

## 计算全息三维显示的数据压缩编码技术

顾华荣

(清华大学 精密仪器系 精密测试技术及仪器国家重点实验室, 北京 100084)

**摘要:** 全息三维显示能够重建真实场景的光场, 提供全部的深度感, 成为真三维显示的最佳解决方案之一。计算全息三维显示只需要知道物光波的数学描述, 可以灵活地控制波前, 显示出虚拟的三维物体。但是, 巨大的数据量和计算量阻碍了计算全息三维显示的实用化。文中提出了计算全息三维显示中多个环节的数据压缩编码技术, 包括三维物体的稀疏采样、全息三维视频压缩编码的优选和参数优化、全息图分形压缩算法, 有效地压缩了数据量, 并利用 GPU 的并行运算能力实现了全息图的快速计算。

**关键词:** 三维显示; 计算全息; 压缩编码

**中图分类号:** O438    **文献标志码:** A    **DOI:** 10.3788/IRLA201847.0603006

## Data compression coding technologies for computer-generated holographic three-dimensional display

Gu Huarong

(State Key Laboratory of Precision Measurement Technology and Instruments, Department of Precision Instrument, Tsinghua University, Beijing 100084, China)

**Abstract:** Holographic three-dimensional (3D) display can reconstruct the light field of a real scene, providing all depth cues and becoming one of the best solutions for true 3D display. Computer-generated holographic 3D display only needs to know the mathematical description of the object light wave, can flexibly control the wavefront, and display a virtual 3D object. However, the huge amount of data and the amount of calculations hinder the practical application of computer-generated holographic 3D display. This paper presented data compression coding techniques for multiple stages in computer-generated holographic 3D display, including sparse sampling of 3D objects, optimization and parameter optimization of holographic 3D video compression coding, holographic fractal compression algorithm, effectively reducing the amount of data. And the parallel computing capabilities of GPUs were utilized to enable fast calculation of holograms.

**Key words:** 3D display; computer-generated holography; compression coding

收稿日期: 2018-05-15; 修订日期: 2018-06-05

基金项目: 国家自然科学基金(61205013)

作者简介: 顾华荣(1981-), 男, 助理研究员, 博士, 主要从事三维显示方面的研究。Email: ghr@mail.tsinghua.edu.cn

## 0 引言

三维显示技术可以再现自然界的三维特性,在医学成像、计算机辅助设计、虚拟现实与增强现实等领域有着广泛的应用,近年来获得了飞速的发展<sup>[1]</sup>。目前成熟的三维显示技术大多基于双目视差原理,存在观看视疲劳<sup>[2]</sup>等问题。全息三维显示能够重建真实场景的光场,提供全部的深度感,解决双目视差的本质缺陷,成为真三维显示的最佳解决方案之一<sup>[3]</sup>。计算全息三维显示采用计算机编码生成全息图,不要求物体的实际存在,只需要知道物光波的数学描述,可以灵活地控制波前,显示出虚拟的三维物体。

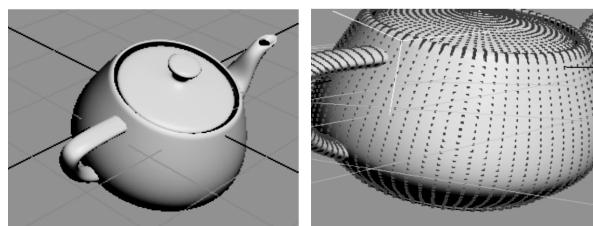
针对高清实时三维显示的需求,当前计算全息三维显示技术的发展面临两大挑战<sup>[4]</sup>:一是如何生成巨大像素数目的全息图,以显示大尺寸、高分辨率、宽视角的高清三维物体;二是如何提高生成计算全息图的速度,以实现虚拟三维物体的实时刷新。在文中,高清三维物体的定义为:在深度方向具有 100~1 000 层,每层的分辨率与现有高清视频的分辨率相同,即 1 920×1 080 或 1 280×720;实时刷新的定义为:每秒更新整个显示区域 25~60 次;暂不考虑彩色问题,即所显示的三维物体是单色的。以 1 920×1 080×1 000 的三维物体和 25 帧/s 的刷新率为例,计算全息图所需像素数目大约为  $2 \times 10^9$ ,所需计算速率大约为  $1 \times 10^{21}$  FLOPS。现有的硬件水平难以达到如此高的要求,因此已实现的计算全息三维显示的物体分辨率都不高,例如韩国光云大学利用 3 块 NVIDIA GTX TITAN 显卡生成的三维视频场景中仅包含 12 666 个点<sup>[5]</sup>。

高清实时计算全息三维显示的实现,一方面依赖于硬件水平的不断提高,包括空间光调制器的分辨率、刷新率、对比度、填充因子的提高和像素尺寸的减小,CPU/GPU 的浮点运算性能的提高,以及数据存储容量和传输速率的提高;另一方面依赖于算法和编码技术的进步,包括设计更快速的计算全息图生成算法,以及开发更先进的数据压缩编码技术。数据压缩编码的目的是利用现有的硬件,在保持三维物体再现质量基本不变的前提下,尽可能地压缩计算数据量和生成数据量。由于全息图通常具有较大的冗余度<sup>[6]</sup>,全息信息的压缩对于提高计算速度和减小数据带宽显得十分重要。

文中针对高清实时计算全息三维显示中数据量大的特点,结合人眼视觉分辨率的限制,使用稀疏采样减少了描述三维物体所需的数据量并提高了全息图的计算速度;将常用视频压缩编码引入计算全息三维视频,分析了编码方式和编码参数对再现质量的影响;提出使用分形压缩算法作为计算全息视频的帧内编码,初步验证了分形压缩算法在高压比下的优越性;基于 CUDA 平台,利用 GPU 的并行运算能力实现了全息图的快速计算。

## 1 稀疏采样

为了计算三维物体的全息图,首先要对物体进行数值描述。可以利用 3DS MAX 软件建立三维模型,然后采用 MAXScript 获取表面采样点的坐标信息。如图 1(a)所示是在 3DS MAX 中建立的茶壶三维模型。对其表面进行采样,采用非均匀采样:曲率小的部分采样点稀疏,曲率大的部分采样点密集一些,细节多的地方多采样,细节少的地方少采样,使得采样得到的坐标信息更加准确,如图 1(b)所示。



(a) 三维模型

(b) 表面采样

(a) 3D model

(b) Sampling on surface

图 1 3DS MAX 中的茶壶

Fig.1 Teapot in 3DS MAX

利用 MAXScript 可以导出茶壶的三维坐标信息。茶壶图的具体信息如下:采样点数 3 554,茶壶实体大小 35 mm×18 mm×25 mm,假定光波长为 532 nm,全息图与三维物体的距离为 1 200 mm。使用点元法<sup>[7-8]</sup>计算得出全息图,如图 2 所示,分辨率为 1 920×1 920。计算硬件为两颗 Intel Xeon X5670 CPU,计算软件为 MATLAB,计算时间为 470 s。使用菲涅尔逆衍射公式进行数字重建,得到茶壶的数字重建图,如图 3(a)所示。利用 HOLOEYE 公司的 PLUTO 纯相位型空间光调制器加载所计算的全息图,光学重建结果如图 3(b)所示。

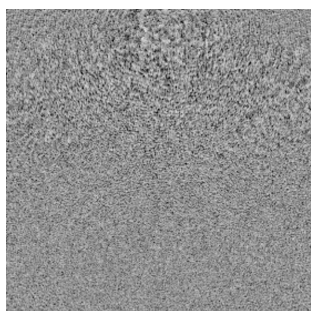
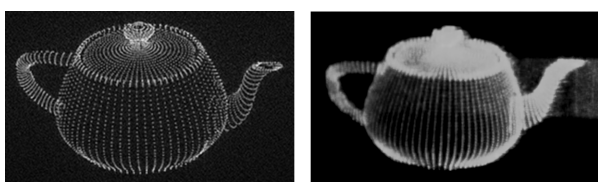


图 2 茶壶的全息图

Fig.2 Hologram of the teapot



(a) 数字重建

(b) 光学重建

(a) Numerical reconstruction

(b) Optical reconstruction

图 3 茶壶的重建图

Fig.3 Reconstruction of the teapot

人眼视觉系统本质上是一个光学系统，存在分辨率限制。根据瑞利判据，当两个物点刚能分辨时，其对入瞳中心的张角为最小分辨角：

$$\theta = 1.22 \frac{\lambda}{D} \quad (1)$$

式中： $\lambda$  为照明光波长； $D$  为入瞳直径。若人眼瞳孔直径为 3 mm，则人眼的角分辨率约为  $2 \times 10^{-4}$  rad。三维物体的观察距离为 1 200 mm，如果两个采样点的距离小于 0.26 mm，那么两个点将不能分辨。对于上述茶壶图来说，采用常规的三维物体数值描述方法存在较多冗余。可以将原先由于人眼有限的分辨能力而观察不到的点删去，以达到数据压缩的目的。经过处理之后得到的采样点数为 2 365。在分辨率不变的情况下，全息图计算时间减少到 350 s。全息图的数字重建结果如图 4(a)所示，采用空间光调制器进行光学重建的结果如图 4(b)所示。可见，不管是数

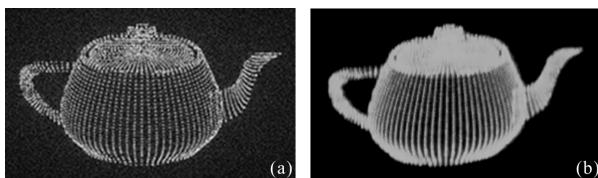


图 4 稀疏采样处理后的茶壶重建图

Fig.4 Reconstruction of the teapot after sparse sampling

字重建还是光学重建，与处理之前的数值描述方法进行比较，虽然采样点数减少了 1/3，但在人眼直接观察时，再现图质量并没有显著的下降，相应的全息图计算时间也减少了 1/3。

## 2 视频压缩编码

连续多帧全息图组成了全息三维视频，可以采用现有的视频编码技术来进行数据压缩。全息三维视频的生成与压缩流程包括：三维模型的构建与数据采样、全息三维视频的计算、全息三维视频的压缩编码、压缩编码的参数调节、全息三维视频的重建和结果比较等，如图 5 所示。

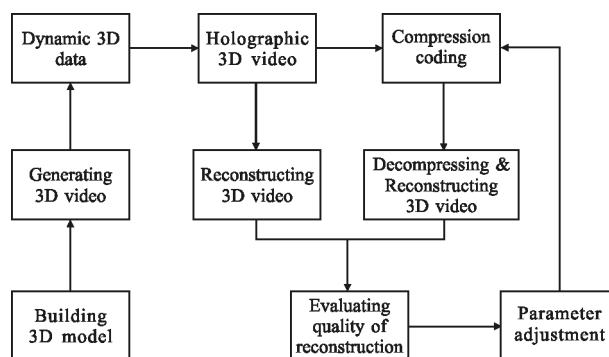


图 5 全息三维视频的生成与压缩流程

Fig.5 Generation and compression process of holographic 3D video

利用 MAXScript 编程生成茶壶动画，茶壶半径为 9，初始位置  $[-117, -18, 0]$ ，标准形状（包含把手、壶嘴，并进行光滑处理）。茶壶运动轨迹是  $x$  轴位移 228，并同时沿  $y$  轴旋转，长度为 100 帧。截取画面中的 25 帧（第 41~65 帧）进行计算，图 6 为茶壶动画的第 41、45、50、55、60、65 帧的截图。

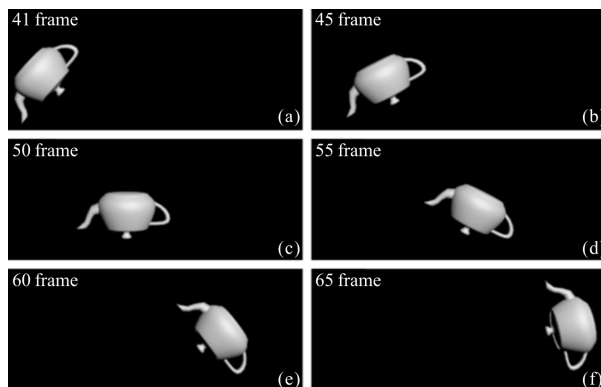


图 6 茶壶动画的第 41、45、50、55、60、65 帧截图

Fig.6 Frames 41, 45, 50, 55, 60, 65 of the teapot video

利用 MAXScript 获取每帧场景采样点的三维坐标,根据点元法计算每帧场景的全息图,将多帧全息图结合在一起成为全息三维视频。设定帧率为5 帧/s,输出未压缩的灰度 AVI 格式视频。图 7 展示了其中第 1、5、10、15、20、25 帧全息图,对应于原始茶壶动画的第 41、45、50、55、60、65 帧。

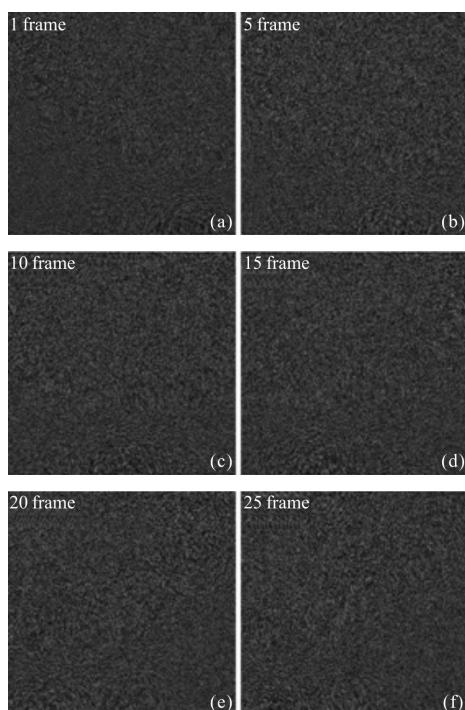


图 7 茶壶的全息三维视频第 1、5、10、15、20、25 帧全息图  
Fig.7 Holograms of frame 1, 5, 10, 15, 20, 25 of the holographic 3D video of the teapot

文中采用现有的 H.264<sup>[9]</sup>、H.265<sup>[10]</sup>和 VP9<sup>[11]</sup>这三种视频编码标准分别对上述生成的全息三维视频进行压缩编码,在不同的码率下比较压缩全息视频重建的茶壶动画图像质量。评价指标选用重建茶壶动画所有帧的平均峰值信噪比(PSNR)和平均结构相似度(SSIM)<sup>[12]</sup>,结果如图 8 和图 9 所示。

从图中可以看出:(1) H.264 和 H.265 压缩全息视频的重建质量在高码率条件下明显好于 VP9,但是在低码率(小于 20 Mbps)时重建质量则比 VP9 差。(2) H.264 和 H.265 的重建质量随码率的降低而衰减的速度比 VP9 快,衰减的主要区域在 60~20 Mbps 之间。而 VP9 在 80 Mbps 以上有一个重建质量的稳定区域。(3) H.264 和 H.265 两者相比,H.264 在几乎全码率范围内都优于 H.265。

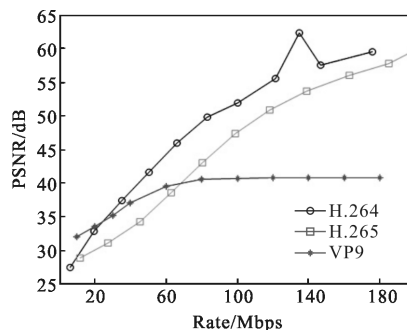


图 8 压缩全息视频重建茶壶动画的帧平均 PSNR  
Fig.8 Average PSNR of the reconstructed teapot video from compressed holographic video

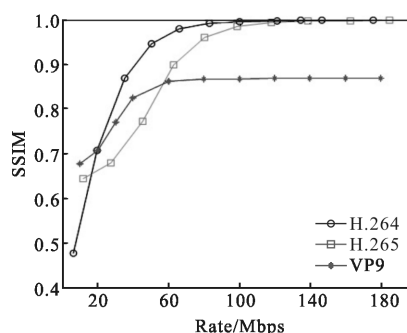


图 9 压缩全息视频重建茶壶动画的帧平均 SSIM  
Fig.9 Average SSIM of the reconstructed teapot video from compressed holographic video

### 3 全息图分形压缩

分形几何图形具有整体与部分以某种方式相似的特征。一幅具有分形结构特征的图像,可以看作是某个平面迭代函数系统 (Iterated Function System, IFS) 的吸引子。分形图像压缩的原理就是通过寻找 IFS 的各项系数实现数据压缩。对于全息图,虽然图像局部和整体之间没有明显的相似性,但是图像局部和局部之间具有相似性,因此可以定义一系列的仿射变换(即平面线性变换及平移)作为 IFS,其定义域块是带编码全息图区域分块的子集,其值域块的总和即为整幅图像。全息图的编码过程如下:

(1) 编码图像分块。首先将待编码的全息图(分辨率 1 920×1 920)划分成尺寸为  $K \times K$  的互不重叠的方块,作为 IFS 的值域块。然后,再从图像中取  $L \times L$  的方块,作为 IFS 的定义域块。为了满足压缩映射条件,定义域块比值域块要大,而且可以重合,一般取  $L=2K$ ,定义域块平移的步长也为  $K$ 。分块示意图如

图 10 所示。

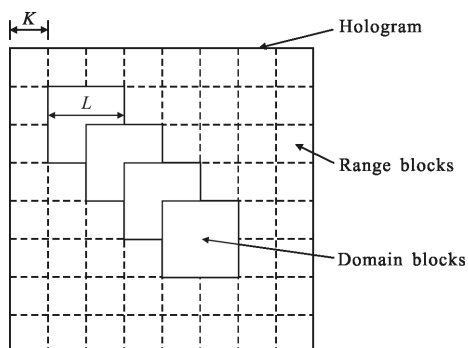


图 10 全息图分形压缩算法分块示意图

Fig.10 Block division diagram for fractal compression of holograms

(2) 寻找最佳压缩变换参数。根据全息图局部的相似性，为每个值域块都在所有定义域块中寻找匹配的定义域块和仿射变换，定义域块的序号和仿射变换系数即为变换参数。

(3) 存储压缩变换参数。记录最佳匹配时值域块对应的定义域块序号和仿射变换系数，最后将所有值域块的变换参数进行量化和编码，就完成了对整幅全息图像的压缩编码。

解码过程相对于编码过程要简单许多，根据 IFS 的不动点定理，只要对任意的初始图像进行 IFS 迭代，最终都可以 IFS 的吸引子即原始图像。这种迭代只需 10 次以上即会收敛。

对如图 11 所示的飞机全息图进行分形图像压缩，原始全息图分辨率  $1920 \times 1920$ ，文件大小为 3.51 MB，压缩后数据量仅为 112 kB，这意味着单帧的文件压缩比就超过 30。对压缩后的全息图进行三维重建，发现明显的中心亮方块，如图 12(a)所示。产生亮方块的原因是高压压缩比下的全息图模糊。可以对压缩后的全息图的灰度进行 Gamma 矫正，提高全息图的对比度，从而降低亮方块的影响，如图 12(b)所示。

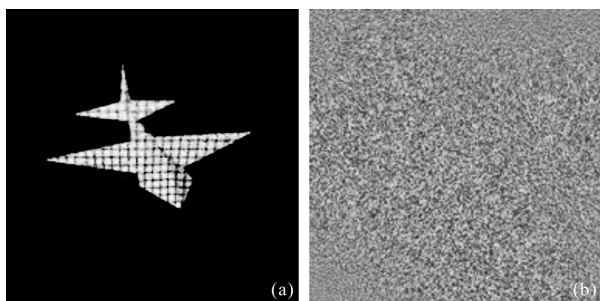
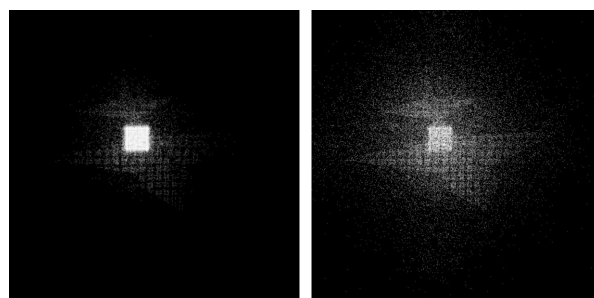


图 11 (a)飞机的三维模型与(b)其全息图

Fig.11 (a) 3D model of a plane and (b) its hologram



(a) 全息图灰度 Gamma 矫正前的重建图像 (b) 全息图灰度 Gamma 矫正后的重建图像

(a) Reconstructed image from hologram before Gamma correction (b) Reconstructed image from hologram after Gamma correction

图 12 飞机全息图分形压缩后的重建图像

Fig.12 Reconstructed images from the fractal compressed hologram of the plane

比较分形压缩全息图 Gamma 矫正后的三维重建与原始全息图的三维重建，计算得到其峰值信噪比为 37.2 dB，结构相似度为 0.76。将上述两个指标与最高压缩比下的 H.264、H.265 和 VP9 压缩全息三维视频重建图像的 PSNR 和 SSIM 进行比较（如表 1 所示），发现分形压缩算法作为计算全息三维视频的帧内编码的表现优于现有的算法。

表 1 高压压缩比下不同编码方法的重建质量

Tab.1 Reconstruction quality of different coding methods with high compression ratio

Coding	Compression ratio	PSNR/dB	SSIM
H.264	22.72	27.5	0.48
H.265	38.46	28.1	0.65
VP9	29.81	32.2	0.68
Fractal compression	32.14	37.2	0.76

## 4 GPU 加速

前述全息图计算时间在百秒量级左右，远远达不到实时显示的需求。GPU 因其有着数量众多的计算核心而在并行计算上有着特有的优势，并具有很强的浮点计算能力，基于 GPU 的通用并行计算技术正受到广泛的关注。CUDA 采用双层并行结构、实现 block 间的数据通信，通过对 C 语言进行扩展使编程难度减低，并让用户能够根据使用需要对 GPU 上

的计算单元进行划分,硬软件相结合,大大提高了并行计算的效率。这为实现快速全息图计算打下了基础。

文中将相位混合算法从 MATLAB 平台移植到 CUDA 平台上。相位恢复算法在 CUDA 上被分为两个部分:一部分是在 CPU 上执行的串行计算;另一部分是在 GPU 上执行的并行计算。CPU 上执行的部分主要是图像处理、数据初始化和设备初始化。GPU 上执行的部分是 GS 算法的迭代部分。将如图 13(a)所示的图像通过 111 次迭代计算生成如图 13(b)所示的全息图,采用 CUDA 平台需要 1 876 ms,而采用 MATLAB 平台需要 17 424 ms,即在当前平台上,使用 CUDA 计算全息图的速度约为使用 MATLAB 计算全息图速度的 9 倍。



图 13 (a)熊猫的灰度图与(b)其全息图

Fig.13 (a) Grayscale image of a panda and (b) its hologram

## 5 结 论

文中提出了几种计算全息三维显示中的数据压缩编码技术:在数值采样环节,利用人眼有限的分辨率,将描述三维物体所需的数据量减少了 1/3,相应的全息图计算时间也减少了 1/3;比较了几种现有的视频压缩编码在压缩全息三维视频时的性能,发现在较高码率时,H.264 的重建图像质量较优,但在较低码率时,VP9 的重建图像质量较优;提出使用分形压缩算法作为计算全息视频的帧内编码,在高压比下,分形压缩全息图的重建图像质量优于 H.264、H.265 和 VP9;利用 GPU 的并行运算将全息图的计算速度提升至利用 CPU 的 9 倍。文中提出的数据压缩编码技术为高清实时计算全息三维显示的实用化

指出了可行的方向。

## 参 考 文 献:

- [1] Geng J. Three-dimensional display technologies[J]. *Advances in Optics and Photonics*, 2013, 5(4): 456–535.
- [2] Huang Yingqing, Su Jian, Chen Yibei, et al. Progress in holographic printing technique [J]. *Infrared and Laser Engineering*, 2018, 47(4): 0406008. (in Chinese)
- [3] Zhou Hao, Gu Jihua, Chen Daqing. Multi-plane imaging in digital holography [J]. *Infrared and Laser Engineering*, 2015, 44(2): 513–518. (in Chinese)
- [4] Xu X W, Solanki S, Liang X N, et al. Full high-definition digital 3D holographic display and its enabling technologies [C]//SPIE, 2010, 7730: 77301C.
- [5] Kwon M W, Kim S C, Yoon S E, et al. Object tracking mask-based NLUT on GPUs for real-time generation of holographic videos of three-dimensional scenes [J]. *Opt Express*, 2015, 23(3): 2101–2120.
- [6] Cai X O, Wang H. On holographic information content and its compression[J]. *Optik*, 2006, 117(3): 131–137.
- [7] Lucente M. Interactive computation of holograms using a look-up table [J]. *Journal of Electronic Imaging*, 1993, 2(1): 28–34.
- [8] Kim S C, Kim E S. Effective generation of digital holograms of three-dimensional objects using a novel look-up table method[J]. *Appl Opt*, 2008, 47(19): D55–D62.
- [9] Wikipedia. H.264/MPEG–4 AVC [EB/OL]. [2018–01–05]. [https://en.wikipedia.org/wiki/H.264/MPEG-4\\_AVC](https://en.wikipedia.org/wiki/H.264/MPEG-4_AVC), 2018.
- [10] Wikipedia. High Efficiency Video Coding [EB/OL]. [2018–03–28]. [https://en.wikipedia.org/wiki/High\\_Efficiency\\_Video\\_Coding](https://en.wikipedia.org/wiki/High_Efficiency_Video_Coding), 2018.
- [11] Wikipedia. VP9 [EB/OL]. [2018–01–01]. <https://en.wikipedia.org/wiki/VP9>, 2018.
- [12] Wang Z, Bovik A C, Sheikh H R, et al. Image quality assessment: from error visibility to structural similarity [J]. *IEEE Transactions on Image Processing*, 2004, 13(4): 600–612.