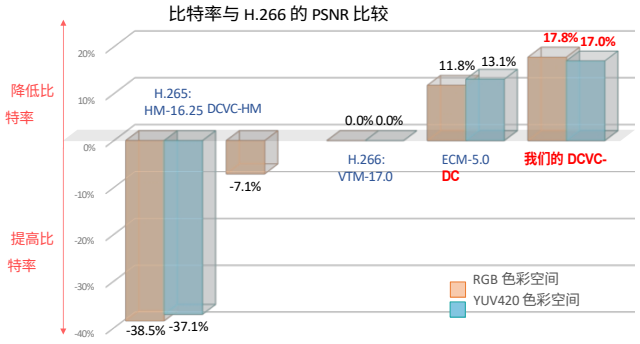
不同语境下的神经视频压缩

李家豪、李斌、卢彦 微软亚洲
研究院
{li.jiahao, libin, yanlu}@microsoft.com

摘要

对于任何视频编解码器来说，编码效率的高低很大程度上取决于当前要编码的信号能否从之前重建的信号中找到相关的上下文。传统的编解码器通过验证更多的上下文带来了可观的编码增益，但却非常耗时。然而，对于新兴的神经视频编解码器（NVC）来说，其上下文仍然有限，导致压缩率较低。为了提高 NVC 的性能，本文建议从时间和空间两个维度增加上下文的多样性。首先，我们引导模型学习分层质量路径，从而提高视频质量。

这丰富了长期且高质量的时间上下文。此外，为了挖掘基于光流的编码框架的潜力，我们引入了基于组的偏移多样性，其中提出了跨组交互，以便更好地挖掘上下文。此外，本文还采用了基于四叉树的分区，以增加并行编码潜代表时的空间上下文多样性。实验表明，我们的编解码器比之前的 SOTA NVC 节省了 23.5% 的比特率。更好的是，我们的编解码器在 RGB 和 YUV420 两种色彩空间中的 PSNR 都超过了正在开发中的下一代传统编解码器/ECM。编码见 https://github.com/microsoft/DCVC。



1. 导言

视频编解码器的原理是，对于当前要编码的信号，编解码器会从以前的重构信号中找到相关内容（如作为上下文的各种预测），以减少时空冗余。相关上下文越多，就能节省更高的比特率。

回顾传统编解码器的发展历程（从 1988 年的 H.261 [17] 到 2020 年的 H.266 [7]），我们会发现编码增益主要来自不断扩展的编码模式，每种模式都采用特定的方

式提取和利用上下文。For example, the numbers of intra prediction directions [42] in H.264, H.265, H.266 分别为 9、35 和 65。如此多的模式可以

图 1. UVG、MCL-JCV 和 HEVC 的平均结果

数据集。所有传统编解码器都使用其最佳压缩比配置。DCVC-HEM [29] 是之前的 SOTA NVC，只发布了 RGB 色彩空间的模型。

在提取不同语境以减少冗余的同时，也带来了巨大的复杂性，因为要使用速率失真优化（RDO）来搜索最佳模式。对一个 1080p 帧进行编码，开发不足的 ECM（下一代传统编解码器的原型）最多需要半个小时 [49]。尽管一些基于 DL 的方法 [24,51,52] 提出了加速传统编解码器的方法，但其复杂度仍然很高。相比之下，神经视频编解码器（NVC）改变了上下文的牵引和利用方式，从手工制作解码标志变为自动学习。

NVC 的主流框架可分为基于残差编码的编解码器 [1, 13, 31, 32, 34, 36, 47, 59, 61] 和基于条件编码的编解码器。[21, 27-29, 33, 38, 50].残差编码明确使用预测帧作为上下文，上下文利用仅限于使用减法去除冗余。相比之下，条件编码会隐式地学习特征域上下文。高维度上下文可以携带更丰富的信息，以促进编码、解码和熵建模。

然而，对于大多数 NVC 而言，情境提取和利用的方式仍然有限，例如，只能使用光流来探索时间相关性。这使得 NVC 很容易受到参数不确定性的影响 [12, 16, 37]，或陷入局部最优状态 [25]。一种解决方案是在 NVC 中加入类似传统编解码器的编码模式 [25]。但这会带来

在使用 RDO 时，计算复杂度很高。因此问题来了：如何更好地学习和使用上下文，同时降低计算成本？

为此，我们基于 DCVC（深度上下文视频压缩）[28] 框架及其后续工作 DCVC-HEM [29]，提出了一种新模型 DCVC-DC，它能有效利用 "多样化上下文"（Diverse Contexts）来进一步提高压缩率。首先，我们引导 DCVC-DC 学习各帧的分层质量模式。在训练过程中，有了这种指导，就能在图像传播过程中隐含地学习到对重建后续帧至关重要的长期但高质量的上下文。这有助于进一步利用视频中的长程时间相关性，有效缓解大多数 NVC 存在的质量下降问题。此外，我们还采用了偏移分集技术[8]来加强基于光流的编解码器，多个偏移可以减少复杂或大型运动的翘曲误差。特别是，受传统编解码器中加权预测的启发，我们将偏移量分为若干组，并提出了跨组融合的方法，以改善对时间上下文的挖掘。

除了时间维度之外，本文还提出了在编码潜表征时增加空间上下文多样性的方法。基于最近的棋盘模型 [19] 和双空间模型 [29, 56]，我们设计了一个基于四叉树的分区来改进分布估计。与[19、29]相比，相关建模的类型更加多样化，因此该模型有更大的机会找到更多相关上下文。

值得注意的是，我们的所有设计都是并行高效的。为了进一步降低计算成本，我们还采用了深度可分离卷积（depth-wise separable convolution）[10]，并为不同分辨率的特征分配了不相等的通道数。实验表明，我们的 DCVC-DC 比以前的 SOTA NVC 效率更高，并将压缩比推向了一个新的高度。与 DCVC- HEM [29] 相比，比特率节省了 23.5%，而 MAC（乘积操作）减少了 19.4%。更妙的是，如图 1 所示，除了 H.266-VTM 17.0，我们的编解码器在 RGB 和 YUV420 两种色彩空间中的表现也优于 ECM-5.0（使用其用于低延迟编码的最佳压缩比配置）。据我们所知，这是第一个能取得如此成就的 NVC。总之，我们的贡献如下

- 我们建议有效地增加语境多样性，以提高 NVC。多样化的语境互为补充，有更大的机会为减少冗余提供良好的参考。
- 从时间维度出发，我们引导模型提取高质量上下文，以缓解质量下降问题。此外，为了更好地挖掘时态上下文，我们还设计了基于组的偏移量。

- 从空间维度出发，我们采用基于四元组的分区进行潜在表示。这为更好地进行熵编码提供了二维空间背景。
- 与之前的 SOTA NVC 相比，我们的 DCVC-DC 节省了 23.5% 的比特率。特别是，我们的 DCVC-DC 在 RGB 和 YUV420 两种色彩空间中都超过了最好的传统编解码器 ECM，这对 NVC 的发展具有重要意义。

2. 相关工作

2.1. 神经图像压缩

大多数神经图像编解码器都基于超先验[4]，即首先使用一些比特为熵编码提供基本上下文。然后，自动回归先验[40] 使用邻近上下文来捕捉空间相关性。最近的研究[18, 26, 44, 45]提出提取全局或长距离上下文来进一步提高性能。这些研究表明，更多样化的上下文为神经图像编码器带来了巨大的编码增益。

2.2. 神经视频压缩

近年来，无损压缩技术也得到了蓬勃发展。开创性的 DVC [34] 遵循传统编解码器。它使用光流网络生成预测帧，然后对其与当前帧的残差进行编码。随后的许多研究也采用了这种基于残差编码的框架，并对其中的模块进行了改进。例如，[31, 43, 47] 提出了运动预测来进一步减少冗余。尺度空间中的运动流估计 [1] 是为处理复杂运动而设计的。Yang 等人[61]利用递归自动编码器提高编码效率。

残差编码以像素域为上下文明确生成预测帧，仅使用减法去除冗余。相比之下，条件编码具有更强的可扩展性。条件的定义、学习和使用方式都可以灵活设计。文献 [33, 38] 设计了时间条件熵模型。[27] 使用条件编码对前景内容进行编码。Li 等人提出了 DCVC [28]，通过学习特征域 con- texts 来增加上下文容量。随后，DCVC-TCM [50] 采用特征传播来提高性能。最近，DCVC-HEM [29] 设计了利用空间和时间上下文的混合熵模型。

然而，与传统编解码器相比，大多数无损压缩技术的编码模式仍有局限性。例如，传统编解码器采用平移/非线性运动模型、几何分割、双预测等模式来处理不同的时间背景[7]。相比之下，现有的 NVC 通常只依赖单一光流，容易受到模型参数中认识不确定性的影响 [12, 16, 37]。最近的研究 [25] 也表明，当编码模式为

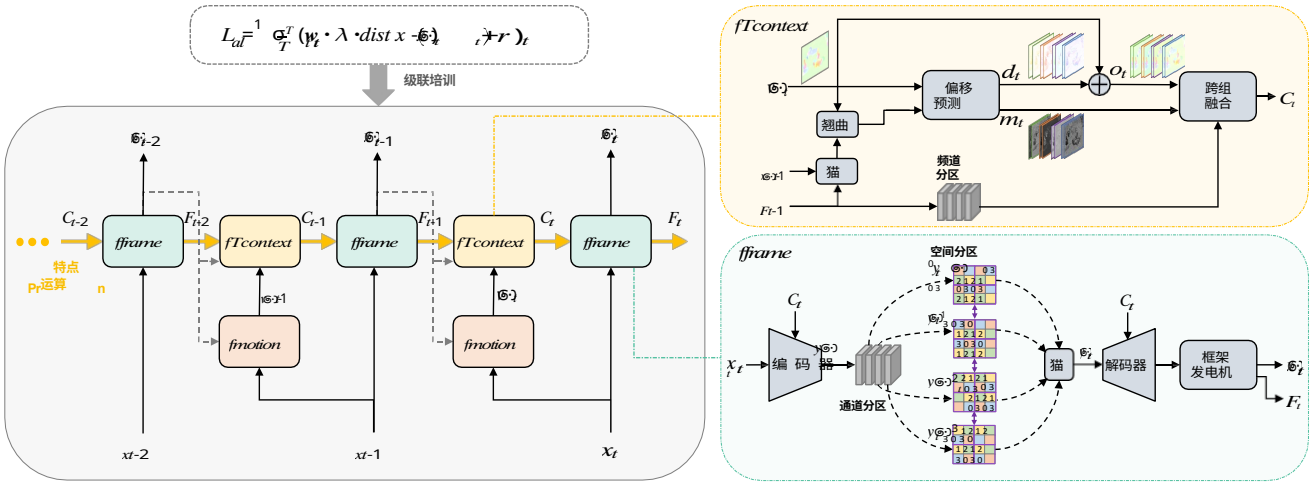


图 2.DCVC-DC 框架概览。x_t 和 $\hat{x}_t$ 分别为输入帧和重建帧。C_t 是学习到的时间上下文，作为对 x_t 进行编码的条件。F_t 是用于下一帧的已传播但未处理的特征。$dist(\cdot)$ 是失真函数，λ 和 w_t 分别是全局权重和帧级权重。$\hat{y}_t$ 空间分区中的数字代表编码顺序索引。

有限。因此，[25] 设计了许多额外的模式，如过渡编解码器，并使用 RDO 搜索最佳模式。这种方法不可避免地会带来巨大的计算成本。相比之下，我们的模型在利用高质量的时间上下文时没有额外的推理成本。我们的偏移多样性和四叉树分区在提供多样性上下文时也是省时的去符号。

3. 建议的方法

3.1. 概述

为了达到更高的压缩比，我们的编解码器采用了更灵活的条件编码而非残差编码。图 2 展示了 DCVC-DC 的框架。值得注意的是，我们的 DCVC-DC 是为低延迟编码而设计的，因为它可以应用于更多场景，如实时通信。如图 2 所示，为了对帧索引为 t 的每一帧 x_t 进行编码，我们的编码流水线包含三个核心步骤：f_{motion}、$f_{T\ context}$ 和 f_{frame}。首先，f_{motion} 使用光流网络估算出运动矢量（MV）v_t，然后将 v_t 编码并解码为 $\hat{v}_t$。其次，基于 $\hat{v}_t$ 和上一帧的传播特征 F_{t-1}，$f_{T\ 上下文}$提取运动对齐的时间上下文特征 C_t。最后，基于 C_t，f_{frame} 将 x_t 编码为量化的潜在代表 $\hat{y}_t$。经过熵编码后，输出帧 $\hat{x}_t$ 通过解码器和帧生成器被重构。与此同时，F_t 也会生成并传播到下一帧。值得注意的是，我们的 DCVC-DC 基于 DCVC-HEM [29]。与 DCVC-HEM 相比，本方案重新设计了模块，从时间（第 3.2 和 3.3 节）和空间（第 3.4 节）两方面利用多样化语境（Diverse Contexts）。

尺寸。

3.2. 分层质量结构

传统编解码器广泛采用分层质量结构，将帧分配到不同的层，然后使用不同的 QPs（量化参数）。这种设计源于可扩展视频编码[48]，但也从两个方面提高了一般低延迟编码的性能。其一，如图 3 所示，周期性地提高质量可减轻错误传播。在间隔预测期间，高质量的参考帧能让编解码器在运动估计时找到更准确的 MV。同时，运动补偿预判定也是高质量的，从而使预测误差更小。另一方面，在多参考帧选择和加权预测机制的作用下，来自最近参考帧和远距离高质量参考帧的预测组合更加合理。文献[30]对帧质量和参考帧选择的多种设置进行了研究，得出的结论是，同时参考最近帧和较远高质量帧的分层质量结构性能最佳。

受传统编解码器成功经验的启发，我们正在思考能否为 NVC 配备分层质量结构，让 NVC 也能享受到这种好处。考虑到最近的神经编解码器[11,29]也支持单一模型中的可变比特率，一种直接的解决方案是沿用传统编解码器，在 NVC 推理过程中直接分配分层 QPs。然而，与传统编解码器使用定义明确的规则来执行运动估计和运动补偿（MEMC）不同，NVC 使用神经网络进行运动估计和运动补偿。

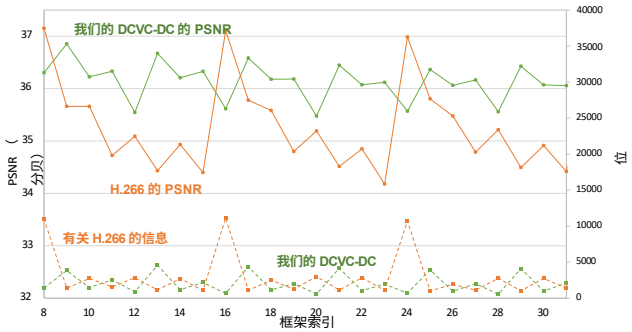


图 3.H.266-VTM 17.0 和我们的 NVC 中的分层质量结构。该示例来自 HEVC D 数据集中的 *BasketballPass* 视频。H.266 的平均 bpp（每像素比特）和 PSNR 分别为 0.056 和 35.10。我们的 DCVC-DC 分别为 0.045 和 36.13。

而 MEMC 通常属于特征域。对于 NVC 来说，这种设计的优势在于它是自动学习的，有更大的潜力实现更好的性能。其不足之处在于，它对分布外质量模式的鲁棒性和泛化能力较弱。因此，如果我们像文献[22]那样在测试过程中直接向 NVC 输入分层 QPs，它可能无法很好地适应分层质量模式，MEMC 可能会获得次优性能。为此，我们建议在训练过程中引导 NVC 学习分层质量模式。具体来说，我们在速率-失真损失中为每帧添加一个权重 w_t，如图 2 所示。w_t 的设置遵循层次结构。在这种分层失真损耗的作用下，高质量的输出帧 $\hat{x}_t$ 和包含许多高频细节的特征 F_t 都会周期性地产生。它们非常有助于提高 MEMC 的效率，进而缓解许多其他 NVC 所面临的误差传播问题。此外，通过对多个帧进行级联训练，还能形成特征提取链。对重建后续帧至关重要的高质量上下文会被自动学习并长期保持。因此，在对 x_t 进行编码时，F_{t-1} 不仅包含从 x_{t-1} 中提取的短期文本，而且还提供了从许多先前帧中不断更新的长期高质量语境。这种多样化的 F_{t-1} 有助于进一步利用许多帧之间的时间相关性，从而提高压缩比。图 3 还显示了我们的 NVC 的质量模式。我们可以看到，我们的 DCVC-DC 实现了更好的质量。平均质量，而比特成本却低于 H.266。

3.3. 基于组的抵消多样性

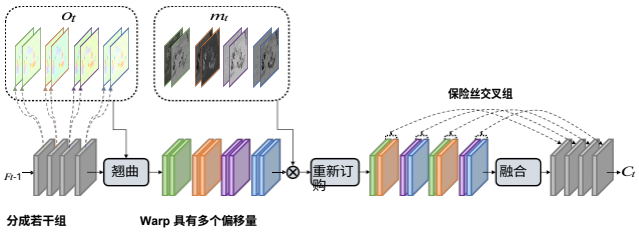


图 4 跨组融合模块跨组融合模块。在本例中，组数 G 为 4，每组的偏移量 N 为 2。

在大多数现有的 NVC 中，$f_{T_上下文}$仅是对单个运动进行翘曲操作。这种基于单一运动的对齐方式对复杂运动或遮挡不具有鲁棒性。研究[9, 54, 58]表明，由于每个位置都有多个离场集来捕捉时间上的对应关系，因此可变形配准在视频还原方面能获得更好的结果。因此，最近它也被应用于 NVC [23]。然而，可变形配准的训练并不稳定，偏移量的溢出会降低其性能[8, 58]。此外，偏移量受限于可变形卷积核的大小。因此，本文采用了更为灵活的设计，即偏移分集 [8]。同时，解码后的 MV 用作基本偏移量，以稳定训练 [9]。

如图 2 所示，我们的 $f_{T_上下文}$由两个核心子模块组成：偏移预测和跨组融合。首先，偏移预测使用解码后的 MV $\hat{v}_t$ 来预测残余偏移 d_t，其中 $\hat{x}_{t-1}$ 和 F_{t-1} 也被扭曲并作为辅助信息输入。d_t 加上基本偏移量 $\hat{v}_t$，得到最终偏移量 o_t。同时，偏移集预测也会生成调制掩码 m_t，它可以被视为反映偏移量置信度的注意力。F_{t-1} 沿信道维度被分为 G 组，每组有独立的 N 个偏移量。因此，总共学习了 $G \times N$ 个偏移。不同的偏移量相互补充，有助于编解码器应对复杂的运动和遮挡。

此外，受改进 CNN 主干网信息流的通道洗牌操作（channel shuffle oper- ation）[62]的启发，我们定制了一种组级交互机制，以进一步挖掘偏移发散性在 NVC 中的潜力。具体来说，在对每个组进行多重偏移翘曲并应用相应的掩码后，我们将在融合前对所有组进行重新排序，如图 4 所示。如果使用 g^j 表示第 i 个组

翘曲，其 j 第 - 个偏移量，即重新排序前的特征值

由于各帧之间的运动不同，直接

是 $g^0, \ldots, g^{N-1}, g^0, \ldots, g^{N-1}$ ，- - - - ，g^0 ，...，g^{N-1} 、

使用未经处理的 F_{t-1} 而不进行运动校准的结果是编解码器很难捕捉到时间上的对应关系。因此，我们沿用现有的 NVC，使用光流网络通过 $f_{T\,context}(F_{t-1}, \hat{V}_t)$ 提取运动对齐的时间上下文 C_t ，其中 $\hat{V}_t$ 是解码后的 MV。然而、

然后，我们将它们重新排序为 $g^0, \ddot{0}, g^0$ G-1 $g^1, \ldots, g^1$ G-1 ，- - - 0 $, g^{N-1}, \ldots,$ G-1 g^{N-1} 、 其中而不是以分组顺序为主。接下来的融合操作将把每 N 个连续的组融合成一个组。因此，在这一过程中，组的重新排序可以实现更多的跨组互动，而不会增加

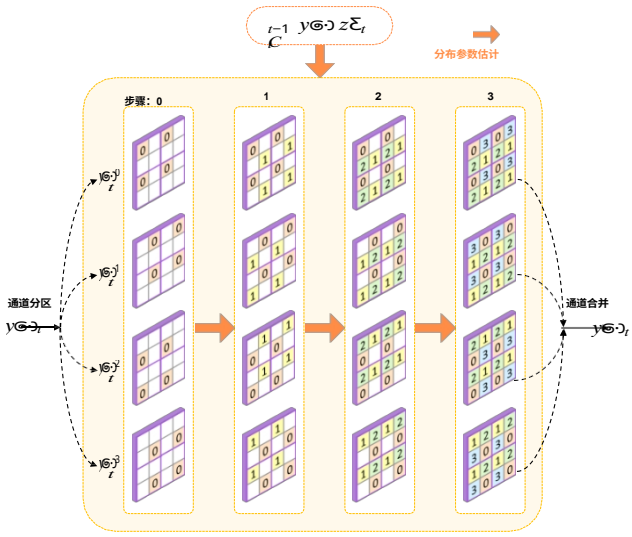


图 5.四叉树分区熵编码。数字表示编码顺序指数。在 4 个编码步骤中，上一帧的 $\hat{\mathbf{y}}_{t-1}$、超先验 $\hat{z}_t$ 和时空背景 c_t 也被用于熵建模。

复杂性。这种设计与传统编解码器中不同参考帧的加权预测具有相似的优势。通过跨组融合，从不同组中提取时空背景的组合更加多样化，进一步提高了偏移多样性的效果。

3.4. 基于四叉树分区的熵编码

如图 2 所示，在通过偏移量发散模块获取时间上下文 C_t 后，输入帧 x_t 将被编码和量化为 $\hat{Y}_t$，条件是 C_t。我们需要估计 $\hat{Y}_t$ 的概率质量函数（PMF），以便对其进行算术编码。在此过程中，如何建立一个准确的熵模型来估计 $\hat{Y}_t$ 的 PMF 对压缩效率至关重要。

许多神经编解码器采用自动回归模型 [40] 作为熵模型。然而，它严重降低了编码速度。相比之下，棋盘模型[19] 建议先对 $\hat{Y}_t$ 的偶数位置进行编码，然后并行使用它们来预测奇数位置的 PMF。最近，双空间模型 [29] 通过利用信道维度的相关性对其进行了改进。然而，与自动回归模型相比，[19, 29] 中用于熵建模的邻域仍然有限。因此，受文献[41, 46] 的启发，本文通过四叉树分区提出了一种更精细的编码方式，即利用不同的空间上下文来改进熵建模。如图 5 所示，我们首先沿信道维度将 $\hat{Y}_t$ 分成四组。然后，在空间维度上将每组划分为不重叠的 2×2 补丁。

整个熵编码分为四个步骤、

每一步都对图 5 中相应索引的不同位置进行编码。在第 0 步，同时对所有补丁中索引为 0 的位置进行编码。值得注意的是，在四组中，索引为 0 的位置彼此不同。因此，每个空间位置都有一个第四通道（即一组）编码。在随后的第 1、第 2 和第 3 步中，前几步编码的所有位置都将用于预测本步编码位置的 PMF，而且每一步都会对不同组的不同空间位置进行编码。

在此过程中，会使用更多不同的邻域。如果考虑一个位置的 8 个空间邻域，自动回归模型 [40] 在不考虑边界区域的情况下，每个位置使用 4 个（左、左上、上、右上）邻域。棋盘模式和双空间模式 [19,29] 在第 0 步和第 1 步分别使用 0 个和 4 个（左、上、右、下）邻域。相比之下，如图 5 所示，我们的 DCVC-DC 在四个步骤中分别使用了 0、4、4 和 8 个邻域。平均而言，DCVC-DC 的邻接数是 [19, 29] 的 2 倍，与自动回归模型相同。但是，我们的模型比自动回归模型更省时，因为每一步中的所有位置都可以并行编码。此外，我们的模型还利用了跨信道相关性，这一点与文献[29]相同，但做了改进。例如，在第三步，对于一个组的一个特定位置，同一位置的其他信道已经在之前的步骤中从不同的组中进行了编码，它们可以用作这一步熵建模的上下文。这有助于进一步减少冗余。总之，我们基于四叉树分区的解决方案使熵编码受益于更细粒度和多样化的上下文，充分挖掘了空间和信道维度的相关性。

3.5. 实施情况

我们的 DCVC-DC 基于 DCVC-HEM [29]，但重点利用了 "多样化上下文"（Diverse Contexts）来进一步提高性能。此外，为了在性能和复杂性之间取得更好的平衡，我们还做了以下改进。首先，考虑到深度可分离卷积（depthwise separable convolution）[10] 在减轻过拟合的同时还能降低计算成本，我们在基本区块设计中广泛使用深度可分离卷积来替代普通卷积。其次，我们对不同分辨率的特征采用不等通道数设置，将分辨率较高的特征分配给较小的通道数，以实现加速。第三，我们在编码器中将部分量化操作移至更高分辨率，这有助于实现更精确的位分配。编码和量化的协调也带来了压缩比的提高。第 4.3 节验证了这些结构优化的有效性，详细的网络工作结构见补充材料。

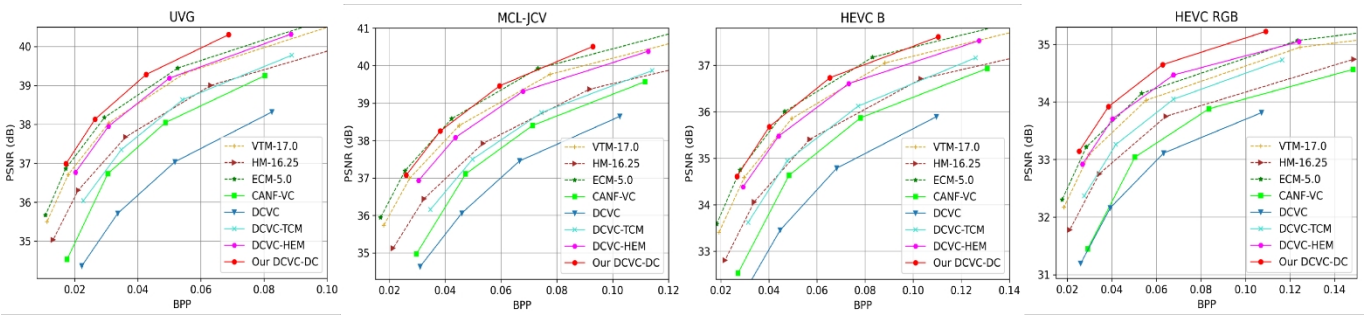


图 6.速率和失真曲线。对比采用 RGB 色彩空间，以 PSNR 测量。更多结果，包括相应的 MS-SSIM 曲线和在 YUV420 色彩空间中的对比，见补充材料。

表 1.在 RGB 色彩空间中使用 PSNR 测量的 BD-Rate (%) 对比。锚点为 VTM-17.0。

	UVG	MCL-JCV	HEVC B	HEVC C	HEVC D	HEVC E	HEVC RGB	平均
VTM-17.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
HM-16.25	36.4	41.5	38.8	36.0	33.7	44.0	39.4	38.5
ECM-5.0	-10.0	-12.2	-11.5	-13.4	-13.5	-10.9	-11.1	-11.8
CANF-VC [21]	73.0	70.8	64.4	76.2	63.1	118.0	79.9	77.9
DCVC [28]	166.1	121.6	123.2	143.2	98.0	266.1	113.4	147.4
DCVC-TCM [50]	44.1	51.0	40.2	66.3	37.0	82.7	24.4	49.4
DCVC-HEM [29]	1.1	8.6	5.1	22.2	2.4	20.5	-9.9	7.1
我们的 DCVC-DC	-19.1	-11.3	-12.0	-10.3	-26.1	-18.0	-27.6	-17.8

特别是，我们的 DCVC-DC 在 YUV420 色彩空间中的表现已经超过了 ECM。与传统编解码器在为不同色彩空间设计编码工具时需要进行大量手工修改不同，DCVC-DC 只需在为 RGB 训练的现有模型基础上进行简单的调整即可。在不改变网络结构的情况下，我们只需对 UV 进行上采样，即可使用统一的 RGB 输入接口。相应地，在获得重建帧后，我们会对 UV 进行下采样。此外，我们的 YUV420 模型只需在 RGB 模型的基础上进行简单的微调训练即可。

4. 实验结果

4.1. 实验设置

数据集。在训练中，我们沿用了大多数现有的 NVC，并使用 Vimeo-90k [60]。在测试 YUV420 视频时，我们使用了 HEVC B~E [6]、UVG [39] 和 MCL-JCV [57]。

它们的原始格式是 YUV420，因此在将其输入 NVC 之前无需进行任何更改。在测试 RGB 视频时，由于这些测试集没有 RGB 格式，现有的大多数 NVC 都使用 BT.601（FFmpeg 的默认设置）将它们从 YUV420 转换为 RGB。实际上，JPEG AI [2,3] 采用的是 BT.709，因为在视觉质量相似的情况下，使用 BT.709 可以获得更高的压缩比。因此，本文沿用 JPEG AI，在测试 RGB 时所有编解码器均采用 BT.709。值得注意的是，在 BT.601 和 BT.709 中，不同编解码器之间的相对比特率比较是相似的。补充说明

terials 显示了使用 BT.601 的结果。此外，在测试 RGB 视频时，我们参考了文献 [29, 50]，还测试了 HEVC RGB 数据集 [15]，由于 HEVC RGB 数据集本身就是 RGB 格式，因此没有改变格式。

测试条件我们沿用 [29, 50] 的方法，对每段视频进行 96 帧测试，并将帧内周期设为 32。使用低延迟编码设置，这与大多数现有作品[1, 28, 34]相同。BD-Rate [5] 用于测量压缩率，负数表示比特率节省，正数表示比特率增加。

我们的基准包括 HM [20] 和 VTM [55]，它们分别代表了最好的 H.265 和 H.266 编码器。特别是，我们还与 ECM [14] 进行了比较，后者是下一代传统编解码器的原型。在编解码器设置方面，我们遵循文献 [29, 50]，在测试 RGB 时进一步使用 10 位作为中间表示，这使得三种传统编解码器的压缩比更好。详细设置见补充材料。至于 NVC 基准，我们与最近的 SOTA 模型进行了比较，包括 CANF-VC [21]、DCVC [28]、DCVC-TCM [50] 和 DCVC-HEM [29]。

模型训练。我们采用多阶段训练方法 [29, 50]。我们的模型还支持单个模型中的可变比特率 [29]，因此在不同的优化步骤中使用不同的 λ 值。我们遵循 [29] 的方法，使用 4 个 λ 值（85、170、380、840）。但与 [29] 在损耗中使用恒定失真权重不同的是，本文建议对失真项的 w_t 进行分级权重设置（即λ值为 *0.5* 时，失真项的权重为 *0.5*）。

表 2.使用 MS-SSIM 测量的 RGB 色彩空间中的 BD 率（%）比较。锚点为 VTM-17.0。

	UVG	MCL-JCV	HEVC B	HEVC C	HEVC D	HEVC E	HEVC RGB	平均
VTM-17.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
HM-16.25	31.1	38.8	36.6	35.2	33.0	41.1	36.6	36.1
ECM-5.0	-9.1	-11.1	-10.2	-11.7	-11.0	-9.9	-9.8	-10.4
CANF-VC [21]	46.5	26.0	43.5	30.9	17.9	173.0	57.7	56.5
DCVC [28]	64.9	27.5	54.4	39.7	15.2	210.4	51.3	66.2
DCVC-TCM [50]	1.0	-10.8	-11.7	-15.2	-29.0	16.7	-22.2	-10.2
DCVC-HEM [29]	-25.2	-36.3	-38.0	-38.3	-48.1	-25.8	-43.6	-36.5
我们的 DCVC-DC	-32.6	-44.8	-47.8	-49.8	-58.2	-45.8	-54.4	-47.6

表 3.用 PSNR 测量的 YUV420 色彩空间的 BD-Rate (%) 对比。锚点为 VTM-17.0。

	UVG	MCL-JCV	HEVC B	HEVC C	HEVC D	HEVC E	平均
VTM-17.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
HM-16.25	36.7	42.5	39.2	33.3	30.0	40.7	37.1
ECM-5.0	-10.6	-13.7	-12.6	-14.7	-14.9	-12.1	-13.1
我们的 DCVC-DC	-17.8	-12.0	-10.8	-12.4	-28.5	-20.4	-17.0

整个损失的定义见图 2）。考虑到训练集 Vimeo-90k 每段视频只有 7 帧，我们参照传统的编解码器设置，将模式大小设为 4。连续 4 帧的 w_t 设置为（0.5, 1.2, 0.5, 0.9）。

4.2. 与以往 SOTA 方法的比较

RGB 色彩空间。表 1 和表 2 分别显示了以 PSNR 和 MS-SSIM 表示的 RGB 视频压缩率对比。从表 1 可以看出，我们的编解码器在每个数据集上都比 VTM 压缩率有显著提高，平均比特率节省了 17.8%。相比之下，其他神经编解码器仍然不如 VTM。如果使用 DCVC-HEM 作为锚，我们的平均比特率节省率为 23.5%。此外，我们的 DCVC-DC 也优于表 1 中的 ECM。如果使用 ECM 作为锚点，平均可节省 6.4% 的比特率。

图 6 显示了速率-失真曲线。从曲线可以看出，我们的 DCVC-DC 在较宽的比特率范围内实现了 SOTA 压缩比。当使用 MS-SSIM 作为质量指标时，我们的 DCVC-DC 有较大的改进。如表 2 所示，DCVC-DC 比 VTM 平均节省 47.6% 的比特率。相比之下，与 VTM 相比，ECM 的相应数字仅为 10.4%。

值得注意的是，表 1 和表 2 使用的是经过 BT.709 转换的 RGB 视频。如果采用 BT.601 转换，相对比特率的节省与 BT.709 转换相似。例如，通过 BT.601 转换，DCVC-DC 比 VTM 平均节省 18.0% 的比特率（PSNR）。有关 BT.601 的更多结果，请参阅补充材料。

YUV420 色彩空间。实际上，传统编解码器主要是在 YUV420 中进行优化的。因此，在

YUV420 对于评估 NVC 相对于传统编解码器的进步也非常重要。相应的结果如表 3 所示。表中的数字是根据 YUV 三个组成部分的加权 PSNR 计算得出的。权重为 (6,1,1)/8，与标准委员会的权重一致[53]。由于大多数 NVC 没有针对 YUV420 发布相应的模型，Ta-ble 3 只报告了我们的 NVC 的数据。我们可以看到，DCVC-DC 比 VTM 平均节省 17.0% 的比特率。如果只考虑 Y 分量，则比 VTM 平均节省 15.3% 的比特率。更好的是，如表 3 所示，我们的 DCVC-DC 在 YUV420 中的平均性能也优于 ECM。这是开发 NVC 的一个重要里程碑。值得注意的是，我们的编解码器在 RGB 和 YUV420 色彩空间中使用了相同的网络结构，在训练过程中只使用了不同的微调。这表明我们的编解码器在针对不同输入色彩空间进行优化时既简单又具有很强的可扩展性。

4.3. 消融研究

为了验证每个组件的有效性，我们进行了全面的消融研究。为简化起见，这里使用的是 RGB 色彩空间的 HEVC 数据集。以 PSNR 表示的平均 BD 速率如图所示。

多样化情境。表 4 显示了对多样化语境有效性的研究。首先，从 M_e 和 M_d 的对比中可以看出，BD 率从 21.3% 降至 14.7%。这一巨大差异显示了我们基于 qaudtree 分区的熵编码的显著编码增益，并验证了通过更细粒度的分区实现多样化空间和信道上下文的优势。

表 4.不同背景下的消融研究。

	M_a	M_b	M_c	M_d	M_e
分层质量结构	✓				
抵消多样性跨组交互	✓	✓			
抵消多样性无交叉组 [8]			✓		
四叉树分区基于模型	✓	✓	✓	✓	
双空间模型 [29]					✓
BD 比率(%)	0.0	8.4	12.1	14.7	21.3

从时间维度出发，我们还设计了分层质量结构和具有跨组交互作用的偏移多样性。在表 4 中，基于 M_d ，我们首先测试了无跨组交互的原始偏移多样性 [8]，即重新移动图 4 中的重排操作。然而，它（M_c ）只能带来 2.6% 的 BD 速率差异。相比之下，通过我们的跨组交互，偏移分集的潜力得到了充分挖掘，M_b 比 M_d 降低了 6.3% 的 BD 速率。最后，基于 M_b ，我们评估了分层质量结构，即 M_a 。8.4% 的差距表明，学习高质量上下文为挖掘多帧时间相关性带来了巨大好处。

结构优化。虽然我们的编解码器能有效利用不同的语境，但我们仍希望在压缩率和计算成本之间取得更好的平衡。因此，我们进一步优化了模型的网络结构。表 5 显示了研究结果。基于 M_a （与表 4 相同），我们首先在编解码器中实现了深度分离卷积。M_h 表明，用深度可分离卷积来取代非恶意卷积不仅能显著减少 MAC，还能带来一定的压缩比改善。

第二个加速是我们使用了不相等的通道编号设置。与许多现有的 NVC 对不同分辨率的特征使用相同的通道号不同，我们建议为低分辨率特征分配较大的通道号，以提高潜在表示能力，而为高分辨率特征使用较小的通道号，以加速模型。M_g 的性能验证了我们改进的有效性。此外，许多正在使用的 NVC 在编码后对低分辨率潜表征进行量化。为了实现更精确的速率调整，本文将部分量化操作移至编码器中的高分辨率处。$_f$ 表明，编码和量化的整合提高了 BD 速率，而 MAC 的变化可以忽略不计。

4.4. 复杂性

复杂性比较见表 6。我们发现 DCVC-DC 的 MAC 降低了 19.4%，这是因为

表 5.结构优化的烧蚀研究

	M_f	M_g	M_h	M_a
高分辨率定量	✓			
不平等的通道设置	✓	✓		
可深度分离的信念	✓	✓	✓	
互助会	2642G	2642G	2939G	3456G
BD 比率 (%)	0.0	1.1	2.4	3.5

表 6.复杂性比较。

	互助会	编码时间	解码时间
DCVC-HEM [29]	3279G	890ms	652ms
我们的 DCVC-DC	2642G	1005ms	765ms

注：测试在 NVIDIA 2080TI 上进行，使用 1080p 作为输入。

29] 相比。然而，实际的编码和解码时间却更长。这是因为目前在相同的 MAC 条件下，深度卷积的计算密度没有普通卷积高。但通过定制优化[35]，未来可以进一步加快计算速度。从另一个角度看，考虑到我们的 DCVC-DC 比之前的 SOTA DCVC-HEM [29]节省了 23.5% 的双倍时间，这种运行时间的增加是值得付出的代价。相比之下，ECM 比其前身 VTM 节省了 13.1%（表 3），但编码复杂度却是 VTM 的 4 倍[49]。

5. 结论和限制

在本文中，我们介绍了如何利用不同的语境来进一步增强 NVC。该模型从时间维度出发，提取长期且高质量的上下文，以减轻误差传播并利用长距离相关性。偏移多样性与跨组互动为处理复杂运动提供了互补的运动配准。从空间维度出发，提出了基于四叉树的细粒度分区，以增加空间上下文多样性。在我们的技术推动下，NVC 的压缩率达到了新的高度。我们的 DCVC-DC 在 RGB 和 YUV420 色彩空间中都超越了 ECM，这是无损压缩技术发展的一个重要里程碑。

在训练过程中，为了学习分层质量模式，我们仍然使用与传统编解码器类似的固定失真权重。这可能不是 NVC 的最佳选择。事实上，强化学习很擅长解决这类时间序列权重决策问题。未来，我们将研究如何利用强化学习来帮助 NVC 在考虑时间依赖性的情况下做出更好的权重决策。

参考资料

[1] Eirikur Agustsson、David Minnen、Nick Johnston、Johannes Balle、Sung Jin Hwang 和 George Toderici。用于端到端优化视频压缩的规模空间流。*IEEE/CVF 计算机视觉与模式识别会议论文集》（CVPR）*，第 8503-8512 页，2020 年。1, 2, 6

[2] E.Alshina, J. Ascenso, T. Ebrahimi, F. Pereira, and T. Richter.[AHG 11] JPEG AI CfP 状态简介。*JVET-AA0047*, 2022.6

[3] Anchors - JPEG-AI MMSP Challenge：//jpegai.github.io/7- anchors/, 2022.已访问：2022-11-02.6

[4] 约翰内斯-巴勒、戴维-明宁、绍拉布-辛格、黄成金和尼克-约翰斯顿。使用尺度超优先级的变异图像压缩。*第六届学习表征国际会议，ICLR*，2018。2

[5] Gisle Bjontegaard.计算 RD 曲线之间的平均 PSNR 差异 .*VCEG-M33*, 2001.6

[6] 常见测试条件和软件参考配置。见 *JCTVC-L1100*，2013 年。6

[7] Benjamin Bross、Ye-Kui Wang、Yan Ye、Shan Liu、Jianle Chen、Gary J Sullivan 和 Jens-Rainer Ohm。通用视频编码（VVC）标准及其应用概述。*IEEE 视频技术电路与系统论文集*，31（10）：3736-3764，2021。1, 2

[8] Kelvin CK Chan、Xintao Wang、Ke Yu、Chao Dong 和 Chen Change Loy.理解视频超分辨率中的可变形对齐。*美国人工智能学会会议论文集*，2021 年。2, 4, 8

[9] Kelvin CK Chan、Shangchen Zhou、Xiangyu Xu 和 Chen Change Loy。BasicVSR++：通过增强传播和对齐提高视频超分辨率。In *Pro- ceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5972-5981, 2022.4

[10] Franc¸ois Chollet.Xception：深度可分离卷积深度学习。In *Proceedings of the IEEE con- ference on computer vision and pattern recognition*, pages 1251-1258, 2017.2, 5

[11] Ze Cui, Jing Wang, Shangyin Gao, Tiansheng Guo, Yihui Feng, and Bo Bai.具有连续速率适应性的非对称增益深度图像合成。*IEEE/CVF 计算机视觉与模式识别大会论文集》*，第 10532-10541 页，2021 年。3

[12] Armen Der Kiureghian 和 Ove Ditlevsen.偶然性还是偶发性？*结构安全》*，31（2）：105-112，2009 年。1, 2

[13] Abdelaziz Djelouah、Joaquim Campos、Simone Schaub-Meyer 和 Christopher Schroers。用于视频编码的神经帧间通信。*IEEE/CVF 计算机视觉国际会议（ICCV）论文集*，2019 年 10 月。1

[14] ECM-5.0. https://vcgit.hhi.fraunhofer.de/ecm/ECM，2022 年。访问日期：2022-11-02。6

[15] D Flynn、K Sharman 和 C Rosewarne。Hevc 范围的通用测试条件和软件参考配置

扩展，jctvc-n1006 号文件。*联合协作组视频编码 ITU-T SG*，2013 年 16 月。6

[16] Yarin Gal.*深度学习中的不确定性*》。剑桥大学博士论文，2016 年。1, 2

[17] Bernd Girod、Eckehard G Steinbach 和 Niko Faerber。H. 263 和 H. 261 视频压缩标准比较。*视频信息系统的标准和通用接口》：评论*》，1995 年。1

[18] Zongyu Guo, Zhizheng Zhang, Runsen Feng, and Zhibo Chen.用于学习图像组合的因果上下文预测。*IEEE 视频电路与系统技术论文集*》，32（4）：2329-2341，2021 年。2

[19] 何代兰、郑耀炎、孙宝成、王艳、秦宏伟。用于高效学习图像压缩的棋盘式上下文模型。*IEEE/CVF 计算机视觉与模式识别会议论文集*》，第 14771-14780 页，2021 年。2, 5

[20] HM-16.25. https://vcgit.hhi.fraunhofer.de/jvet/HM/, 2022.访问日期：2022-11-02。6

[21] Yung-Han Ho、Chih-Peng Chang、Peng-Yu Chen、Alessan- dro Gnutti 和 Wen-Hsiao Peng。Canf-vc：用于视频压缩的条件增强归一化流。*欧洲计算机视觉会议*，2022 年。1, 6, 7

[22] Zhihao Hu, Guo Lu, Jinyang Guo, Shan Liu, Wei Jiang, and Dong Xu.超优先模式预测的粗到细深度视频编码。*IEEE/CVF 计算机视觉与模式识别会议论文集*》，第 5921-5930 页，2022 年。4

[23] Zhihao Hu, Guo Lu, and Dong Xu.FVC：实现特征空间深度视频压缩的新框架。In *Pro- ceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1502-1511, 2021.4

[24] Yan Huang, Li Song, and Ebroul Izquierdo.CNN 加速的内部视频编码，上限在哪里？In *2019 Picture Coding Symposium (PCS)*, pages 1-5.IEEE, 2019.1

[25] Shuai Huo, Dong Liu, Li Li, Siwei Ma, Feng Wu, and Wen Gao.走向混合优化视频编码》，*arXiv preprint arXiv:2207.05565*, 2022.1, 2, 3

[26] Jun-Hyuk Kim, Byeongho Heo, and Jong-Seok Lee.用于学习图像组合的全局和局部联合分层先验。*IEEE/CVF 计算机视觉与模式识别会议论文集*》，第 5992- 6001 页，2022 年。2

[27] The´o Ladune、Pierrick Philippe、Wassim Hamidouche、Lu Zhang 和 Olivier De´forges.用于灵活学习视频压缩的条件编码。*神经压缩：神经压缩：从信息论到应用--* 2021 年 *ICLR 研讨会*。1, 2

[28] Jiahao Li、Bin Li 和 Yan Lu.深度上下文视频通信。*神经信息处理系统进展*》，34，2021。1, 2, 6, 7

[29] Jiahao Li, Bin Li, and Yan Lu.用于神经视频压缩的混合时空建模。In *Proceed-ings of the 30th ACM International Conference on Multime- dia*, pages 1503-1511, 2022.1, 2, 3, 5, 6, 7, 8

[30] Chong Soon Lim, SMT Naing, V Wahadaniah, and X Jing.低延迟约束下B图片的参考列表。

文件 JCTVC-D093，ITU-T/ISO/IEC 视频编码联合工作组（JCT-VC），韩国大邱，2011 年。3

[31] Jianping Lin, Dong Liu, Houqiang Li, and Feng Wu.M-LVC：用于学习视频压缩的多帧预测。*IEEE/CVF 计算机视觉与模式识别会议论文集*，2020 年。1, 2

[32] 刘豪杰、卢明、马占、王帆、谢志煌、曹勋和王尧。使用多尺度运动补偿和时空上下文模型的神经视频编码。*IEEE 视频技术电路与系统论文集*，2020 年。1

[33] Jerry Liu、Shenlong Wang、Wei-Chiu Ma、Meet Shah、Rui Hu、Pranaab Dhawan 和 Raquel Urtasun。高效视频压缩的条件编码。*欧洲计算机视觉会议*，第 453-468 页。Springer, 2020.1, 2

[34] Guo Lu、Wanli Ouyang、Dong Xu、Xiaoyun Zhang、Chunlei Cai 和 Zhiyong Gao。DVC：端到端深度视频通信框架。In *Proceedings of the IEEE/CVF Con- ference on Computer Vision and Pattern Recognition*, pages 11006-11015, 2019.1, 2, 6

[35] Gangzhao Lu, Weizhe Zhang, and Zheng Wang.优化 GPU 上的深度可分离卷积操作。*IEEE 并行与分布式系统论文集*，33（1）：70- 87，2021。8

[36] Guo Lu、Xiaoyun Zhang、Wanli Ouyang、Li Chen、Zhiyong Gao 和 Dong Xu。用于视频压缩的端到端学习框架。*电气与电子工程师学会模式分析与机器智能事务*，2020 年。1

[37] Wufei Ma, Jiahao Li, Bin Li, and Yan Lu.不确定性感知的集合深度视频压缩》，2021.1, 2

[38] Fabian Mentzer、George Toderici、David Minnen、Sung-Jin Hwang、Sergi Caelles、Mario Lucic 和 Eirikur Agusts- son。Vct： *ArXiv preprint arXiv:2206.07307*, 2022.1, 2

[39] Alexandre Mercat、Marko Viitanen 和 Jarno Vanne。UVG 数据集：用于视频编解码器分析和开发的 50/120fps 4k 序列。*第 11 届 ACM 多媒体系统会议论文集*》，第 297-302 页，2020 年。6

[40] David Minnen, Johannes Balle´, and George D Toderici.联合自回归和分层先验的学习图像压缩。*神经信息处理系统进展*》，2018年第31期。2, 5

[41] Ken M Nakanishi、Shin-ichi Maeda、Takeru Miyato 和 Daisuke Okanohara。神经多尺度图像压缩 *计算机视觉-ACCV 2018：14th Asian Conference on Computer Vision, Perth, Australia, December 2-6, 2018, Re-vised Selected Papers, Part VI 14*, pages 718-732.Springer, 2019.5

[42] 乔纳森-普法夫、阿列克谢-菲利波夫、刘山、赵昕、陈建乐、圣地亚哥-德-卢桑-埃尔南德斯、托马斯-维甘德、瓦西里-鲁菲茨基、阿达尔什-克里希南-拉马苏布拉莫尼安、吉尔特-范德奥维拉。VVC 中的内部预测和模式编码。*IEEE 视频技术电路与系统论文集*》，31（10）：3834-3847，2021 年。1

[43] Reza Pourreza、Hoang Le、Amir Said、Guillaume Sautiere 和 Auke Wiggers。 提升神经视频编解码器

arXiv preprint arXiv:2208.04303, 2022.2

[44] 钱一辰、林明、孙秀玉、谭志宇和金蓉。Entroformer：基于变换器的学习型图像压缩熵模型。*ArXiv 预印本 arXiv:2202.05492*, 2022.2

[45] 钱一辰、谭志宇、孙秀玉、林明、李东阳、孙振红、李浩、金蓉。利用全局参考学习精确的图像压缩熵模型》，*arXiv preprint arXiv:2010.08321*, 2020.2

[46] Scott Reed、Aa¨ron Oord、Nal Kalchbrenner、Sergio Go´mez Colmenarejo、Ziyu Wang、Yutian Chen、Dan Belov 和 Nando Freitas。并行多尺度自回归密度计算。*国际机器学习大会*，第 2912-2921 页。PMLR，2017。5

[47] Oren Rippel、Alexander G Anderson、Kedar Tatwawadi、San- jay Nair、Craig Lytle 和 Lubomir Bourdev。ELF-VC：有效学习的灵活速率视频编码。*IEEE/CVF 计算机视觉国际会议（ICCV）论文集*》，第 14479-14488 页，2021 年 10 月。1, 2

[48] Heiko Schwarz、Detlev Marpe 和 Thomas Wiegand。分层 B 图片和 MCTF 的分析。In *2006 IEEE International Conference on Multimedia and Expo*, pages 1929-1932.IEEE, 2006.3

[49] Vadim Seregin、Jie Chen、Fabrice Leannec 和 Kai Zhang。JVET AHG 报告：ECM 软件开发（AHG6）。*JVET-AA0006*, 2022.1, 8

[50] Xihua Sheng、Jiahao Li、Bin Li、Li Li、Dong Liu 和 Yan Lu.用于学习视频压缩的时间上下文挖掘 *电气和电子工程师学会多媒体期刊*，2022 年。1, 2, 6, 7

[51] Hui Su, Mingliang Chen, Alexander Bokov, Debargha Mukherjee, Yunqing Wang, and Yue Chen.针对 av1 的机器学习加速变换搜索。In *2019 Picture Coding Symposium (PCS)*，pages 1-5.IEEE, 2019.1

[52] Hui Su, Chi-Yo Tsai, Yunqing Wang, and Yaowu Xu.用于视频编码的机器学习加速分区搜索。In *2019 IEEE International Conference on Image Pro- cessing (ICIP)*, pages 2661-2665.IEEE, 2019.1

[53] Gary J Sullivan 和 Jens-Rainer Ohm.视频编码联合协作组（JCT-VC）第四次会议报告，韩国大邱，2011 年 1 月 20-28 日。*文件 JCTVC-D500，韩国大邱*，2011 年。7

[54] 田亚鹏、张玉伦、傅云、徐晨亮。TDAN：用于视频超分辨率的时态可变形配准网络。*IEEE/CVF 计算机视觉与模式识别会议论文集*》，第 3360-3369 页，2020 年。4

[55] VTM-17.0. https://vcgit.hhi.fraunhofer.de/jvet/VVCSoftware_VTM/, 2022.访问时间：2022-11-02.6

[56] Guo-Hua Wang, Jiahao Li, Bin Li, and Yan Lu.EVC：带掩码的实时神经图像压缩。*国际学习代表会议*，2023 年。2

[57] Haiqiang Wang, Weihao Gan, Sudeng Hu, Joe Yuchieh Lin, Lina Jin, Longguang Song, Ping Wang, Ioannis Katsavouni- dis, Anne Aaron, and C-C Jay Kuo.MCL-JCV：基于 JND

H.264/AVC 视频质量评估数据集。In *2016 IEEE International Conference on Image Processing (ICIP)*, pages 1509-1513.IEEE, 2016.6

[58] Xintao Wang, Kelvin CK Chan, Ke Yu, Chao Dong, and Chen Change Loy.EDVR：利用增强型可变形卷积网络进行视频修复。In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pat- tern Recognition* Workshops, pages 0-0, 2019.4

[59] Chao-Yuan Wu, Nayan Singhal, and Philipp Krahenbuhl.通过图像插值实现视频压缩。*欧洲计算机视觉会议论文集》（ECCV）*，第 416-431 页，2018 年。1

[60] 薛天帆、陈百安、吴佳俊、魏东来、William T Freeman。面向任务流的视频增强。*国际计算机视觉杂志》（IJCV）*，127（8）：1106-1125，2019.6

[61] Ren Yang、Fabian Mentzer、Luc Van Gool 和 Radu Timofte。使用递归自动编码器和递归概率模型的视频压缩学习。*IEEE 信号处理选题期刊》*，15（2）：388-401，2021。1, 2

[62] Xiangyu Zhang, Xinyu Zhou, Mengxiao Lin, and Jian Sun.Shufflenet：适用于移动设备的高效卷积神经网络。In *Proceedings of the IEEE con- ference on computer vision and pattern recognition*, pages 6848-6856, 2018.4