
目录

摘要.....	1
9.1 性能分析.....	1
9.2 译码设置.....	1
9.2.1 译码器软件.....	1
9.2.2 测试条件.....	2
9.2.3 预测结构.....	2
9.2.3.1 全帧内.....	2
9.2.3.2 随机存贮	2
9.2.3.3 低延迟 P 图 (LDP)	3
9.2.3.4 低延迟 B 图(LDB).....	4
9.2.4 测试序列.....	4
9.2.5 测试用例和位深度.....	4
9.2.6 失真率曲线	6
9.2.7 R-D 最优化.....	7
9.3 客观性能分析	12
9.4 主观性能分析	16
9.4.1 测试方法学	16
9.4.1.1 DSIS(双边刺激受损程度).....	17
9.4.1.2 DSCQS(双边刺激连续质量等级)	17
9.4.1.3 训练项目	17
9.4.1.4 场景浏览	18
9.4.2 主观质量评估测试.....	18

9.4.2.1 测试环境	18
9.4.2.2 测试方法学	18
9.4.2.3 数据集.....	19
9.4.2.4 训练项目	19
9.4.2.5 测试课程	19
9.4.2.6 结果分析	20
9.4.2.7 曲线图结果失真率	21
9.4.2.8 平均比特率的差异	21
9.5 编码器产品质量性能分析	23
9.5.1 测试条件.....	23
9.5.1.1 编码器设定	23
9.5.1.2 主观评估条件.....	24
9.5.1.3 测试序列	24
9.5.2 主观质量评估结果.....	24
9.5.3 结果	25
9.6 结论.....	25
鸣谢:	26
参考文件:	26

摘要

在本章中，给出了 HEVC（推荐 ITU-T H.265|ISO/IEC 23008-2）AVC（推荐 IUT-T H.264|ISO/IEC 14996-10）的性能分析在主观和客观两方面比较的质量评估。由于 HEVC 的复杂性增加，从失真率上来讲，讲述了如何去选择最优编码参数。注意到，在 HEVC 和 AVC 进行主观和客观的质量评估实验的时候，我们采用了统一的方法。我们所有的研究评估结果表明 HEVC 的编码效率大约是 AVC 的两倍还要高。

9.1 性能分析

HEVC 的性能分析通常是一项复杂的工程，因为它可以基于很多种不同的方式进行，例如，压缩效率、复杂度、视觉效果、失真率最优化应用、延迟、鲁棒性等等。本章的所有内容意在表达把复杂度、速率失真最优化应用和延迟考虑在内，从主观和客观质量评估去比较 HEVC 和 AVC 的压缩效率。注意的是，质量方法都很重要；单单依赖客观的质量评估，在一些情况下，可能会低估比特率从而影响我们的压缩效率分析。尽管主观质量评估很难直接进行，但从另一方面来说观察者可以直接用视觉经验把它关联起来。该章节的组织如下。9.1 节提供背景信息和该节要遵守的基本结构。在 9.2 节中，根据复杂度和延迟要求，考虑不同编码器设置来描述编码器的设置和测试条件。而且，该节全部内容在讲，使用到的测试队列列表，测试案例和描述的非标准 R-D 最优化工具，这些明显使编码效率大大提高。9.3 节中，研究了 HEVC 和 AVC 参考实现的客观质量评估。9.4 节，我们展示了 HEVC 客观质量测试和 HEVC、AVC 视觉评定的结果。9.5 节，在 4K 视频流背景下的应用下，讲述了一个非正式主观视频质量和比较 HEVC 和 AVC 编码器的产品质量。结论在 9.6 节中。

9.2 译码设置

为了研究 HEVC 和 AVC 的性能评估，需要建立一个定义好的编码器设置和测试环境。在本章节中，我们将描述，我们研究中用到的 HEVC 和 AVC 的参考编码软件（SW）。除此之外，我还将描述各种不同的编码器设置和预测结构，这些在编码效率、复杂度、延迟方面将适用于不同的应用需求。

9.2.1 译码器软件

在标准 HEVC 中，参考软件叫做 HM（HM 测试模型，参考软件）[15]，它已经发展成为一种作为更深入改进和研究的通用 SW 软件平台。使用 SVN 服务器，参考软件主要维护于两站点[16]。HHI（Heinrich Hertz Institute）维护主 SVN 服务器和 BBC（British Broadcasting Corporation）维护网站镜像存储库。AVC 的参

考软件叫做 JM，它已经发展成为 AVC 性能评估的通过测试平台。参考软件 JM 主要维护 SVN 服务器[6]。在本章中，为了比较 HEVC 和 AVC 的编码性能，HM12.1 和 JM18.5 两款软件各自用于 HEVC 和 AVC 译码器。

9.2.2 测试条件

在 HEVC 规范的开发过程中，通用测试条件的建立提供了定义好的平台，在该平台上编码评估工具设备得以执行。根据 CTC，要使 HEVC 的性能评估得以执行，需要遵守详细的 CTC 主键。

9.2.3 预测结构

为了性能评估，CTC 定义了如下的预测结构

1. 全帧内 (AI)
2. 随机存贮 (RA)
3. P 图低延迟 (LDP)
4. B 图低延迟 (LDB)

在这些配置中，QP(量化参数)值可以通过添加”QP 偏移“值来修改。意思就是，CTC 定义第一张图片 (QPI 是 I 张图片的 QP，用 I 图定义下一张) 的 QP 和下一张图的 QP 可以导出为 QP， $QP = (QPI + QPoffset)$ ，根据图片类型或是图片临时 ID 来决定 QP 偏移值。一张 I 图指的是这张图无需请求其他解码图片的预测数据就可以单独被解码。一张 P 图，通常来讲，需要的图片像素数据来源于其他 I、P 或是 B 图，产生每一张被预测的像素块。一张 B 图，通常来讲，需要的图片像素数据来源于其他 I、P 或是 B 图，产生每一张被预测的像素块。

9.2.3.1 全帧内

在这个配置中，每一张图片被编码成一张 I 图。因为没使用图内预测结构，因此，它适用于低延迟和更高比特率的应用。QP 偏移在这个配置中是 0，因为在整个队列中 QP 值保持恒定。图 9.1 展示了一个预测结构的例子。

9.2.3.2 随机存贮

在这个配置中，要使用到一个分层的 B 图结构[21]。图 9.2 显示的是一个预测结构的例子。通过间接的分层预测结构来实现编码效率比其他配置的编

码效率更高。但是，由于图片重排序，使得它的延迟更大。为了尽可能的控制传输的错误和随机存贮的易用性，I 图要周期性的插入。每一张图片的 QP 偏移值汇总再图 9.2 中。

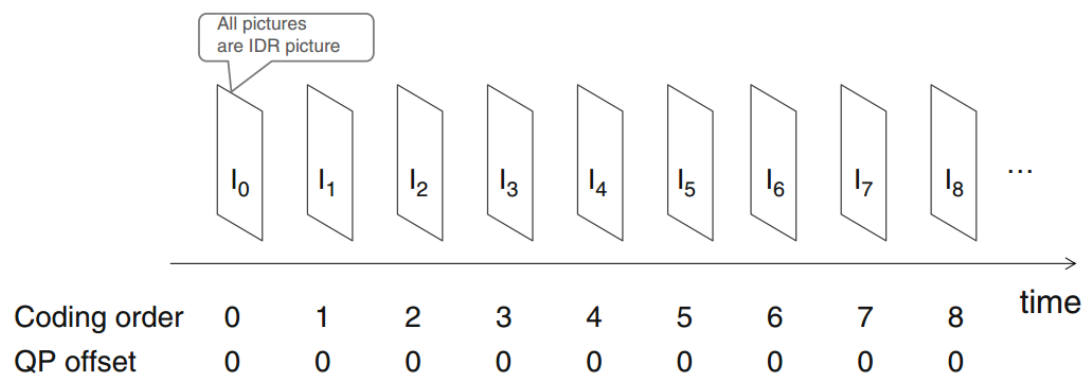


图 9.1 单内部配置的预测结构

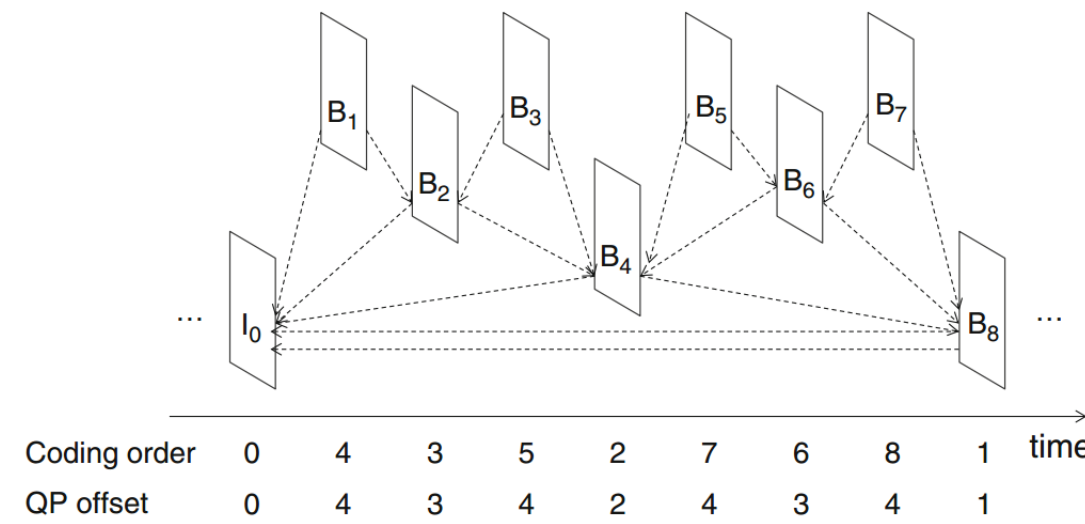


图 9.2 随机访问配置的预测结果

9.2.3.3 低延迟 P 图 (LDP)

在该配置中，第一张图片被编码成 I 图，后面的图片被编码成 P 图。因为不允许重排序图片，仅通过的图被用于预测，在该配置中编码延迟可能会变小。图 9.3 显示了一个预测结构的例子。QP 偏移值汇总在图 9.3 中。

9.2.3.4 低延迟 B 图(LDB)

在这个配置中，类似之前的配置，不允许重排序图片。第一张图被编码成 I 图，随后的图片被编码成 B 图。而且，因为只有通过的 B 图被用作预测，编码延迟降低，与 LDP 类似，但是可以达到更高的编码效率（由于 bi 预测结构）。每一张图片的偏移值汇总在图 9.3 中。

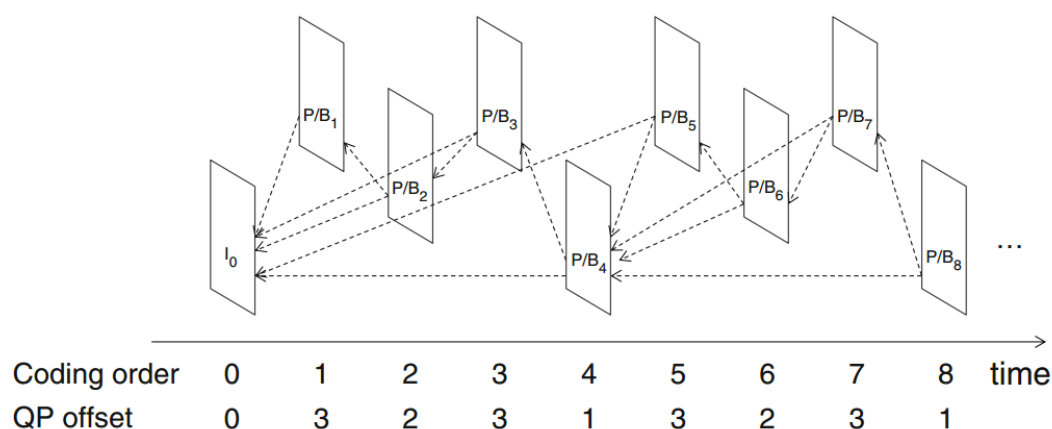


图 9.3 低延迟 P 和 B 帧帧配置的预测结构

9.2.4 测试序列

测试系列是根据图片的大小和应用来定义的，他们被分类成六类（从 A 类到 F 类）。A 类具有比 1080 PHDTV 更高像素的序列集合。这些系列被用来评估 4K/8K 视频编码性能。为了减少计算时间，图片被裁成 2560X1600 像素大小的图片。B 类是用于 1080P HDTV 的编码性能评估，它是包含 HDTV 序列集合，该序列集合是一张 1920X1080 像素大小的图片。C 类和 D 类分别是具有 832x480 和 416x240 像素大小的图片的测试序列的集合。这两类测试序列是用于评估移动端应用的编码性能。E 类具有 280x720 像素大小图片的测试序列集合。它用来评估低延迟应用的编码性能，例如，视频通话。除此之外，CTC 定义的 F 类序列，用于评估非相机获取内容的编码性能，比如视屏屏幕内容，比方说文字和电脑图像内容。测试序列列在表 9.1 中。除了在 CTC 中定义测试序列以外，4K 的测试序列列在表 9.2 中，该表用于本章的客观和主观质量性能分析。

9.2.5 测试用例和位深度

定义 MAIN 和 MAIN 10 两个用例来评估 8 位和 10 位的视屏编码性能。所有的测试用例列在表 9.3 中。

在 MAIN 10 配置中，一个 8 位的视屏首先通用左移 2 位转换成一个 10 位的视频，然后再被编码成 10 位的视频。同样，在 MAIN 的配置中，一个 10 位视

频首先通过右移 2 位转成 8 位的视频，然后再编码成 8 位的视频。“optional（可选项）”这个词在表 9.3 中的意思是使用一个确定的序列类（比如 F 类），或是不需要但推荐确定的预测结构（比如 LDP）。在本张中，MAIN 配置用于评估表 9.3 中的“optional”用列。

表 9.1 测试序列

Class	Sequence name	Frame count	Frame rate (fps)	Bit depth
A	Traffic	150	30	8
A	PeopleOnStreet	150	30	8
A	Nebuta	300	60	10
A	SteamLocomotive	300	60	10
B	Kimono	240	24	8
B	ParkScene	240	24	8
B	Cactus	500	50	8
B	BQTerrace	600	60	8
B	BasketballDrive	500	50	8
C	RaceHorses	300	30	8
C	BQMall	600	60	8
C	PartyScene	500	50	8
C	BasketballDrill	500	50	8
D	RaceHorses	300	30	8
D	BQSquare	600	60	8
D	BlowingBubbles	500	50	8
D	BasketballPass	500	50	8
E	FourPeople	600	60	8
E	Johnny	600	60	8
E	KristenAndSara	600	60	8
F	BaskeballDrillText	500	50	8
F	ChinaSpeed	500	30	8
F	SlideEditing	300	30	8
F	SlideShow	500	20	8

表 9.2 4K 测试序列

Resolution	Sequence name	Frame count	Frame rate (fps)	Bit depth
3,840 × 2,160	Book ^a	500	50	10
3,840 × 2,160	BT709Birthday ^b	500	50	10
3,840 × 2,160	HomelessSleeping ^c	600	60	10
3,840 × 2,160	Manege ^d	600	60	8
4,096 × 2,048	Traffic	300	30	8

表 9.3 通用测试条件下测试案例的汇总

Prediction structure	Class					
	A	B	C	D	E	F
AI	Main/Main10	Main/Main10	Main/Main10	Main/Main10	Main/Main10	Optional
RA	Main/Main10	Main/Main10	Main/Main10	Main/Main10	N/A	Optional
LDB	N/A	Main/Main10	Main/Main10	Main/Main10	Main/Main10	Optional
LDP	N/A	Optional	Optional	Optional	Optional	Optional

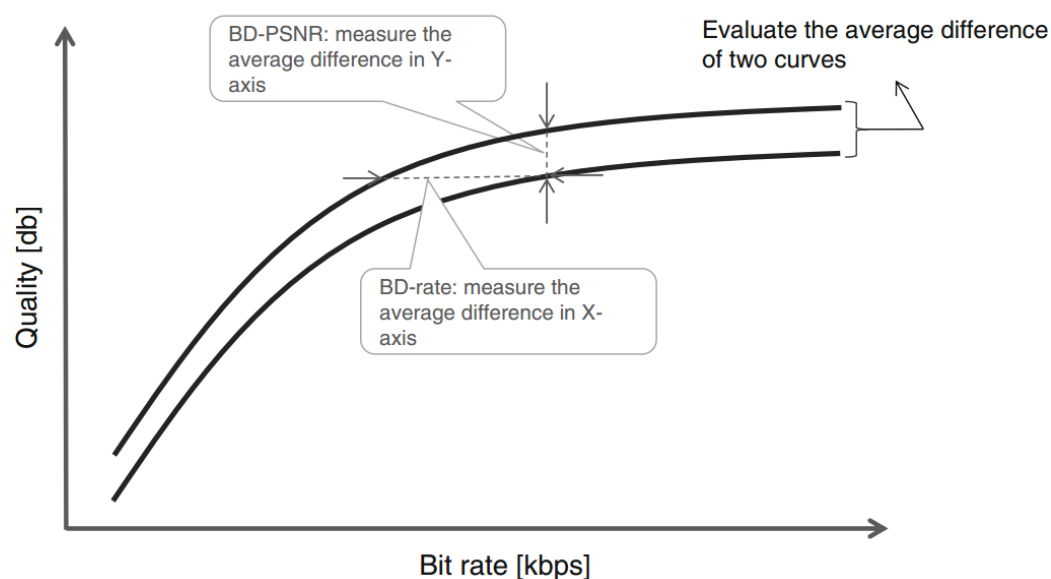


图 9.4 R-D 曲线图的一个样本

9.2.6 失真率曲线

当评估一个视频编解码的编码性能的时，会用到 R-D 曲线图（失真率曲线图）。R-D 曲线图是通过绘制编码的结果生成的，在图中的比特率对应他们的结果质量。水平轴表示比特率，垂直轴表示失真方式或是视频编码质量。通常来讲，较高的压缩率会导致比特率降低；所以，图片质量通常会降低。另一方面，低的压缩比可以提升图片质量，但会以增加比特率作为代价。因此高效率编码的编解码可以降低比特率来达到更高图片质量，R-D 曲线会向左上方移动，如图 9.4 所示。

作为图片质量的客观测量工具 PSNR（峰值信噪比）被广泛的应用。PSNR 可以通过下面的等式来计算。

$$PSNR = 10\log_{10} \frac{(2^{bitdepth} - 1)^2 * W * H}{\sum_i \{O_i - D_i\}^2}$$

bitdepth: 每一个像素的位深

W: 水平像素的数量

H: 垂直像素点的数量

O_i: 参考图片的像素值

D_i: 解码图片的像素值

i: 像素地址

PSNR 工具用于计算每一个 YCbCr 组件. 在 YCbCr 域中, 人类视觉系统相比于色度 (Cb 或是 Cr) 对亮度 (Y) 更加敏感; 因此, 在实际中, 对于客观质量测试方法 PSNR 的亮度是一个更加重要的公制。

为了比较参考编码解码器的编码性能。被评估的图片, 要计算它们两幅 R-D 曲线图的均差。比特率的均差 (在水平方向的差异) 是指 BD 率, PSNR 的均差是指 BD PSNR[1]。

为了计算 BD 率和 BD PSNR, 两幅 R-D 曲线 (对应参考和测试的编解码器) 图近似由下面三次项表达式绘制。

$$PSNR = a + b * (bit\ rate) + c * (bit\ rate)^2 + d * (bit\ rate)^3$$

以上等式中的 a-d 参数可以通过四个数据点 (PSNR 和比特率点) 推导。该多项式逼近式允许我们通过统合两幅曲线图在水平方向上的不同点来推导出 BD 率, 通过统合两幅曲线图在垂直方向上的差异推导出 BD PSNR (请看图 9.4)。

在 HEVC 标准化工作中, BD 率和 BD PSNR 已经被广泛应用于评估编码工具。然而, 我们都知道这样的逼近式有时候可能导致严重的错误, 特别对于大的图片的时候 (比如, A 类序列)。为了更进一步的提升逼近式的精确度, 提出的分段三次插值法可作为一种替代[2]。

9.2.7 R-D 最优化

HEVC 编码工具的灵活性源于这样的事实, 即它包含了很多已经增加的编码工具, 优于早期时候提出的视频编码标准, 例如 AVC。增加的灵活性让编码器自适应决定块独立的下列的编码参数:

1. 四叉树结构的编码单元 (CU), 分区模式的预测单元 (PU), 四叉树结构的转换单元 (TU)

2. 帧内 PU 预测模式

3. 对于移动估计，有内部 PU 运动参数和参考列表索引或指数。

4. 对量化过程，有失真率的最优量化（RDOQ）。

为了提升编码效率，最优编码参数（或叫做语法元素值）是一个好的编码器的主要功能和分化点的选择。寻找最优编码参数传统做法是通过失真率来决定的：它能够在图片编码块和比特数目之间做一个折衷。失真的结果是由于使用的比特数产生的。失真率最优化问题通常可以用如下公式表示：

$$\min_{(\text{coding parameters})} (D) \text{ subject to } R \leq R_T \quad (9.1)$$

公式中的参数

D=失真

R=率（信号编码参数所需的位数）

R_t=目标率

以上最小化是结束编码参数的组合集，失真项用于量化原始和重组块之间的保真度。原则上，失真度要么依赖于数学上的方法要么把感知机制考虑进去来测量。感知指标与观察者的感知经验密切相关，但定义它们具有挑战性，是由于要建模涉及到人类视觉系统的各种生理成分的复杂性。从另一方面来讲，基于数学距离的客观质量测量，更容易推导，在很多环境下它们仍然可以在主观质量和利用率之间做一个好的折衷。而且，他们与内容无关，指的是，相同的误差分布在不同的内容上可以产生类似的客观质量公制。例如，基于客观质量公制的距离包括均方误差（MSE），峰值信噪比（PSNR），绝对误差之和（SAD）。约束最优化问题在 9.1 式中可以通过引入非负的拉格朗日乘子 λ 转化成无约束的最优化问题，拉格朗日乘数把 R 与 D 结合成拉克朗日代价函数，方式如下：

$$\min_{(\text{coding parameters})} J = (D + \lambda * R) \quad (9.2)$$

注意 λ 的变动，另一方面来说，类似于 ‘knob’：改变 λ 值可以在减少速率和失真率之间做一个折衷。例如， $\lambda = 0$ ，在 (9.2) 式中，对应的是失真率最小。相反，选择较大的 λ 值，对用的是速率最小。自然问题就会出现， λ 应该选什么值呢。sullivan 和 wiegand[22]还有 ohm 等人[19]，通过建立 λ 和量化步长大小 Q 回答了这个问题。

$$\lambda = c * Q^2 \quad (9.3)$$

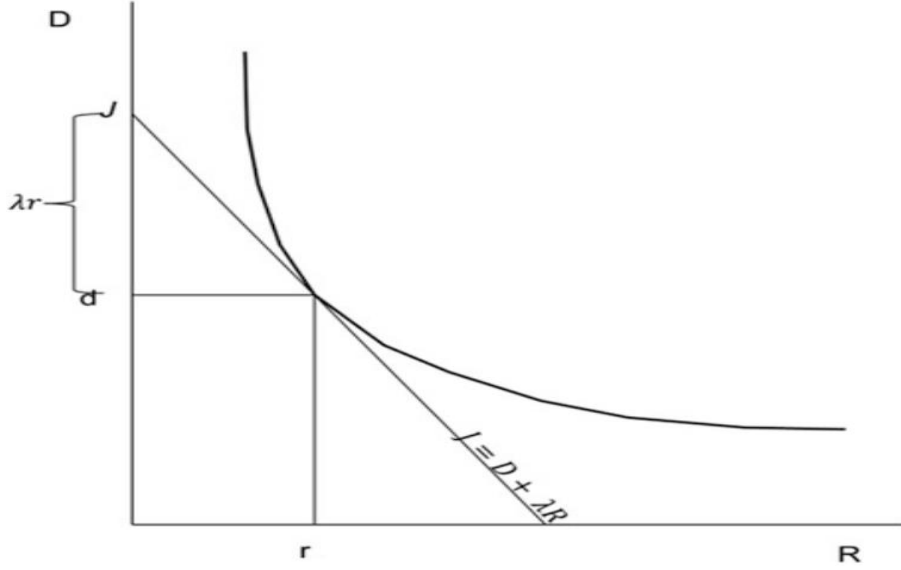


图 9.5 典型的 R-D 曲线和 J 与斜率 λ 的函数关系

在 AVC 和 HEVC 中，量化步长大小 Q 可以通过量化参数 QP 来控制， Q 是正比于 $2^{(QP-12)/6}$ 和比例常数， C 是根据编码模式决定的。

基于 (9.2) 式最小化图像的例子在图 9.5 中展示，图中一行表示针对典型的失真率曲线绘制出来的拉格朗日代价函数，即是 R 非递增凸函数。失真率曲线图与 λ 斜率平面波第一次交叉处就是 J 的最小值 [20]。

有很多可选的方法来取得 R-D 代价最优化。例如，可以最小化水平帧失真或是最小化平均帧失真来接受大量视频帧。这些之前提到的方法不是那么实用，因为他们会导致大量的复杂性和延迟。反而，像 [15, 19] 中描述的，(9.2) 式最小化单独的每一个像素块以 4 个步骤实现。（1）确定模式（2）帧内预测模式估计（3）模式估计（4）量化。因此，对于每一个块代价函数的详细预计算和与每一个编码参数的组合有联系：块的最优化 R-D 方法是 R-D 代价函数最小的组合。假定，尽管空间/时间独立可能存在于块与块之间（比如，当前预测模块是基于过去重组像素块），但在实际应用中使块通常可以忽略的 [19]。我们现在来简单描述最优化失真的步骤。

A、B 块大小一样， $S_A(i, j)$ 和 $S_B(i, j)$ 各自表示他们的 $(i, j)^{th}$ 。为了测试失真，我们使用如下指定的公制 [15]。

$$\text{Sum of Square Error (SSE)} = \sum_{i,j} (s_A(i, j) - s_B(i, j))^2 \quad (9.4)$$

$$\text{Sum of Absolute Difference} = \sum_{i,j} |s_A(i, j) - s_B(i, j)| \quad (9.5)$$

$$\text{Hadamard Transformed SAD (SATD)} = \sum_{i,j} |HT(i, j)| \quad (9.6)$$

式 (9.6) 中的 $HT(i, j)$ 是一个块的系数 $(i, j)^{th}$ ，该块通过哈达玛变换在 A 与 B 块之间的不同块获得的。JCT-VC [15] 指定如下的 λ 值：

$$\lambda_{mode} = \alpha * W_k * 2^{((QP-12)/3.0)} \quad (9.7)$$

$$\lambda_{pred} = \sqrt{\lambda_{mode}} \quad (9.8)$$

$$\omega_{chroma} = 2^{((QP-QP_{chroma})/3.0)} \quad (9.9)$$

$\alpha = 1.0 - Clip3(0.0, 0.5, 0.05 * number_of_B_frames)$ for referenced pictures

$$\alpha = 1.0 \quad \text{for non-referenced pictures} \quad (9.10)$$

其中

$$Clip3(x, y, z) = \begin{cases} x; & z < x \\ y; & z > y \\ z; & otherwise \end{cases}$$

感兴趣的读者可以参考[15] 中的 W_k 的求导以及色度的 λ 值。

基于查找编码参数来决定 CU 水平编码，最小代价函数 J_{mode} 在 (9.1) 中。

$$J_{mode} = (SSE_{luma} + \omega_{chroma} * SSE_{chroma}) + \lambda_{mode} * R_{mode} \quad (9.11)$$

失真项 SSE_{luma} 和 SSE_{chroma} 各自对应于 SSE 原始的和重构的 CU 块的亮度和色度。类似， R_{mode} 表示帧内 CU 水平用的总位数或内部模式发送信号用的总位数，CU 内的 PU 划分用的总位数，假使帧内模式的 PU 预测模式用的总位数或是假使内部模式 PU 运动参数用的总位数，TU 二叉树划分，最终要求的位数表示量化残余转换系数水平。

为了找到最好的内部 CU 编码代价， J_{mode} 用于估计所有可能的 PU 划分模式（比如， $2N \times 2N$, $N \times N$, $2N \times N$, $N \times 2N$, $n1 \times 2N$, $nR \times 2N$ ），然后选择给定的最小编码代价的 PU 划分。

每一个内部 PU 划分的运动估计是基于如 (4.12) 式所示的最小化内部预测代价来完成的。

$$mp^* = arg \min_{mp \in MP} D_{mp} + \lambda_{pred} * R_{mp} \quad (9.12)$$

对于一个给定的参考图表，通过设定 MP 可以得到最小化值，所有可能的运动参数组合叫运动矢量，其关联参考指标。(9.12) 式中的最小化任务被分成两个部分：整体采样精度和子采样精度。对于整体采样精度，失真项 D_{mp} 对应于原始的 PU 块与它的移动补偿参考块之间的 SAD。然而对于子采样运动搜索，失真项 D_{mp} 表示原始和子采样移动补偿参考块之间差异块的 SATD。 R_{mp} 项表示所需传输 mp 被编码位数的估计。

对于预测位，(9.12) 式中最小代价函数变成一个联合优化问题，问题可以通过应用迭代算法来解决[5]。对于每一个参考列表（L0 和 L1），可以单独的获取两个最优运动参数来初始化迭代参数。当在互补列表（L1）上对子像素运动搜索的时候，为了更加精炼和合并成本最小化，可以是 L0 列表中的运动参数保持为常数。一旦最小代价得以实现，与 L1 列表相关的运动参数会被保持为常数，L0 列表的运动参数通过计算联合代价最小值来调整。这些“乒乓”像迭代进程一样被一直执行着，直到收敛为止。

对于内部的 PU 预测，一个最小进程的两步被执行：

第一步中，根据 (9.13) 中预测代价函数的最小值，候选的帧内预测模式的一个最低预测代价数将被选中。

$$p^* = \arg \min_{p \in P} D_p + \lambda_{pred} * R_p \quad (9.13)$$

在 (9.13) 式中，失真项 D_p 表示原始块和使用帧内预测模式 P 的预测块之间的 SATD 值， R_p 表示所需发送信号模式 P 的编码位目。planar, DC 和所有的 33 个角度预测方向的组合，通过设定 P 得出最小化值。

在第二步中，包含第一步中候选的帧内预测模式的列表增加了三个最有可能的模式，否则该模式已经不存在列表中。在扩增的列表中，最优帧内预测模式是存在于给定的候选帧内预测模式中的最小 J_{mode} 值。

注意到，如果块大小更大或等于被传送信号的 PCM 编码块大小，那么 HEVC 允许一个 CU 块的 PCM 编码。对于 PCM J_{mode} 评估，当输入和 PCM 编码像素有相同位深时，失真项 SSE_{luma} 和 SSE_{chroma} 被置成 0。 R_{mode} 项包需求的信号发送 PCM 模式和 PCM 编码像素的所有位。

最后，在 CU 递归树的每一级水平中，通过使用 CU 水平决定模式可以获得一个编码树单元（CTU）编码水平决定模式。

在量化进程阶段中，从 R-D 方面看[17]，量化优化失真率的目标是转换系数水平的调整。从最小化进程的总体概念来看，在一个已经转换好的块中对于给定的位置 K, 假定 C_k 是最后一个非零系数; 那么对于每一个转换系数水平 L_i , 在位置 $I=K-1 \dots 0$ 中，RDOQ 试图查找优化转换系数水平， l_i^* 即最小化代价函数， $J_k(L_i)$ 如下：

$$J_k(l_i) = D_k(l_i) + \lambda * R_k(l_i) \quad (9.14)$$

为了计算简便， L_i 的可能值被限制是要么是 0 要是截断的 L_i , 要么是舍弃 L_i (比如， L_{floor} 和 $L_{ceiling}$)。失真项 $D_k(l_i)$ 由于量化误差，在转换域中被当作正常的 SSE 计算， $R_k(L_i)$ 表示用作传输水平 L_i 的位数。最佳优化方法是在所有可能的位置 K 中，最小值 J_k 位置 K 的轻量化转换水平的矢量。

9.3 客观性能分析

本节总结 HEVC 和 AVC 的编码效率对比结果。测试条件汇总在表 9.4 中，用于比较的所有编码器设定列在表 9.3 中。测试序列的结果在表 9.1 中，总结归纳在表 9.5、9.6、9.7 和 9.8 中。随机存储 Main 案例中，HEVC 的编码效率高
于 AVC 编码效率的 42.7%。然而案例中 MAIN 的所有帧内编码效率提高只有
21.9%，这表明帧内图编码效率相比于预测图编码效率提升的幅度要低（B 或 P
图）。 举个例子，四类序列的 R-D 曲线显示在图 9.6、9.7 和 9.8 中。

表 9.4 测试环境

Test conditions	
Encoder	HM12.1 (HEVC) and JM18.5 (AVC)
Test sequences	All sequences defined in Table 9.1
Bit depth	8-bit (Main configuration)
Prediction structure	AI, RA, LDP and LDB
QPI	22, 27, 32 and 37

表 9.5 HEVC 和 AVC 编码性能之间的比较（所有帧内的 Main）

	All Intra Main		
	Y (%)	U (%)	V (%)
Class A	−23.6	−21.1	−19.9
Class B	−22.7	−22.1	−21.7
Class C	−19.7	−20.8	−21.1
Class D	−16.4	−17.0	−17.7
Class E	−28.8	−27.1	−27.1
Overall	−21.9	−21.4	−21.2
Class F	−28.6	−25.3	−26.2

表 9.6 HEVC 和 AVC 编码性能之间的比较（随机存储的 Main）

	Random Access Main		
	Y (%)	U (%)	V (%)
Class A	−42.6	−42.5	−44.5
Class B	−47.7	−43.3	−42.5
Class C	−35.5	−34.7	−35.4
Class D	−33.9	−36.6	−37.9
Class E	−56.0	−54.4	−55.8
Overall	−42.7	−41.7	−42.5
Class F	−53.1	−52.2	−53.9

表 9.7 HEVC 和 AVC 编码性能之间的比较（低延迟 B 图 Main）

	Low Delay B Main		
	Y (%)	U (%)	V (%)
Class A	−38.6	−24.9	−26.7
Class B	−42.1	−34.4	−35.3
Class C	−32.7	−32.2	−32.9
Class D	−29.9	−31.2	−33.0
Class E	−44.1	−39.2	−38.8
Overall	−36.6	−32.6	−33.6
Class F	−33.9	−35.3	−37.8

表 9.8 HEVC 和 AVC 编码性能之间的比较（低延迟 P 图的 Main）

	Low Delay P Main		
	Y (%)	U (%)	V (%)
Class A	−29.3	−31.4	−33.4
Class B	−38.1	−37.3	−39.6
Class C	−32.4	−38.2	−38.5
Class D	−30.1	−38.2	−38.6
Class E	−44.4	−44.9	−44.7
Overall	−35.3	−38.0	−38.9
Class F	−35.3	−39.5	−40.5

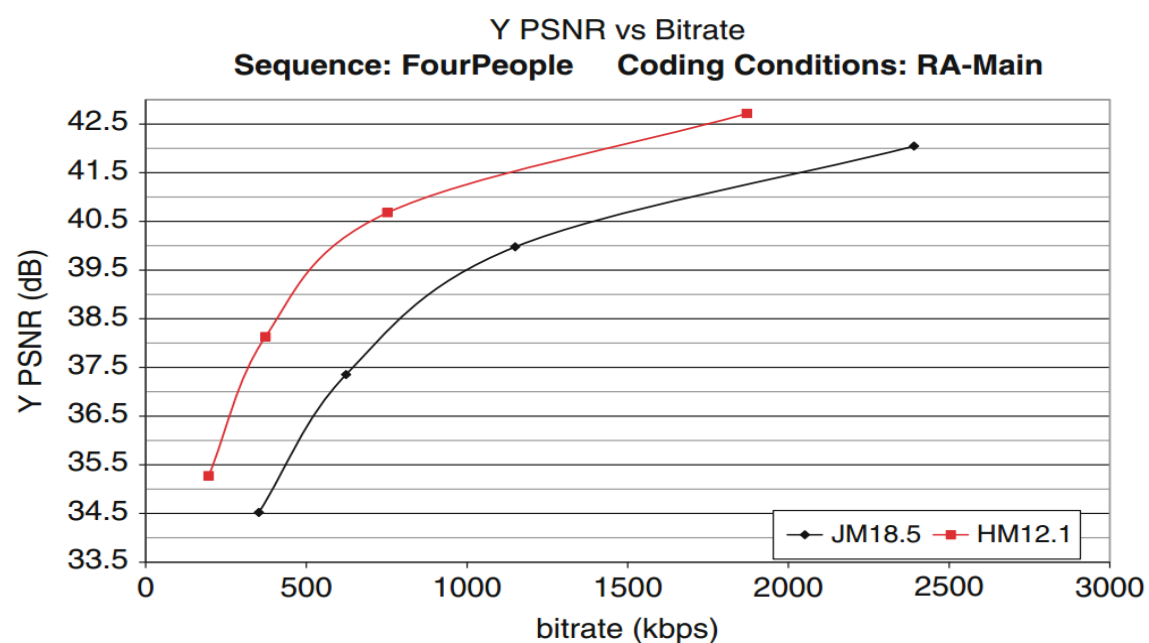


图 9.6 Y 的 R-D 曲线（4 类，RA 主体）

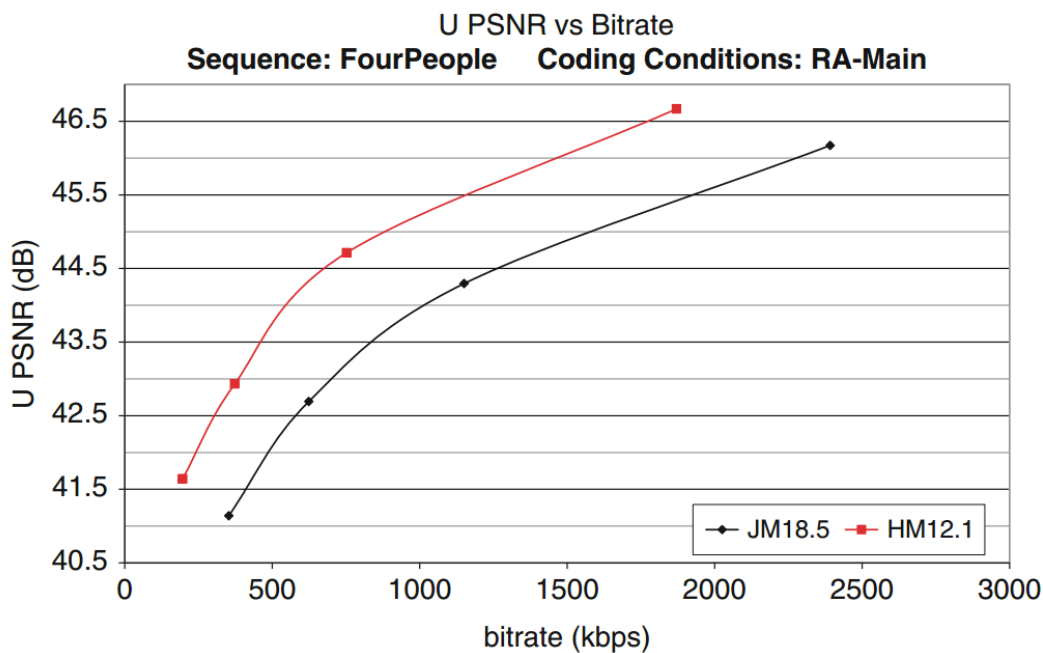


图 9.7 U 的 R-D 曲线图（4 类，RA 主体）

表 9.2 的 4K 测试序列结果汇总在表 9.9 中（只有随机获取的 Main）。我们观察到 HEVC 的编码效率提高到了 76%。

除此之外，HEVC 图片编码性能仍然是基于帧内编码，与 JPEG 和 AVC 相关的帧内编码汇总在[18]中。结果显示由于 HEVC 帧内编码比特率减小各自大概是 44%和 32%。通过主观和客观评估比较 HEVC 与 JPEG 和 JPEG2000 的帧内编码汇总在[9]中。评估结果表明，HEVC 帧内编码平均比特率优于图片编码，其比特率减少了从 16%到 43%不等（相比于 JPEG）。

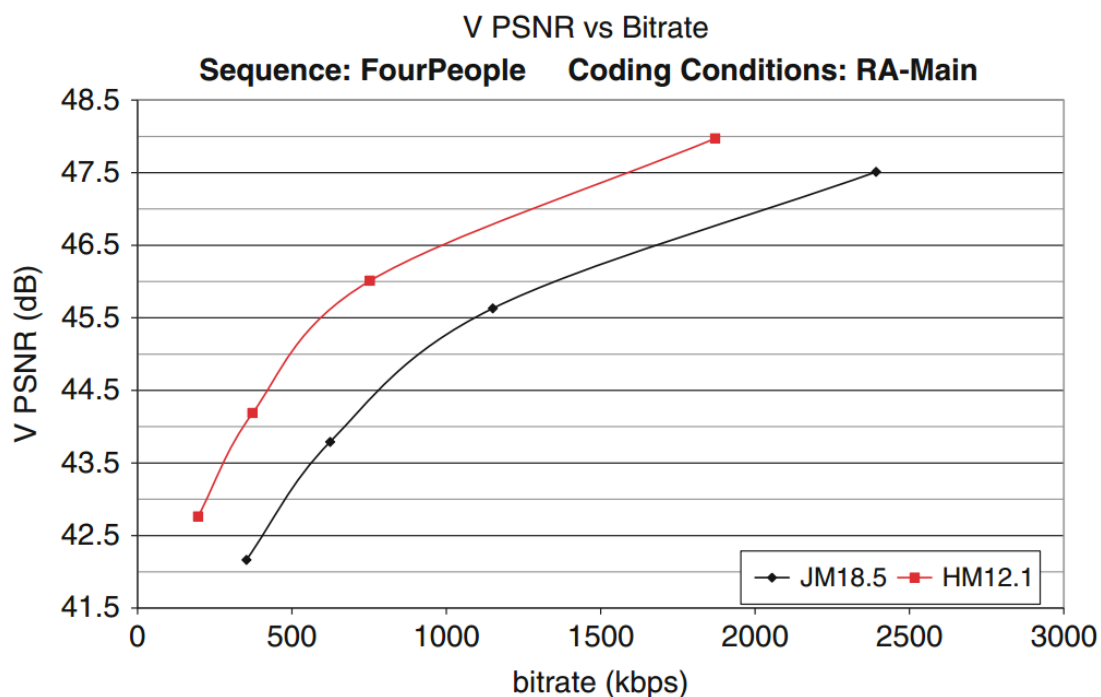


图 9.8 V 的 R-D 曲线图（4 类，RA-Main）

表 9.9 HEVC 和 AVC 的编码性能比较（随机存取的 Main）

	Random Access Main		
	Y (%)	U (%)	V (%)
Book	−58.8	−55.2	−56.7
BT709Birthday	−60.4	−54.3	−57.6
HomelessSleeping	−75.9	−79.6	−83.5
Menage	−33.3	−34.1	−36.2
Traffic	−41.4	−43.7	−43.6

9.4 主观性能分析

因为主观视频内容评估和观众感知经验有着直接的联系，它能很好地被认为是一个更可靠的编解码器测试平台。因此对于研究主观评估测试来说它很重要，测试方法学被定义为能普遍接受的指导方法和实现方式，例如，在推荐中描述的这些 ITU-R BT. 500[11-14]。在下面的小节中，我将更加详尽的描述用于参考的主观评估测试结果的测试方法学和测试环境。

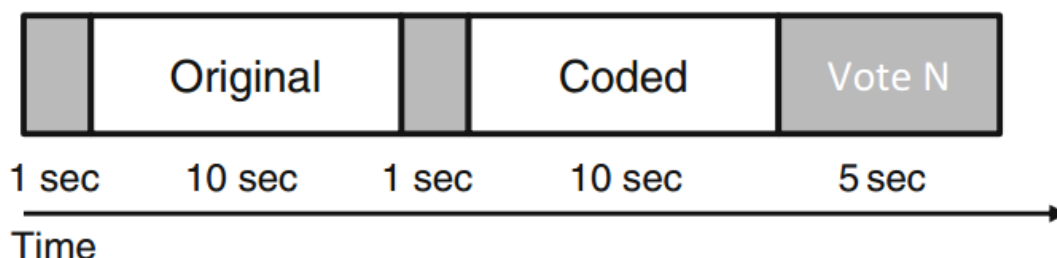


图 9.9 基本测试单元的 DSIS (BTC)

9.4.1 测试方法学

首先，我们提供经常用到的主观质量评估方法的一个简明的辅导。通常来说，有两种广泛使用的方法来实验视觉评估测试：双边刺激和单边刺激。在双边刺激测试中，受试者评估两个视频剪辑参考（原始）与受损（被编码的）之间的质量或是质量变化。在单边刺激侧记测试中，受试者只评估受损（被编码的）视频剪辑的质量。我们现在描述前面的两个例子，双边刺激受损程度(DSIS)和双边刺激连续质量程度（DSCQS）。

9.4.1.1 DSIS(双边刺激受损程度)

这种方法用于被评估的材料显示视觉质量的更宽范围，包括所有的质量范围（不包括受损的）。这里有两个 DSIS 的变异体：变异体 I 和变异体 2。变异体 I 的基本测试单元（BTC）的结构见图 9.9。他由两个连续的视频剪辑演示组成。原始（参考）视频剪辑最先被演示，接着是受损（被编码的）版本的视频剪辑的演示。然后消息显示 5 秒要求观察者做出选择。

预期的观众从一个被定义好的质量评估等级中把他们的视觉质量分数填在答案单上，比如质量等级被分成 5 个等级，从 1（非常讨厌）到 5（没有察觉）。在 DSIS 的变异体 II 中，原始（参考）视频剪辑和受损（被编码的）版本的视频剪辑片段被播放两次，让后让观众选择。在 9.4.2 节中实验的视觉测试评。描述的 DSIS 变异体 1 的方法学较早被选。

9.4.1.2 DSCQS(双边刺激连续质量等级)

双重刺激连续质量等级用于不可能播放所有范围的质量等级的案例中。使用这个方法，对于每个 BTU，原始（参考）和被编码的（受损的）视频剪辑的样品随机的播放两次。第二次演示结束之后，观众被要求各自给每两个原始和两个被编码的视频剪辑打分。注意的是，由于随机演示，观众是没能优先的识别哪个剪辑视频属于原始的和哪个是受损的视频剪辑。如图 9.10 所示，

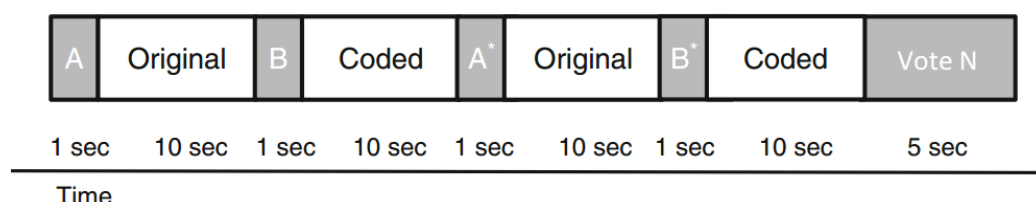


图 9.10 基本测试单元 DSIS（BTC）

DSCQS 方法的 BTC 结构包含两个连续的演示片段。首先，在中灰色的屏幕中间显示一个字母 A 1 秒钟，紧接着是 10 秒剪辑视频演示-它们要么是原始的要么是受损的剪辑视频。然后，一个中灰色屏幕上显示字母 B，紧接着是 10 秒的剪辑视频演示。相反，在第二轮演示期间类似的过程一直重复着，字母 A、B 变成字母 A*、B*。最后，显示 5 秒的消息来提示观众要做选择了。

9.4.1.3 训练项目

视觉测试结果可能高依赖于参与者适当的训练。为了让观众熟悉测试程序，在开始主观评估测试之前，观众简单了解测试程序和参与训练项目是非常重要的

的。还有，用于训练的剪辑视频和用作真实测试的剪辑视频不能一样。编码受损视频应该类似出现在测试中的材料。但在训练项目中，应该包括三个 BTCs（最坏质量、中等质量、最好质量）让观众了解测试视频的质量等级。

9.4.1.4 场景浏览

在实验室举行观看会议，一般的内部灯光必须是低电平且显示器后面应该纺织均匀的灯光。灯光的强度指定是 ITU-R BT. 500[11, 14]。不能有光源直接反射到荧幕上或引起反射。天花板，楼层和实验室的墙面必须用无反射的材料制作而成（地毯和天鹅绒等）且应该尽可能把颜色调整成 CIE 光源标准颜色 D65（日光发光体，6500K）。浏览室必须被保护防止外部视觉或是音频污染。

9.4.2 主观质量评估测试

本章节主要汇报 EPPFL MMSPG 测试实验室的主观质量评估结果，该实验室满足由 ITU-R BT. 500 解决后的视觉数据主观评估的推荐条件。值得注意的是，在进行如表 8 描述的主观评估测试时，本章节中的测试执行方法学会大大受益于其经验获得。

9.4.2.1 测试环境

测试房间由一个控制的灯光系统装备，该灯光系统拥有 6500K 颜色，温度和周围环境亮度是屏幕亮度最大值的 15%，因此在测试区域内的展现是所有的背景墙颜色和窗帘颜色是中灰色的。实验室环境调整通过避免外部因素无意影响来确保主观测试结果的再现性。

为了显示测试刺激物，将用到两个具有 2560X1600 像素的原始分辨率的艺卓 CG301W LCD 显示器。根据如下的描述这两个显示器通过一个 X-Rite 爱色丽 I1 display pro 校色仪来校准。全范围的 sRGB，D65 白色点，120cd/m² 光，和最小黑色阶。

这个实验需要两个受试物，每个受试物通过显示器评估测试材料。受试物设在显示器中央的正交中心的行上，处在距图片高度 2.2 倍的位置处，大致相当于两个相邻像素之间的 1 弧分的视觉角度，如表 13。

9.4.2.2 测试方法学

双重刺激受损规模方法论（DSIS Variant I）如早期描述的那样，选中受损部分来测试。使用到一个 5 阶受损程度（5：细微的，4：可接受的嘈杂，3：轻微嘈杂，2：嘈杂，1：非常嘈杂）。这些受试物通过视频序列对展现出来，第一

个视频序列总是参考帧，第二个视频序列是被评估的帧。每一对序列对展现完之后，接下来是 5 秒的选择时间。受试者将被问到一次刺激和二次刺激的受损率是多少，为了表达这些判断，项目中将使用定义好的评定等级表来修饰判断结果。

9.4.2.3 数据集

本实验使用表 9.2 的 5 个视频序列，他们具有不同的视觉特性、分辨率和帧率。所有的序列被贮存在原始视频文件中，被逐渐扫描，YCbCr4:2:0 的颜色采样。这些序列被压缩成 HEVC 和 AVC 格式。对于每一个序列，编解码需要选择 4 个量化参数，这样总的会有 40 次刺激测试。

通过使用 Sintel39 系列（分辨率是 3840x1744）来产生 5 个训练样品和通过专业的观察者手动选择，这样确保样品的质量是评估范围所有等级中最具有代表性的。

原始的序列被分割成与显示器一样的分辨率，紧紧保留中间部分，10bit 的序列被裁剪成 8Bit 的序列。

9.4.2.4 训练项目

实验之前，一份同意书提交给受试者签名，通过口头知道讲解他们的任务是什么。除此之外，还组织一项训练课程来让受试者熟悉评估流程。

9.4.2.5 测试课程

对于单次测试课程由于总的测试样品太大，所有实验被分成两部分，第一部分大约 13 分钟。课程测试期间，受试者有 10 分钟的休息。测试材料被随机分布于两个测试课程中。

三对虚拟受试者（一个高质量，一个低质量，一个中等质量）的分数不包括在结果里面，但包括在每部分开始时的测试中，以便稳定测试者的评定等级。为了减少上下文切换的影响，显示的刺激顺序被随机使用在不同的受试者群组的排列中，因此相同的内容从不连续显示。

总共 18 位无经验的受试者（6 个女性和 12 个男性）参加了这个实验。他们的年龄在 18 到 27 岁之间，平均年龄是 23.4 岁。所有的受试者各自通过 Snellen 和 ISHIHARA 图表筛选出正确的视觉敏锐度和色彩视觉。

9.4.2.6 结果分析

主观结果通过首次检测和移除严重偏离其他人的分数的受试者分数来加工。根据附表 14 的 3.2.1 节描述指导，检测的异常值被加工。在这个研究中，需要检测一个异常值。然后，需要计算每一个刺激测试的平均主观分数，因为这个平均值覆盖有效受试者的整个平均等级，关联 95% 自信区间，估算出学生的分数分布。

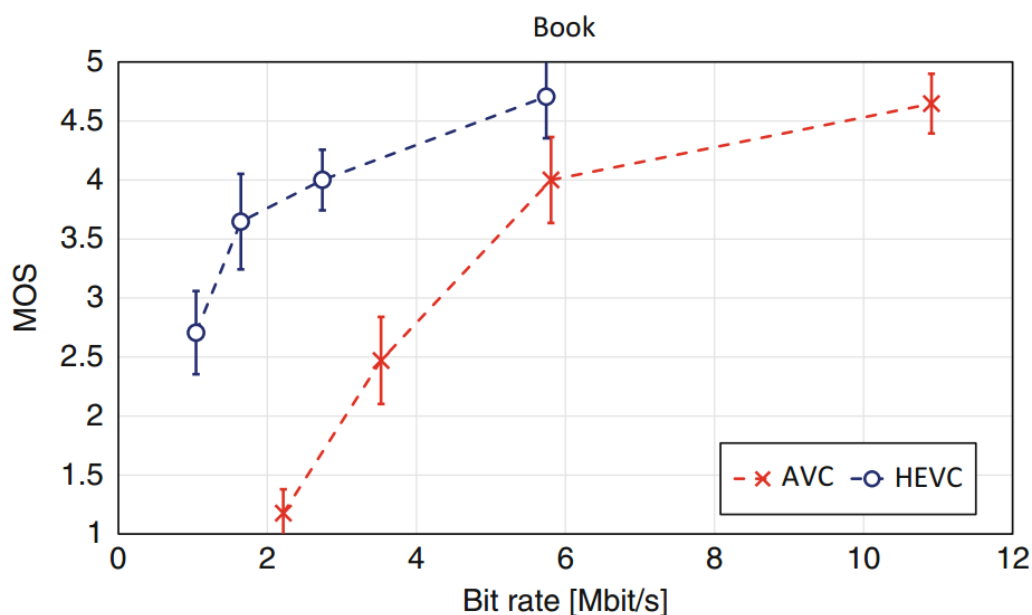


图 9.11 R-D 曲线（预定）

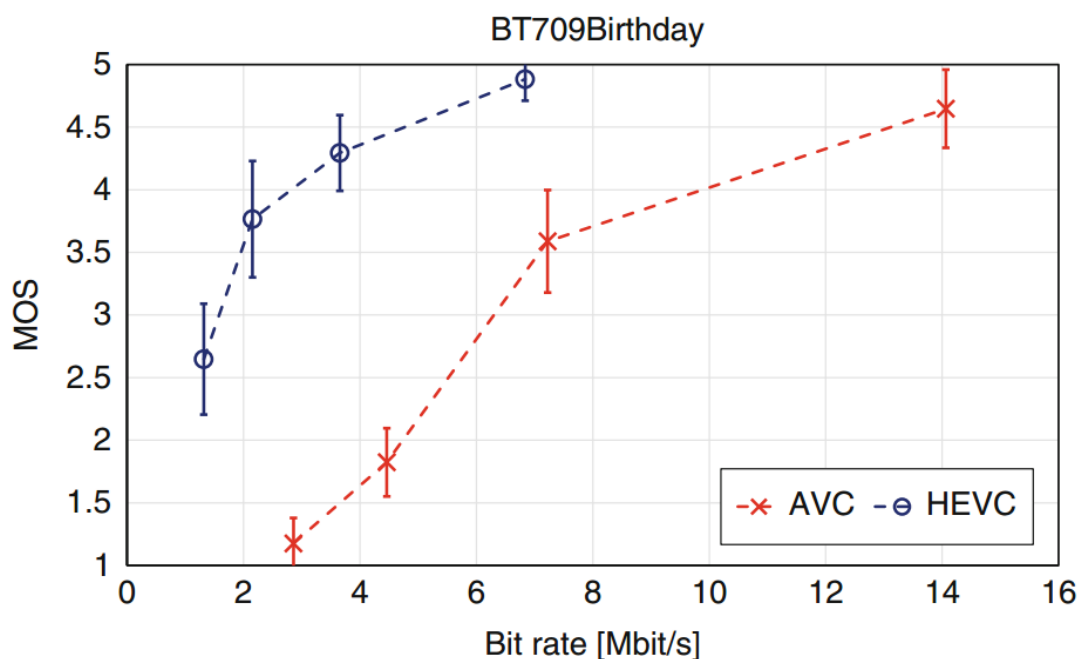


图 9.12 R-D 曲线（BT709 诞生）

9.4.2.7 曲线图结果失真率

通过图 9.11, 9.12, 9.13, 9.14 和 9.15 中的主观质量评估来获得 R-D 曲线图。从这些图片中可以看出 HEVC 视觉质量相比与 AVC 有大量的提升，特别是在较低的比特率上面。

9.4.2.8 平均比特率的差异

HEVC 和 AVC 平均比特率的差异是通过使用附表 7 中的模式来计算的。这个模型是 Bjøntegaard 模型的一个延伸模型：

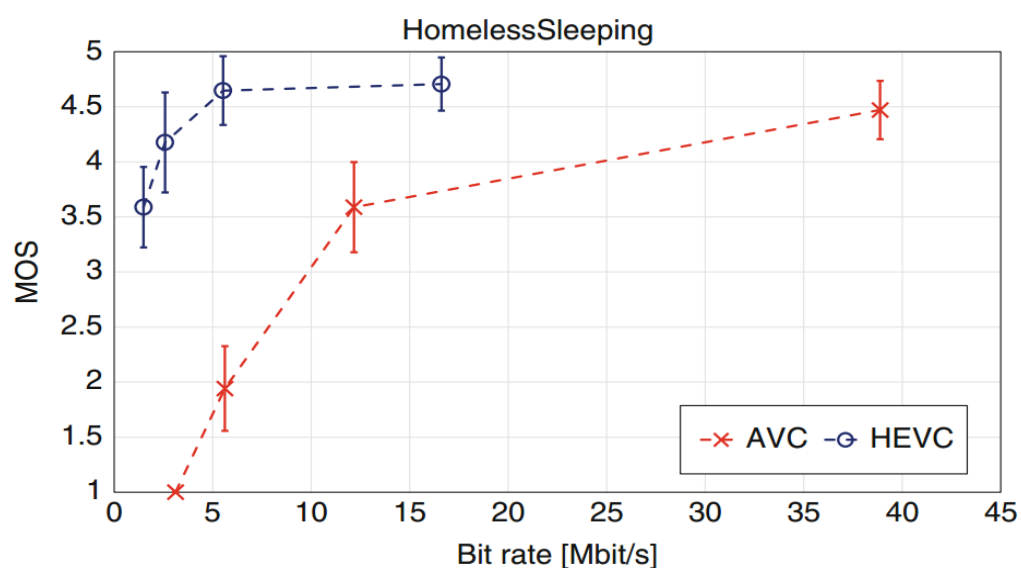


图 9.13 R-D 曲线图（无家可归者睡眠）

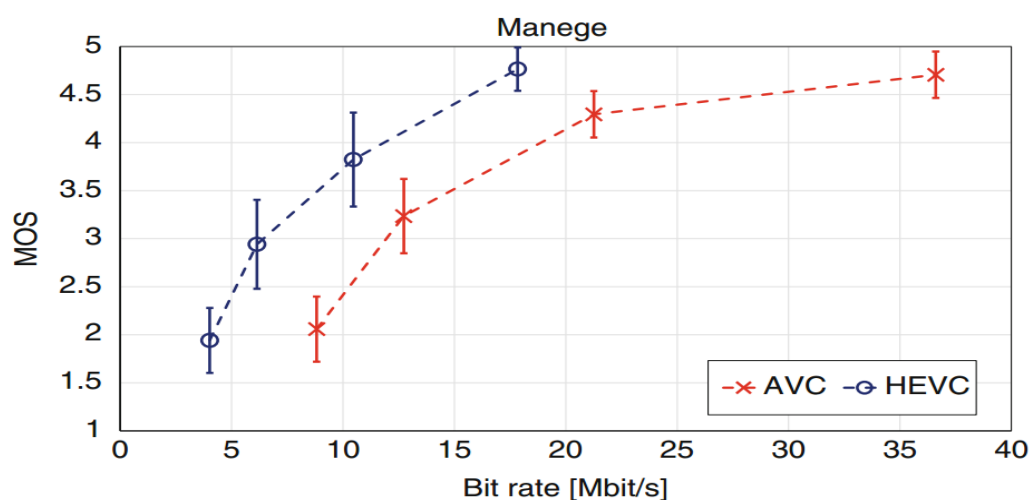


图 9.14 R-D 曲线图（马术练习者）

对于主观分数： ΔR 是通过 MOS 来计算； $[\Delta R_{\min}, \Delta R_{\max}]$ 提供 ΔR 的一个置信期间，然后决定考虑用哪个置信区间下标计算主观分数；置信区间下标获取会考虑 MOS 在整个评定等级的分部，表 9.10 所有值合适的成分。

对于 CTC 测试序列的视觉质量评估感兴趣的读者可以参考附件 19，里面是主观测试结果的汇总。这份报告结果表明在视频测试序列例子中，使比特率失真率下降 50% 是可以被实现的。

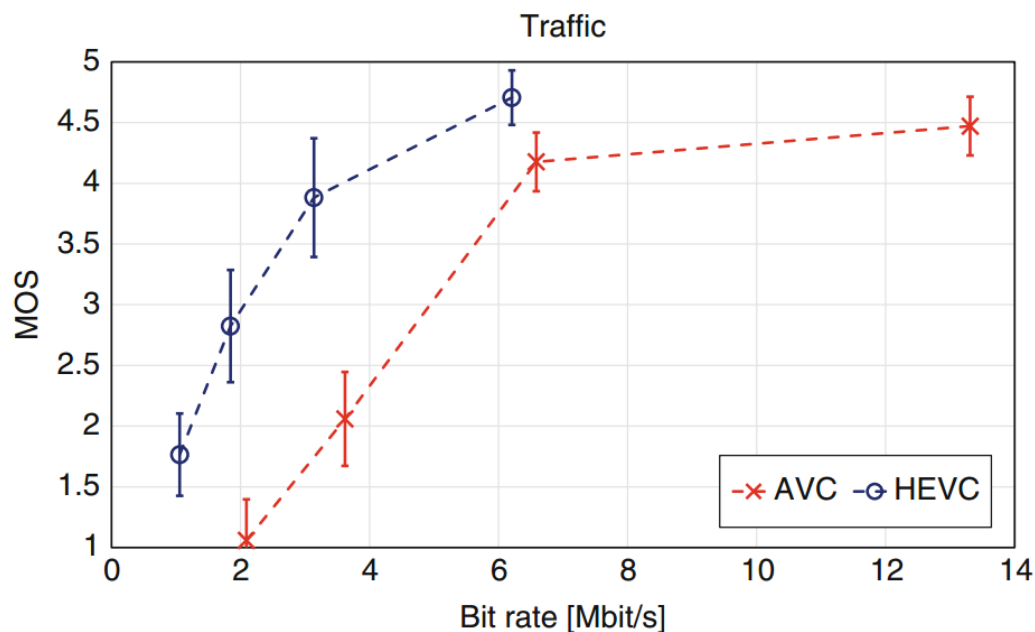


图 9.5 R-D 曲线图（交通运输情况）

表 9.10 测试位流的比特率差异

Sequence	ΔR (%)	$[\Delta R_{\min}, \Delta R_{\max}]$	Confidence index (%)
Book	-62	$[-75 \%, -51 \ %]$	100
BT709Birthday	-71	$[-81 \ %, -60 \ %]$	100
HomelessSleeping	-87	$[-94 \ %, -71 \ %]$	100
Manege	-43	$[-60 \ %, -17 \ %]$	88
Traffic	-55	$[-68 \ %, -39 \ %]$	100
Average	-64	$[-76 \ %, -48 \ %]$	98

9.5 编码器产品质量性能分析

本章节描述 eBrisk-UHD 和 x264 编码器非标准主观质量的比较结果，他们分别各自配置自己，保持与 HEVC 主要轮廓和 AVC 高度轮廓一致。本章节对编码器比较的描述，是为了对本章前面讨论主观质量对比起到一个补充的作用，本章中用到 HEVC 和 AVC 编码器的参考使。

9.5.1 测试条件

在这一节中，测试环境包括编码器配置和评估条件（比如：序列表示的细节，观看设备，光线条件等），和测试描述的视频序列。

表 9.11 主观质量比较在播放顺序中用到的视频序列（从顶部到底部）

Resolution	Sequence name	Frame count	Frame rate (fps)	Bit depth
3,840 × 2,160	IntoTrees	400	50	8
3,840 × 2,160	OldTownCross	400	50	8
3,840 × 2,160	ParkJoy	400	50	8
3,840 × 2,160	DucksTakeOff	400	50	8

9.5.1.1 编码器设定

编码器配置是为了高效率编码控制。特别是，HEVC 编码器被配置成主要轮廓，是为了使用类似于 9.2.3.2 节描述的预测结构，该预测结构的图片帧内周期设置为 48 帧。AVC 编码器配置是为了使用默认的参数值，除此之外这些参数还要被明确设定（比如：按键返回入口函数，使用指定的帧内周期，设定为 48 帧）。每个视频序列会有好几个 AVC 编码器被执行，这些视频序列每个的量化参数值不一样。在每个测试中，编码器产生的平均比特率，相当于 HEVC 编码器产生的比特率的 2.5 倍（比如：HEVC 编码序列的比特率大约低于 AVC 编码序列 60%）。

9.5.1.2 主观评估条件

27 位志愿者即观看受试者参加了主观质量评估实验。其中 15 位观看者有一点或是没有先前视频序列评估经历。观看场景被布置在一个用 XBR55X900A 55 UHD 索尼 Bravia LED 显示器装置的微暗房间中。主观评估的额外关键要素和视频演示方法被列在如下：

1. 未经训练的观众每次都与每个观看者一起参与坐在椅子上，椅子的位置距显示器和中央约 1.5 米。
2. 对于列在表 9.11 的四个测试序列中的每一个，HEVC 和 AVC 编码的视频序列被裁剪成水平维度，并拼接在一起。被比较的编码器的相对位置是随机分配的，实验是以双盲的方式被执行。
3. 序列显示在他们原始的空间分辨率中；然而，显示分辨率被设定成每秒 30 帧（比如：序列显示帧分辨率是他们原始帧率 50fps 的 60%）。
4. 视频序列给每位观众显示 1-3 次数，根据 5 度的量表，要求观众评估出并行编码的相对质量。5 度的量表：左边好，左边有点好，中立，右边有点好，右边好。播放每个视频序列衔接处之间留有大约 30 秒的时候给观众记录他们的选择。共七个视频对播放，每个观看部分大约持续 7 分钟的时间。

9.5.1.3 测试序列

在主观质量评估实验中，要用到表 9.11 中描述的 4 个视频序列。8 位 YCbCr4:2:0 的 4K 视频序列有各种在流媒体应用程序环境中遇到的典型特点，这些通常是容易获得或是用于视频编码测试目的的特点来选择出来。所有序列的原始帧率都是 50 帧每秒。4K 序列以 30 帧每秒的形式来播放；显示器高帧率具有播放 4K 的内容的能力。播放顺序遵循表 9.11 排列，从顶部到底部。

9.5.2 主观质量评估结果

表 9.12 是主观视频质量评估的结果。每一个视频序列，HEVC 和 AVC 编码器的平均比特率是按着观众的视频序列评估值给出的结果，即根据 9.5.1.2 节描述的 5 度量表。使用固定的 QP 编码每一个视频序列。选择编码的 QP 值来产生的序列是好的，但不是为了得到特别高的视频质量，是为了避免出现两种编码都能产生不可识别的优秀质量的场景，或者两种编码都能产生大量的编码垃圾。

9.5.3 结果

表 9.12 展现的主观结果表明，观看者要么没有偏好要么喜欢 HEVC 编码的视频，该视频的比特率大约低于所有尝试实验的 69.4% 的 AVC 编码比特率的 60%。

表 9.12 使用 HEVC 和 AVC 编码器进行编码序列的主观视野比较结果（B 最好，SB 比较好，NP 无参考价值）

Sequence name	HEVC QP	HEVC bit rate	AVC QP	AVC bit rate	HEVC B	HEVC SB	NP	AVC SB	AVC B
InToTrees	34	4.5 Mbps	35	11.9 Mbps	21	6	0	0	0
OldTownCross	32	3.1 Mbps	34	7.8 Mbps	14	3	3	4	3
ParkJoy	38	11.9 Mbps	38	29.0 Mbps	0	2	7	10	8
DucksTakeOff	37	15.4 Mbps	39	37.6 Mbps	12	4	3	6	2
Selection count totals					47	15	13	20	13
Percent totals (%)					43.5	13.9	12.0	18.5	12.0

这些结果与参考文件[8]中主观结果始终保持一致。除此之外，对于确定的 4K 视频序列，HEVC 相对于 AVC 的编码效率增益在早期的研究中已经报告出（比如：参考文件[10]），在研究中已经使用相同视频序列的高质量（较低分辨率）重采样版本，从中可以看出对于 4K 视频序列 HEVC 相对与 AVC 的编码效率增益更大。这些比较表明，相比与 AVC 观察到的 4K 视频视频序列，HEVC 的编码效率增益增加了，但不能单独通过内容的差异来解释。

9.6 结论

在本章中，给出 HEVC 和 AVC 在主观和客观质量评估方面的比较结果。由于 HEVC 的复杂度增加，讲述在率失真方面选取最优编码参数方法。特别关心的是，当研究 HEVC 和 AVC 主观和客观质量评估的时候我们使用了一个统一的方法。主观和客观测试结果表明，HEVC 的压缩效率相比于 AVC 而言有明显的提升。更具体的说，比特率失真率，基于 CTC 测试序列的客观评估表明，HEVC 相比于 AVC 提升的整个性能中，其中 AI 提升了 22%，RA 提升了 43%，LDB 提升了 37%，LDP 提升了 35%。此外，通过使用非 CTC 测试序列，我们观察到在编码性能上提升了 76%，结果在表 9.9 中。主观评估测试结果表明，甚至一个更高的比特率，节省掉 55%-87% 的比特率是可以实现的。非正规的视觉质量评估测试结果可以确定，在视频流应用方面，HEVC 在压缩能力上有大幅度的提升，这远远超过 AVC 的压缩能力。可见，HEVC 相比于 AVC 在编码性能增益通常是随着增加视频分辨率而提高，增加的分辨率至少达到 4K 分辨率。

鸣谢:

关于 9.4 节, 作者想感谢 4 位合作人, 英国广播公司 (BBC), Kamerawerk 股份有限公司 ([Kamerawerk GmbH](#)) 和 Technicolor 集团, 在视觉测试中他们提供了原始的视屏测试序列。我还特别感谢和感激日本电信公司 NTT Docomo 的 T. K TAN 的帮助和支持, 因为在准备 9.4 节的主观性能评估用到的编码视频测试序列就是他们帮忙准备的。在 EPFL 上执行主观评估成为可能, 这归功于瑞士国家科学研究基金会 (FN 200021-143696-1), EC 资助的卓越网络视频, 和欧洲网络公司 COST Action IC1003 QUALINET 在多媒体系统和服务质量方面的经验。感谢通 Hiromi Nemoto 的努力, 在 EPFL 平台上建立和实施的主观评估。

参考文件:

1. Bjøntegaard G (2001) Calculation of average PSNR differences between RD-curves, ITU-T SG16 Q6 Video Coding Experts Group (VCEG), Document VCEG-M33, Austin, Apr. 2001
2. Bossen F (2011) Excel template for BD-rate calculation based on Piece-wise Cubic Interpolation, JCT-VC Reflector
3. Bossen F (2013) Common test conditions and software reference configurations, Joint Collaborative Team on Video Coding (JCT-VC), Document JCTVC-L1110, Geneva, Jan. 2014
4. Cover TM, Thomas JA (1991) Elements of information theory, Chapter 13. Wiley, New York
5. Flierl M, Girod B (2003) Generalized B pictures and the draft H.264/AVC video compression standard. IEEE Trans Circuits Syst Video Technol 13(7):587 - 597
6. H.264/MPEG-4 AVC Reference Software, Joint Model 18.5:
<http://iphome.hhi.de/suehring/tml/download/jm18.5.zip>

-
7. Hanhart P, Ebrahimi T (2014) Calculation of average coding efficiency based on subjective quality scores. *J Visual Commun Image Represent* 25(3):555 – 564
 8. Hanhart P, Rerabek M, De Simone F, Ebrahimi T (2012) Subjective quality evaluation of the upcoming HEVC video compression standard. In: *Proc. SPIE. 8499, Applications of Digital Image Processing XXXV*, no. 84990V, Oct. 2012
 9. Hanhart P, Rerabek M, Korshunov P, Ebrahimi T (2013) Subjective evaluation of HEVC intra coding for still image compression. In: *Seventh international workshop on Video Processing and Quality Metrics for Consumer Electronics (VPQM)*, Scottsdale, Arizona
 10. Horowitz M, Kossentini F, Mahdi N, Xu S, Guermazi H, Tmar H, Li B, Sullivan GJ, Xu J (2012) Informal subjective quality comparison of video compression performance of the HEVC and H.264/MPEG-4 AVC standards for low-delay applications. In: *Proc. SPIE. 8499, Applications of Digital Image Processing XXXV*, no. 84990W, Oct. 2012
 11. ITU-R Rec. BT.500-11 (2006) Methodology for the subjective assessment of the quality of television pictures
 12. ITU-T Rec. P.910 (2008) Subjective video quality assessment methods for multimedia applications
 13. ITU-R BT.2022 (2012) General viewing conditions for subjective assessment of quality of SDTV and HDTV television pictures on flat panel displays
 14. ITU-R BT.500-13 (2012) Methodology for the subjective assessment of the quality of television pictures
 15. McCann K, Bross B, Han WJ, Kim IK, Sugimoto K, Sullivan GJ (2013), High Efficiency Video Coding (HEVC) Test Model 13 (HM 13) Encoder Description, Joint Collaborative Team on Video Coding (JCT-VC), Document JCTVC-01002, Geneva, Oct. 2013
 16. Joint Collaborative Team on Video Coding (JCT-VC) of ITU-T SG 16 WP 3 and ISO/IEC JTC 1/SC 29/WG 11. HM12.1 Reference Software: [https://hevc.hhi.fraunhofer.de/svn/svnHEVCSoftware/\(mainsite\)](https://hevc.hhi.fraunhofer.de/svn/svnHEVCSoftware/(mainsite)) and [http://hevc.kw.bbc.co.uk/svn/jctvc-a124/\(mirrorsite\)](http://hevc.kw.bbc.co.uk/svn/jctvc-a124/(mirrorsite))
 17. Karczewicz M, Ye Y, Chong I (2008) Rate distortion optimized quantization, ITU-T SG16 Q6 Video Coding Experts Group (VCEG), Document VCEG-AH21, Antalya, Jan. 2008

-
18. Nguyen T, Marpe D (2012) Performance analysis of HEVC based Intra coding for still image compression. In: PCS2012, May 2012, pp 233 - 236
 19. Ohm J-R, Sullivan GJ, Schwarz H, Tan TK, Wiegand T (2012) Comparison of the coding efficiency of video coding standards - including High Efficiency Video Coding (HEVC). IEEE Trans Circuits Syst Video Technol 22(12):1669 - 1684
 20. Ortega A, Ramchandran K (1999) Rate-distortion methods for image and video compression:an overview. IEEE Signal Process J 23 - 50
 21. Schwarz H, Marpe D, Wiegand T (2005) Hierarchical B pictures, Joint Video Team (JVT), Document JVT-P014, Poznan, July 2005
 22. Sullivan GJ, Wiegand T (1999) Rate-distortion optimization for video compression. IEEE Signal Process J 74 -90
 23. VideoLANx264 SW library (2013)
<http://www.videolan.org/developers/x264.html>. Version core 135 r2345, 30 July 2013