arXiv:2304.03076v1 [eess.IV] 6 Apr 2023

快速 QTMT P unerquicklich

用于 VVC 帧内编码的 U-Net 框架分区

赵赞* 黄蕾蕾‡ *ShuShi Chen** *Xiantao Zhang**
*Zhenghui Zhao** 尹海兵† 樊一博*

*中国上海，复旦大学 ‡中国上海，华东师范大学
*中国杭州，阿里巴巴集团 †中国杭州，杭州电子科技大学

摘要

通用视频编码（VVC）通过采用众多复杂的编码工具显著提升了编码效率，但也带来了更高的复杂度，尤其是灵活的四叉树加多类型树（QTMT）块划分。本文提出一种基于深度学习的算法，用于加速 VVC 帧内编码中的 QTMT 分区。该方案通过提前终止可能性较低的帧内预测模式和分区，在 BD-BR 增幅可忽略不计的情况下，大幅缩短编码时间。首先，本文提出一种重新设计的 U-Net，作为网络的基础框架。其次，我们设计了一种质量参数（QP）融合网络，用于调节量化参数对分区结果的影响。最后，我们采用精细化后处理策略，以更好地平衡编码性能与复杂度。实验结果表明，该方案优于当前最先进的方法，复杂度降低了 44.74% 至 68.76%，BD-BR 增幅为 0.60% 至 2.33%。

索引词— VVC、帧内编码、U-Net、复杂度

1. 引言

应用范围的扩大和数据的爆炸式增长给视频编码带来了新的挑战。在此背景下，VVC [1]发布了。VVC采用QTMT技术 [2], , 除了高效视频编码（HEVC） [3]已支持的四叉树（QT）分区外，还增加了二叉树水平/垂直（BTH/BTV）和三叉树水平/垂直（TTH/TTV）分区。在VVC测试模型（VTM）中，大小为128×128的编码树单元必须先进行QT分区以进行帧内编码。随后，通过耗时的率失真优化（RDO）过程确定最优分区结构，最小子块尺寸为4。其复杂度来自两个过程：

本工作部分由国家自然科学基金项目（编号：62031009）资助，部分由阿里巴巴创新研究计划（AIR）资助，部分由复旦大学-长春光机所联合基金（FC2019-001）资助，部分由复旦大学-中兴联合实验室资助，部分由工程与技术学院的先导项目资助。
丹大学(gyy2021-001)部分资助，并得到CCF-阿里巴巴青年学者创新研究基金的部分资助。（通讯作者：LeileiHuang，ll- huang@cee.ecnu.edu.cn）

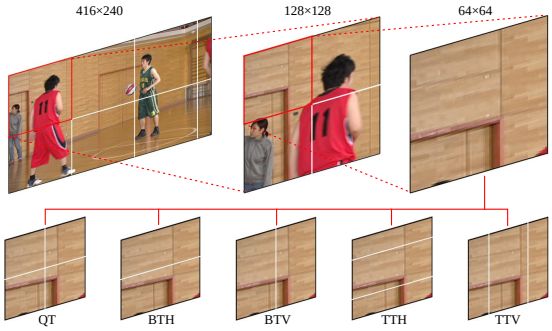

图1。VTM中QTMT分区的一个具体示例。

执行帧内预测和进行各种分区。图1展示了QTMT分区的示例。

大量研究 [4–6] 阐明了HEVC中的快速分区，但它们无法直接应用于VVC。快速VVC分区算法正受到越来越多的关注，相关文献大致可分为两类：一类基于统计分析 [7–11] ，另一类采用卷积神经网络（卷积神经网络） [12–15]。

在基于统计分析的方法中，会计算与分区结果密切相关的有用特征，并将其输入分类器。这些方法所提出的分类器各不相同。Fan 等 [7] 提出了一种决策方法，只需将特征值与给定的阈值进行比较即可。机器学习也被用于设计决策树 [8, 9], LGBM [10], 和支持向量机 [11]等分类器。这些分类器通常从上到下递归调用，因此无法并行处理。此外，如果某个特征仅在 VVC 中可用，例如 QTMT 深度级别 [10], , 则此类方法无法直接迁移到其他视频编码标准。

基于 CNN 的方法也受到了广泛关注。有一种本质上类似的方法，用于直接获取所有 4×4 子块的分区概率 [12, 13]。该方法不存在序列数据依赖，但由于网络设计简单，且采用了未考虑是否执行帧内预测的相对粗略的后处理策略，其性能较为一般。有人提出了一种多阶段退出 CNN 模型，用于在每个阶段决定是否划分编-

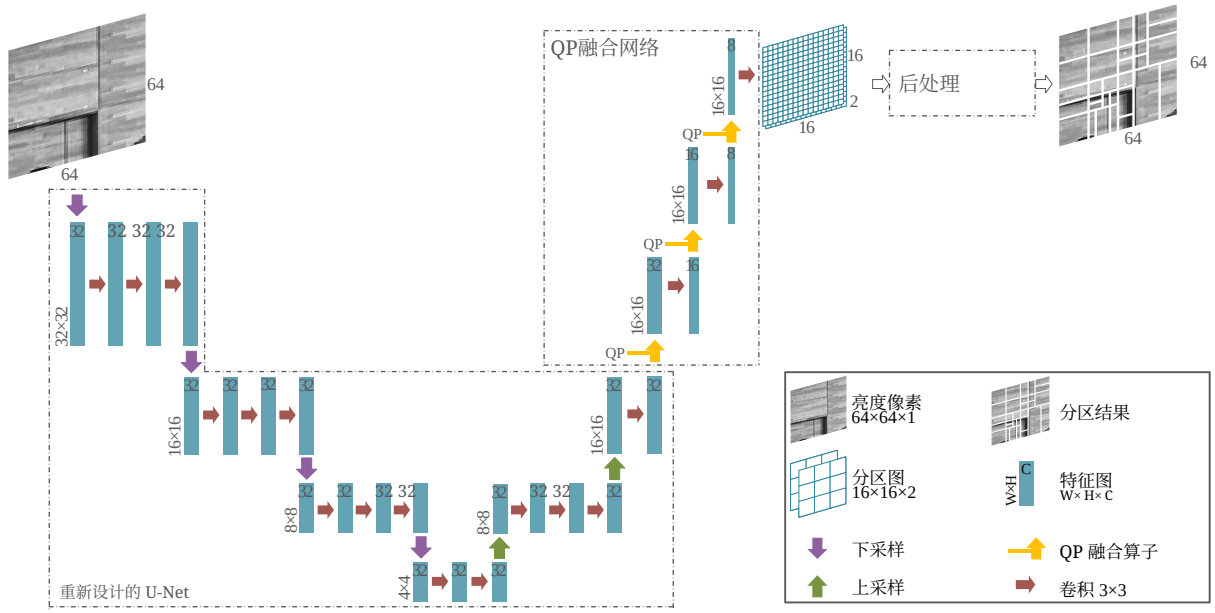

图 2。整体框架。虚线框表示关键组件，包括重新设计的 U-Net、QP 融合网络和后处理流程。

码单元（CU），从而大幅减少编码时间[14]。然而，该方法也受限于较高的网络复杂度以及缺乏并行性。Wu 等 [15]提出了一种全卷积网络，通过分层网格图在编码时间和质量之间取得了较好的折中。不过，该方法针对大小为 64×64 的编码单元（CU）和其他尺寸的编码单元（CU）分别采用了两个不同的网络，这使得网络训练和部署更具挑战性。此外，由于网络未考虑量化参数（QPs）对分区的影响，该方法仍有提升空间。

本文提出了一种基于卷积神经网络的 VVC 帧内编码快速分区算法。由于该方案仅需输入原始像素和量化参数，因此具有高度并行性，且易于进行硬件实现。另一个优势是，该方案可迁移至其他视频编码标准，因为它不需要任何与 VVC 密切相关的中间编码信息。简而言之，我们的具体贡献总结如下：（1）采用重新设计的 U-Net [16]作为网络的基础框架，以有效提取纹理信息；（2）提出一种 QP 融合网络，以调节量化参数对分区结果的影响；（3）采用精细化后处理策略，进一步优化算法，最终取得超越当前先进方法的性能。

2. 提出的方法

2.1. 概述

图 2 展示了所提出的快速分区算法流程。该框架由三个关键部分组成：一个重新设计的 U-Net、一个 QP 融合网络和一个后处理流程。前两者以原始亮度像素和 QP 为输入，并输出所有 4×4子块的划分概率，称为“分区图”。后者以该图为输入，并输出最终的 QTMT 分区。

2.2. 重新设计的 U-Net 结构

块划分任务与图像处理中的传统语义分割任务相似。在这两种任务中，都会根据内容纹理等信息，将当前图像划分为互不重叠的子块。然而，块划分任务的分割边缘是规则的，而语义分割任务的分割边缘则可能不规则。因此，我们对 U-Net 进行了大幅改造。U-Net 是一种广泛用于语义分割的经典网络，改造后的网络更适用于块划分任务。改造后的 U-Net 能够更充分地提取并融合 64×64 输入块中不同子块的特征。

输出被修改为形状为 16×16×2的分区图，其中包含 4×4 子块的划分概率。垂直和水平划分概率分别记录在两个通道中。尽管分区图的底行和右列是冗余的，但为了保持特征图的规整性，选择了 16×16×2 的形状，而不是16×15和 15×16这两种不同的形状。

如图2所示，该网络结构相对于原始 U-Net还包含以下变化：(1) 减少卷积核的数量，以降低网络的复杂度；(2) 移除桥接结构，be-

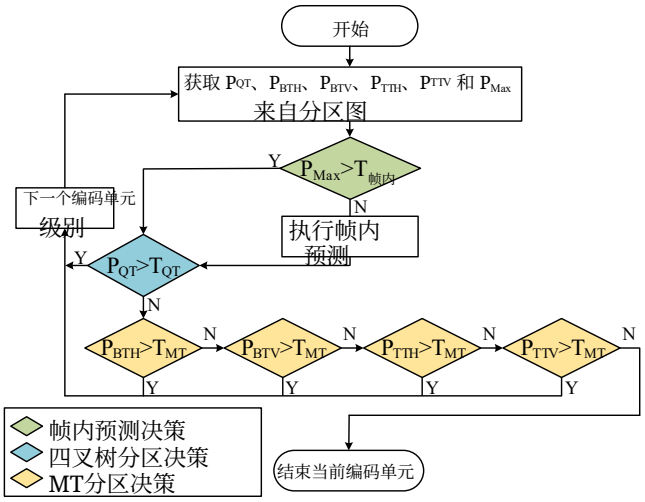

图 3。后处理流程图。

因为它会增加复杂度，却不会对特定任务的最终性能产生明显影响；（3）使用步幅为 2 的卷积神经网络进行下采样，并使用子像素卷积神经网络 [17] 进行上采样；（4）采用“same”填充，确保所有特征图的尺寸都是 4 的整数倍。

2.3. QP 融合网络

量化参数对分区有显著影响。如果仅将量化参数调节混入后处理流程，就无法充分探究量化参数的作用。因此，在重新设计的 U-Net 输出之前加入了 QP融合网络。如图2所示，QP融合网络由三个组件级联而成，每个组件都包含一个量化参数融合算子和一个卷积层。首先，将输入特征图的一半除以归一化后的量化参数，然后通过卷积以大小为3×3的卷积核进行特征融合。消融实验表明，三层级联结构优于其他量化参数融合结构。

2.4. 精细化后处理策略

在重新设计的 U-Net 和 QP 融合网络之后，建议采用精细化后处理策略。后处理流程如图3所示。首先，根据分区图计算五种分区各自的概率及其最大值。总共提出三个不同的阈值：阈值 T_{Intra} 用于决定是否执行帧内预测，阈值 T^{QT} 和 T_{MT} 分别用于确定是否跳过 QT分区和 MT分区。如果 P_{Max} 大于 T_{Intra}，则此时很可能会执行某种分区，因此无需进行帧内预测。

表1。后处理流程中的阈值设置

配置	T_{Intra}	T_{QT}	T_{MT}
我们的：*“C1”*	0.9	$0.1 \sim 0.0$	$0.05 \sim 0.0$
我们的：*“C2”*	0.8	$0.2 \sim 0.1$	$0.15 \sim 0.1$
Our: “C3”	0.7	$0.3 \sim 0.2$	$0.25 \sim 0.2$

此后，如果特定分区的可能性大于相应阈值，则执行该分区；否则，RDO过程将提前终止。

随着分区深度的增加，编码单元的大小会减小，分区图中用于计算划分概率的值也会减少，从而降低结果的稳健性。随着分区深度的增加，阈值T^{QT} 和T_{MT} 应逐渐减小，使更多较小块的潜在分区能够进入常规RDO过程，从而避免编码质量显著下降。

3. 实验

3.1. 实验设置

DIV2K [18]和 RAISE [19] 用于创建训练集。训练框架为 TensorFlow 15.0 [20]。采用二元交叉熵作为损失函数，并使用 Adam[21]作为优化器。模型训练 120 个周期，初始学习率为 1e-3，每 10 个周期降低 20%。

所提出的方法在参考软件 VTM 7.0 中实现。后处理中的阈值设置如表1所示。测试序列包括 A-E 类的全部 22 个视频序列。使用 VTM 中的编码器 intra vtm.cfg 作为编码器配置文件，量化参数设为 22、27、$\bar{3}2$ 和 $\bar{3}7$。分别测量 BD-BR [22]和复杂度降低ΔT，以评估性能。 ΔT 的计算方式如下：

$$\Delta \mathrm{T} = \frac{1}{4}\sum_{i=1}^{4}\frac{T_{anchor}(i) - T_{test}(i)}{T_{anchor}(i)} \tag{1}$$

其中，$T_{anchor}(i)$ 和 $T_{test}(i)$ 分别表示在 ith QP 和 $\{22, 27, 32, 37\}$ 条件下，使用原始 VTM 和所提出的快速算法进行编码所需的时间。为验证所提出方法的可扩展性，还在 VTM 15.0 中进行了额外测试。

3.2. 性能

在 VTM 7.0 中测试的性能如表 2 所示。编码复杂度降低了 44.74% 至 68.76%，而 BD-BR 损失仅为 0.60% 至 2.33%。与 Saldanha等人 [10], 的工作相比，我们的方法在所有评估序列上都表现更佳。A1 类的实验结果表明，我们的方案 *“C2”* 优于

表 2. 与当前先进方法相比，VTM 7.0 中快速分区算法的性能

序列	TCSVT-21 [10]		TCSVT-22 [15]		我们的：*"C1"*		我们的：*"C2"*		我们的：*"C3"*	
	BD-BR (%)	ΔT (%)	BD-BR (%)	ΔT (%)	BD-BR (%)	ΔT (%)	BD-BR (%)	ΔT (%)	BD-BR (%)	ΔT (%)
A1 类	1.05	53.40	1.37	56.52	0.50	48.89	1.05	58.75	1.89	63.97
A2 类	1.11	52.63	1.26	62.63	0.46	48.23	1.11	61.07	2.00	68.82
B 类	1.40	58.26	1.09	51.11	0.60	49.53	1.31	64.75	2.35	73.33
C 类	1.65	51.99	0.95	41.27	0.61	38.54	1.37	55.88	2.50	67.06
D 类	1.24	49.92	1.00	38.00	0.53	36.79	1.26	55.09	2.16	64.58
E类	2.07	58.45	1.32	47.54	0.89	47.98	1.82	63.97	3.11	73.75
平均值	**1.42**	**54.20**	**1.14**	**48.76**	**0.60**	**44.74**	**1.32**	**59.95**	**2.33**	**68.76**

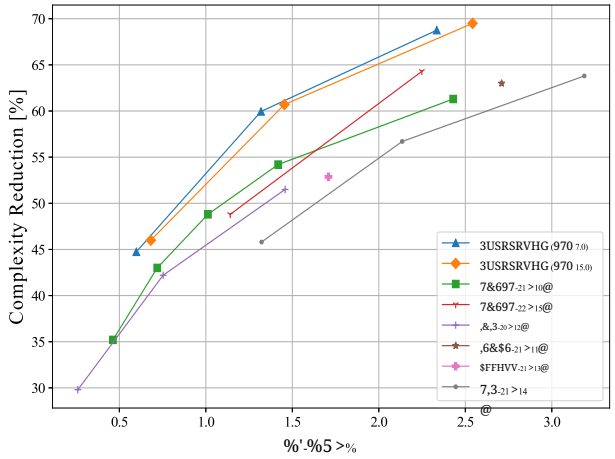

图 4. 所提方法的 BD-BR 损失和复杂度降低，并与相关工作进行比较。

Wu等人 [15], 的工作，编码时间进一步节省了 2.23%，BD-BR 损失降低了 0.32%。

为进一步直观展示所提方法的性能，图 4 绘制了性能对比曲线，其中包括若干重要研究 [10–15]。我们的方法显然在 BD-BR 损失与复杂度降低之间实现了更好的折中。VTM 7.0 和 VTM 15.0 中的性能非常相似，这是因为版本更新并未改变 QTMT块划分结构。不过，VTM 15.0 增加了许多新工具，整体编码复杂度更高，因此 VTM 7.0 中的性能略好。

3.3. 消融实验

为验证 QP融合网络的有效性，在改变 QP融合网络结构的同时，保持重新设计的 U-Net 和后处理策略不变。最终结构（所提方法）如图所示。

表 3。QP融合网络的消融实验结果

消融实验	组件数量		性能	
	QP 融合算子	卷积 3×3	BD-BR (%)	ΔT (%)
S1	0	0	1.37	56.62
S2	0	3	1.39	56.73
S3	1	3	1.35	57.82
所提出的	3	3	**1.31**	**58.57**

图 2. 另外三种用于比较的结构（*S1*、*S2* 和*S3*）以及 B 类、C 类和 D 类的平均性能如表 3 所示。可以观察到，所提出的结构性能最佳。*S1* 被用作测试锚点，不包含 QP 融合算子或卷积层。所提出的 结构与*S2*中的训练参数数量相同，从而排除了网络复杂度对性能的影响。对比所提出的结构与仅包含一个 QP 融合算子的*S3* 的结果表明，增加 QP 融合算子能够带来更优的性能。然而，更多的级联 QP 融合算子反而会导致计算过于复杂，因此最终选择了三层级联结构。

4. 结论

本文提出了一种基于 CNN 的方法，用于加速 VVC 帧内编码中 QTMT 划分结构的决策过程。首先，我们通过改进 U-Net 构建骨干网络。随后，我们设计了 QP 融合网络和精细化后处理策略，以进一步提升性能。在 BD-BR 损失为 0.60% ～ 2.33% 的情况下，所提出的快速算法可节省 44.74% ～ 68.76% 的编码时间。未来，我们计划将快速分区算法扩展到帧间编码，并完成硬件实现。

5. 参考文献

[1] B. Bross、Y.-K. Wang、Y. Ye、S. Liu、J. Chen、G. J. Sul- livan 和 J.-R. Ohm，“通用视频编码（VVC）标准及其应用综述”，*IEEE* 视频技术电路与系统学报，第 31 卷，第 10 期，第 3736–3764 页，2021 年。

[2] Y.-W. Huang、C.-W. Hsu、C.-Y. Chen、T.-D. Chuang、S.-T. Hsiang、C.-C. Chen、M.-S. Chiang、C.-Y. Lai、C.-M. Tsai、Y.-C. Su、Z.-Y. Lin、Y.-L. Hsiao、O. Chubach、Y.-C. Lin 和 S.-M. Lei，“一种采用四叉树加二叉树-三叉树编码块结构及先进编码技术的 VVC 提案”，*IEEE* 视频技术电路与系统汇刊，第 30 卷，第 5 期，第 1311–1325 页，2020 年。

[3] G. J. Sullivan、J.-R. Ohm、W.-J. Han 和 T. 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。[4] L. Shen、Z. Zhang 和 Z. Liu，“HEVC 帧内编码的有效编码单元尺寸决策”，*IEEE* 图像处理汇刊，第 23 卷，第 10 期，第 4232–4241 页，2014 年。[5] B. Min 和 R. C. C. Cheung，“一种用于 HEVC 帧内编码器的快速编码单元尺寸决策算法”，*IEEE* 视频技术电路与系统汇刊，第 25 卷，第 5 期，第 892–896 页，2015 年。[6] Z. Chen、J. Shi 和 W. Li，“学习式 HEVC 快速帧内编码”，*IEEE* 图像处理汇刊，第 29 卷，第 5431–5446 页，2020 年。[7] Y. Fan、J. Chen、H. Sun、J. Katto 和 M. Jing，“一种用于 VVC 帧内预测的快速 QTMT 分区决策策略”，*IEEE Access*，第 8 卷，第 107 900–107911 页，2020 年。[8]H. Yang、L. Shen、X. Dong、Q. Ding、P. An 和 G. Jiang，“通用视频编码的低复杂度 CTU 分区结构决策与快速帧内模式决策”，*IEEE* 视频技术电路与系统汇刊，第 30 卷，第 6 期，第 1668–1682 页，2019 年。[9] X. Dong、L. Shen、M. Yu 和 H. Yang，“通用视频编码的快速帧内模式决策算法”，*IEEE* 多媒体汇刊，第 24 卷，第 400–414 页，2021 年。

[10] M. Saldanha、G. Sanchez、C. Marcon 和 L. Agostini，“使用轻量级梯度提升机的可配置 VVC 帧内编码快速块划分”，*IEEE* 视频技术电路与系统汇刊，第 32 卷，第 6 期，第 3947–3960 页，2021 年。

[11] G. Wu、Y. Huang、C. Zhu、L. Song 和 W. Zhang，“基于支持向量机的 VVC 帧内编码快速编码单元划分算法”，载于 *2021* 年 *IEEE* 国际电路与系统研讨会（*ISCAS*），2021 年，第 1–5 页。

[12] A. Tissier、W. Hamidouche、J. Vanne、F. Galpin 和 D. Menard，“面向卷积神经网络的 VVC 帧内编码器复杂度降低”，载于*2020 IEEE* 图像处理国际会议（*ICIP*），2020 年，第 3139–3143 页。

[13]Q. Zhang、R. Guo、B. Jiang 和 R. Su，“一种基于 DenseNet 网络的 VVC 快速编码单元决策算法”，*I EEE Access*，第 9 卷，第 119 289–119297 页，2021 年。

[14] T. Li、M. Xu、R. Tang、Y. Chen 和 Q. Xing，“Deep-QTMT：一种用于 VVC 帧内模式快速基于 QTMT 的编码单元划分的深度学习方法”，*IEEE* 图像处理汇刊，第 30 卷，第 5377–5390 页，2021 年。[15] S. Wu、J. Shi 和 Z. Chen，“HG-FCN：用于 VVC 快速帧内编码的分层网格全卷积网络”，*IEEE* 视频技术电路与系统汇刊，第 32 卷，第 8 期，第 5638–5649 页，2022 年。[16] O. Ronneberger、P. Fischer 和 T. Brox，“U-Net：用于生物医学图像分割的卷积网络”，载于医学图像计算与计算机辅助干预国际会议。Springer，2015 年，第 234–241 页。[17] W. Shi、J. Caballero、F. Husz´ar、J. Totz、A. P. Aitken、R. Bishop、D. Rueckert 和 Z. Wang，“使用高效亚像素卷积神经网络实现实时单幅图像和视频超分辨率”，载于*IEEE* 计算机视觉与模式识别会议论文集，2016 年，第 1874–1883 页。[18] E. Agustsson 和 R. Timofte，“NTIRE 2017 单幅图像超分辨率挑战赛：数据集与研究”，载于*IEEE* 计算机视觉与模式识别会议研讨会论文集，2017 年，第 126–135 页。[19]D.-T. Dang-Nguyen、C. Pasquini、V. Conotter 和 G. Boato，“RAISE：用于数字图像取证的原始图像数据集”，载于第六届 *ACM* 多媒体系统会议论文集，2015 年，第 219–224 页。[20] “TensorFlow”，2022 年。 [在线]。网址：https://www. tensorflow.org/[21] D. P. Kingma 和 J. Ba，“Adam：一种随机优化方法”，载于国际学习表征会议，2015 年。[22] G. Bjontegaard，“RD 曲线之间平均峰值信噪比差异的计算”，*VCEG-M33*，2001 年。