

PAPER • OPEN ACCESS

Fast Implementation of Image Structural Similarity Algorithm

To cite this article: Yiwen Xu *et al* 2019 *IOP Conf. Ser.: Mater. Sci. Eng.* **533** 012057

View the [article online](#) for updates and enhancements.

Fast Implementation of Image Structural Similarity Algorithm

Yiwen Xu, Zhongwen Zhuang, Weimeng Li, Ying Fang^a

College of Physics and Information Engineering, Fuzhou University, Fuzhou
China

Abstract. The traditional Structural SIMilarity (SSIM) index has higher computation complexity compared with Peak-Signal-to-Noise-Ratio (PSNR). In order to solve this problem, we propose a two-stage optimization algorithm to reduce the computation complexity of SSIM. First, floating-point operations in the calculation process of SSIM are converted to fixed-point ones. Then, the repeated calculation rule of the iteration operation in SSIM is analyzed and the repeated operations are subsequently removed via modifying SSIM calculation method. Experiment results show that, in comparison with the original SSIM algorithm, the proposed algorithm obtains 24.12% of computation time reduction with negligible accuracy decrease.

1. Introduction

With the rapid development of digital image field, Image Quality Assessment (IQA), as one of the important aspects in the image processing, has attracted more and more researchers' attention and achieved remarkable results. It evaluates the performance of the algorithm by analyzing the various characteristics of the image and comparing the degree of distortion.

To our best knowledge, Peak-Signal-to-Noise-Ratio (PSNR) is one of the classic objective metrics in IQA, which is widely used in the image processing. It measures the degree of the distortion in the image by calculating the errors of the corresponding pixels between the original image and the distorted image. However, PSNR does not take into account the visual characteristics of the human eyes [1]. For example, the human eyes are more sensitive to the contrast difference of the lower frequency. In addition, the sensitivity to the contrast of the luminance is higher than that of the contrast of the chromaticity. Furthermore, the perception of the region by the human eyes is often affected by the surrounding areas. Therefore, the evaluation result of PSNR may be inconsistent with the Mean Opinion Score (MOS). Considering the characteristics of the Human Visual System (HVS), Wang *et al.* [2] proposed Structural SIMilarity (SSIM) index based on the characteristics of the human eyes that are sensitive to structural edges. It firstly extracted the brightness, contrast and structure information of the original image and the distorted image. Then, it calculated the similarities of the three features, respectively. Finally, the similarities were integrated to obtain the overall metric. SSIM ranges from 0 to 1. The larger the value is, the greater the similarity between the original image and the distorted image. Compared with the traditional PSNR, SSIM has obvious advantages in accuracy and practicability.

In practical applications, SSIM usually adds a window to calculate the local SSIM. Then it traverses the image and calculates the average value as SSIM of the entire image. As a result, this process generates a lot of repetitive operations and dramatically increases the calculation complexity of SSIM. In order to avoid unnecessary repetitive operations, in this paper, we calculate the SSIM of each image after performing block processing on the image. In addition, there are a large number of floating-point calculations in the SSIM



calculation process, which need much more calculation time than fixed-point calculations. Thereby, we optimize the SSIM formula by converting floating-point operations to fixed-point ones without affecting the accuracy of SSIM.

2. Advantages of Structural Similarity Algorithm

In the past few decades, Video Quality Assessment (VQA) [3-5] has developed rapidly and played an important role in the video processing. VQA is a part of IQA. VQA and IQA have two main roles. The one is to be used as a test tool to select the best quality video/image from multiple reconstructed videos/images, or to select the best encoding algorithm. The other is to be used as an optimization tool to embed VQA and IQA metrics into the encoder to optimize the algorithm for optimal coding efficiency. According to the conclusion in [6-7], the performance of SSIM is significantly better than that of PSNR in terms of saturation at high rate. During the encoding process, as the Bit Rate (BR) increases, the quality of the reconstructed video or image obtained by the encoder is also increasing. However, the quality of the reconstructed video or image will be close to that of the original video or image when the BR is increased to a certain extent. It means that increasing the BR will no longer affect the visual quality. The ideal IQA or VQA model needs to accurately reflect this saturation effect. PSNR is still rising at high BRs while SSIM can reach saturation. This is because SSIM does not guide the encoder to allocate bits to the position where the spatio-temporal perceptual quality is saturated during the optimization process for video or image coding. Therefore, SSIM can more accurately reflect the degree of distortion of the video or image compared to PSNR.

In addition, SSIM takes the characteristics of HVS into consideration. Thus, the performance of SSIM is also significantly superior to PSNR in terms of perceptual quality correlation [8-10].

From the above analyses, it can be seen that SSIM is better than PSNR in terms of the overall performance. PSNR only takes into account pixel error between the original image and the reconstructed image, while SSIM considers three different properties, including brightness, contrast and structure. It indicates that the computation complexity of SSIM is significantly higher than that of PSNR, which has negative effects on its applications in realtime video/image encoding. To solve this problem, we propose a modified SSIM calculation method to reduce its calculation time.

3. The Proposed Algorithm

The proposed algorithm optimizes the SSIM from two aspects to reduce its calculation time: converting floating-point operations to fixed-point operations and reducing the number of repeated operations.

3.1. Convert Floating-Point Operations to Fixed-Point Operations

The SSIM is calculated as follows [2]:

$$SSIM(x, y) = \frac{(2\mu_x\mu_y + C_1)(2\sigma_{xy} + C_2)}{(\mu_x^2 + \mu_y^2 + C_1)(\sigma_x^2 + \sigma_y^2 + C_2)} \quad (1)$$

where μ_x and μ_y represent the average of the grayscale value of image x and image y , respectively. δ_x and δ_y represent the variance of the grayscale value of image x and image y , respectively. δ_{xy} represents the covariance of the grayscale value of image x and image y . C_1 , C_2 and C_3 , which aim at maintaining stability, are constants. It can be seen from equation (1) that μ_x , μ_y , δ_x , δ_y , δ_{xy} are all floating-point numbers. It is well known that, floating-point operation consumes more time than fixed-point operation. So we can reduce the calculation time of SSIM via converting floating-point operations to fixed-point ones. However, it brings heavy impact on the result of SSIM if just cast a floating-point number to an integer number. Hence, in this paper, we reduce the calculation results of floating-point type by changing the calculation formula, so as to reduce the calculation time. The specific process is as follows. Assume that the size of the window is $w \times w$, x_i and y_i represent the grayscale values of the i -th pixel of image x and image y , respectively. Then μ_x and δ_x can be represented by x_i and y_i as shown in equation (2) and equation (3).

$$\mu_x = \frac{\sum x_i}{w^2} \quad (2)$$

$$\delta_x^2 = \frac{\sum (x_i - \mu_x)^2}{w^2} = \frac{(\sum x_i - 2x_i\mu_x + \mu_x^2)}{w^2} = \frac{\sum x_i^2}{w^2} - \left(\frac{\sum x_i}{w}\right)^2 \quad (3)$$

In a similar way, μ_y , δ_y and δ_{xy} also can be represented by x_i and y_i . Therefore, equation (1) can be described as:

$$SSIM(x, y) = \frac{2\sum x_i \sum y_i + w^4 C_1}{((\sum x_i)^2 + (\sum y_i)^2 + w^4 C_1)} \times \frac{2(w^2 \sum x_i y_i - \sum x_i \sum y_i) + w^4 C_2}{w^2((\sum x_i^2) + (\sum y_i^2)) - (\sum x_i)^2 + (\sum y_i)^2 + w^4 C_2} \quad (4)$$

After the above conversion, all the variables in equation (4) are fixed-point numbers, which means that all the floating-point operations in SSIM, except for the last process, are converted to fixed-point operations.

3.2. Reduce the Number of Iteration Operations

In the process of SSIM calculation, the window needs to traverse the whole image. In addition, each traversal can only move one row or one column of the current window. As a result, a large number of repeated operations are generated in the adjacent windows, which leave the gap of reducing computation complexity of SSIM.

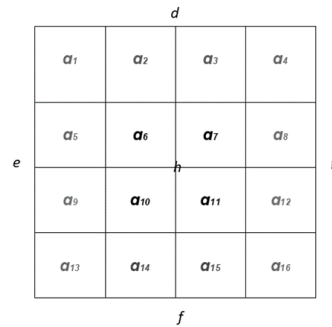


Figure 1. An example of repeated calculation in traditional SSIM

To demonstrate the derivation of computation complexity reduction, we use 3×3 size window traversing in 4×4 size image as an example. As shown in Fig.1, when the 3×3 window slide from the window centered at a_6 to the window centered at a_7 , there will be 2×3 parts (that is, $a_2, a_3, a_6, a_7, a_{10}, a_{11}$ as show in Fig.1) being repeatedly operated. Apparently, removing the repeated operations means reducing the calculation complexity of SSIM.

First, we divide the repeatedly calculated values into five parts, named as d, e, f, g and h , where

$$\begin{cases} d = a_2 + a_3 \\ e = a_5 + a_9 \\ f = a_{14} + a_{15} \\ g = a_8 + a_{12} \\ h = a_6 + a_7 + a_{10} + a_{11} \end{cases} \quad (5)$$

Then, we denote the calculation results of 3×3 windows centered at a_6, a_7, a_{10} and a_{11} as X_1, X_2, X_3 and X_4 , respectively. Combining equation (5), it is easy to obtain the following relationships:

$$\begin{cases} X_1 = a_1 + e + d + h \\ X_2 = a_4 + d + h + g \\ X_3 = a_{13} + e + h + f \\ X_4 = a_{16} + f + g + h \end{cases} \quad (6)$$

From equation (5) and (6), it can be seen that the repeated calculations in the iteration operations of the traditional SSIM algorithm are removed.

After the above two steps, there will be no floating point calculations except for the last process in the SSIM calculation formula and the repeated calculations will be removed in the window calculation.

4. Experiments and Results

The process of converting the floating-point operations to fixed-point ones in our proposal algorithm may bring calculation errors to SSIM results. In order to verify the accuracy of the proposed algorithm, we calculate the SSIM values of typical images (*Aerial_city*, *Boston*, *Bridge*, *et al.* as shown in Fig.2) from live image database [11]. The results are shown in Table 1.

It can be seen from Table 1 that the deviation between the SSIM values calculated by the traditional algorithm and the proposed algorithm are so small that can be negligible.

In order to verify the effect of the proposed algorithm on calculation complexity, we use an 11×11 window to traverse the images with different resolutions, which are 128×128, 256×256, and 512×512. The calculation time and time decrease of the traditional algorithm and the proposed algorithm are shown in Table 2.

Table 1. Accuracy of the Proposed Algorithm

Image	SSIM values		Relative error (%)
	Traditional algorithm	Proposed algorithm	
<i>Aerial_city</i>	0.77642	0.77629	0.016
<i>Boston</i>	0.80971	0.80962	0.011
<i>Bridge</i>	0.79235	0.79231	0.005
<i>butter_flower</i>	0.73245	0.73113	0.180
<i>cactus</i>	0.81772	0.81215	0.681
<i>child_swimming</i>	0.83214	0.83163	0.061
<i>couple</i>	0.75564	0.75181	0.506
<i>family</i>	0.71998	0.71633	0.506
<i>fisher</i>	0.79877	0.79134	0.930

Table 2. Complexity Reduction of the Proposed Algorithm

Image resolution	Traditional algorithm (s)	Proposed algorithm	
		(s)	(%)
128x128	0.464	0.355	23.49
256x256	0.792	0.602	23.98
512x512	1.097	0.824	24.88
average	/	/	24.12

From Table 2, it can be seen that the calculation time of SSIM increases with the increase of the image resolution. The reason is that the number of iteration operations in SSIM calculation is increase with the image resolution. The more times the window is traversed, the longer the calculation time is required. The results in Table 2 also show that, compared with the traditional algorithm, the calculation time of our proposal algorithm is reduced by 24.12% in average. The result of the table 1 and the table 2 show that, the propose method improve the computational efficiency and not affect the calculation accuracy.

In summary, compared to the tradition algorithm, the proposed algorithm reduces the calculation time to a certain extent under the condition that the accuracy is basically unchanged. The proposed algorithm is effective in reducing the calculation time in the case of different image resolutions. Experimental results prove that after two steps of optimization, the proposed algorithm is better than traditional algorithm which has better practicability.

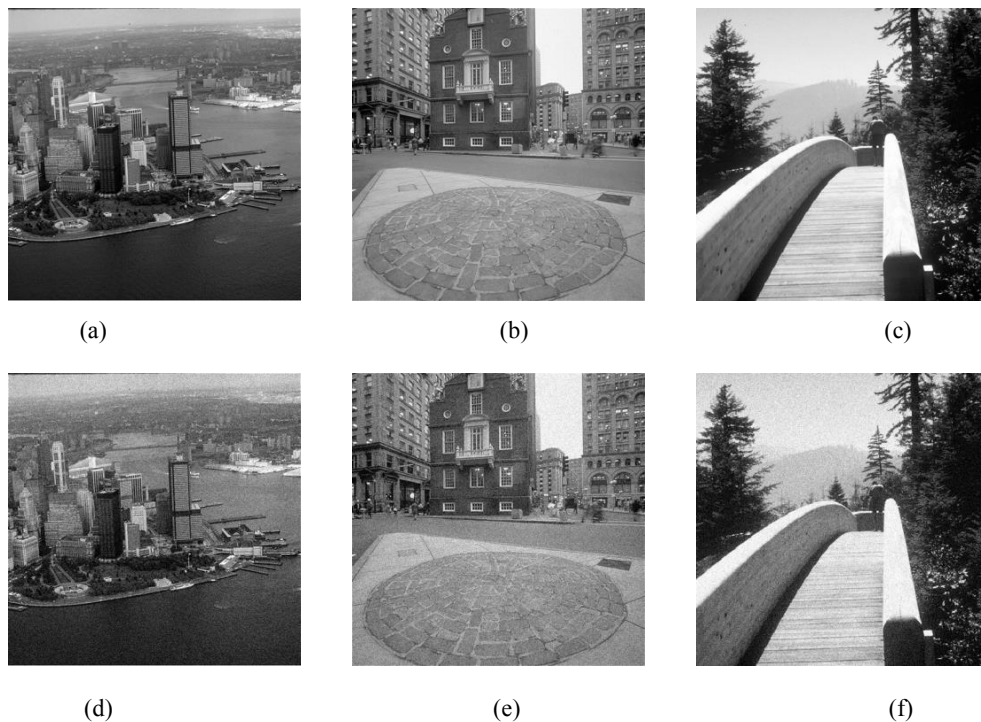


Figure 2. Live image database samples. (a)-(c) are original images. (d)-(f) are distorted images.

5. Conclusions

In this paper, we propose a modified SSIM algorithm to reduce the computation complexity of traditional SSIM algorithm. The proposed algorithm is composed of two stages, one is the conversion from floating-point operations to fixed-point ones, and the other is reduction of the number of iteration operations in SSIM calculation. The experimental results show that the proposed algorithm can effectively reduce the calculation time of SSIM without decrease of its accuracy. The proposed optimization scheme can be applied to various image quality evaluations.

This method reduces the calculation time to a certain extent. However, there is still much room for improvement. In future works, we will optimize the algorithm by further analyzing the image structure and studying on the application of SSIM in image quality evaluation to reduce more computation time.

References

- [1] Z. Wang, 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, Jan. 2009.
- [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, Apr. 2004.

- [3] Z. Wang, A. C. Bovik, "A universal image quality index," IEEE Signal Processing Letters, vol. 9, no.3, pp. 81-84, Mar. 2002.
- [4] Z. Wang, L. Lu, and A.C. Bovik, "Video quality assessment based on structural distortion measurement," Signal Processing: Image Communication. Special issue on Objective Video Quality Metrics, vol. 19, no.2, pp. 121-132, Feb. 2004.
- [5] Z. Wang, E. P. Simoncelli, and A. C. Bovik, "Multi-scale structural similarity for image quality assessment," Proc IEEE Asilomar Conference on Signals, vol. 2, no. 2, pp.1398-1402, Nov. 2003.
- [6] T. Zhao , K. Zeng, A. Rehman, and Z. Wang. "On the use of SSIM in HEVC," Asilomar Conference on Signals, Systems and Computers, 2013, pp. 1107–1111.
- [7] S. Wang, A. Rehman, Z. Wang, et al. "SSIM–motivated rate–distortion optimization for video Coding," IEEE Transactions on Circuits and Systems for Video Technology, vol. 22, no. 4, pp. 516–529, Apr. 2012.
- [8] A. Rehman, "SSIM-inspired quality assessment, compression, and processing for visual communications", Ph.D. Thesis. University of Waterloo: Canada, 2013.
- [9] 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, Dec. 2011.
- [10] 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, Apr. 2012.
- [11] R. L. DeValois and K. K. DeValois, Spatial Vision, Oxford University Press, Oxford, UK, 1990.