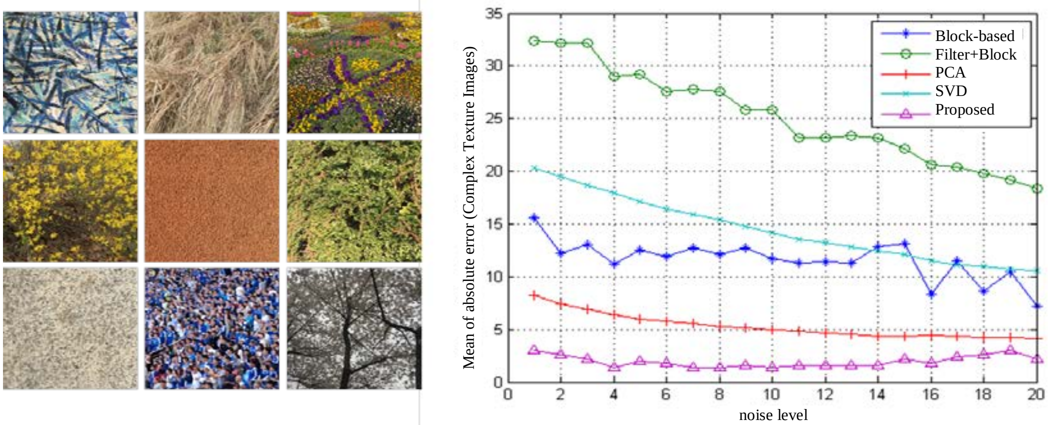
Fig. 6. 3D graphical analysis of (a) homogeneous image, (b) complex textured image, and (c) MC-rendered image

3.2 Experimental Results

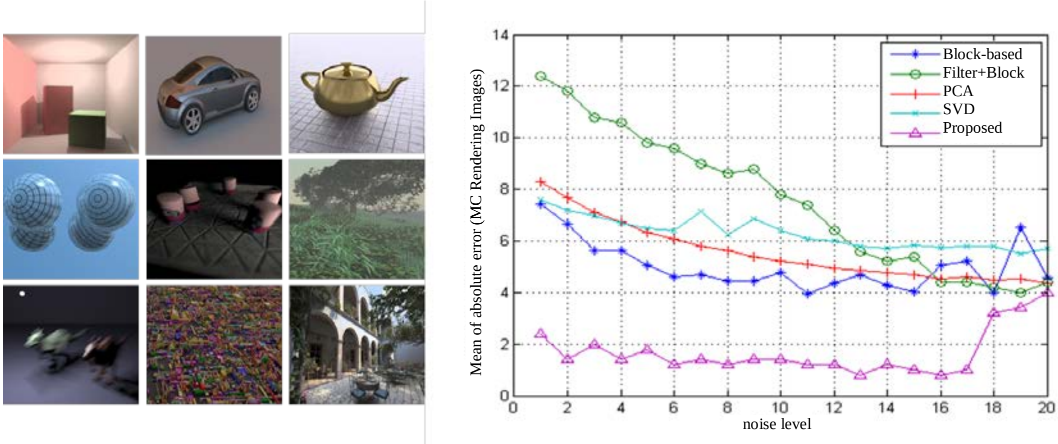
The results of the proposed algorithm were compared with those of four other algorithms representative of block-based [6], hybrid (filter- and block-based) [8], SVD [11], and PCA algorithms [14], respectively. Thirty test images were used, comprising homogenous images, complex textured images, and MC-rendered images. Examples of the images are shown in **Fig. 7**. To evaluate the performance of each algorithm, the estimation error $E_k = |\sigma_n - \sigma_e|$ was calculated for different noise levels between 0 and 20 (increments of 1). The estimation error E_k is the difference between the true and estimated noise levels, and the average value for each group of images was calculated. **Fig. 7** compares the performance results for the three groups of test images. As can be observed, the conventional noise level estimation algorithms performed well because there were many homogeneous regions. However, underestimation or overestimation began to appear at low and high noise levels. Conversely, the proposed algorithm performed well and better at both low and high noise levels. In the cases of the complex textured and MC-rendered images, as shown in **Figs. 7**(b) and (c), the conventional algorithms overestimated at low noise levels and underestimated at high noise levels. However, the proposed algorithm maintained its good performance. The estimation error of the proposed algorithm was confirmed to be lower than those of the other noise level estimation algorithms for all noise levels and all three image types. The proposed noise level estimation method is thus significantly better for complex textured and MC-rendered images.



(a)



(b)



(c)

Fig. 7. Comparison of the results of various noise level estimation algorithms for (a) homogeneous images, (b) complex textured images, and (c) MC-rendered images

The peak signal-to-noise ratios (PSNRs) of the restored images were measured using the proposed method, conventional noise estimation methods, and the general bilateral filtering method. **Table 1** summarizes the results. As can be seen, the proposed method produced better PSNR gains compared to the other methods for all three types of the test images.

Table 1. PSNR of the test images

Image type	PSNR (dB)				
	Block-based	Filter + Block	PCA	SVD	Proposed
Homogeneous	34.3265	34.5678	34.3648	34.5876	34.6348
Complex-Structure	27.7783	27.4972	27.9456	28.0290	28.2095
MC rendered	28.9424	29.8578	29.2289	29.5327	30.0160

4. Conclusion

This paper proposed and described a novel noise level estimation algorithm. The proposed algorithm uses histogram modification to suppress the variance in the original noise-free image without affecting the noise level. The advantage of the method is that it enables more accurate determination of the noise level in complex textured images and images containing spatially variant noise, to which current noise level estimation algorithms are generally difficult to apply. In addition, the application of the proposed method requires fewer computational resources because it is not necessary to detect the homogeneous regions or segment the image to strengthen localization. Experimental results confirmed the higher accuracy of the proposed method for noise level estimation.

References

[1] F. Russo, "A method for estimation and filtering of Gaussian noise in images," *IEEE Transactions on Instrumentation and Measurement*, vol. 52, no. 4, pp. 1148-1154, 2003. Article (CrossRef Link).

[2] T.A. Nguyen, W. S. Song, and M. C. Hong, "Spatially adaptive denoising algorithm for a single image corrupted by Gaussian noise," *IEEE Transactions on Consumer Electronics*, vol. 56, no. 3, pp. 1610-1615, 2010. Article (CrossRef Link).

[3] M. Ghazal and A. Amer, "Homogeneity localization using particle filters with application to noise estimation," *IEEE Transactions on Image Processing*, vol. 7, no. 7, pp. 1788-1796, 2011. Article (CrossRef Link).
[4] K. Konstantinides, B. Natarajan, and G. S. Yovanof, "Noise estimation and filtering using block-based singular value decomposition," *IEEE transactions on image processing*, vol. 6, no. 3, pp. 479-483, 1996. Article (CrossRef Link).
[5] A. Bosco, A. Bruna, G. Messina, and G. Spampinato, "Fast method for noise level estimation and integrated noise reduction," *IEEE Transactions on Consumer Electronics*, vol.51, no. 3, pp. 1028 - 1033, 2005. Article (CrossRef Link).
[6] M. Ghazal, A. Amer, and A. Ghrayeb, "A real-time technique for spatio-temporal video noise estimation," *IEEE transactions on Circuits and Systems for Video Technology*, vol. 17, no. 12, pp. 1690-1699, 2007. Article (CrossRef Link).
[7] A. Amer and E. Dubois, "Fast and reliable structure-oriented video noise estimation," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 15, no. 1, pp. 113-118, 2005. Article (CrossRef Link).
[8] D.H. Shin, R.H. Park, S.J. Yang, and J.H. Jung, "Block-based noise estimation using adaptive Gaussian filtering," *IEEE Transactions on Consumer Electronics*, vol. 51, no. 1, pp. 218-226, 2005. Article (CrossRef Link).
[9] S.M. Yang and S.C. Tai, "Fast and reliable image-noise estimation using a hybrid approach," *Journal of Electronic Imaging*, vol. 19, no. 3, pp. 033007-033007, 2010. Article (CrossRef Link).
[10] L. Zhang, W. Dong, D. Zhang, and G. Shi, "Two-stage image denoising by principal component analysis with local pixel grouping," *Pattern Recognition*, vol. 43, no. 4, pp. 1531-1549, 2010. Article (CrossRef Link).
[11] S. Pyatykh, J. Hesser, and L. Zheng, "Image Noise Level Estimation by Principal Component Analysis," *IEEE Transactions on Image Processing*, vol.22, no. 2, pp. 687 - 699, 2013. Article (CrossRef Link).
[12] K. W. Jorgensen and L. K. Hansen, "Model selection for Gaussian kernel PCA denoising," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 23, no. 1, pp.163-168, 2012. Article (CrossRef Link).
[13] X. Liu, M. Tanaka, and M. Okutomi, "Single-image noise level estimation for blind denoising," *IEEE Transactions on Image Processing*, vol. 22, no. 12, pp. 5226-5237, 2013. Article (CrossRef Link).
[14] W. Liu and W. Lin, "Additive white Gaussian noise level estimation in SVD domain for images," *IEEE Transactions on Image Processing*, vol. 22, no. 3, pp. 872-883, 2013. Article (CrossRef Link).
[15] V. Zlokolica, A. Pizurica, and W. Philips, "Noise estimation for video processing based on spatio-temporal gradients," *IEEE Signal Processing Letters*, vol. 13, no. 6, pp. 337-340, 2006. Article (CrossRef Link).
[16] N.K. Kalantari and P. Sen, "Removing the Noise in Monte Carlo Rendering with General Image Denoising Algorithms," *Computer Graphics Forum*, vol.32, no. 2, pp. 93-102, 2013. Article (CrossRef Link).
[17] P. Sen and S. Darabi, "On filtering the noise from the random parameters in Monte Carlo rendering," *ACM Transactions on Graphics*, vol.31, no. 3, 2012. Article (CrossRef Link).
[18] Y. Luo, T. Guan, B. Wei, H. Pan, and J. Yu, "Fast terrain mapping from low altitude digital imagery," *Neurocomputing*, vol. 156, pp. 105-116, 2015. Article (CrossRef Link).
[19] T. Guan, Y. He, J. Gao, J. Yang, and J. Yu, "On-device mobile visual location recognition by integrating vision and inertial sensors", *IEEE Transactions on Multimedia*, vol.15, no.7, pp. 1688-1699, 2013. Article (CrossRef Link).
[20] T. Guan, Y.F. He, L.Y. Duan, and J.Q. Yu, "Efficient BOF generation and compression for on-device mobile visual location recognition," *IEEE Multimedia*, vol. 21, no.2, pp. 32-41, 2014. Article (CrossRef Link).
[21] A. Iones, A. Krupkin, M. Sbert, and S. Zhukov, "Fast, realistic lighting for video games," *IEEE computer graphics and applications*, vol. 23, No. 3, pp. 54-64, 2003. Article (CrossRef Link).

[22] K. Rank, M. Lendl, and R. Unbehauen, “Estimation of image noise variance,” *IEE Proceedings on Vision, Image and Signal Processing*, vol. 146, no. 2, pp. 80-84, 1999. Article (CrossRef Link).
[23] K.N. Plataniotis, D. Androutsos, and A.N. Venetsanopoulos, “Adaptive fuzzy systems for multichannel signal processing,” in *Proc. of the IEEE*, , vol.87, no. 9, pp. 1601-1622, 1999. Article (CrossRef Link).

I-Gil Kim obtained a B.S. degree in Electrical Engineering from the Hongik University in 1999, and an M.S. degree in Electrical Engineering from the University of Southern California in 2002. In 2008, he was awarded an E.E degree (post-master’s degree in Electrical Engineering) by the University of Florida. Between 2003 and 2005, he was a senior research engineer in the R&D Group, Digital Media Division, Samsung Electronics Co., Ltd. He is currently a principal research engineer at the KT (Korea Telecom) Institute of Convergence Technology. His research interests include computer vision, virtual reality, and game programming.