Constant-Time Gaussian Filtering for Acceleration of Structure Similarity

Tomohiro Sasaki*, Norishige Fukushima†, Yoshihiro Maeda‡,
Kenjiro Sugimoto*, and Sei-ichiro Kamata*

*Waseda University, †Nagoya Institute of Technology, ‡Tokyo University of Science

***Abstract*—In this paper, we propose an acceleration method of structural similarity (SSIM) and its multi-scaled version, called MS-SSIM. The calculation process of SSIM and MS-SSIM includes multiple Gaussian filters, and the cost of the filter is dominant for the entire process; thus, to accelerate SSIM/MS-SSIM, we replace Gaussian filtering using convolution with sliding DCT. Gaussian filter based on sliding DCT is faster than the usual convolution method. Besides, its computational complexity does not depend on the filter window length. Also, naive implementations of SSIM and MS-SSIM scan image many times for the pixel-wise operation; however, these operations can be incorporated into Gaussian filtering. Thus, we optimize the processing pipeline to achieve high cache-efficiency. As a result, the proposed SSIM computation was accelerated by 6.36 times and MS-SSIM by 8.11 times faster than the conventional approach.**

***Index Terms*—SSIM, fast image quality assessment, constant-time Gaussian filtering, sliding DCT, acceleration**

I. Introduction

Image quality assessment (IQA) is essential for image processing. The quality metrics are used for the evaluation and optimization of various image processing methods. IQA has essential roles in the following research topics, e.g., image coding, denoising, deblurring, super-resolution, image synthesis, and high-dynamic-range imaging. Due to the availability of a reference image, the objective IQA has three categories; full reference (FR), no-reference (NR), and reduced-reference (RR) methods. In this paper, we focus on FR methods.

The well-known FR metric is the mean-squared error (MSE) or its log-scaled value of the peak-signal-to-noise ratio (PSNR). This metric compares degraded signals with ideal signals via pixel by pixel. This method is fast and straightforward; however, it does not correlate well with perceived human-visual-quality.

IQA research community continually improves the quality metrics. The representatives are noise quality measure (NQM) [1], structure similarity (SSIM) [2], information fidelity criterion (IFC) [3], and its improved version named visual information fidelity (VIF) [4], visual signal-to-noise ratio (VSNR) [5], most apparent distortion (MAD) [6], Riesz-transform based Feature similarity metric (RFSIM) [7], gradient similarity (GSIM) [8], feature similarity (FSIM) [9], spectral residual based similarity (SR-SIM) [10], internal generative mechanism (IGM) [11], gradient magnitude similarity deviation (GMSD) [12], visual saliency-induced (VSI) [13], and perceptual similarity (PSIM) [14].

SSIM is a milestone in the recent history of the IQA researches; thus, there are many reports for SSIM [15]–[18]. Also, there are various variants of SSIM. Universal image quality (UIQ) [19] is extended to SSIM at first. The multi-scaled version of SSIM, named MS-SSIM [20], is then extended. The gradient-based SSIM is also reported in [21]. For shift-invariant issues, the complex-wavelet-based method (CW-SSIM) [22] is proposed. For improving a pooling function, combined percentile-fixation-SSIM or FP-SSIM is proposed [23]. The function is also extended as information content weighting SSIM (IW-SSIM) [24]. SSIMplus [25] is the state-of-the-art for video quality metric. Moreover, there are various image processing applications based on human visual system (HVS) with SSIM [26]–[29]. The survey of IQA and its applications are summarized in the following papers [30]–[33].

For image processing based on SSIM, real-time processing is required even for IQA. For example, video coding is performed while optimizing video quality. This optimization often employ SSIM. In this situation, real-time processing of SSIM is required. In addition, acceleration of SSIM can reduce the learning cost of deep learning, such as Generative Adversarial Networks (GAN) [34]. Rouse and Hemami [35] analyze that the luminance component in SSIM has less importance than the other; thus, the computation of the luminance factor can be omitted. The following research is Fast SSIM [36], [37]. In the Fast SSIM, (1) the luminance component of each block was computed by using an integral image, (2) the contrast and structure components of each block were computed based on 2×2 Roberts gradient operators, (3) the Gaussian-weighting window used in the contrast and structure components was replaced with an integer approximation. Both approaches require algorithmic changing; thus, the response of the SSIM index map becomes mostly changed. For this reason, Fast SSIM is inappropriate to apply universally, that is, efficiency of applying Fast SSIM should be verified for each application. Also, the main bottleneck in computing SSIM is low-pass filtering of Gaussian filtering (GF); however, the cost is not well resolved.

This work was supported by JSPS KAKENHI JP17H01764, 18K18076, 18K19813, 19K24368.

International Conference on Image Processing and Robot (ICIPRoB), 2020.

To suppress the distortion in the index map, and to improve the performance of Gaussian filtering, we accelerate the SSIM index by using sliding DCT for Gaussian blurring [38]–[43]. The Gaussian filtering with the sliding approach has constant order in filtering kernel radius; thus, the filtering performance can be improved in the higher scale case. Also, we optimize the image-processing-pipeline of SSIM computation to have highly cache efficiency.

The contribution of this paper is threefold:

1) Reduction of computational complexity to $O(1)$ by replacing naive convoluted GF with constant time GF based on sliding DCT.
2) Optimization of the computing pipeline of both SSIM and MS-SSIM to get high cache efficiency.
3) Verification of the accuracy between original SSIM and proposed one.

II. Related Works

A. Structural Similarity (SSIM) Index

SSIM is one of the IQA, which considered the structural similarity of images. This metric is computed by compering not the pixel-wise error like PSNR but the local-area error between two images.

Let $\boldsymbol{A}$ and $\boldsymbol{B}$ are reference image and distorted image, respectively. Image area is denoted by $\mathcal{S} \subset \mathbb{Z}^2$. Then the SSIM value of a local area whose center is a focusing pixel $\boldsymbol{p}$ is computed from following;

$$\mathrm{SSIM}(\boldsymbol{p},\boldsymbol{A},\boldsymbol{B})=[l(\boldsymbol{p},\boldsymbol{A},\boldsymbol{B})]^{\alpha}[c(\boldsymbol{p},\boldsymbol{A},\boldsymbol{B})]^{\beta}[s(\boldsymbol{p},\boldsymbol{A},\boldsymbol{B})]^{\gamma}, \quad (1)$$

where $l(\boldsymbol{p},\boldsymbol{A},\boldsymbol{B})$, $c(\boldsymbol{p},\boldsymbol{A},\boldsymbol{B})$, and $s(\boldsymbol{p},\boldsymbol{A},\boldsymbol{B})$ are

$$l(\boldsymbol{p},\boldsymbol{A},\boldsymbol{B}) = \frac{2\mu_{A\boldsymbol{p}}\mu_{B\boldsymbol{p}} + C_1}{{\mu_{A\boldsymbol{p}}}^2 + {\mu_{B\boldsymbol{p}}}^2 + C_1}, \quad (2)$$

$$c(\boldsymbol{p},\boldsymbol{A},\boldsymbol{B}) = \frac{2\sigma_{A\boldsymbol{p}}\sigma_{B\boldsymbol{p}} + C_2}{{\sigma_{A\boldsymbol{p}}}^2 + {\sigma_{B\boldsymbol{p}}}^2 + C_2}, \quad (3)$$

$$s(\boldsymbol{p},\boldsymbol{A},\boldsymbol{B}) = \frac{\sigma_{AB\boldsymbol{p}} + C_3}{\sigma_{A\boldsymbol{p}}\sigma_{B\boldsymbol{p}} + C_3}. \quad (4)$$

Here, $\mu_{A\boldsymbol{p}}$ and $\sigma_{A\boldsymbol{p}}$ represent average and variance of a local area on image $\boldsymbol{A}$, whose center is $\boldsymbol{p}$, respectively. Let be $C_1 = (K_1 L)^2$ and $C_2 = (K_2 L)^2$, where L is the dynamic range of the pixel values. These constant values roles as prevention for zero division. $K_1, K_2 << 1$ are small constant values, and $K_1 = 0.01, K_2 = 0.03$ are used in many cases. SSIM ≤ 1, and higher value is better. Furthermore, for simplicity, we set $\alpha = \beta = \gamma = 1$ and $C_3 = C_2/2$. Then, (1) is redefined as follows;

$$\mathrm{SSIM}\left(\boldsymbol{p},\boldsymbol{A},\boldsymbol{B}\right)=\frac{(2\mu_{A\boldsymbol{p}}\mu_{B\boldsymbol{p}}+C_1)(2\sigma_{AB\boldsymbol{p}}+C_2)}{(\mu_{A\boldsymbol{p}}^2+\mu_{B\boldsymbol{p}}^2+C_1)(\sigma_{A\boldsymbol{p}}^2+\sigma_{B\boldsymbol{p}}^2+C_2)}. \quad (5)$$

In this paper, we use this form to compute SSIM.

Normally, there are many cases for evaluation of entire images. In this case, we use mean SSIM (MSSIM) or average pooling with SSIM. The MSSIM averages the total values of an SSIM index map on each pixel.

$$\mathrm{MSSIM}(\boldsymbol{A},\boldsymbol{B}) = \frac{1}{|\mathcal{S}|}\sum_{\boldsymbol{p}\in\mathcal{S}} \mathrm{SSIM}(\boldsymbol{p},\boldsymbol{A},\boldsymbol{B}), \quad (6)$$

where, $|\mathcal{S}|$ denotes the total number of pixels in the image.

B. Multi-scaled Structural Similarity (MS-SSIM) Index

MS-SSIM is proposed to handle the variations of image resolution and viewing conditions more flexible. MS-SSIM with the number of the scale M is defined as follows;

$$\begin{aligned}&\text{MS-SSIM}(\boldsymbol{p},\boldsymbol{A},\boldsymbol{B}) = \\ &[l_M(\boldsymbol{p},\boldsymbol{A},\boldsymbol{B})]^{\alpha_M}\prod_{j=1}^{M}[c_j(\boldsymbol{p},\boldsymbol{A},\boldsymbol{B})]^{\beta_j}[s_j(\boldsymbol{p},\boldsymbol{A},\boldsymbol{B})]^{\gamma_j}. \quad (7)\end{aligned}$$

To simplify, in many cases, parameters are set to $\alpha = \beta = \gamma$. In addition, to enable comparison between different parameters, parameters are normalized as $\sum_{j=1}^{M}\gamma_j = 1$.

C. Sliding DCT for Gaussian Filtering

The computational complexity of direct convolution with Gaussian kernel for an image is $O(W)$ per pixel, where W is filtering window length. For acceleration, some $O(1)$ algorithms are proposed. These methods can smooth image in constant-time; thus, computational complexity is independent of filter window length. Among them, a method based on discrete cosine transform (DCT) is high-performance from the viewpoint of accuracy and speed. This DCT based GF has many applications. For example, bilateral filtering [44] is accelerated [45]–[47]. Also, guided image filtering [48] is accelerated [49] as well.

In this paper, we also use the DCT based GF in the calculation of SSIM. Two cascades of 1-D GF realize 2-D GF because of its separability. Therefore, we consider 1-D GF below. 1-D Gaussian kernel $h_n \in \mathbb{R}$ is defined as;

$$h_n := \eta^{-1} e^{-\frac{n^2}{2\sigma^2}}, \sum_{n=-N+1}^{N-1} e^{-\frac{n^2}{2\sigma^2}}, \quad (8)$$

where $\sigma \in \mathbb{R}_+$ is a spatial scale parameter, and $\eta \in \mathbb{R}$ is normalized factor. Then, DCT representation of kernel is

$$h_n = \sum_{k=0}^{N-1} \hat{h}^{(k)} C_n^{(k)} \quad (9)$$

$$C_n^{(k)} := \cos\left(\frac{2\pi}{T}(k+k_0)(n+n_0)\right), \quad (10)$$

where $\hat{h}^{(k)}$ is the k-th weight coefficient. Therefore, the convolution of kernel and input signal x is defined as;

$$(x*h)_t = \sum_{n=-N+1}^{N-1} x_{t+n}h_n = \sum_{k=0}^{N-1}\hat{h}^{(k)} \sum_{n=-N+1}^{N-1} x_{t+n}C_n^{(k)} \quad (11)$$

$$= \sum_{k=0}^{N-1}\hat{h}^{(k)}\hat{x}_t^{(k)}, \hat{x}_t^{(k)} = \sum_{n=-N+1}^{N-1} x_{t+n}C_n^{(k)}, \quad (12)$$

where $\hat{x}_t^{(k)}$ is a short-time transform coefficient of the input sequence x_t at time t. Then, $\hat{h}^{(k)}$ is commutable form the following recursion formula in constant-time.

$$\hat{x}_{t-1}^{(k)} + \hat{x}_{t+1}^{(k)} = 2C_{1-n_0}^{(k)}\hat{x}^{(k)} + x_{t-N}C_{-N+1}^{(k)} + x_{t+N}C_{+N-1} \tag{13}$$

$$- x_{t-N+1}C_{-N}^{(k)} - x_{t+N-1}C_{+N}^{(k)}. \tag{14}$$

This relationship among three terms is called second shift property, and the operation, which computes $\hat{x}_{t+1}^{(k)}$ from this formula in constant-time, is called sliding transform. Here, spectrum of Gaussian is also Gaussian, and it attenuates exponentially. Therefore, k can be truncated at a few terms. DCT is classified into DCT-1 to DCT-8 according to the differences of three parameters T, k_0, n_0. In this paper, we use DCT-5.

III. Proposed Method

A. Acceleration of SSIM

In this section, we propose a fast SSIM implementation. First, in the calculation process of SSIM, the bottleneck is GF, because others are pixel-wise operations. Hence, by replacing the naive convoluted GF with DCT based GF, the calculation amount is reduced. Its total complexity is reduced to $O(1)$. As a result, we achieve an acceleration of SSIM.

Second, we consider the computing pipeline of SSIM. To obtain μ and σ, which are necessary for computing SSIM, we cannot ignore the data dependency shown in Fig. 1. In the naive implementation, each arrow is shown in Fig. 1 is calculated as one process as a part of the image processing pipeline of SSIM. That is, for each GF and pixel-wise calculation is treated as independent processes, and every process re-scans images from top to bottom. This multiple scanning causes low memory-access-efficiency and increases the cache miss. Therefore, we try to realize efficient implementation by improving the calculation pipeline efficiency by considering the data dependency.

Let us consider separable filtering in the order of horizontal filtering and then vertical filtering. In step 1, we consider horizontal filtering. As shown in Fig. 2, we copy one line from image $\boldsymbol{A}$ and $\boldsymbol{B}$ to each line-buffer, respectively. At the same time, we calculate $\boldsymbol{A^2}$, $\boldsymbol{B^2}$, and $\boldsymbol{AB}$, and then write to each line-buffer. In step 2, apply 1-D GF to the five line-buffers, which is given in the previous process, and then write back to the $\boldsymbol{A}, \boldsymbol{B}$. Here, writing to $\boldsymbol{A}, \boldsymbol{B}$ is in-place processing, i.e., we share input and output memory space. In the other three processes, we write the line-buffers to three image areas, which allocated in advance. After steps 1 and 2, we move to the next rows, and then we repeat the process until the last row. For each filtering, we simultaneously process five filters at one scan-line. In other words, we use one for-loop, not five for-loops. When horizontal process is finished, we obtain $g_1(\boldsymbol{A})$, $g_1(\boldsymbol{B})$, $g_1(\boldsymbol{A}^2)$, $g_1(\boldsymbol{B}^2)$, and $g_1(\boldsymbol{AB})$, where $g_1(\cdot)$ denotes 1-D GF.

Next, remaining vertical process is then just applied to five images, i.e., $g_1(\boldsymbol{A})$, $g_1(\boldsymbol{B})$, $g_1(\boldsymbol{A}^2)$, $g_1(\boldsymbol{B}^2)$, and $g_1(\boldsymbol{AB})$.

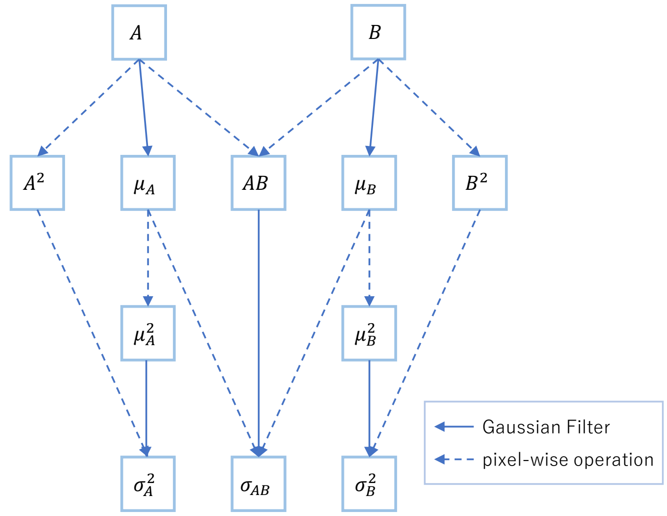

Fig. 1: Data dependency of SSIM. $\boldsymbol{A}, \boldsymbol{B}$: two input images. Solid arrows represent Gaussian Filters, and dashed arrows represent pixel-wise operations.

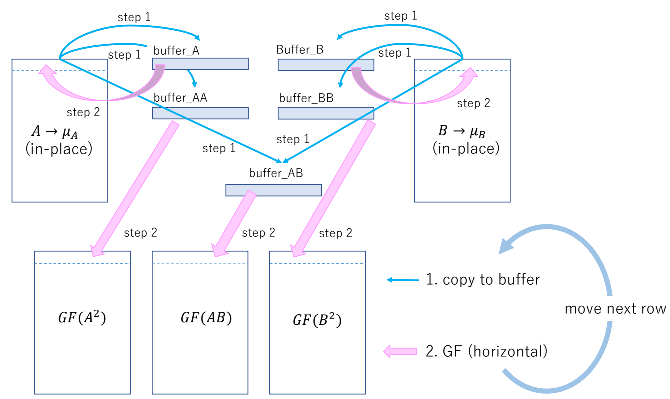

Fig. 2: Horizontal process of SSIM. First, copy image to line buffers from $\boldsymbol{A}, \boldsymbol{B}$. Second, filter line-buffers and then write to images. Next, move to the next row to repeat this process.

We also merge the five filters into one-loop. Note that, at this time, we do not write the output of 5 GF results to the five image area but directly write to an output image with calculating SSIM. Because after obtaining the five filtered results, resulting processes are pixel-wise operations, and the operations do not have a dependency. In this method, we can reduce the number of scans according to incorporate the pixel-wise calculation into GF.

B. Acceleration of MS-SSIM

Next, we optimize MS-SSIM. Convolution GF is replaced with DCT based GF as well. Also, the main idea is almost the same as the proposed SSIM, so we overview the process. A term $l(\cdot)$ in (7) are calculated first. Then, two terms $c(\cdot), s(\cdot)$ are calculated at the same time. In this two-step, pixel-wise calculations are incorporate into GF as well as proposed SSIM.

TABLE I: The number of for-loops for naive and proposed SSIM and MS-SSIM computation. M is the number of scales.

method	SSIM	MS-SSIM
naive	27	$10 + 24M$
proposed	2	$2M$

IV. Experimental Results

In our experiments, Intel Core i7-7500U CPU 2.70 GHz (2 cores /4 threads) with main memory 8 GB on Windows 10 64-bit were employed. We implemented all methods in C++ by using OpenCV [51], and also we vectorized the code by AVX. Note that the code is worked for a single thread, not for multi-threads. The parameter σ of GF using in SSIM is fixed to 1.5, and a kernel size of naive implementation is fixed to 11×11. The approximate number of terms of DCT based GF is fixed to 3.

For preliminary, we evaluated the accuracy of the approximated Gaussian filtering based on DCT by mean squared error (MSE). In the comparison of a naive convolution GF with DCT based GF, the difference between both resulting Gaussian filters is 5.15×10^{-7}. The result is a low enough error.

Next, we evaluated the accuracy and speed of the proposed SSIM and MS-SSIM. Here, we assume that the OpenCV implementation is naive, and this naive implementation is the ground truth result. Then, we evaluated the error between the ground truth and the proposed method by MSE. Notice that the purpose of this experiment is to evaluate how close the DCT based SSIM is to the original SSIM, and it is not a proposal of IQA, which is superior in accuracy to SSIM. The efficiency of SSIM has been already justified. Input is grayscale Lenna image, and Lenna image distorted by Gaussian blur ($\sigma = 0.99999$, kernel size $= 7 \times 7$) or JPEG compression (QP=30). In MS-SSIM, $M = 5$ and $\beta_1 = \gamma_1 = 0.0448$, $\beta_2 = \gamma_2 = 0.2856$, $\beta_3 = \gamma_3 = 0.3001$, $\beta_4 = \gamma_4 = 0.2363$, $\alpha_5 = \beta_5 = \gamma_5 = 0.1333$ were used.

The result are shown in Tables I,II. Table I indicates that the proposed approach dramatically reduces the number of image scan loops; thus, the cache efficiency is improved. Table II shows that the proposed SSIM accelerates 6.36 times faster than the naive implementation, and the proposed MS-SSIM accelerates 8.11 times faster than the naive implementation. Besides, MSEs of both SSIM and MS-SSIM index maps are low, and the accuracy of SSIM is almost the same as that of a naive implementation. It can be said that the naive implementation can be reproduced in the accuracy. The SSIM/MS-SSIM index maps are shown in Figs. 3, 4. It can be visually confirmed that there is no difference. Just in case, the same experiment was done with Kodak Photo CD [50] including 24 images, except for the calculation time. These images were also converted to grayscale in advance. The average values of the results obtained from each image are shown in Table III. First, by comparing MSSIM/MMS-SSIM, the accuracy is high as in the case of using Lenna image. Second, although average of error by MSE is higher than Lenna image, is's still high accuracy.

V. Conclusion

In this paper, we proposed an acceleration method for SSIM and MS-SSIM. The proposed method was based on sliding DCT, which has constant time property for Gaussian filtering. Also, we optimized the computing pipeline of SSIM to have high cache efficiency. Experimental results showed that the proposed SSIM and MS-SSIM accelerate 6.36 times and 8.11 times faster than the naive implementation, respectively.

As our future work, we will verify the relationship between subjective and objective assessment scores of the accelerated method via various image data set, such as LIVE [52], TID2008 [53], TID2013 [54], Categorical image quality (CSIQ) database [55], KADID-10k Image Database [56], and ESPL Synthetic Image Database [57].

References

[1] N. Damera-Venkata, T. D. Kite, W. S. Geisler, B. L. Evans, and A. C. Bovik, “Image quality assessment based on a degradation model,” *IEEE Transactions on Image Processing*, vol. 9, no. 4, pp. 636–650, 2000.

[2] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli, “Image quality assessment: from error visibility to structural similarity,” *IEEE Transactions on Image Processing*, vol. 13, no. 4, pp. 600–612, 2004.

[3] H. R. Sheikh, A. C. Bovik, and G. de Veciana, “An information fidelity criterion for image quality assessment using natural scene statistics,” *IEEE Transactions on Image Processing*, vol. 14, no. 12, pp. 2117–2128, 2005.

[4] H. R. Sheikh and A. C. Bovik, “Image information and visual quality,” *IEEE Transactions on Image Processing*, vol. 15, no. 2, pp. 430–444, 2006.

[5] D. M. Chandler and S. S. Hemami, “Vsnr: A wavelet-based visual signal-to-noise ratio for natural images,” *IEEE Transactions on Image Processing*, vol. 16, no. 9, pp. 2284–2298, 2007.

[6] E. C. Larson and D. M. Chandler, “Most apparent distortion: full-reference image quality assessment and the role of strategy.” *Journal of Electronic Imaging*, vol. 19, no. 1, p. 011006, 2010.

[7] L. Zhang, L. Zhang, and X. Mou, “Rfsim: A feature based image quality assessment metric using riesz transforms,” in *Proc. IEEE International Conference on Image Processing (ICIP)*, 2010, pp. 321–324.

[8] A. Liu, W. Lin, and M. Narwaria, “Image quality assessment based on gradient similarity,” *IEEE Transactions on Image Processing*, vol. 21, no. 4, pp. 1500–1512, 2012.

[9] L. Zhang, L. Zhang, X. Mou, and D. Zhang, “Fsim: A feature similarity index for image quality assessment,” *IEEE Transactions on Image Processing*, vol. 20, no. 8, pp. 2378–2386, 2011.

[10] L. Zhang and H. Li, “Sr-sim: A fast and high performance iqa index based on spectral residual,” in *Proc. IEEE International Conference on Image Processing (ICIP)*, 2012, pp. 1473–1476.

[11] J. Wu, W. Lin, G. Shi, and A. Liu, “Perceptual quality metric with internal generative mechanism,” *IEEE Transactions on Image Processing*, vol. 22, no. 1, pp. 43–54, 2013.

[12] W. Xue, L. Zhang, X. Mou, and A. C. Bovik, “Gradient magnitude similarity deviation: A highly efficient perceptual image quality index,” *IEEE Transactions on Image Processing*, vol. 23, no. 2, pp. 684–695, 2014.

[13] L. Zhang, Y. Shen, and H. Li, “Vsi: A visual saliency-induced index for perceptual image quality assessment,” *IEEE Transactions on Image Processing*, vol. 23, no. 10, pp. 4270–4281, 2014.

[14] K. Gu, L. Li, H. Lu, X. Min, and W. Lin, “A fast reliable image quality predictor by fusing micro- and macro-structures,” *IEEE Transactions on Industrial Electronics*, vol. 64, no. 5, pp. 3903–3912, 2017.

[15] D. Brunet, E. R. Vrscay, and Z. Wang, “On the mathematical properties of the structural similarity index,” *IEEE Transactions on Image Processing*, vol. 21, no. 4, pp. 1488–1499, 2012.

[16] A. Hore and D. Ziou, “Image quality metrics: PSNR vs. SSIM,” in *Proc. International Conference on Pattern Recognition (ICPR)*. IEEE, 2010.

[17] Z. Wang and A. C. Bovik, “Mean squared error: Love it or leave it? a new look at signal fidelity measures,” *IEEE Signal Processing Magazine*, vol. 26, no. 1, pp. 98–117, 2009.

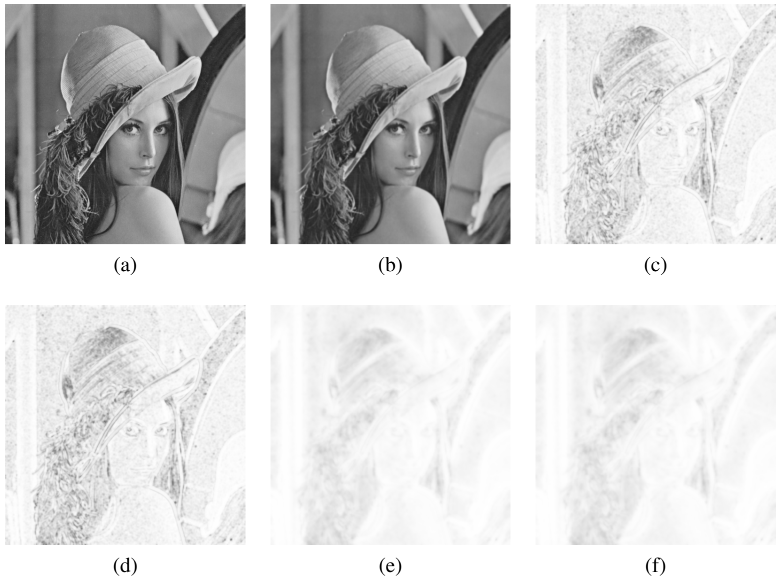

Fig. 3: (a) Original image; (b) distorted image created by Gaussian blurring ($\sigma = 0.99999$, kernel size $= 7 \times 7$); (c) naive SSIM index map; (d) proposed SSIM index map; (e) naive MS-SSIM index map; (f) proposed MS-SSIM index map.

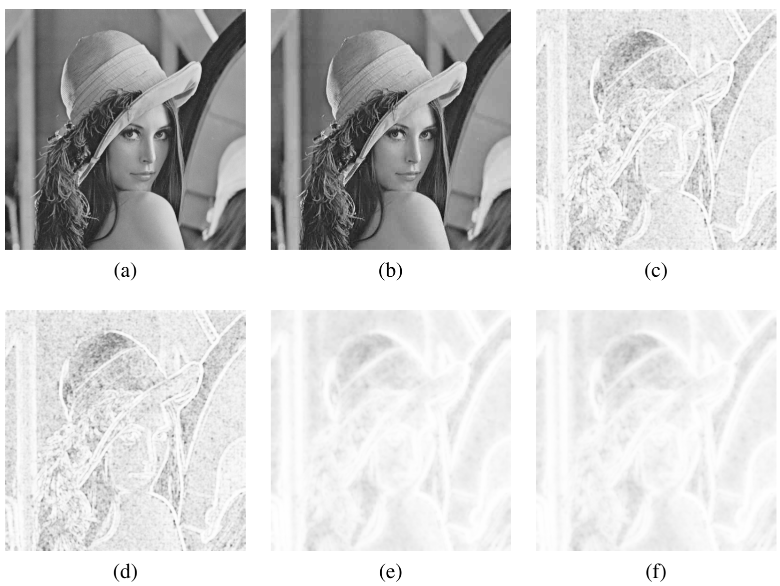

Fig. 4: (a) Original image; (b) distorted image created by JPEG compression (QP = 30); (c) naive SSIM index map; (d) proposed SSIM index map; (e) naive MS-SSIM index map; (f) proposed MS-SSIM index map.

[18] R. Dosselmann and X. D. Yang, "A comprehensive assessment of the structural similarity index," *Signal, Image and Video Processing*, vol. 5, no. 1, pp. 81–91, 2011.

[19] Z. Wang and A. C. Bovik, "A universal image quality index," *IEEE Signal Processing Letters*, vol. 9, no. 3, pp. 81–84, 2002.

[20] Z. Wang, E. P. Simoncelli, and A. C. Bovik, "Multiscale structural similarity for image quality assessment," in *Proc. Asilomar Conference on Signals, Systems Computers*, vol. 2, 2003, pp. 1398–1402 Vol.2.

[21] G. Chen, C. Yang, and S. Xie, "Gradient-based structural similarity for image quality assessment," in *Proc. International Conference on Image Processing (ICIP)*, 2006, pp. 2929–2932.

[22] M. P. Sampat, Z. Wang, S. Gupta, A. C. Bovik, and M. K. Markey, "Complex wavelet structural similarity: A new image similarity index," *IEEE Transactions on Image Processing*, vol. 18, no. 11, pp. 2385–2401, 2009.

[23] A. K. Moorthy and A. C. Bovik, "Visual importance pooling for image quality assessment," *IEEE Journal of Selected Topics in Signal Processing*, vol. 3, no. 2, pp. 193–201, 2009.

[24] Z. Wang and Q. Li, "Information content weighting for perceptual image quality assessment," *IEEE Transactions on Image Processing*, vol. 20, no. 5, pp. 1185–1198, 2011.

[25] A. Rehman, K. Zeng, and Z. Wang, "Display device-adapted video quality-of-experience assessment," in *Proc. SPIE 9394, Human Vision and Electronic Imaging*, 2015.

[26] K. Ma, Z. Duanmu, H. Yeganeh, and Z. Wang, "Multi-exposure image fusion by optimizing a structural similarity index," *IEEE Transactions on Computational Imaging*, vol. 4, no. 1, pp. 60–72, 2018.

[27] S. Wang, A. Rehman, K. Zeng, J. Wang, and Z. Wang, "Ssim-motivated two-pass vbr coding for hevc," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 27, no. 10, pp. 2189–2203, 2017.

TABLE II: Result of SSIM and MS-SSIM for Gaussian blur and JPEG compression degradation. MSE compares the error between naive and proposed implementation. The calculation time [ms] was averaged over 100 trails. Input is grayscale Lenna image.

	SSIM					MS-SSIM				
	Gaussian blur		JPEG compression			Gaussian blur		JPEG compression		
method	MSSIM	MSE	MSSIM	MSE	Time	MMS-SSIM	MSE	MMS-SSIM	MSE	Time
naive	0.914066	-	0.896569	-	21.01	0.958201	-	0.950684	-	141.14
proposed	0.914033	7.0×10^{-7}	0.896527	1.3×10^{-6}	3.30	0.958451	3.0×10^{-7}	0.951038	5.0×10^{-7}	17.40

TABLE III: Result of SSIM and MS-SSIM for Gaussian blur and JPEG compression degradation. MSE compares the error between naive and proposed implementation. Result is average of Kodak Photo CD [50].

	SSIM				MS-SSIM			
	Gaussian blur		JPEG compression		Gaussian blur		JPEG compression	
method	MSSIM	MSE	MSSIM	MSE	MMS-SSIM	MSE	MMS-SSIM	MSE
naive	0.849193	-	0.879107	-	0.901135	-	0.930397	-
proposed	0.849143	8.0×10^{-6}	0.879064	4.0×10^{-6}	0.900682	3.0×10^{-5}	0.930974	3.0×10^{-6}

[28] T. Zhao, J. Wang, Z. Wang, and C. W. Chen, "Ssim-based coarse-grain scalable video coding," *IEEE Transactions on Broadcasting*, vol. 61, no. 2, pp. 210–221, 2015.

[29] S. Wang, A. Rehman, Z. Wang, S. Ma, and W. Gao, "Ssim-inspired divisive normalization for perceptual video coding," in *IEEE International Conference on Image Processing (ICIP)*, 2011, pp. 1657–1660.

[30] A. Beghdadi, M.-C. Larabi, A. Bouzerdoum, and K. Iftekharuddin, "A survey of perceptual image processing methods," *Signal Processing: Image Communication*, vol. 28, no. 8, pp. 811 – 831, 2013.

[31] W. Lin and C.-C. J. Kuo, "Perceptual visual quality metrics: A survey," *Journal of Visual Communication and Image Representation*, vol. 22, no. 4, pp. 297 – 312, 2011.

[32] L. Zhang, L. Zhang, X. Mou, and D. Zhang, "A comprehensive evaluation of full reference image quality assessment algorithms," in *Proc. IEEE International Conference on Image Processing (ICIP)*, 2012, pp. 1477–1480.

[33] D. M. Chandler, "Seven challenges in image quality assessment: Past, present, and future research," *ISRN Signal Processing*, no. 905685, pp. 1–53, 2013.

[34] 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 27*, Z. Ghahramani, M. Welling, C. Cortes, N. D. Lawrence, and K. Q. Weinberger, Eds. Curran Associates, Inc., 2014, pp. 2672–2680. [Online]. Available: http://papers.nips.cc/paper/5423-generative-adversarial-nets.pdf

[35] D. M. Rouse and S. S. Hemami, "Understanding and simplifying the structural similarity metric," in *Proc. IEEE International Conference on Image Processing (ICIP)*, 2008, pp. 1188–1191.

[36] M. Chen and A. C. Bovik, "Fast structural similarity index algorithm," in *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2010, pp. 994–997.

[37] M.-J. Chen and A. C. Bovik, "Fast structural similarity index algorithm," *Journal of Real-Time Image Processing*, vol. 6, no. 4, pp. 281–287, 2011. [Online]. Available: https://doi.org/10.1007/s11554-010-0170-9

[38] K. Sugimoto and S. Kamata, "Fast image filtering by dct-based kernel decomposition and sequential sum update," in *Proc. IEEE International Conference on Image Processing (ICIP)*, 2012, pp. 125–128.

[39] ——, "Fast gaussian filter with second-order shift property of dct-5," in *Proc. IEEE International Conference on Image Processing (ICIP)*, 2013, pp. 514–518.

[40] K. Sugimoto and S. Kamata, "Efficient constant-time gaussian filtering with sliding dct/dst-5 and dual-domain error minimization," *ITE Transactions on Media Technology and Applications*, vol. 3, no. 1, pp. 12–21, 2015.

[41] K. Sugimoto, S. Kyochi, and S. Kamata, "Universal approach for dct-based constant-time gaussian filter with moment preservation," in *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2018, pp. 1498–1502.

[42] T. Yano, K. Sugimoto, Y. Kuroki, and S. Kamata, "Acceleration of gaussian filter with short window length using dct-1," in *Proc. Asia-Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA ASC)*, 2018, pp. 129–132.

[43] N. Fukushima, Y. Maeda, Y. Kawasaki, M. Nakamura, T. Tsumura, K. Sugimoto, and S. Kamata, "Efficient computational scheduling of box and gaussian fir filtering for cpu microarchitecture," in *Proc. Asia-Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA ASC)*, 11 2018, pp. 875–879.

[44] C. Tomasi and R. Manduchi, "Bilateral filtering for gray and color images," in *Proc. International Conference on Computer Vision (ICCV)*, 1998, pp. 839–846.

[45] K. Sugimoto and S. Kamata, "Compressive bilateral filtering," *IEEE Transactions on Image Processing*, vol. 24, no. 11, pp. 3357–3369, 2015.

[46] K. Sugimoto, N. Fukushima, and S. i. Kamata, "Fast bilateral filter for multichannel images via soft-assignment coding," in *Proc. Asia-Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA)*, 2016.

[47] K. Sugimoto, N. Fukushima, and S. Kamata, "200 fps constant-time bilateral filter using svd and tiling strategy," in *Proc. IEEE International Conference on Image Processing (ICIP)*, 2019, pp. 190–194.

[48] K. He, J. Sun, and X. Tang, "Guided image filtering," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 35, no. 6, pp. 1397–1409, 2013.

[49] N. Fukushima, K. Sugimoto, and S. Kamata, "Guided image filtering with arbitrary window function," in *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2018, pp. 1523–1527.

[50] "PhotoCD PCD0992," http://r0k.us/graphics/kodak/.

[51] G. Bradski and A. Kaehler, *Learning OpenCV: Computer vision with the OpenCV library*. " O'Reilly Media, Inc.", 2008.

[52] H. R. Sheikh, M. F. Sabir, and A. C. Bovik, "A statistical evaluation of recent full reference image quality assessment algorithms," *IEEE Transactions on Image Processing*, vol. 15, no. 11, pp. 3440–3451, 2006.

[53] N. Ponomarenko, V. Lukin, A. Zelensky, K. Egiazarian, M. Carli, and F. Battisti, "Tid2008 - a database for evaluation of full-reference visual quality assessment metrics," *Advances of Modern Radioelectronics*, vol. 10, pp. 30–45, 01 2009.

[54] N. Ponomarenko, L. Jin, O. Ieremeiev, V. Lukin, K. Egiazarian, J. Astola, B. Vozel, K. Chehdi, M. Carli, F. Battisti, and C.-C. J. Kuo, "Image database tid2013: Peculiarities, results and perspectives," *Signal Processing: Image Communication*, vol. 30, pp. 57 – 77, 2015. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0923596514001490

[55] E. Larson and D. Chandler, "Most apparent distortion: Full-reference image quality assessment and the role of strategy," *J. Electronic Imaging*, vol. 19, p. 011006, 01 2010.

[56] H. Lin, V. Hosu, and D. Saupe, "Kadid-10k: A large-scale artificially distorted iqa database," in *International Conference on Quality of Multimedia Experience (QoMEX)*, 2019, pp. 1–3.

[57] D. Kundu, L. K. Choi, A. C. Bovik, and B. L. Evans, "Perceptual quality evaluation of synthetic pictures distorted by compression and transmission," *Signal Processing: Image Communication*, vol. 61, pp. 54 – 72, 2018. [Online]. Available: http://signal.ece.utexas.edu/%7Ebevans/synthetic/"