

Image Quality Assessment (IQA)是一个快速，精确，可靠 的测量视频/图像质量的基于C的库。

它实现了很多流行的算法比如 MS-SSIM, SIMM, MSE 和 PSNR。

其提供的方法在iqa.h中，如下所示：

[cpp]  

```
1.  /*
2.  * Copyright (c) 2011, Tom Distler (http://tdistler.com)
3.  * All rights reserved.
4.  *
5.  * The BSD License
6.  *
7.  * Redistribution and use in source and binary forms, with or without
8.  * modification, are permitted provided that the following conditions are met:
9.  *
10. * - Redistributions of source code must retain the above copyright notice,
11. *   this list of conditions and the following disclaimer.
12. *
13. * - Redistributions in binary form must reproduce the above copyright notice,
14. *   this list of conditions and the following disclaimer in the documentation
15. *   and/or other materials provided with the distribution.
16. *
17. * - Neither the name of the tdistler.com nor the names of its contributors may
18. *   be used to endorse or promote products derived from this software without
19. *   specific prior written permission.
20. *
21. * THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS"
22. * AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
23. * IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
24. * ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE
25. * LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR
26. * CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF
27. * SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS
28. * INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN
29. * CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE)
30. * ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE
31. * POSSIBILITY OF SUCH DAMAGE.
32. */
33.
34. #ifndef IQA_H
35. #define IQA_H
36.
37. #include "iqa_os.h"
38.
39. /**
40.  * Allows fine-grain control of the SSIM algorithm.
41.  */
42. struct iqa_ssim_args {
43.     float alpha;    /**< luminance exponent */
44.     float beta;     /**< contrast exponent */
45.     float gamma;    /**< structure exponent */
46.     int L;          /**< dynamic range (2^8 - 1)*/
47.     float K1;       /**< stabilization constant 1 */
48.     float K2;       /**< stabilization constant 2 */
49.     int f;          /**< scale factor. 0=default scaling, 1=no scaling */
50. };
51.
52. /**
53.  * Allows fine-grain control of the MS-SSIM algorithm.
54.  */
55. struct iqa_ms_ssim_args {
56.     int wang;       /**< 1=original algorithm by Wang, et al. 0=MS-SSIM* by Rouse/Hemami (default). */
57.     int gaussian;   /**< 1=11x11 Gaussian window (default). 0=8x8 linear window. */
58.     int scales;     /**< Number of scaled images to use. Default is 5. */
59.     const float *alphas; /**< Pointer to array of alpha values for each scale. Required if 'scales' isn't 5. */
60.     const float *betas; /**< Pointer to array of beta values for each scale. Required if 'scales' isn't 5. */
61.     const float *gammas; /**< Pointer to array of gamma values for each scale. Required if 'scales' isn't 5. */
62. };
63.
64. /**
65.  * Calculates the Mean Squared Error between 2 equal-sized 8-bit images.
66.  * @note The images must have the same width, height, and stride.
67.  * @param ref Original reference image
68.  * @param cmp Distorted image
69.  * @param w Width of the images
70.  * @param h Height of the images
71.  * @param stride The length (in bytes) of each horizontal line in the image.
72.  * This may be different from the image width.
73.  * @return The MSE.
74.  */
75. float iqa_mse(const unsigned char *ref, const unsigned char *cmp, int w, int h, int stride);
```

```

76.
77. /**
78.  * Calculates the Peak Signal-to-Noise-Ratio between 2 equal-sized 8-bit
79.  * images.
80.  * @note The images must have the same width, height, and stride.
81.  * @param ref Original reference image
82.  * @param cmp Distorted image
83.  * @param w Width of the images
84.  * @param h Height of the images
85.  * @param stride The length (in bytes) of each horizontal line in the image.
86.  *           This may be different from the image width.
87.  * @return The PSNR.
88.  */
89. float iqa_psnr(const unsigned char *ref, const unsigned char *cmp, int w, int h, int stride);
90.
91. /**
92.  * Calculates the Structural SIMilarity between 2 equal-sized 8-bit images.
93.  *
94.  * See https://ece.uwaterloo.ca/~z70wang/publications/ssim.html
95.  * @note The images must have the same width, height, and stride.
96.  * @param ref Original reference image
97.  * @param cmp Distorted image
98.  * @param w Width of the images
99.  * @param h Height of the images
100. * @param stride The length (in bytes) of each horizontal line in the image.
101. *           This may be different from the image width.
102. * @param gaussian 0 = 8x8 square window, 1 = 11x11 circular-symmetric Gaussian
103. * weighting.
104. * @param args Optional SSIM arguments for fine control of the algorithm. 0 for
105. * defaults. Defaults are a=b=g=1.0, L=255, K1=0.01, K2=0.03
106. * @return The mean SSIM over the entire image (MSSIM), or INFINITY if error.
107. */
108. float iqa_ssim(const unsigned char *ref, const unsigned char *cmp, int w, int h, int stride,
109.               int gaussian, const struct iqa_ssim_args *args);
110.
111. /**
112.  * Calculates the Multi-Scale Structural SIMilarity between 2 equal-sized 8-bit
113.  * images. The default algorithm is MS-SSIM* proposed by Rouse/Hemami 2008.
114.  *
115.  * See https://ece.uwaterloo.ca/~z70wang/publications/msssim.pdf and
116.  * http://foulard.ece.cornell.edu/publications/dmr_hvei2008_paper.pdf
117.  *
118.  * @note 1. The images must have the same width, height, and stride.
119.  * @note 2. The minimum image width or height is 2^(scales-1) * filter, where 'filter' is 11
120.  * if a Gaussian window is being used, or 9 otherwise.
121.  * @param ref Original reference image
122.  * @param cmp Distorted image
123.  * @param w Width of the images.
124.  * @param h Height of the images.
125.  * @param stride The length (in bytes) of each horizontal line in the image.
126.  *           This may be different from the image width.
127.  * @param args Optional MS-SSIM arguments for fine control of the algorithm. 0
128.  * for defaults. Defaults are wang=0, scales=5, gaussian=1.
129.  * @return The mean MS-SSIM over the entire image, or INFINITY if error.
130.  */
131. float iqa_ms_ssim(const unsigned char *ref, const unsigned char *cmp, int w, int h, int stride,
132.                  const struct iqa_ms_ssim_args *args);
133.
134. #endif /* _IQA_H */

```

源代码下载：<http://download.csdn.net/detail/leixiaohua1020/6376741>

SourceForge项目页面：<http://sourceforge.net/projects/iqa/>

项目官方页面：<http://tdistler.com/iqa/>

版权声明：本文为博主原创文章，未经博主允许不得转载。 <https://blog.csdn.net/leixiaohua1020/article/details/12564709>

文章标签：[IQA](#) [开源](#) [PSNR](#) [SSIM](#) [图像质量](#)

个人分类：[视频质量评价](#)

所属专栏：[视频质量评价](#)

此PDF由spygg生成,请尊重原作者版权!!!

我的邮箱:liushidc@163.com