arXiv:2512.16101v1 [cs.MM] 18 Dec 2025

三动态　　面向用户生成内容视频压缩的预处理框架　　N

Fei Zhao[1]、Mengxi Guo[2], Shijie Zhao[2]†、Junlin Li[2],Li Zhang[2],Xiaodong Xie[1]*

[1]北京大学计算机科学学院[2]字节跳动

摘要

近年来，用户生成内容（UGC）已成为互联网流量的主要来源。然而，与传统编码测试视频相比，UGC 视频表现出更高的多样性和更丰富的特征。这种差异给机器学习算法在更广泛的 UGC 场景中优化编码的有效性带来了挑战。为解决这一问题，我们提出了一种面向 UGC 的三动态预处理框架。首先，我们采用自适应因子调节预处理强度。其次，采用自适应量化级别对编解码器模拟器进行微调。第三，我们利用自适应λ权衡来调整率失真损失函数。在大规模测试集上的实验结果表明，我们的方法取得了卓越的性能。

索引词— 深度学习、视频编码、预处理、用户生成内容

1. 引言

随着社交媒体和视频分享平台的快速发展，用户生成内容（UGC）已成为视频流媒体服务互联网流量中一个重要且不断扩大的组成部分。为解决这一问题，许多研究人员致力于开发高效的视频编码算法。尽管人们已为 H.264/AVC [1], H.265/HEVC [2], 和 H.266/VVC [3], 等视频编码标准的开发做出了一些努力，但不断增长的流媒体内容仍需要更高效的视频压缩优化。

与传统客观度量相比，感知视频编码是一条前景广阔的研究方向，与人类视觉系统（HVS）的感知密切契合。这项系统性研究不仅包括视频质量评估（VQA）方法，例如视频多方法评估融合（VMAF） [4], , 还涉及针对标准视频编解码器的感知优化算法开发 [5, 6, 7]。在这些研究中，预处理是一种很有价值的方法，无需修改编解码器即可提高编码效率。

*字节跳动实习生。 †通讯作者。

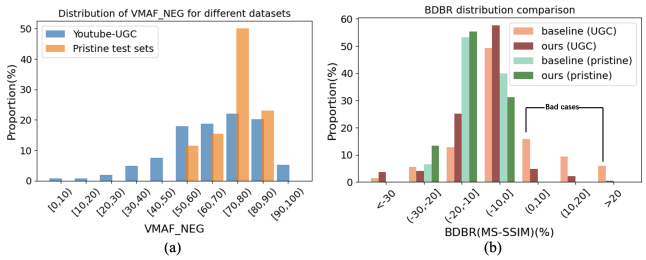


图 1: (a). 在 1500kbps 下对不同数据集进行编码后 VMAF NEG 的分布 (b). 基线方法与我们的方法相比的 BDBR 性能分布。

随着深度学习的兴起，利用深度神经网络的预处理技术在感知率优化方面取得了进展。与运动补偿时间滤波（MCTF） [8], 等传统预处理技术相比，深度预处理在原始测试集 [7], 上的感知度量方面，带来了更大的比约恩特加德增量比特率（BDBR） [9] 收益。这展现了深度学习在视频预处理中的潜力。然而，将深度预处理应用于用户生成内容数据时会遇到问题。如图1(a)所示，基线模型是我们基于 [7]复现的深度预处理器。该模型在原始测试集上表现良好，但也产生了不良案例，即在 Youtube-UGC [10]中，BDBR（多尺度结构相似性）> 0 为负的案例占总数的比例高达26%。这一现象源于用户生成内容视频的时空复杂度高于原始测试集，导致编码行为存在不可忽视的差异。如图1(b)所示，我们采用 VMAF NEG 作为失真度量，对 Youtube-UGC 和原始测试集（JVET B类、XIPH [11], 等）中以 1,000kbps 编码的视频分布进行比较。显然，用户生成内容视频比原始测试集具有更高的多样性。因此，以往的深度预处理方法缺乏感知视频序列空间和时间信息的能力，难以有效处理大量用户生成内容视频。

为解决上述问题，本研究提出了一种用于压缩用户生成内容视频的三动态预处理（TDP）框架。该框架由

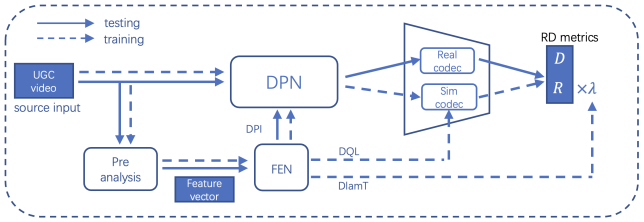


图 2：所提出方法的整体流程。请注意，训练时我们使用完整的 TDP（DPI、DQL 和 DlamT），而测试时仅使用 DPI。训练时使用编解码器模拟器，测试时则使用真实编解码器。

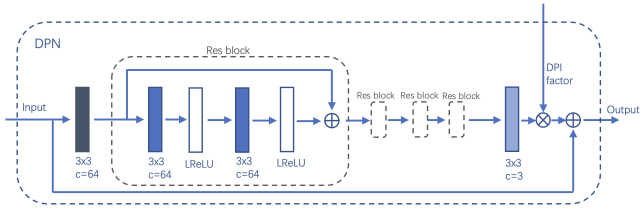


图 3：所提出的 DPN 结构，其中包含四个基本残差块。通过采用残差连接架构，我们可以通过乘法因子动态微调处理强度。

三个组件构成，分别是动态处理强度（DPI）、动态量化级别（DQL）和动态 λ 权衡（DlamT）。这些组件利用训练阶段对源内容预先分析得到的特征，感知视频序列的时空复杂度，从而有效训练深度预处理网络。实验结果表明，我们的流程使预处理网络在 VMAF NEG 上实现了 7.14% 的 BDBR 降低，在 VMAF 上实现了 12.03% 的 BDBR 降低。此外，我们的方法还能显著减少 YouTube 用户生成内容数据集中的不良案例数量。

2. 所提出的方法

如图 2 所示，我们的 TDP 框架由预分析和三部分动态方案组成：DPI、DQL 和 DlamT。训练时使用全部三个 TDP 组件，测试时则仅应用 DPI。下文各小节将详细说明 TDP 中的这些组件。

2.1. 预分析

我们对输入视频进行预分析以生成特征向量，从而指导训练期间后续的三动态预处理。这包括提取帧级空间信息（SI）和时间信息（TI）[12],，包括最大值、平均值和标准差。我们还在指定比特率下使用预设快速 模式的 x264 [13]，以获取帧级 QP。所有特征拼接在一起，形成一个 7 维特征向量，再由双层多层感知机（MLP）特征提取网络（FEN）处理，生成动态强度因子 f_d，并通过 DPI 将其应用于深度预处理网络。值得注意的是，QP 特征还会进一步用于 DQL 和 DlamT。

2.2. 动态处理强度

我们对预处理优化过程进行建模。当对视频 x进行以率失真（RD）为导向的预处理 $\mathcal{P}$时，如果指定了目标码率，主要目标之一是最小化整体系统失真，表示为 $min(D_{proc+codec} = |x - \mathcal{C}(\mathcal{P}(x))|)$。在某些特定场景中，尤其是当输入视频序列的时空复杂度极低时，编码器 $\mathcal{C}$ 几乎以无损方式运行。在这种情况下，失真 D_{codec} 趋近于零，因此：

$$D_{codec} = |x - \mathcal{C}(x)| \to 0; \tag{1}$$

$$D_{proc+codec} \to |\mathcal{C}(x) - \mathcal{C}(\mathcal{P}(x))|. \tag{2}$$

在这种情况下，$\mathcal{C}(x) \to \mathcal{C}(\mathcal{P}(x))$，这表明 $\mathcal{P}(x)$趋近于 x。这意味着预处理强度应降至零。正如以往方法所采用的那样，使用固定的预处理强度显然缺乏合理性。为解决这一问题，我们提出一种动态处理强度（DPI）方案。我们将动态预处理网络（DPN）的网络结构设计为残差连接形式，如图 3 所示。DPN 负责对视频序列进行预处理，并输出残差处理掩码 x_m。为了使预处理能够调节处理强度，我们对来自 FEN 的动态缩放因子中 x_m 的幅值进行调整，具体如下：

$$\mathcal{P}(x) = f_d \times x_m + x. \tag{3}$$

根据前文分析，该因子与处理强度同义，并且与输入视频的时空特征密切相关。基于这一点，我们使用第 2.1 节所述、由特征导出的因子 f_d 来控制处理强度。

2.3. 动态量化级别

量化因子是预处理器训练过程中的关键超参数。较高的量化级别会使预处理器感知到更显著的量化失真，进而提高训练所得预处理器的处理强度，反之亦然。在此背景下，我们的目标是根据输入视频的质量调整量化因子。为此，我们提出一种动态量化级别（DQL）方案，通过动态调整量化

因子 f_q，在训练过程中调整编解码器模拟器的量化级别。如第 2.1 节所述，先以特定比特率进行快速编码，以获取输入视频的帧级 QP，然后将 QP 值裁剪到范围 [1,50]，作为 f_q 的值，并将其应用于编解码器模拟器，即 $f_q = clip(QP)$。

2.4. 动态 Lambda 权衡

我们的优化损失函数为率失真损失，表示如下：

$$L = D + \lambda \times R, \tag{4}$$

其中，D 是源视频 x 与压缩后的预处理视频 $\mathcal{C}$（$\mathcal{P}$（x））之间的误差，R 是用于编码预处理视频的比特数。值得注意的是，λ至关重要，因为它表示比特率与失真之间的权衡。以往基于深度学习的预处理或压缩工作 [7, 14] 通常使用固定的 λ，这使得神经网络难以处理极端情况。为解决这一问题，我们提出了动态 Lambda 权衡（DlamT）策略。具体而言，受 [2, 3], 启发，我们将 λ 和 QP 的对数结果建模为固定的正相关线性映射，即 $\log_{10}(\lambda_{adpt}) = k \times QP + b$。根据大量实验结果，我们设置 $k = 0.12$ 和 $b = -8$。在此设置下，我们将自适应 Lambda λ_{adpt} 值限制在（1e-8，1e-2], ）范围内，该范围对应于帧级 QP，其范围为（0, 50]。需要注意的是，λ_{adpt} 值的范围取决于所选 D 和 R 的取值。在我们的实验中，我们采用 MS-SSIM 损失 [15] 作为失真 D 的度量，并使用 Balle 等人的可学习的因子化先验熵模型 [16, 17] 来计算 R。最终的率失真损失表示为：

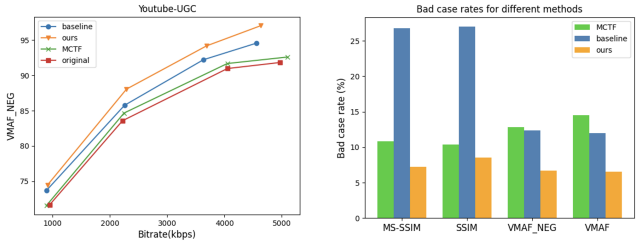
$$L = D(\mathcal{C}(\mathcal{P}(x)), x) + \lambda_{adapt} \times R. \tag{5}$$

3. 实验

3.1. 实现

为了让训练数据更好地代表 UGC 领域，我们从 Youtube-8M [18] 生成训练数据集，其中包含 20,000+ 个子序列，每个子序列由三帧组成。对于测试数据集，我们采用目前公开可用、最具代表性的学术 UGC 数据集 YouTube-UGC。测试时，我们使用 YouTube-UGC 的 1080p 子集中来自 13 个类别的 266 个视频。该数据集具有复杂的时空复杂度分布 [10],，足以代表 UGC 场景。

由于没有可用的开源深度预处理实现，我们参考 [7] 构建了基线。对于编解码器模拟器，我们根据 [7] 中的描述以及 [17]提供的框架实现了一个版本。如图1(b)所示，我们的基线取得了



(a) (b)

图 4：(a) VVC 在 Youtube-UGC 上的 VMAF NEG 率失真曲线示意图；(b) VVC 在 Youtube-UGC 数据集上不同方法的坏例率。

在 CTC 数据集上的性能与论文报告的结果相当。对于预分析的实现，我们使用 Sobel 滤波器提取 SI/TI 特征 [12]，并采用 x264 快速编码预设，以 1,500 kbps 的恒定比特率进行编码。整个流程使用 PyTorch 深度学习框架实现，并在使用 4 个 NVIDIA A100-SXM-80GB GPU 训练 20 个 epoch 后进行测试。

在测试方面，我们用标准编解码器替代编解码器模拟器，以获得率失真结果。对于标准编解码器，我们使用 VVC（vvenc）[19], 、H.265（x265）[2] 和 H.264（x264）[1] 的开源版本，通过预设 中等模式，并采用广泛使用的目标比特率 [1000, 2500, 4000, 5000] kbps 进行恒定比特率（CBR）编码。我们使用比约恩特加德增量比特率（BDBR）[9] 来衡量率失真性能，其中锚点是使用标准编解码器对未经预处理的视频进行压缩。

3.2. 实验结果

为评估所提出方法的整体性能，我们进行了率失真性能测试，将该方法与基线方法以及 VVC 的一款现有内部工具 MCTF 进行比较。所有这些方法均与直接对源视频进行编码（不进行任何预处理或预滤波）的结果作比较。评估指标方面，我们采用了四种广泛使用的感知度量：多尺度结构相似性、结构相似性、VMAF NEG 和 VMAF。率失真结果见表1；总体而言，我们的方法大幅优于其他方法。相应地，图 4(a) 展示了 VMAF NEG 的率失真曲线示例。对于 LyricVideo 和 How-to 等时空复杂度较低的类别，我们提出的方法在多尺度结构相似性和结构相似性指标上表现出显著提升。我们认为，这是因为所提出的 TDP 能够有效感知此类视频独特的时空特征，并据此进行最佳调整，从而相较于基线获得显著的性能增益。值得注意的是，在某些类别中，我们的方法在 VMAF 指标上的增益不如基线。这可能是因为预处理算法对 VMAF 指标存在轻微的负面影响 [20]。我们的方法在其他指标上的表现则更有力地证明了其优越性。

表 1：使用 VVC 时，YouTube 用户生成内容数据集上的率失真性能（BDBR%）。

BDBR	MCTF [8]				基线				Ours			
类别	MS-SSIM	SSIM	VMAF_NEG	VMAF	MS-SSIM	SSIM	VMAF_NEG	VMAF	MS-SSIM	SSIM	VMAF_NEG	VMAF
动画	-2.57	-2.8	-2.12	-2.88	-2.77	-2.29	-6.91	**-12.65**	**-4.11**	**-8.47**	**-8.24**	-12.56
翻唱歌曲	-1.87	-1.89	-1.69	-1.7	-1.1	-2.31	-4.48	-7.41	**-2.49**	**-5.99**	**-7.33**	**-10.24**
游戏	-2.29	-2.64	-2.33	-2.3	-5.93	-8.07	-5.41	-11.52	**-9.17**	**-13.15**	**-10.58**	**-14.95**
教程	-0.47	-0.48	-0.42	-0.35	8.05	7.46	6.76	-4.85	**-0.71**	**-3.57**	**-2.56**	**-10.85**
讲座	**-4.7**	-4.04	**-5.0**	-6.67	-2.16	-3.84	-0.97	**-12.21**	-3.41	**-13.9**	-4.15	-9.68
现场音乐	-1.7	-2.42	-1.30	-1.36	4.88	10.69	-4.17	-6.74	**-6.42**	**-8.62**	**-9.46**	**-11.07**
LyricVideo	**-0.4**	-0.45	-0.41	-0.37	12.43	8.53	-0.81	**-19.73**	-0.13	**-1.24**	**-3.14**	-6.8
音乐视频	-1.86	-1.93	-1.39	-1.3	-1.04	-0.17	-2.82	-5.49	**-4.64**	**-5.44**	**-5.08**	**-7.16**
新闻片段	-2.11	-2.1	-2.06	-2.01	-0.45	-4.15	-8.33	-20.18	**-3.22**	**-9.68**	**-11.62**	**-20.38**
竖屏视频	-0.48	-0.48	-0.34	-0.26	5.66	5.36	-1.2	**-12.52**	**-0.66**	**-2.26**	**-3.72**	-5.85
体育	-1.3	-1.32	-1.34	-1.29	-3.21	-5.18	-6.3	-9.41	**-3.81**	**-8.24**	**-8.52**	**-10.27**
电视片段	-1.22	-1.21	-1.1	-0.95	-0.33	-2.32	-6.5	-13.99	**-2.87**	**-6.75**	**-8.48**	**-14.89**
Vlog	-2.75	-2.85	-1.96	-1.82	-2.09	-1.32	-4.49	-7.0	**-4.38**	**-5.42**	**-6.0**	**-7.64**
Avg	-1.88	-1.95	-1.75	-1.86	-0.21	-1.03	-3.79	-10.61	**-4.07**	**-7.78**	**-7.14**	**-12.03**

表 2：H.264 和 H.265 的率失真性能。

编解码器	方法	多尺度结构相似性	SSIM	VMAF NEG	VMAF
H.264	基线	-0.76	-2.8	-6.03	-11.5
	Ours	-3.08	-8.46	-7.84	-12.84
H.265	基线	-1.21	-2.29	-5.8	-9.64
	Ours	-3.98	-8.09	-8.11	-12.35

表3：不同控制方案的消融研究。

DPI	DQL	DlamT	多尺度结构相似性	结构相似性	VMAF	负向增强增益	VMAF
✓	×	×	-0.28	-1.77	-4.01		-10.25
×	✓	×	-1.26	-2.45	-4.55		-10.39
×	×	✓	-0.97	-2.89	-5.64		-10.2
✓	✓	✓	-4.07	-7.78	-7.14		-12.03

我们计算了每种评估指标对应的不良案例比例（BDBR >0）。如图 4(b) 所示，我们的方法显著优于 MCTF 和基线方法，同时不良案例比例也明显降低。表 2 还列出了我们的方法在较早的标准编解码器 H.264(x264) 和 H.265(x265) [21], 下取得的优异性能结果。

此外，我们开展了测试，以评估所提出的 TDP 机制中各个部分的独立影响。表3所示结果表明，尽管每个部分单独都能带来一定益处，但显然，将它们结合起来才能充分发挥其潜在性能。

3.3. 进一步讨论

我们开展了额外实验，以分析源视频特征、 f_q与率失真性能之间的关系。我们参考 [10] 中的方法来计算面向编码的时空复杂度。我们将归一化的时空复杂度划分为十组，每组宽度为 0.1。对于每组视频，我们找出相应的 f_q值，并分别评估其率失真性能。

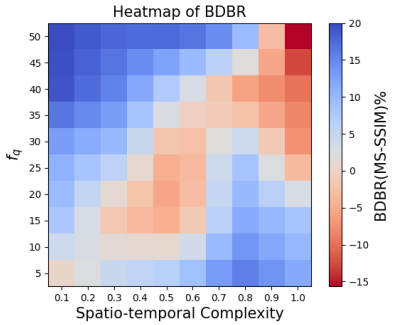


图5：不同时间复杂度与 f_q 下的 BD-Rate 性能分布热力图。

结果如图 5 所示。显然，时空复杂度较低的视频往往在较小的 f_q 值下获得更好的性能，反之亦然。这也解释了为什么采用固定强度的基线方法不可避免地会在时空复杂度较高的视频中产生不良案例。这一观察结果与我们之前关于处理器强度的分析一致。

4. 结论

本文介绍了一种用于用户生成内容视频压缩的三动态预处理框架。该框架能够感知视频序列的时空复杂度，并借助动态处理强度、动态训练量化级别和动态损失权衡，有效处理用户生成内容数据。实验结果表明，我们提出的方法能够

在多种常用感知度量上始终优于基准方法。

5. 参考文献

[1] ThomasWiegand、GaryJSullivan、Gisle Bjontegaard 和 Ajay Luthra，“H.264/AVC 视频编码标准概述”，*IEEE*视频技术电路与系统汇刊，第 13 卷，第 7 期，第 560–576 页，2003 年。

[2] Gary J Sullivan、Jens-Rainer Ohm、Woo-Jin Han 和 Thomas Wiegand，“高效视频编码（HEVC）标准概述”，*IEEE*视频技术电路与系统汇刊，第 22 卷，第 12 期，第 1649–1668 页，2012 年。

[3] Benjamin Bross、Jianle Chen、Jens-Rainer Ohm、Gary J Sullivan 和 Ye-Kui Wang，“AVC 之后国际视频编码标准化的发展及多功能视频编码（VVC）概述”，*IEEE* 会议录，第 109 卷，第 9 期，第 1463–1493 页，2021 年。

[4] Christos G Bampis、Zhi Li 和 Alan C Bovik，《用于全参考视频质量评估的时空特征整合与模型融合》，*IEEE*视频技术电路与系统汇刊，第 29 卷，第 8 期，第 2256–2270 页，2018 年。

[5]Zhengyi Luo、Chen Zhu、Yan Huang、Rong Xie、Li Song 和 C-C Jay Kuo，《基于分段指标耦合的 VMAF 导向感知编码》，*IEEE*图像处理汇刊，第 30 卷，第 5109–5121 页，2021 年。

[6] Madhukar Bhat、Jean-Marc Thiesse 和 Patrick Le Callet，《基于人类视觉系统的视频编码感知预处理》，载于 *2019* 年第 *27* 届欧洲信号处理会议（*EUSIPCO*）。IEEE，2019 年，第 1–5 页。

[7] Aaron Chadha 和 Yiannis Andreopoulos，“用于视频编码的深度感知预处理”，载于*IEEE/CVF*计算机视觉与模式识别会议，2021年，第14852–14861页。

[8] Jack Enhorn、Rickard Sj¨oberg 和 Per Wennersten，“一种基于双边滤波的视频编码时间预滤波器”，载于*2020 IEEE*国际图像处理会议（*ICIP*）。IEEE，2020年，第1161–1165页。

[9] G. Bjøntegaard，“率失真曲线之间平均峰值信噪比差异的计算”，文档 VCEG-M33，ITU-TSG16/Q6（VCEG），2001年4月。

[10] YilinWang、Sasi Inguva 和 Balu Adsumilli，“用于视频压缩研究的 YouTube 用户生成内容数据集”，载于*2019 IEEE*第*21* 届多媒体信号处理国际研讨会（*MMSP*）。IEEE，2019年，第1–5页。

[11]xiph.org，“Xiph.org视频测试媒体 [derf 的合集],” https://media.xiph.org/video/derf/。

[12] 电话装置与本地线路，《多媒体应用的主观视频质量评估方法》，网络，第 910 卷，第 37 期，第 5 页，1999 年。

[13] Loren Merritt 和 Rahul Vanam，《x264：一种高性能 H.264/AVC 编码器》，在线*] http://neuron2.net/library/avc/overview-x264 v8 5.pdf*，2006。

[14] Guo Lu、Wanli Ouyang、Dong Xu、Xiaoyun Zhang、Chunlei Cai 和 Zhiyong Gao，《DVC：一种端到端深度视频压缩框架》，载于*IEEE/CVF*计算机视觉与模式识别会议录，2019年，第11006–11015页。

[15] Zhou Wang、Eero P Simoncelli 和 Alan C Bovik，《用于图像质量评估的多尺度结构相似性》，载于第三十七届阿西洛马信号、系统与计算会议，*2003*。IEEE，2003年，第2卷，第1398–1402页。

[16] Johannes Ball´e、David Minnen、Saurabh Singh、SungJin Hwang 和 Nick Johnston，《采用尺度超先验的变分图像压缩》，*arXiv* 预印本 *arXiv:1802.01436*，2018 年。

[17] Johannes Ball´e、Valero Laparra 和 Eero P Simoncelli，《端到端优化图像压缩》，*arXiv* 预印本 *arXiv:1611.01704*，2016 年。

[18] Sami Abu-El-Haija、Nisarg Kothari、Joonseok Lee、Paul Natsev、George Toderici、Balakrishnan Varadarajan 和 Sudheendra Vijayanarasimhan，《YouTube-8M：大型视频分类基准》，*arXiv* 预印本 *arXiv:1609.08675*，2016 年。

[19]Adam Wieckowski、Jens Brandenburg、Tobias Hinz、Christian Bartnik、Valeri George、Gabriel Hege、Christian Helmrich、Anastasia Henkel、Christian Lehmann、Christian Stoffers 等，《VVEnc：一种开放且经过优化的 VVC 编码器实现》，载于*2021 IEEE* 国际多媒体与博览会会议研讨会（*ICMEW*）。IEEE，2021 年，第 1–2 页。

[20] Maksim Siniukov、Anastasia Antsiferova、D mitriy Ku- likov 和 Dmitriy Vatolin，“破解 VMAF 和 VMAF NEG：对不同预处理方法的脆弱性”，载于*2021*年第四届人工智能与云计算会议论文集，2021 年，第 89–96 页。

[21] Qiang Hu、Xiaoyun Zhang、Zhiyong Gao 和 Jun Sun，“x265编码器的分析与优化”，载于*2014*年*IEEE*视觉通信与图像处理会议。IEEE，2014 年，第 502–505 页。