J Real-Time Image Proc (2011) 6:281–287
DOI 10.1007/s11554-010-0170-9

ORIGINAL RESEARCH PAPER

Fast structural similarity index algorithm

Ming-Jun Chen · Alan C. Bovik

Received: 17 January 2010 / Accepted: 12 July 2010 / Published online: 4 August 2010

Abstract The development of real-time image and video quality assessment algorithms is an important direction on which little research has focused. Towards this end, we present a design of real-time implementable full-reference image/video quality algorithms that are based on the Structural SIMilarity (SSIM) index and multi-scale SSIM (MS-SSIM) index. The proposed algorithms, which modify SSIM/MS-SSIM to achieve speed of execution, were tested on the LIVE Image Quality Database and LIVE Video Quality Database. The experimental results show that the performance of the new, fast algorithms is commensurate with that of SSIM and MS-SSIM, but with much lower computational complexity. Indeed, the proposed Fast MS-SSIM algorithm is 10 times faster (lower complexity) than the MS-SSIM algorithm, while the proposed Fast SSIM is 2.68 times faster than SSIM without parallel computing optimization.

Keywords Real time · Video quality assessment · SSIM · Image quality assessment · Low complexity algorithm

1 Introduction

With the increasing prevalence of digital images and videos, people live in an era rich in digital visual information. Effective systems for automatic image quality differentiation are thus urgently needed to help manage the abundance of available digital visual content. In the field of full-reference (FR) image and video quality assessment research, where it is assumed that the algorithm has a reference (pristine) image available to it, a variety of popular algorithms have been proposed and studied. For instance, "Yonsei" has been recommended by the VQEG group as an FR quality assessment method in the J.144 document [1], while the Structural SIMilarity (SSIM) index [2] has been widely used in many FR image quality assessment applications. A number of algorithms have been derived from SSIM: Multi-scale SSIM (MS-SSIM) [3], Percentile Pooling SSIM (P-SSIM) [3], Complex-Wavelet SSIM (CW-SSIM) [5], Gradient-based Structural Similarity (G-SSIM) [6], and Three-Component Weighted SSIM [7]. All these derivative algorithms aim to improve predictive accuracy of SSIM relative to human subjectivity, but inevitably increase the computational complexity.

The surge of mobile applications has created a demand for low complexity algorithms that can run on handheld and other mobile devices. Multimedia services, such as video on demand (VOD) and IPTV are already available on mobile devices, further necessitating algorithms that have low complexity. Here we focus on reducing the computational complexity of the popular SSIM index, and propose a low complexity version of the algorithm which we call *Fast SSIM*. Fast SSIM performs at a level comparable to SSIM in terms of correlation with human subjectivity, as measured on the LIVE Image Quality Database. We also extend the Fast SSIM concept to the Multi-Scale SSIM index, which has better prediction performance than single scale SSIM. The resulting Fast MS-SSIM algorithm also performs commensurate with that of MS-SSIM in terms of agreement with human subjectivity.

M.-J. Chen (✉) · A. C. Bovik
Laboratory for Image and Video Engineering (LIVE),
The University of Texas, Austin, USA
e-mail: mjchen@mail.utexas.edu

A. C. Bovik
e-mail: bovik@ece.utexas.edu

The rest of this paper is organized as follows. Section 2 reviews the SSIM and MS-SSIM indices. In the following sections, the specific elements of the SSIM index are considered in light of their computational complexity, and how they can be optimized to achieve more efficient performance, thus yielding Fast SSIM. Section 3 describes the development of the luminance term of Fast SSIM, while in Sect. 4, the contrast term and the structural terms of FAST SSIM and Fast MS-SSIM are elaborated. Section 5 explains how optimization can be applied to Fast SSIM and Fast MS-SSIM. Experiments are presented in Sect. 6, and concluding remarks are offered in Sect. 7.

2 Structural similarity index

We review the SSIM and MS-SSIM indices.

2.1 Single scale structural similarity index

Based on the hypothesis that the HVS is highly adapted for extracting structural information, the SSIM algorithm contains three terms that capture different aspects of the similarity (or lack thereof) between two non-negative signals $\mathbf{x}$ and $\mathbf{y}$: luminance $l(\mathbf{x}, \mathbf{y})$, contrast $c(\mathbf{x}, \mathbf{y})$, and structure $s(\mathbf{x}, \mathbf{y})$:

$$l(\mathbf{x},\mathbf{y}) = \frac{2\mu_{\mathbf{x}}\mu_{\mathbf{y}} + C_1}{\mu_{\mathbf{x}}^2 + \mu_{\mathbf{y}}^2 + C_1} \tag{1}$$

$$c(\mathbf{x},\mathbf{y}) = \frac{2\sigma_{\mathbf{x}}\sigma_{\mathbf{y}} + C_2}{\sigma_{\mathbf{x}}^2 + \sigma_{\mathbf{y}}^2 + C_2} \tag{2}$$

$$s(\mathbf{x},\mathbf{y}) = \frac{\sigma_{\mathbf{xy}} + C_3}{\sigma_{\mathbf{x}}\sigma_{\mathbf{y}} + C_3} \tag{3}$$

where $C_1 = (K_1L)^2$, $C_2 = (K_2L)^2$, and $C_3 = C_2/2$ are small constants; L is the dynamic range of the pixel values, and $K_1 \ll 1$ and $K_2 \ll 1$ are scalar constants. The constants C_1, C_2, and C_3 ensure stability when the denominator approaches zero. Combining the three terms, the common form of SSIM is:

$$\begin{aligned}\mathrm{SSIM}(\mathbf{x},\mathbf{y}) = & \; [l(\mathbf{x},\mathbf{y})]\,[c(\mathbf{x},\mathbf{y})]\,[s(\mathbf{x},\mathbf{y})] \\ = & \; \frac{(2\mu_{\mathbf{x}}\mu_{\mathbf{y}} + C_1)(2\sigma_{\mathbf{xy}} + C_2)}{\left(\mu_{\mathbf{x}}^2 + \mu_{\mathbf{y}}^2 + C_1\right)\left(\sigma_{\mathbf{x}}^2 + \sigma_{\mathbf{y}}^2 + C_2\right)}\end{aligned} \tag{4}$$

The overall SSIM index value between signals $\mathbf{x}$ and $\mathbf{y}$ is defined as the average of all the SSIM index values calculated within a (commonly) 11 × 11 isotropic Gaussian weighting window passed over the image, although other "pooling" strategies than averaging exist [4]. The Gaussian weighting window prevents artifacts arising from a discontinuous truncation of the local image patches when computing the local values of the SSIM index map.

2.2 Multi-scale structural similarity index

The distance between the image and the observer affects the observer's perceived image quality. The results of subjective impressions of quality can vary as the viewing distance changes. In addition, images are naturally multi-scale, and both distortions and image features possess multi-scale attributes. For these reasons, the Multi-scale SSIM (MS-SSIM) index was developed.

In MS-SSIM, quality assessment is performed on multiple scales of the reference and the distorted images. Low-pass filtering and down-sampling are applied iteratively (as shown in Fig. 1), and elements of the SSIM index are applied at each scale, indexed from 1 (original image) through the finest scale M obtained after M − 1 iterations.

At each scale i, the contrast and structure terms are calculated: cj(x, y) and sj(x, y), respectively. The luminance term is computed only at scale M and represented as l_M(x, y). The overall quality evaluation is obtained by combining the measurement over scales:

$$\begin{aligned}&\mathrm{MS\text{-}SSIM}(\mathbf{x},\mathbf{y}) \\ &= [l_M(\mathbf{x},\mathbf{y})]^{\alpha_M}\prod_{j=1}^{M}[c_j(\mathbf{x},\mathbf{y})]^{\beta_j}[s_j(\mathbf{x},\mathbf{y})]^{\gamma_j}\end{aligned} \tag{5}$$

where typically $M = 5$, and the exponents σ_M, β_j, γ_j are selected such that $\sigma_M = \beta_j = \gamma_j$ and $\sum_{j=1}^{M}\gamma_j = 1$ [3].

3 Luminance term

As noted by Rouse and Hemami [8], the luminance term of the SSIM index often plays a less significant perceptual role in predicting visual quality than the other terms. They proposed eliminating it to reduce complexity. We choose to preserve the luminance term since images may suffer from a luminance bias, even if not all image quality databases explicitly include such distortions. Nevertheless, we have sought to expend as little computation as possible on the luminance term.

The luminance term in Fast SSIM utilizes an 8 × 8 square window, and an integral image technique [9] to compute the luminance similarity between the reference and test images.

By using the so-called integral image, extraction of the mean value of the pixels within a square window can be made quite efficient. As shown in Fig. 2, the value of the integral image at (x, y) is the sum of the pixels values above and to the left of (x, y), and including the value at (x, y).

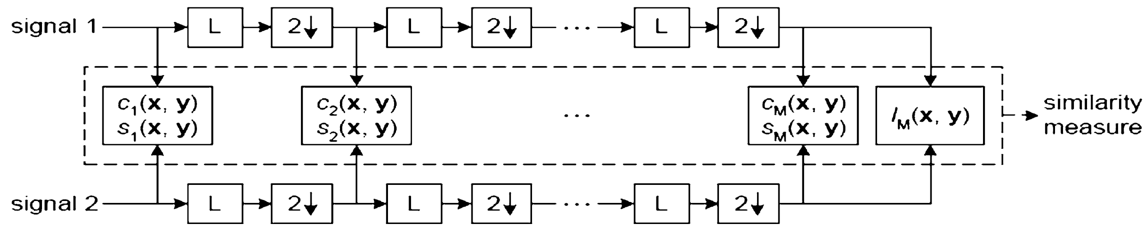

Fig. 1 Multi-scale SSIM. L low-pass filtering, 2↓ down-sampling by 2

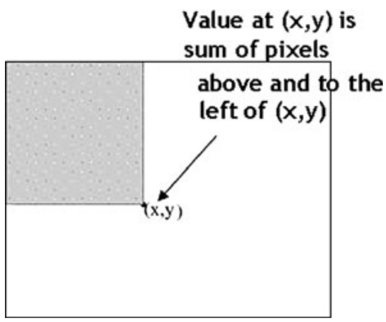

Fig. 2 Illustration of integral image

Computing the sum over any rectangular area can be achieved with only two additions and one subtraction. As shown in Fig. 3, the sum of the pixel values within the rectangle D can be computed using four array references. In Fig. 3, there are four pixels marked as 1, 2, 3 and 4 and their values are defined as V_1, V_2, V_3 and V_4, respectively. The value of the integral image at location 1 (V_1) is the sum of the pixels in rectangle A. The value at location 2 (V_2) is A + B, at location 3 (V_3) is A + C, and at location 4 (V_4) is A + B + C + D. The sum of the values in region D can be computed as $V_4 + V_1 - (V_2 + V_3)$.

Using the integral image [9] and a square window, the complexity of computing the luminance term is reduced considerably. Assuming the window size is $n \times n$, then to compute the mean pixel values of this window, the standard SSIM index algorithm (using a Gaussian weighted window) requires n^2 multiplies and $(n^2 - 1)$ additions,

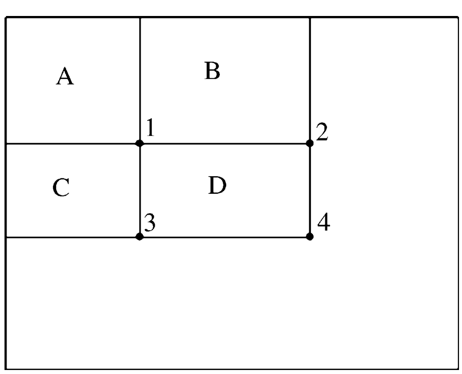

Fig. 3 How to compute sum value over region D in integral image domain

while the proposed Fast SSIM algorithm only requires three additions and one subtraction. Meanwhile, the computation needed to generate the integral image is negligible.

Assume we compute the SSIM and Fast SSIM indices on an image (width = w, height = h) with an $n \times n$ window size. To compute an integral image we need (wh-1) addition operations. The overall reduction in computation is the product of the computational reduction per window by the number of windows, where the computational reduction per window is n^2 multiplies + $(n^2 - 4)$ additions -1 subtraction, while the number of windows is $(w - n + 1)(h - n + 1)$. Hence, applying the integral image to calculate the SSIM luminance term reduces the computational complexity dramatically.

4 Contrast and structure terms

Computation of the variance term is the most time-consuming part of the SSIM algorithm. In order to lower the complexity, we substitute the variance with a gradient value in Fast SSIM. Following Field [10], we note that while images of real-world scenes vary greatly in their absolute luma and chroma distributions, the gradient magnitudes of natural images generally obey heavy tailed distribution laws. Indeed, some no-reference image quality assessment algorithms [11], [12] use the gradient image to assess blur severity. Similarly, the performance of the gradient-based SSIM index suggests that applying SSIM on the gradient magnitude may yield slightly higher quality assessment performance. The gradient is certainly responsive to image variations. Moreover, the gradient magnitude has low complexity and is amenable to integer-only implementation.

We generate the gradient image using the Roberts gradient templates depicted in Fig. 4.

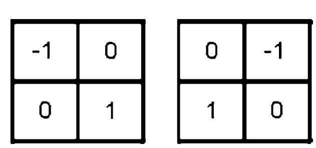

Fig. 4 Roberts gradient templates

The gradient magnitude is approximated by[1]

$$|\nabla I| = \max\{|\nabla i|, |\nabla j|\} + (1/4)\min\{|\nabla i|, |\nabla j|\} \tag{6}$$

where ∇i and ∇j are the Roberts template responses in the two orthogonal directions. This approximation is based upon a simple expansion of the gradient.

Assuming the image has dimensions $N_x \times N_y$, the contrast $c(\mathbf{x}, \mathbf{y})$ and structure $s(\mathbf{x}, \mathbf{y})$ terms of the Fast SSIM index algorithm are then defined:

$$c(\mathbf{x},\mathbf{y}) = \frac{(2\mu_{G\mathbf{x}}\mu_{G\mathbf{y}} + C_2)}{\left(\mu_{G\mathbf{x}}^2 + \mu_{G\mathbf{y}}^2 + C_2\right)} \tag{7}$$

$$s(\mathbf{x},\mathbf{y}) = \frac{(\mu_{GxGy} + C_3)}{(\mu_{G\mathbf{x}}\mu_{G\mathbf{y}} + C_3)} \tag{8}$$

where $C_3 = C_2/2$, and

$$\mu_{G\mathbf{x}} = \frac{1}{N_\mathbf{x}N_\mathbf{y}}\sum_{x=1}^{N_\mathbf{x}}\sum_{y=1}^{N_\mathbf{y}}|\nabla\mathbf{x}(x,y)| \tag{9}$$

$$\mu_{GxGy} = \frac{1}{N_\mathbf{x}N_\mathbf{y}}\sum_{x=1}^{N_\mathbf{x}}\sum_{y=1}^{N_\mathbf{y}}|\nabla\mathbf{x}(x,y)||\nabla\mathbf{y}(x,y)| \tag{10}$$

and where $|\nabla\mathbf{x}|$ and $|\nabla\mathbf{y}|$ are the gradient magnitude values of the images $\mathbf{x}$ and $\mathbf{y}$, estimated using the approximation (6).

The Fast SSIM index between $\mathbf{x}$ and $\mathbf{y}$ is then:

$$\text{Fast-SSIM}(\mathbf{x},\mathbf{y}) = \frac{(2\mu_\mathbf{x}\mu_\mathbf{y} + C_1)(2\mu_{GxGy} + C_2)}{\left(\mu_\mathbf{x}^2 + \mu_\mathbf{y}^2 + C_1\right)\left(\mu_{G\mathbf{x}}^2 + \mu_{G\mathbf{y}}^2 + C_2\right)}. \tag{11}$$

In MS-SSIM, the contrast and structural terms are calculated over multiple scales. Therefore, the Fast MS-SSIM index between signal $\mathbf{x}$ and $\mathbf{y}$ is defined as:

$$\text{Fast-MS-SSIM}(\mathbf{x},\mathbf{y}) = [l_M(\mathbf{x},\mathbf{y})]^{\alpha_M}\prod_{j=1}^{M}[c_j(\mathbf{x},\mathbf{y})]^{\beta_j}[s_j(\mathbf{x},\mathbf{y})]^{\gamma_j} \tag{12}$$

where $M = 5$, and the exponents σ_M, β_j, γ_j are selected such that $\sigma_M = \beta_j = \gamma_j$ and $\sum_{j=1}^{M}\gamma_j = 1$.

Another modification is using an integer approximation to the Gaussian weighting window. In this way, the computation of the contrast and the structure terms uses only integer operations. Figure 5 shows the window. In addition, because the contrast and the structure terms are more important than the luminance term, allowing for a flexible integer window design can help improve the quality evaluation ability of the algorithm without increasing the computation complexity. The simplifications provide that all computations are reduced to integer operations, with square roots eliminated.

[1] An accurate truncated expansion approximation learned by author ACB whilst lecturing at Texas Instruments in the 1990s.

0	0	0	1	1	0	0	0
0	0	1	2	2	1	0	0
0	1	2	4	4	2	1	0
1	2	4	8	8	4	2	1
1	2	4	8	8	4	2	1
0	1	2	4	4	2	1	0
0	0	1	2	2	1	0	0
0	0	0	1	1	0	0	0

Fig. 5 8 × 8 integer approximation to Gaussian window

The other modification applied to create the Fast MS-SSIM algorithm is the use of sub-sampling. We suggest that the contrast and the structure terms need not be computed at the original scale in Fast MS-SSIM. Since humans are less sensitive to localized higher spatial frequencies, skipping computation of the contrast and the structure terms at the finest scale can increase the computation speed dramatically without lowering performance. The experiment results given in the Sect. 6 support this assumption.

5 Optimization

Optimization is an essential process when implementing an algorithm for industrial applications, especially for real-time applications. A natural approach would be to use parallel computing optimization on the Fast SSIM and MS-SSIM index algorithms to achieve better performance. In order to demonstrate the potential performance of an optimized version of our proposed algorithm, data-level parallelization and frame-level parallelization are implemented in this study.

In our fast algorithms, about 80% of the computation is consumed by the contrast and the structure terms once the luminance term is simplified. Since most of the operations in the computation of the contrast and the structure terms in Fast SSIM and MS-SSIM are integer-only, Fast SSIM and MS-SSIM are amenable to Single Instruction Multiple Data (SIMD) optimization. Also, since Fast SSIM and MS-SSIM do not currently use any dependency between frames, it is natural to conduct frame-level parallelization. In our implementation, the computation of (7) and (8) are optimized using SIMD, while multi-threading is applied at the frame level.

6 Experimental results

The LIVE Image Quality Database [13] and the LIVE Video Quality Database [14–16] were used in the following experiment.

The LIVE Image Quality Database includes difference mean opinion score (DMOS) subjective scores for each image and six types of distortions. The distortions include JPEG2000 compression distortion (227 images), JPEG compression distortion (233 images), white noise (174 images), Gaussian blur (174 images), and fast-fading channel noise (174 images).

The LIVE Video Quality Database also includes DMOS subjective scores for 150 distorted videos created from 10 high-quality reference videos (15 distorted videos per reference). The database includes four different distortion types: MPEG-2 compression, H.264 compression, simulated transmission of H.264 compressed bit streams through error-prone IP networks, and simulated transmission of H.264 compressed bit streams through error-prone wireless networks.

6.1 Performance: quality assessment

To provide a quantitative performance evaluation, we abide by the guidelines on experiments design published by the Video Quality Experts Group (VQEG) [17]; we use the Spearman Rank Order Correlation Coefficient (SROCC), the linear correlation coefficient (LCC) (after non-linear regression), and the root mean-squared error (RMSE) (after non-linear regression) as objective measures of algorithm performance relative to human subjectivity. Higher correlation values between the scores predicted by the tested algorithm and human DMOS scores indicate that the ratings supplied by the tested algorithm better predict those of human subjects'.

Following non-linear regression, the LCC and RMSE between the subjective and the objective scores are calculated as measures of the *prediction accuracy* of our proposed algorithms. The non-linearity chosen to fit the data is a five-parameter logistic function (a logistic function with an added linear term, and constrained to be monotonic) given by:

$$\mathrm{Quality}(x) = \beta_1 \mathrm{logistic}(\beta_2, (x - \beta_3)) + \beta_4 + \beta_5$$

$$\mathrm{logistic}(\tau, x) = \frac{1}{2} - \frac{1}{1 + \exp(\tau x)} \tag{13}$$

The LCC value between two random vectors X and Y is

$$\mathrm{LCC}(X, Y) = \frac{\sum_{i=1}^{n} (X_i - \bar{X})(Y_i - \bar{Y})}{\sqrt{\sum_{i=1}^{n} (X_i - \bar{X})^2}\sqrt{\sum_{i=1}^{n} (Y_i - \bar{Y})^2}} \tag{14}$$

where $\bar{X}$ and $\bar{Y}$ are the means of X and Y, and n is their lengths. We calculated the LCC between the predicted quality scores and the DMOS scores.

Prediction monotonicity of IQA algorithms is measured by the SROCC. The SROCC value between the predicted scores X and the subjective scores DMOS is

$$\mathrm{SROCC}(X, \mathrm{DMOS}) = 1 - \frac{6\sum_{i=1}^{n} d_i}{n(n^2 - 1)} \tag{15}$$

where $d_i = x_i - \mathrm{DMOS}_i$, and the two variables x_i and DMOS_i are the ranks of X_i (the predicted score of image i) and the DMOS score of image i, respectively.

Table 1 clearly demonstrates that the Fast SSIM algorithm suffers no performance loss in terms of its subjective quality prediction capability on images. The SROCC, LCC, and RMSE values of SSIM and Fast SSIM are very close.

The performance numbers on the LIVE Image Quality Database for Fast MS-SSIM are shown in Table 2. The modifications in Fast MS-SSIM are the same as the modification in Fast SSIM, except that we propose to skip the analysis on the contrast and the structure terms at the finest scale. Table 2 shows that the SROCC, LCC, and RMSE values of Fast MS-SSIM and Fast MS-SSIM with subsampling are very close, although slightly lower than MS-SSIM.

Table 3 and Table 4 are the performance numbers on the LIVE Video Quality Database using a square window. In our experiments, using a square window yields better performance than using the integer Gaussian window, while the integer Gaussian window has better performance than the square window on the still image database. Of course, this is as measured on a single database suggest possible advantages of allowing flexible window design in the computation of the various SSIM terms.

From Table 3 and Table 4, the performance gap between perceptual-based algorithms (SSIM, MS-SSIM, Fast SSIM, and Fast MS-SSIM) and peak signal-to-noise

Table 1 Experimental result of SSIM and Fast SSIM on LIVE image quality database

Algorithm	SROCC	LCC	RMSE
PSNR*	0.8755	0.8709	13.4265
SSIM	0.9244	0.9299	8.5045
Fast SSIM (*Luminance term optimized only*)	0.9233	0.9413	7.8065
Fast SSIM (*Fully optimized*)	0.9214	0.9373	8.0554

* The performance numbers for PSNR are from [18]

Table 2 Experimental result of MS-SSIM and MS Fast SSIM on LIVE image quality database

Algorithm	SROCC	LCC	RMSE
PSNR*	0.8755	0.8709	13.4265
MS-SSIM	0.9429	0.9439	7.6327
Fast MS-SSIM (*Luminance term optimized only*)	0.9425	0.9226	8.9153
Fast MS-SSIM (*Fully optimized*)	0.9409	0. 9369	8.0787

* The performance numbers for PSNR are from [18]

Table 3 Experimental result of SSIM and Fast SSIM on LIVE Video Quality Database

Algorithm	SROCC	LCC	RMSE
PSNR*	0.3684	0.4085	–
SSIM	0.4953	0.5228	9.3573
Fast SSIM (*Luminance term optimized only*)	0.4954	0.5176	9.3816
Fast SSIM (*Luminance, contrast, and structure terms optimized*)	0.4949	0.5191	9.3825

* The performance numbers for PSNR are from [14]

Table 4 Experimental result of MS-SSIM and MS Fast SSIM on LIVE Video Quality Database

Algorithm	SROCC	LCC	RMSE
PSNR*	0.3684	0.4085	–
MS-SSIM	0.7593	0.7634	7.0899
Fast MS-SSIM (*Luminance term optimized only*)	0.7546	0.7625	7.1025
Fast MS-SSIM (*Full-optimized*)	0.7049	0.7131	7.6955
Fast MS-SSIM (*Full-optimized +sub-sampling*)	0.6991	0.7118	7.7103

Full-optimized here means luminance, contrast, and structure terms are optimized with fast algorithm

* The performance numbers for PSNR are from [14]

ratio (PSNR) becomes very obvious, particularly on videos. Table 3 illustrates that the performance of Fast SSIM is almost the same as SSIM. Table 4 shows that both MS-SSIM and Fast MS-SSIM are much better than PSNR as measured by SROCC and LCC scores. Although the performance of Fast MS-SSIM is a little bit lower than MS-SSIM, it still has excellent performance. In addition, by comparing the SROCC and LCC scores of Fast MS-SSIM and Fast MS-SSIM with sub-sampling (the Fast MS-SSIM algorithm we proposed), it can be seen that skipping the higher spatial frequency information has little impact on performance.

6.2 Performance: speed

The performance numbers on speed were tested on the 768×432 progressive video with 250 frames (the format used in the LIVE Video Quality Database). All experiments were conducted on an Intel Core 2 Duo 2.2 GHz platform, except the experiment result of the multi-threading optimization, which was run on an Intel Core 2 Quad Q6600 platform.

Tables 5 and 6 are the performance numbers for execution speed for each step. In Table 5, we can see the improvement from SSIM to Fast SSIM is 168% (from 3.42 to 9.17 fps). Thus Fast SSIM is 2.68 times faster than SSIM. For optimization, Intel SSE2 instructions were implemented to calculate the mean and the correlation of the gradient images to demonstrate the improvement on applying data-level parallelization. As shown in Table 5, Fast SSIM with SIMD enhances the performance from 9.7 to 16.6 fps. Finally, with multi-threading optimization, Fast SSIM reaches 57.83 fps on an Intel Core 2 Quad platform, which qualifies the algorithm for real-time applications.

Table 5 Experimental result of Fast SSIM computational speed on LIVE Video Quality Database

Algorithm	Speed (fps)
SSIM	3.42
Fast SSIM (*Luminance term optimized only*)	5.09
Fast SSIM	9.17
Fast SSIM (*SIMD*)	16.6
Fast SSIM (*SIMD + multi-threading*)	57.83

Table 6 Experimental result of Fast MS-SSIM computational speed on LIVE Video Quality Database

Algorithm	Speed (fps)
MS-SSIM	2.54
Fast MS-SSIM (*Luminance term optimized only*)	3.74
Fast MS-SSIM	6.4
Fast MS-SSIM (***Proposed algorithm** using sub-sampling*)	25.31
Fast MS-SSIM (*SIMD + sub-sampling*)	35.34
Fast MS-SSIM (*SIMD + sub-sampling + multi-threading*)	121.97

The performance numbers on speed for Fast MS-SSIM are shown in Table 6. Table 6 shows that the proposed Fast MS-SSIM algorithm is close to 10 times faster than MS-SSIM (from 2.54 to 25.31 fps). With some optimization skills the algorithm can easily achieve speeds adequate for real-time application. For instance, in our optimization, we achieve about 122 fps in a Quad Core machine.

7 Concluding remarks

In this paper we proposed Fast SSIM and Fast MS-SSIM index algorithms and verified their performance on the LIVE Image Quality Database and LIVE Video Quality Database. The experimental results show that the proposed algorithms not only have competitive performance with SSIM and MS-SSIM for assessing image quality, but have much lower computational complexity. The proposed Fast MS-SSIM algorithm is about 10 times faster than the MS-

SSIM algorithm yielding close to real-time performance (25.31 fps) without optimization. Further, the proposed algorithm achieves real-time performance with simple optimization.

References

1. LC-Text: Recommendation J.144 (Rev.1)—Objective perceptual video quality measurement techniques for digital cable television in the presence of a full reference
2. Wang, Z., Bovik, A.C., Sheikh, H.R., Simoncelli, E.P.: Image quality assessment: from error visibility to structural similarity. IEEE Trans. Image Process **13**(4), 600–612 (2004)
3. Wang, Z., Simoncelli, E.P., Bovik, A.C.: Multi-scale structural similarity for image quality assessment. IEEE Asilomar Conf. Signals Syst. Comput. **2**, 1398–1402 (2003)
4. Moorthy, A.K., Bovik, A.C.: Visual importance pooling for image quality assessment. IEEE J. Sel. Top. Signal Process **3**(2), 193–201 (2009)
5. Wang, Z., Simoncelli, E.P.: Translation insensitive image similarity in complex wavelet domain, IEEE. Intl. Conf. Acoust. Speech Signal Process **2**, 573–576 (2005)
6. Chen, G.H., Yang C.L., Xie, S.L.: Gradient-based structural similarity for image quality assessment. Proc. ICIP, 2929–2932 (2006)
7. Li, C., Bovik, A.C.: Three-component weighted structural similarity index. In: SPIE Conference on image quality and system performance, Jan 19–22, 2009, San Jose, California (2009)
8. Rouse, D.M., Hemami, S.S.: Understanding and simplifying the structural similarity metric. IEEE Int. Conf. Image Process (ICIP), 1188–1191 (2008)
9. Porikli, F.: Integral histogram: a fast way to extract histograms in Cartesian spaces. IEEE CVPR **1**, 829–836 (2005)
10. Field, D.: What is the goal of sensory coding? Neural Comput. **6**, 559–601 (1994)
11. Chen, M.-J., Bovik, A.C., No-reference image blur assessment using multi-scale gradient. Int. Workshop Qual. Multimed. Experience, 70–74 (2009)
12. Renting, L., Zhaorong, L., Jiaya, J., Image partial blur detection and classification. IEEE CVPR, 1–8 (2008)
13. Sheikh, H.R., Wang, Z., Cormack, L.K., Bovik, A.C.: LIVE image quality assessment database," Release 2, [online]: Available at: http://live.ece.utexas.edu/research/quality/subjective.htm
14. Seshadrinathan, K., Soundararajan, R., Bovik, A. C., Cormack, L. K.: Study of subjective and objective quality assessment of video. IEEE Trans. Image Process **19**(6):1427–1441 (2010)
15. Seshadrinathan, K., Soundararajan, R., Bovik, A.C., Cormack, L.K.: A subjective study to evaluate video quality assessment algorithms. Proc. SPIE vol. 7527, 75270H (2010)
16. LIVE Video Quality Database, available at http://live.ece.utexas.edu/research/quality/live_video.html
17. VQEG, Final report from the video quality experts group on the validation of objective quality metrics for video quality assessment
18. Sheikh, H.R., Sabir, M., Bovik, A.C.: A statistical evaluation of recent full reference image quality assessment algorithms. IEEE Trans. Image Process. **15**, 3440–3451 (2006)

Author Biographies

Ming-Jun Chen earned a B.S. and an M.S. in Computer Science from the National Cheng-Kung University, Taiwan, in 2001 and 2003, respectively. After graduation, he joined CyberLink as a software engineer specializing in audio and video processing. He left CyberLink in June 2008 when he was a senior engineer. Then he came to the University of Texas at Austin and joined LIVE lab in fall, 2008. Currently he is pursuing a doctorate degree under the supervision of Dr. Alan C. Bovik. His research interests include video/image quality assessment, image/video/audio compression, and video/audio content analysis.

Alan C. Bovik is the Curry/Cullen Trust Endowed Chair Professor at The University of Texas at Austin. He is well known for his fundamental work in non-linear image processing, perceptual image and video processing and computational modeling of visual perception. Al is particularly noted for his pioneering work on robust image processing, multi-resolution image processing, image and video quality assessment, and computational modeling of visual perception. He is also noted for innovations in engineering education, including his popular books and the widely used Signal, Image and Video Audiovisual (SIVA) Demonstration Gallery. Al has also been a major contributor to engineering service, including innovating and creating the IEEE International Conference on Image Processing, first held in Austin, Texas, in November, 1994, and playing an instrumental role in proposing and creating the IEEE Transactions on Image Processing, for which he served as Editor-in-Chief for 6 years. Al has received a number of major awards from the IEEE Signal Processing Society, including: the Education Award (2008); the Technical Achievement Award (2005), the Distinguished Lecturer Award (2000); and the Meritorious Service Award (1998). He is a Fellow of the IEEE, a Fellow of the Optical Society of America (OSA), and a Fellow of the Society of Photo-Optical and Instrumentation Engineers (SPIE).