



## (12) 发明专利

(10) 授权公告号 CN 111095932 B

(45) 授权公告日 2022.06.07

(21) 申请号 201880060233.7

(22) 申请日 2018.08.29

(65) 同一申请的已公布的文献号

申请公布号 CN 111095932 A

(43) 申请公布日 2020.05.01

(30) 优先权数据

17306105.2 2017.08.29 EP

(85) PCT国际申请进入国家阶段日

2020.03.17

(86) PCT国际申请的申请数据

PCT/US2018/048427 2018.08.29

(87) PCT国际申请的公布数据

W02019/046356 EN 2019.03.07

(73) 专利权人 交互数字麦迪逊专利控股公司

地址 法国巴黎

(72) 发明人 A.罗伯特 F.勒林内克 F.厄本

(74) 专利代理机构 北京市柳沈律师事务所

11105

专利代理人 叶齐峰

(51) Int.Cl.

H04N 19/70 (2006.01)

H04N 19/56 (2006.01)

H04N 19/533 (2006.01)

H04N 19/57 (2006.01)

H04N 19/52 (2006.01)

H04N 19/119 (2006.01)

H04N 19/103 (2006.01)

H04N 19/46 (2006.01)

H04N 19/513 (2006.01)

(56) 对比文件

CN 101189882 A, 2008.05.28

CN 104662906 A, 2015.05.27

CN 101969568 A, 2011.02.09

CN 101310534 A, 2008.11.19

EP 2461565 A2, 2012.06.06

Jianle Chen et al..Algorithm

Description of Joint Exploration Test Model 6 (JEM 6).《Joint Video Exploration Team (JVET) of ITU-T SG 16 WP 3 and ISO/IEC JTC 1/SC 29/WG11》.2017,

审查员 李维维

权利要求书2页 说明书8页 附图3页

(54) 发明名称

使用帧速率上转换工具用于改进的压缩/解压缩的方法和装置

(57) 摘要

通过在视频比特流中使用用于控制视频图像的子部分和子处理两者的标志,改进帧速率上转换功能的管理。代替控制帧速率上转换的单个标志,至少一个实施例使用附加信令来确定编码器或解码器是否应该对特定视频编码部分实施帧速率上转换,或者是否实施与帧速率上转换相关联的处理之一。

CN 111095932 B

300

开始

301

执行帧速率上转换

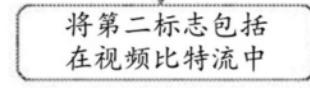
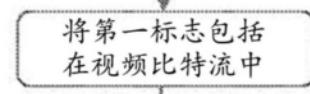
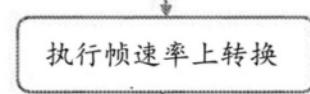
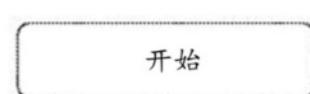
310

将第一标志包括  
在视频比特流中

320

将第二标志包括  
在视频比特流中

330



1.一种方法,包括:

使用一个或多个帧速率上转换(FRUC)子工具对视频图像的一部分执行FRUC;  
将第一标志包括在视频比特流中;以及  
将第二标志包括在所述视频比特流中;以及  
将第三标志包括在所述视频比特流中,  
其中,所述第一标志指示向译码器信令通知比特流以进一步从初始的FRUC工具中启用FRUC合并子工具,

所述第二标志指示向译码器信令通知比特流以对于在所述译码器侧的所述视频图像的所述部分进一步启用所述启用的FRUC合并子工具的子部分,以及

所述第三标志指示向译码器信令通知比特流以进一步启用所述启用的FRUC合并子工具的双边模板匹配代价函数。

2.一种方法,包括:

解析视频比特流以识别是否设置了第一标志;  
解析所述视频比特流以识别是否设置了第二标志;  
解析所述视频比特流以识别是否设置了第三标志;以及  
在设置了所述第一标志的条件下,使用从初始的FRUC工具中启用的FRUC合并子工具对视频图像的一部分执行帧速率上转换(FRUC),

其中,在设置了所述第一标志和所述第二标志的条件下,使用从初始的FRUC工具中启用的FRUC合并子工具和进一步启用的所述FRUC合并子工具的子部分对所述视频图像的所述部分执行帧速率上转换(FRUC),

其中,在设置了所述第一标志和所述第三标志的条件下,使用从初始的FRUC工具中启用的FRUC合并子工具和进一步启用的所述FRUC合并子工具的双边模板匹配代价函数对所述视频图像的所述部分执行帧速率上转换(FRUC)。

3.一种装置,包括:

存储器,和  
处理器,被配置为:  
使用一个或多个帧速率上转换(FRUC)子工具对视频图像的一部分执行FRUC;  
将第一标志包括在视频比特流中;以及  
将第二标志包括在所述视频比特流中;以及  
将第三标志包括在所述视频比特流中,  
其中,所述第一标志指示向译码器信令通知比特流以进一步从初始的FRUC工具中启用FRUC合并子工具,

所述第二标志指示向译码器信令通知比特流以对于在所述译码器侧的所述视频图像的所述部分进一步启用所述启用的FRUC合并子工具的子部分,以及

所述第三标志指示向译码器信令通知比特流以进一步启用所述启用的FRUC合并子工具的双边模板匹配代价函数。

4.一种装置,包括:

存储器,和  
处理器,被配置为:

- 解析视频比特流以识别是否设置了第一标志；  
解析所述视频比特流以识别是否设置了第二标志；  
解析所述视频比特流以识别是否设置了第三标志；以及  
基于所述第一标志、所述第二标志和所述第三标志，使用一个或多个帧速率上转换(FRUC)子工具对视频图像的一部分执行FRUC，  
在设置了所述第一标志的条件下，使用从初始的FRUC工具中启用的FRUC合并子工具对视频图像的一部分执行帧速率上转换(FRUC)，  
其中，在设置了所述第一标志和所述第二标志的条件下，使用从初始的FRUC工具中启用的FRUC合并子工具和进一步启用的所述FRUC合并子工具的子部分对所述视频图像的所述部分执行帧速率上转换(FRUC)，  
其中，在设置了所述第一标志和所述第三标志的条件下，使用从初始的FRUC工具中启用的FRUC合并子工具和进一步启用的所述FRUC合并子工具的双边模板匹配代价函数对所述视频图像的所述部分执行帧速率上转换(FRUC)。  
5. 根据权利要求1或2所述的方法，或者根据权利要求3或4所述的装置，其中，所述视频图像的所述部分是编码单元(CU)并且合并块的子部分是所述CU的子块。  
6. 根据权利要求1所述的方法，或者根据权利要求3所述的装置，还包括：  
在所述比特流中包括第四标志，所述第四标志指示向译码器信令通知比特流以进一步从初始FRUC工具启用自适应运动矢量预测(AMVP)子工具。  
7. 根据权利要求2的方法，或者根据权利要求4的装置，还包括：  
解析所述视频比特流以识别是否设置了第四标志；  
在设置了所述第一标志和所述第四标志的条件下，使用从所述初始的FRUC工具中启用的FRUC合并子工具和从所述初始的FRUC工具中启用的自适应运动矢量预测(AMVP)子工具对所述视频图像的所述部分执行帧速率上转换(FRUC)。  
8. 根据权利要求1或2所述的方法，或者根据权利要求3或4所述的装置，其中所述标志的任一个位于视频参数集中。  
9. 根据权利要求1或2所述的方法，或者根据权利要求3或4所述的装置，其中所述标志任一个位于序列参数集中。  
10. 根据权利要求1或2所述的方法，或者根据权利要求3或4所述的装置，其中所述标志任一个位于图片参数集中。  
11. 一种非暂时性计算机可读介质，具有代码储存在其上，当执行所述代码时，使得处理器执行根据权利要求1和5至10中任一项所述的方法。  
12. 一种非暂时性计算机可读介质，具有代码储存在其上，当由计算机执行所述代码时，使得计算机执行根据权利要求2和5至10中任一项所述的方法。

## 使用帧速率上转换工具用于改进的压缩/解压缩的方法和装置

### 技术领域

[0001] 本原理涉及视频压缩，并且更具体地，涉及执行视频编码和解码。

### 背景技术

[0002] 已经进行了许多尝试来提高基于块的编解码器的编码效率。帧速率上转换(Frame Rate Up-Conversion, FRUC)是一种工具，其允许在没有任何信息的情况下(即没有补充语法的情况下)，推导运动矢量预测量。FRUC处理是完全对称的，因为在解码侧执行与在编码侧相同的操作。

[0003] 在使用若干种子工具时，该工具能用一个标志来完全开/关(如表2所示)：

[0004] AMVP (Advanced Motion Vector Prediction, 高级运动矢量预测) 块使用一个模板匹配代价函数，并且没有信令。

[0005] 合并(merge) 块可以在相同的处理中使用子部分细化，使用两个不同的模板匹配代价函数，并具有一些信令(开/关模板/双边)。

[0006] 表1提供了FRUC工具以及不同子工具在联合视频探索小组的联合探索模型4 (JEM 4) (ITU-T VCEG (Q6/16)) 和ISO/IEC MPEG (JTC 1/SC 29/WG11) 上的总体性能。

	随机接入 Main 10				
	在 HM-16.6-JEM-4.0 上(并行)			EncT	DecT
	Y	U	V		
FRUC	-3.60%	-3.72%	-3.92%	139%	142%
AMVP	-0.85%	-0.77%	-0.85%	107%	110%
Merge	-2.85%	-3.13%	-3.33%	128%	133%
子部分	-0.43%	-0.52%	-0.59%	109%	118%
双边	-1.16%	-1.35%	-1.44%	116%	115%

[0007]

	低延迟 B Main10				
	在 HM-16.6-JEM-4.0 上(并行)			EncT	DecT
	Y	U	V		
FRUC	-2.41%	-3.28%	-3.39%	161%	162%
AMVP	-0.41%	0.17%	-0.28%	113%	116%
Merge	-2.09%	-3.65%	-3.79%	132%	136%

子部分	-0.48%	-0.99%	-0.79%	114%	117%
双边	-1.03%	-2.06%	-2.10%	120%	120%

[0008]	低延迟 P Main10				
	在 HM-16.6-JEM-4.0 上(并行)			EncT	DecT
Y	U	V	EncT	DecT	
FRUC	-1.38%	-1.33%	-1.53%	128%	119%
AMVP	-0.16%	-0.12%	-0.45%	115%	109%
Merge	-1.27%	-1.48%	-1.32%	114%	109%
子部分	-0.21%	-0.42%	-0.45%	104%	104%
双边	0.00%	0.00%	0.00%	106%	100%

[0009] 表1 FRUC工具和子工具在JEM4[2]上的性能

[0010] 此外,这些子工具中的若干种子工具使用参数。其中一些已经在表2所示的语法中,但是其他的没有。

[0011] 在表2中,sps\_use\_FRUC\_mode是整个FRUC工具的开/关标志,FRUC\_refine\_filter允许更改子像素插值滤波器,FRUC\_refine\_range\_in\_pel定义用于细化的最大整数像素范围,并且FRUC\_small\_blk\_refine\_depth定义FRUC Merge块的子部分的最大深度(即它们的最小尺寸)。

[0012]

...	
sps_use_FRUC_mode	u (1)
if (sps_use_FRUC_mode) {	
FRUC_refine_filter	ue (v)
FRUC_refine_range_in_pel	ue (v)
FRUC_small_blk_refine_depth	ue (v)
}	

[0013] 表2当前FRUC工具的SPS语法

## 发明内容

[0014] 现有技术的这些和其他缺点和弱点通过本描述的实施例来解决,这些实施例针对一种用以管理由FRUC工具提供的编码效率与其复杂性之间的折衷的方法和装置。

[0015] 根据所述实施例的一个方面,提供了一种方法。该方法包括对视频图像的一部分进行编码的步骤,包括对视频图像的一部分执行帧速率上转换;将第一标志包括在视频比特流中,该第一标志指示帧速率上转换;以及将第二标志包括在所述视频比特流中,所述第二标志进一步指示所述帧速率上转换。

[0016] 根据所述实施例的另一方面,提供了一种方法。该方法包括对视频图像的一部分进行解码的步骤,包括解析视频比特流以识别第一标志;解析视频比特流以识别第二标志,以及基于第一标志和第二标志对视频图像的一部分执行帧速率上转换。

[0017] 根据所述实施例的另一方面,提供了一种装置。该装置包括存储器和处理器。处理器可以被配置为编码视频信号的一部分;对视频图像的一部分执行帧速率上转换;将第一标志包括在视频比特流中,该第一标志指示帧速率上转换;以及将第二标志包括在所述视频比特流中,所述第二标志进一步指示所述帧速率上转换。

[0018] 根据所述实施例的另一方面,提供了一种装置。该装置包括存储器和处理器。该处理器可以被配置为解码视频信号的一部分:解析视频比特流以识别第一标志;解析视频比特流以识别第二标志,以及基于第一标志和第二标志对视频图像的一部分执行帧速率上转换。

[0019] 从下面结合附图阅读的示例性实施例的详细描述中,本原理的这些和其他方面、特征和优点将变得显而易见。

## 附图说明

[0020] 图1示出了可以应用所提出的实施例的典型视频编码器的框图。

[0021] 图2示出了可以应用所提出的实施例的典型视频解码器的框图。

[0022] 图3示出了使用所描述的实施例进行编码的方法的一个实施例。

[0023] 图4示出了使用所描述的实施例进行解码的方法的一个实施例。

[0024] 图5示出了使用所描述的实施例进行编码或解码的装置的一个实施例。

## 具体实施方式

[0025] 本文描述的实施例的领域是视频压缩,意图提高现有技术视频编码方案的视频压缩效率。

[0026] 最近在联合探索模型(JEM)中引入的示例性编码工具被称为FRUC(帧速率上转换),或者也称为模式匹配运动推导,并且旨在解码器侧基于运动块的运动矢量预测量推导。

[0027] 帧速率上转换(FRUC)工具旨在针对模板匹配代价在候选集合当中寻找最佳运动矢量预测量(MVP)。识别的最佳候选然后向着最小模板匹配代价而被细化。

[0028] 对于每种类型的块,FRUC处理都是相似的:对整个块执行一个处理,然后,对于某些特定的块,还可以对子部分执行第二处理。这些处理之间的主要区别在于候选的初始列表和可用模板匹配代价函数。

[0029] 为了管理该FRUC工具的性能与其复杂性之间的折衷,有可能向解码器通知允许或不允许哪些FRUC处理(或子处理,即处理的一部分)。

[0030] 本文描述的实施例所解决的问题是管理由FRUC工具提供的编码效率与其复杂性之间的折衷,例如,在JEM视频编解码器中。

[0031] FRUC工具应用于所有块(Merge和AMVP),并在Merge块的子部分或子块级别进行细化。对于AMVP块,只有一个模板匹配代价函数(“模板”)可用。对于Merge块及其子部分,测试两个不同的模板匹配代价函数,“模板”和“双边”。

[0032] 模板匹配通过寻找当前图片中的模板(当前编码单元的顶部和/或左边邻近块)和参考图片中具有与模板相同尺寸的块之间的最佳匹配来推导当前编码单元的运动信息。

[0033] 双边匹配通过在两个参考图片中沿着当前编码单元的运动轨迹寻找两个块之间的最佳匹配来推导当前编码单元的运动信息。

[0034] Merge块的子部分是子块。在FRUC Merge中,首先在CU(编码单元)级别应用FRUC工具,然后将该CU划分为子块,并且再次用与CU相同的模板匹配代价函数对每个子块应用FRUC工具。

[0035] 当SPS (Sequence Parameter Set, 序列参数集) FRUC标志为开时, FRUC工具在编码侧被使用/测试, 并且在解码侧被启用。

[0036] 对于AMVP, FRUC工具允许寻找以相同方式在编码和解码时推导出的AMVP候选, 而无需信令。

[0037] 对于Merge, 在编码侧测试FRUC Merge模式, 然后有一些信令指示是否应该使用FRUC, 并且如果使用FRUC, 则用该模板匹配代价函数(有三种状态: 关/模板/双边)。

[0038] 对于Merge CU, 当激活FRUC工具(在SPS级别)时, 第一标志信令通知Merge模式, 然后第二标志信令通知FRUC状态。解码器读取这些标志, 从而知道是否必须为该CU使用FRUC工具。

[0039] CU->Merge标志=0=>AMVP(用FRUC)

[0040] =1=>Merge+FRUC模式=关=>经典Merge

[0041] =模板/双边=>FRUC Merge+模板匹配代价函数

[0042] 换句话说, 当前, 在SPS(序列参数集)级别的标志使能是否使用FRUC工具。当激活时, 编码和解码侧的每个AMVP块进入FRUC工具, 而无需信令, 并且编码侧的每个Merge块在具有一些信令(关/开模板/双向)的情况下进入FRUC工具。

[0043] 本文的方面所提出的解决方案包括添加一些高级语法, 以允许管理由FRUC工具提供的编码效率与其复杂性之间的折衷。

[0044] 如表1所示, FRUC工具的整体性能是巨大的, 在JEM4中, 它带来RA(Random Access, 随机接入)中的3.60%的BD速率增益、LDB(低延迟-B)中的2.41%和LDP(低延迟-P)中的1.38%, 但是在编码和解码时也具有关于RA中的大约40%、LDB中的60%和LDP中的30%的大量复杂性。

[0045] 在该FRUC工具中, 有若干子处理, 每个子处理都带来编码增益, 但也增加了复杂性, 但是它们都在SPS级别用唯一标志来激活。

[0046] 能够通过使用一些新的高级语法来指示将使用哪些子工具以及使用哪些参数或者在子块或子部分的基础上激活特定工具来管理编码增益与复杂性之间的折衷是有利的。例如, 禁用FRUC Merge块的子部分允许节省10-15%的编码时间(RA/LDB)和将近20%的解码时间, 而达到小于0.5%的BD速率损失。

[0047] 在第一实施例中, 可以使用来自表3的语法, 用一个标志从初始FRUC工具启用/禁用FRUC AMVP子工具。

[0048]	<pre>if (sps_use_FRUC_mode) {     FRUC_amvp }</pre>	
--------	---	--

[0049] 表3启用/禁用FRUC AMVP的标志的语法

[0050] 在第二实施例中, 可以使用来自表4的语法, 用一个标志从初始FRUC工具启用/禁用具有子部分和两个模板匹配代价函数的整个FRUC Merge子工具。

[0051]	<pre>if (sps_use_FRUC_mode) {     FRUC_merge }</pre>	
--------	--	--

[0052] 表4启用/禁用整个FRUC Merge的标志的语法

[0053] 在第三实施例中,可以使用来自表5的语法,用一个标志从初始FRUC工具启用/禁用FRUC Merge子工具的子部分。

[0054]	if (sps_use_FRUC_mode) { FRUC_merge_subblock }	
--------	--	--

[0055] 表5启用/禁用FRUC Merge子部分的标志的语法

[0056] 在第四实施例中,可以使用来自表6的语法,用一个标志从初始FRUC工具启用/禁用FRUC Merge子工具的双边模板匹配代价函数。

[0057]	if (sps_use_FRUC_mode) { FRUC_merge_bilateral }	
--------	---	--

[0058] 表6启用/禁用FRUC Merge的双边模板匹配代价函数的标志的语法

[0059] 在第五实施例中,可以用一个标志从初始FRUC工具启用/禁用整个FRUC Merge子工具,然后只有当FRUC Merge子工具被启用时,也可以用一个标志来启用/禁用FRUC Merge子工具的子部分或双边模板匹配代价函数。

[0060] 为此,语法应遵循表7或表8,其中FRUC\_merge\_xxx代表FRUC\_merge\_subblock或FRUC\_merge\_binary或两者。

[0061]	if (sps_use_FRUC_mode) { FRUC_merge if (FRUC_merge) { FRUC_merge_xxx } }	u(1)
--------	---	------

[0062] 表7启用/禁用整个FRUC Merge和当FRUC Merge被启用时启用/禁用子部分和/或双边模板匹配代价函数的标志的语法

[0063]	if (sps_use_FRUC_mode) { FRUC_merge }	u(1)
--------	---	------

[0064]	if(sps_use_FRUC_mode && FRUC_merge ) { FRUC_merge_xxx }	u(1)
--------	--	------

[0065] 表8启用/禁用整个FRUC Merge和当FRUC Merge被启用时启用/禁用子部分和/或双边模板匹配代价函数的标志的独立语法

[0066] 在第六实施例中,可以从初始FRUC工具启用/禁用FRUC AMVP和整个FRUC Merge子工具,每个子工具利用一个标志。然后,也可以使用来自表9或表10的语法,用一个标志来启用/禁用FRUC Merge子工具的子部分和双边模板匹配代价函数。

[0067]	if (sps_use_FRUC_mode) {	
--------	--------------------------	--

FRUC_amvp	u(1)
FRUC_merge	u(1)
if (FRUC_merge) {	
FRUC_merge_subblock	u(1)
FRUC_merge_bilateral	u(1)
}	
}	

[0068] 表9启用/禁用FRUC AMVP和整个FRUC Merge和当FRUC Merge被启用时启用/禁用子部分和/或双边模板匹配代价函数的标志的语法

[0069]	if( sps_use_FRUC_mode ) {	
	FRUC_amvp	u(1)
	FRUC_merge	u(1)
[0070]	}	
[0071]	if (sps_use_FRUC_mode&&FRUC_merge) {	
	FRUC_merge_subblock	u(1)
	FRUC_merge_bilateral	u(1)
	}	

[0072] 表10启用/禁用FRUC AMVP和整个FRUC Merge和当FRUC Merge被启用时启用/禁用子部分和/或双边模板匹配代价函数的标志的独立语法

[0073] 如果FRUC\_amvp与FRUC\_merge标志一起使用,则至少一个必须为“真”(否则,当两个标志都为“假”时,这与关闭所有FRUC工具一样(sps\_use\_FRUC\_mode==“假”)。

[0074] 此外,可以观察到前四个实施例的任何其他组合。

[0075] 在第七实施例中,还可以添加语法中还没有的一些参数,以便更精细地管理编码增益与复杂性之间的折衷。

[0076] 这些参数可以是但不限于:细化搜索模式(菱形、六边形、十字等)、细化循环的数量、模板函数的模板匹配尺寸、在细化期间应用于运动矢量代价的权重、要评估的候选的最大数量、模板匹配代价的最大值、或其他此类参数。

[0077] 如示例性表11所示,可以为所有处理或每种块(AMVP、Merge)或子部分或任何组合指定这些参数中的每一个。

[0079]	if (sps_use_FRUC_mode) {	
	FRUC_refine_pattern	ue(v)
	FRUC_refine_pattern_sub	ue(v)
	FRUC_refine_loops	ue(v)
	FRUC_refine_loops_sub	ue(v)
	FRUC_temp_size_amvp	ue(v)
	FRUC_temp_size_merge	ue(v)
	FRUC_nb_cand_amvp	ue(v)

FRUC_nb_cand_merge	ue (v)
FRUC_nb_cand_sub	ue (v)
...	
}	

[0080] 表11若干个FRUC参数的语法示例

[0081] 在第八实施例中,从实施例1至6引入的任何标志可以与来自实施例7的任何参数耦合。

[0082] 在第九实施例中,这种高级语法可以发生在不同的NAL单元中,如:作为一般信息的“简档、层、级别”中的VPS(视频参数集)、SPS(序列参数集)、PPS(图片参数集)或SH(条带头)。

[0083] 所使用的语法位置取决于所需的粒度,即在VPS和SPS级别它将对整个序列激活,在PPS级别对一组帧激活,在SH级别对每个帧独立激活。

[0084] 来自实施例1、实施例2、实施例3、实施例4、表7的实施例5、表9的实施例6和实施例7的语法可以在比特流中占据任何位置。来自表8的实施例5和表10的实施例6的语法的第一部分也可以出现在任何位置,但是第二部分应该跟随在其后,或者在相同的级别或者在更深的级别。例如,可以在SPS级别针对整个序列定义AMVP/Merge,而在SH级别针对每个帧定义子块/双边。

[0085] 图3示出了用于在编码器中对视频图像的一部分执行帧速率上转换的方法300的一个实施例。该方法从开始框301开始,并前进到框310,以对视频图像的一部分执行帧速率上转换。控制从框310前进到框320,以将第一标志包括在视频比特流中,第一标志指示帧速率上转换。控制从框320前进到框330,以将第二标志包括在视频比特流中,第二标志进一步指示所述帧速率上转换。第二标志可以用于指示帧速率转换处理的附加特征,诸如块或编码单元的子部分的性能,或者实施帧速率上转换处理的一个或多个子处理。

[0086] 图4示出了用于在解码器中对视频图像的一部分执行帧速率上转换的方法400的一个实施例。该方法从开始框401开始,并开始于框410,以解析视频比特流以识别第一标志。控制从框410前进到框420,以解析视频比特流以识别第二标志。控制从框420前进到框430,以基于第一标志和第二标志对视频图像的一部分执行帧速率上转换。

[0087] 图5示出了用于对视频图像中的块进行编码或解码的装置500的一个实施例。该装置包括处理器510和存储器520。处理器510被配置用于编码,以执行图3的步骤,即对视频图像的一部分执行帧速率上转换;将第一标志包括在视频比特流中,第一标志指示帧速率上转换;以及将第二标志包括在视频比特流中,第二标志进一步指示帧速率上转换。

[0088] 当处理器510被配置用于解码时,它执行图4的步骤,即通过解析视频比特流以识别第一标志来解码视频比特流;解析视频比特流以识别第二标志,以及基于第一标志和第二标志对视频图像的一部分执行帧速率上转换。

[0089] 附图中所示的各种元件的功能可以通过使用专用硬件以及能够与适当软件相关联地执行软件的硬件来提供。当由处理器提供时,功能可以由单个专用处理器、由单个共享处理器或由多个单独的处理器提供,其中一些处理器可以被共享。此外,术语“处理器”或“控制器”的显式使用不应被解释为专门指能够运行软件的硬件,并且可以隐含地包括但不限于数字信号处理器(“DSP”)硬件、用于存储软件的只读存储器(“ROM”)、随机存取存储器

(“RAM”) 和非易失性存储器。

[0090] 也可以包括传统的和/或定制的其他硬件。类似地,图中所示的任何开关都只是概念性的。它们的功能可以通过程序逻辑的操作、通过专用逻辑、通过程序控制和专用逻辑的交互或者甚至手动来实现,具体技术可以由实施者选择,如从上下文中更具体地理解的。

[0091] 本描述说明了本原理。因此,应当理解,本领域的技术人员将能够设计各种布置,尽管在此没有明确描述或示出,但是这些布置体现了本发明的原理,并且被包括在其范围内。

[0092] 本文所列举的所有示例和条件语言都是为了教学目的,以帮助读者理解本发明的原理和发明人为推进本领域所贡献的概念,并被解释为不限于这些具体列举的示例和条件。

[0093] 此外,本文列举本原理的原理、方面和实施例的所有陈述,以及其具体示例,都意图涵盖其结构和功能等同物。此外,这种等同物意图包括当前已知的等同物以及将来开发的等同物,即无论结构如何,开发的执行相同功能的任何元件。

[0094] 因此,例如,本领域技术人员将会理解,本文呈现的框图表示体现本原理的说明性电路的概念图。类似地,将会理解,任何流程图表、流程图、状态转换图、伪代码等表示可以基本上在计算机可读介质中表示并由计算机或处理器执行的各种处理,无论是否明确示出了这样的计算机或处理器。

[0095] 在其权利要求中,表示为用于执行特定功能的装置的任何元件意图涵盖执行该功能的任何方式,包括例如a) 执行该功能的电路元件的组合,或者b) 任何形式的软件,因此包括固件、微码等,与用于运行该软件以执行该功能的适当电路相结合。由这样的权利要求定义的本原理在于这样的事实,即由各种列举的装置提供的功能以权利要求所要求的方式被组合和集合在一起。因此,可以认为能够提供这些功能的任何装置都等同于本文所示的那些装置。

[0096] 说明书中对本原理的“一个实施例”或“实施例”以及其他变型的引用意味着结合该实施例描述的特定特征、结构、特性等被包括在本原理的至少一个实施例中。因此,在整个说明书的不同地方出现的短语“在一个实施例中”或“在实施例中”以及任何其他变型不一定都指同一实施例。

[0097] 总之,上述实施例示出了对视频图像的一部分执行帧速率上转换的改进方法和装置。在至少一个实施例中,编码器可以向解码器信令通知是否仅对视频图像的一部分或编码单元的子部分使用帧速率上转换。另外,提供标志以在编码器或解码器中使用帧速率上转换处理的子处理。

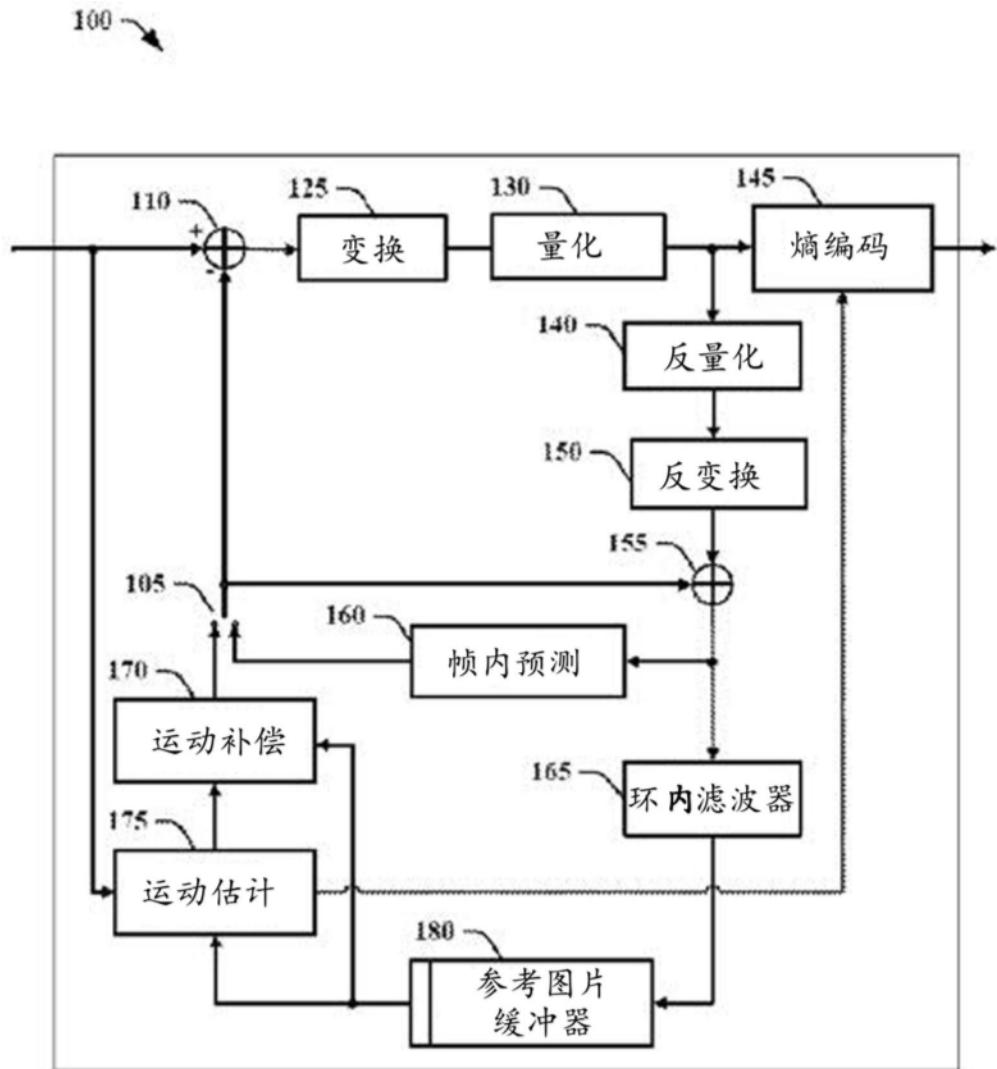


图1

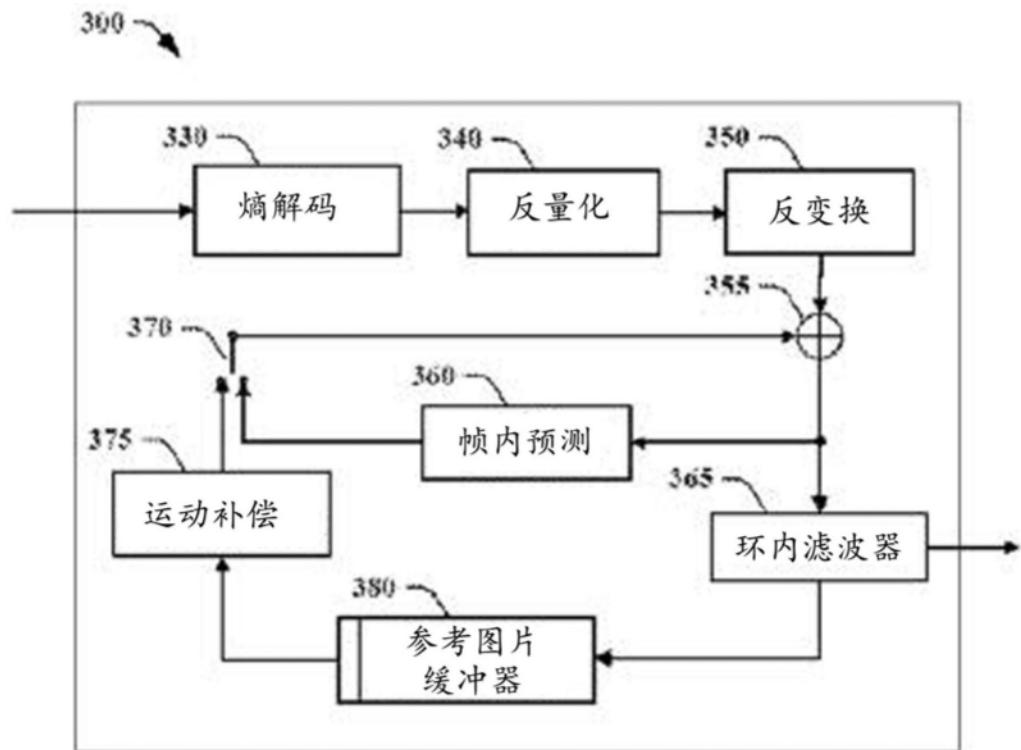


图2

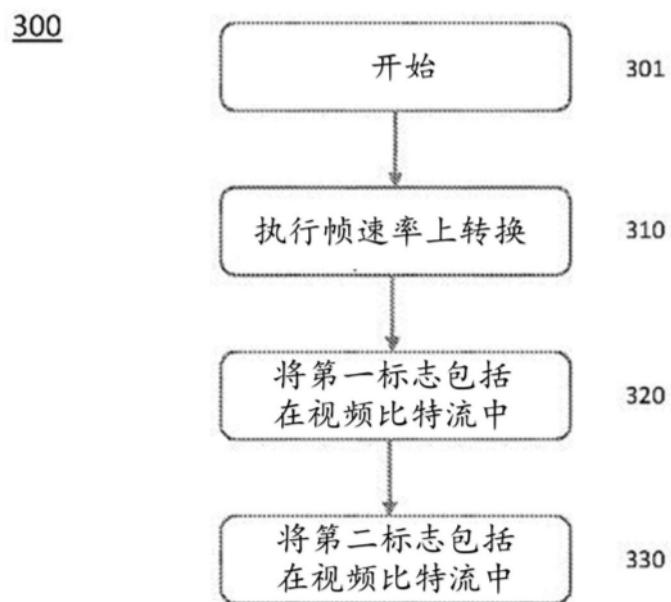


图3

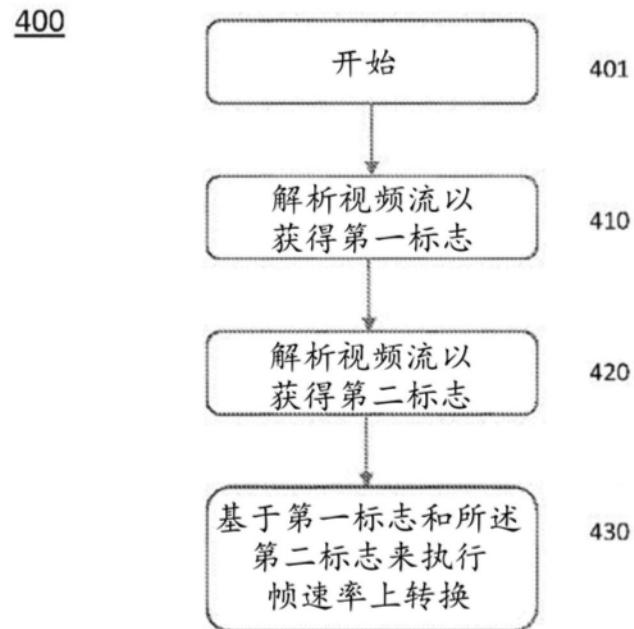


图4

500

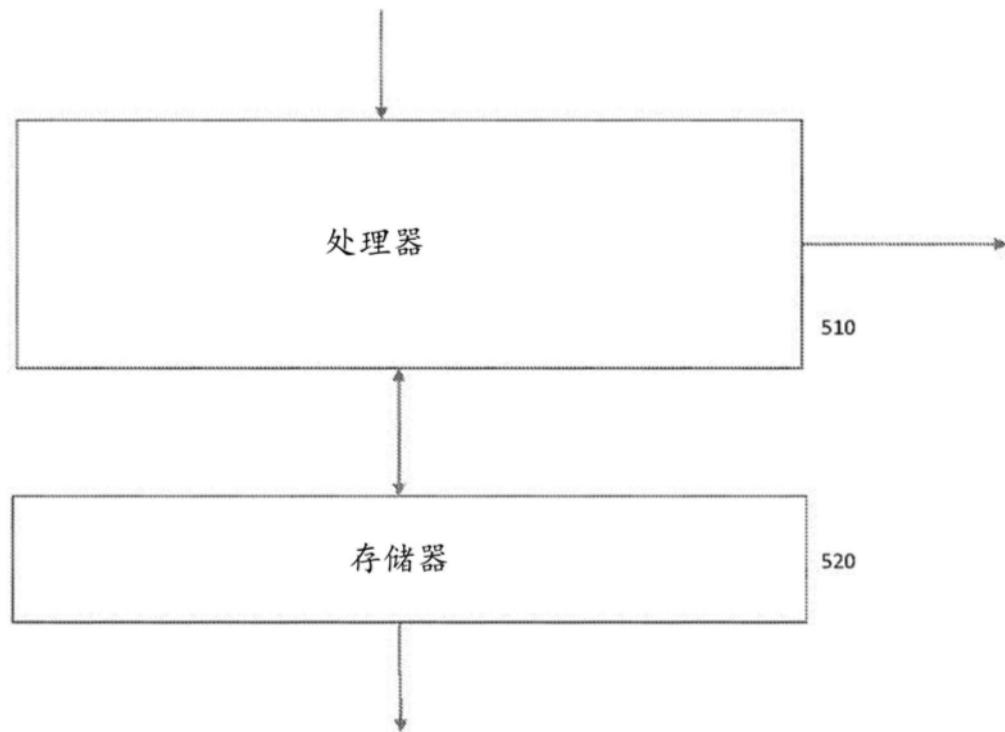


图5