

[19] Patents Registry [11] 1213402 B
The Hong Kong Special Administrative Region
香港特別行政區
專利註冊處

[12] **STANDARD PATENT SPECIFICATION**
標準專利說明書

[21] Application no. 申請編號 [51] Int. Cl.
16101202.4 H04N
[22] Date of filing 提交日期
02.02.2016

[54] METHODS AND APPARATUS FOR VIDEO ENCODERS AND DECODERS
用於視頻編碼器和解碼器的方法和裝置

[30] Priority 優先權 01.07.2009 US 61/222,177	[73] Proprietor 專利所有人 Thomson Licensing 湯姆森特許公司 1-5, rue Jeanne d'Arc 92130 Issy-les-Moulineaux FRANCE
[43] Date of publication of application 申請發表日期 30.06.2016	[72] Inventor 發明人 ZHENG, Yunfei 鄭雲飛 XU, Qian 許茜 LU, Xiaoan 呂小安 YIN, Peng 尹鵬 SOLE, Joel J·索爾 ABBAS, Adeel A·阿巴斯
[45] Date of publication of grant of patent 批予專利的發表日期 30.08.2019	[74] Agent and / or address for service 代理人及/或送達地址 Liu, Shen & Associates Room 3716, 37th Floor Sun Hung Kai Centre 30 Harbour Road Wanchai, Hong Kong
CN Application no. & date 中國專利申請編號及日期 CN 201510242361.1 29.06.2010	
CN Publication no. & date 中國專利申請發表編號及日期 CN 104853207 19.08.2015	
Date of grant in designated patent office 指定專利當局批予專利日期 26.06.2018	



(12)发明专利

(10)授权公告号 CN 104853207 B

(45)授权公告日 2018.06.26

(21)申请号 201510242361.1

(51)Int.Cl.

(22)申请日 2010.06.29

H04N 19/176(2014.01)

(65)同一申请的已公布的文献号

H04N 19/70(2014.01)

申请公布号 CN 104853207 A

H04N 19/147(2014.01)

(43)申请公布日 2015.08.19

H04N 19/196(2014.01)

(30)优先权数据

H04N 19/96(2014.01)

61/222,177 2009.07.01 US

H04N 19/11(2014.01)

(62)分案原申请数据

H04N 19/463(2014.01)

201080038907.7 2010.06.29

H04N 19/136(2014.01)

(73)专利权人 汤姆森特许公司

(56)对比文件

地址 法国伊西莱穆利诺

US 5060285 A, 1991.10.22,

(72)发明人 郑云飞 许茜 吕小安 尹鹏

CN 101273641 A, 2008.09.24,

J.索尔 A.阿巴斯

US 2008123977 A1, 2008.05.29,

审查员 孙婧

(74)专利代理机构 北京市柳沈律师事务所

权利要求书2页 说明书14页 附图9页

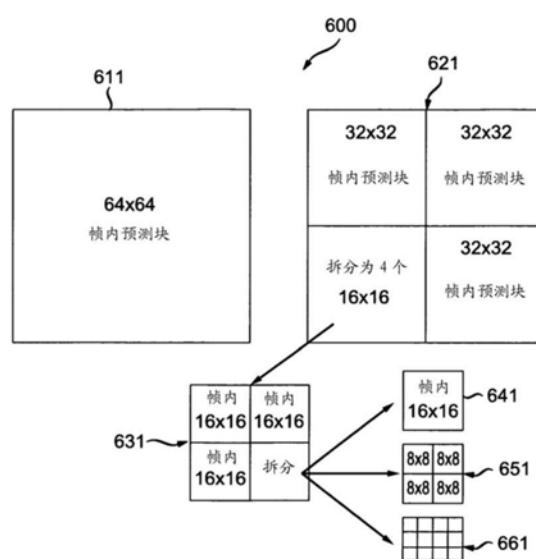
代理人 吕晓章 叶齐峰

(54)发明名称

用于视频编码器和解码器的方法和装置

(57)摘要

提供了用于视频编码器和解码器的对大块的帧内预测进行信令的方法和装置。装置包括视频编码器(400)，所述视频编码器通过对用于画面中的至少一个大块的帧内预测进行信令来编码所述至少一个大块的画面数据。通过选择基本编码单元尺寸并且分配用于基本编码单元尺寸的单个空间帧内划分类型来对帧内预测进行信令。该单个空间帧内划分类型是可从多个空间帧内划分类型中选择的。所述至少一个大块具有比基本编码单元的块尺寸大的大块尺寸。帧内预测是分层级的帧内预测并且通过以下操作中的至少一个而对至少一个大块执行：将大块尺寸拆分为基本编码单元尺寸以及从基本编码单元尺寸合并到大块尺寸。



1. 一种视频编码装置中的方法,所述方法包括:

通过确定要为画面中的大块执行帧内预测来编码所述大块的画面数据,其中所述大块具有比基本编码单元尺寸大的块尺寸,所述大块尺寸是 32×32 和 64×64 中的一个,且所述基本编码单元尺寸是 16×16 ,

其中对于所述大块,通过以下操作信令所述帧内预测:

编码二进制拆分信令语法元素,所述二进制拆分信令语法元素指定是否所述大块被进一步拆分为四个尺寸相等的子块;

在所述二进制拆分信令语法元素指定所述大块不被进一步拆分的情形下,编码用于所述大块的帧内预测模式;

否则在所述二进制拆分信令语法元素指定所述大块被进一步拆分的情形下:

对于每个子块,在所述子块是 32×32 的情形下,编码指定是否所述 32×32 的子块被进一步拆分为四个基本编码单元尺寸相等的块的二进制拆分信令语法元素,以及在所述指定是否所述 32×32 的子块被进一步拆分为四个基本编码单元尺寸相等的块的二进制拆分信令语法元素指定所述 32×32 的子块不被进一步拆分的情形下,编码用于所述 32×32 的子块的帧内预测模式;以及

对于每个子块,在所述子块是 16×16 的情形下,编码单个空间帧内划分类型,所述单个空间帧内划分类型是从多个空间帧内划分类型中可确定的。

2. 如权利要求1所述的方法,还包括:

编码至少一个二进制合并信令语法元素,所述至少一个二进制合并信令语法元素指定是否基本编码单元尺寸的块被合并为大尺寸的块。

3. 如权利要求1所述的方法,其中空间帧内划分类型表和帧内预测模式表中的至少一个是预先存储的并由所述编码方法使用以编码所述大块。

4. 如权利要求1所述的方法,其中空间帧内划分类型表和帧内预测模式表中的至少一个是由所述编码方法使用一个或多个高级语法元素编码的,并由所述编码方法使用以编码所述大块。

5. 一种视频解码装置中的方法,包括:

通过确定要为画面中的大块执行帧内预测来解码所述大块的画面数据,

其中,所述大块具有比基本编码单元尺寸大的块尺寸,所述大块尺寸是 32×32 和 64×64 中的一个,且所述基本编码单元尺寸是 16×16 ,

其中对于所述大块,通过以下操作信令所述帧内预测:

解码二进制拆分信令语法元素,所述二进制拆分信令语法元素指定是否所述大块被进一步拆分为四个尺寸相等的子块;

在所述二进制拆分信令语法元素指定所述大块不被进一步拆分的情形下,解码用于所述大块的帧内预测模式;

否则在所述二进制拆分信令语法元素指定所述大块被进一步拆分的情形下:

对于每个子块,在所述子块是 32×32 的情形下,解码指定是否所述 32×32 的子块被进一步拆分为四个基本编码单元尺寸相等的块的二进制拆分信令语法元素,以及在所述指定是否所述 32×32 的子块被进一步拆分为四个基本编码单元尺寸相等的块的二进制拆分信令语法元素指定所述 32×32 的子块不被进一步拆分的情形下,解码用于所述 32×32 的子块

的帧内预测模式；以及

对于每个子块，在所述子块是 16×16 的情形下，解码单个空间帧内划分类型，所述单个空间帧内划分类型是从多个空间帧内划分类型中可确定的。

6. 如权利要求5所述的方法，还包括：

解码至少一个二进制合并信令语法元素，所述至少一个二进制合并信令语法元素指定是否基本编码单元尺寸的块被合并为大尺寸的块。

7. 如权利要求5所述的方法，其中空间帧内划分类型表和帧内预测模式表中的至少一个是预先存储的并由所述解码方法使用以解码所述大块。

8. 如权利要求5所述的方法，其中空间帧内划分类型表和帧内预测模式表中的至少一个是由所述解码方法使用一个或多个高级语法元素接收的，并由所述解码方法使用以解码所述大块。

用于视频编码器和解码器的方法和装置

[0001] 本申请是申请日为2010年6月29日、申请号为201080038907.7、发明名称为“用于视频编码器和解码器的对大块的帧内预测进行信令的方法和装置”的发明专利申请的分案申请。

[0002] 相关申请的交叉引用

[0003] 本申请要求2009年7月1日提交的美国临时申请序列号No.61/222,177(代理案号No.PU090082)的权益,通过引用将其内容全部并入于此。

技术领域

[0004] 本原理一般地涉及视频编码和解码,并且更具体地涉及用于视频编码器和解码器的对大块的帧内预测进行信令(signal)的方法和装置。

背景技术

[0005] 多数现代视频编码标准采用各种编码模式来有效地减少空间域和时间域中的相关度。例如,在国际标准化组织/国际电工委员会(ISO/IEC)运动画面专家组-4(MPEG-4)第10部分高级视频编码(AVC)标准/国际电信联盟电信分部ITU-T H.264推荐(下文的“MPEG-4 AVC标准”)中,可以帧内编码或者帧间编码画面。在帧内画面中,以帧内模式编码所有宏块,由此利用画面内的空间相关度。帧内模式可以被归类为以下三种类型:INTRA4×4;INTRA8×8;INTRA16×16。INTRA4×4和INTRA8×8支持9种帧内预测模式,INTRA16×16支持4种帧内预测模式。

[0006] INTRA 4×4和INTRA8×8支持以下9种帧内预测模式:垂直预测、水平预测、DC预测、对角下/左预测、对角下/右预测、垂直-左预测、水平-下预测、垂直-右预测,以及水平-上预测。INTRA16×16支持以下4种帧内预测模式:垂直预测、水平预测、DC预测,以及平面预测。转到图1,由参考标号100总地指示INTRA4×4和INTRA8×8预测模式。在图1中,参考标号0指示垂直预测模式、参考标号1指示水平预测模式、参考标号3指示对角下/左预测模式、参考标号4指示对角下/右预测模式、参考标号5指示垂直-右预测模式、参考标号6指示水平-下预测模式、参考标号7指示垂直-左预测模式、参考标号8指示垂直-上预测模式。未示出作为INTRA4×4和INTRA8×8预测模式一部分的DC模式。转到图2,由参考标号200总地指示INTRA16×16预测模式。在图2中,参考标号0指示垂直预测模式、参考标号1指示水平预测模式、参考标号3指示平面预测模式。未示出作为INTRA16×16预测模式一部分的DC模式。

[0007] INTRA4×4使用4×4离散余弦变换(DCT)。INTRA8×8使用8×8变换。INTRA16×16使用级联的4×4变换。为了进行信令,INTRA4×4和INTRA8×8共享相同的宏块类型(mb_type)0并且通过变换尺寸标志(transform_8×8_size_flag)来区分。然后,通过最可能的模式(如果必要,可能利用其余模式)来对在INTRA4×4或INTRA8×8中帧内预测模式的选取进行信令。对于INTRA16×16,在mb_type中对所有帧内预测模式连同编码块图案(cbp)类型进行信令,其使用1到24的mb_type值。表1示出了用于帧内编码码片(I码片)的宏块类型的详细的信令。如果尺寸大于16×16的更大的块用于帧内预测,则面对如下的若干可能的问题。

[0008] (1) 如果通过在MPEG-4 AVC标准中简单地扩展mb_type来增加INTRA32×32或者INTRA64×64预测，则其将对这两种新模式造成太多的开销，并且另外，将不允许帧内预测的分级类型。如下解释帧内预测的分级类型的示例。如果32×32块用作大块并且允许子划分为16×16，则对于每个16×16子划分，应允许INTRA4×4、INTRA8×8、INTRA16×16。

[0009] (2) 如果更大的变换(诸如16×16变换)而不是级联的变换用于INTRA16×16，则不能应用当前的信令。

[0010] (3) 应对一个帧内划分类型内部的帧内预测模式给出不同的优先级。

[0011] 表1

mb_type	mb_type 的名称	transform_size _8x8_flag	Mb划分预测模式 (mb_type, 0)	Intra16x16- 预测模式	编码块 图案色度	编码块 图案色度
0	I_NxN	0	Intra_4x4	na	方程式 7-33	方程式 7-33
0	I_NxN	1	Intra_8x8	na	方程式 7-33	方程式 7-33
1	I_16x16_0_0_0	na	Intra_16x16	0	0	0
2	I_16x16_1_0_0	na	Intra_16x16	1	0	0
3	I_16x16_2_0_0	na	Intra_16x16	2	0	0
4	I_16x16_3_0_0	na	Intra_16x16	3	0	0
5	I_16x16_0_1_0	na	Intra_16x16	0	1	0
6	I_16x16_1_1_0	na	Intra_16x16	1	1	0
7	I_16x16_2_1_0	na	Intra_16x16	2	1	0
8	I_16x16_3_1_0	na	Intra_16x16	3	1	0
9	I_16x16_0_2_0	na	Intra_16x16	0	2	0
10	I_16x16_1_2_0	na	Intra_16x16	1	2	0
11	I_16x16_2_2_0	na	Intra_16x16	2	2	0
12	I_16x16_3_2_0	na	Intra_16x16	3	2	0
13	I_16x16_0_0_1	na	Intra_16x16	0	0	15
14	I_16x16_1_0_1	na	Intra_16x16	1	0	15
15	I_16x16_2_0_1	na	Intra_16x16	2	0	15
16	I_16x16_3_0_1	na	Intra_16x16	3	0	15
17	I_16x16_0_1_1	na	Intra_16x16	0	1	15
18	I_16x16_1_1_1	na	Intra_16x16	1	1	15
19	I_16x16_2_1_1	na	Intra_16x16	2	1	15
20	I_16x16_3_1_1	na	Intra_16x16	3	1	15
21	I_16x16_0_2_1	na	Intra_16x16	0	2	15
22	I_16x16_1_2_1	na	Intra_16x16	1	2	15
23	I_16x16_2_2_1	na	Intra_16x16	2	2	15
24	I_16x16_3_2_1	na	Intra_16x16	3	2	15
25	I_PCM	na	na	na	na	na

[0013] MPEG-4 AVC标准的扩展中存在与对大的运动(帧间)划分进行信令有关的一些现

有技术方法。关于第一种现有技术方法来描述在MPEG-4 AVC标准的扩展中怎样对大的运动(帧间)划分进行信令的一个示例。第一种现有技术方法描述了怎样为使用分级编码结构的 32×32 块或者 64×64 块进行信令。

[0014] 此外,除了MPEG-4 AVC标准中现有的运动划分尺寸(16×16 、 16×8 、 8×16 、 8×8 、 8×4 、 4×8 和 4×4)之外,也已经提出使用 32×32 、 32×16 和 16×32 划分的用于MPEG-4 AVC标准的扩展的帧间编码。转到图3,通过参考标号300总地指示用于 32×32 块中的运动划分。划分包括 32×32 、 32×16 、 16×32 和 16×16 。 16×16 划分可以进一步被划分为尺寸 16×16 、 16×8 、 8×16 和 8×8 的划分。此外, 8×8 划分可以进一步被划分为尺寸 8×8 、 8×4 、 4×8 和 4×4 的划分。

[0015] 对于每个 32×32 的块,以对MPEG-4 AVC标准的其它模式执行的方式类似的方式使用mb32_skip_flag来对SKIP模式或者DIRECT模式进行信令。另外,MPEG-4 AVC标准中的 $M \times N$ ($M=8$ 或 16 而 $N=8$ 或 16)划分的原始mb_type还用于对 32×32 块中的 $2M \times 2N$ 划分进行信令。如果 32×32 的mb32_type指示使用 16×16 划分,则通过使用与MPEG-4 AVC标准中的macroblock_layer()相同的语法元素,以光栅扫描顺序来对四个 16×16 块进行信令。可以进一步以四叉树方式将每个 16×16 块从尺寸 16×16 向下划分为尺寸 4×4 。

[0016] 对于宏块尺寸 64×64 ,在 32×32 块中使用的划分之上添加了以下划分: 64×64 、 64×32 和 32×64 。由此,在块尺寸 32×32 之上的宏块划分中添加了超过一个分级的层。MPEG-4 AVC标准中的 $M \times N$ ($M=8$ 或 16 而 $N=8$ 或 16)宏块划分的原始的mb_type用于对 64×64 宏块中的 $4M \times 4N$ 宏块划分进行信令。如果 32×32 宏块划分用于 64×64 块,则每个 32×32 块将以与上述相同的方式被处理。

[0017] 然而,现有文献没有解决怎样对大的帧内模式进行信令,其中大的帧内模式被定义为意指涉及具有等于或者大于 32×32 的尺寸的划分块的帧内预测。

发明内容

[0018] 通过本原理解决现有技术的这些和其它缺陷和缺点,本原理针对用于视频编码器和解码器的对大块的帧内预测进行信令的方法和装置。

[0019] 根据本原理的一个方面,提供了一种装置。该装置包括视频编码器,所述视频编码器通过对用于画面中的至少一个大块的帧内预测进行信令来编码所述至少一个大块的画面数据。通过选择基本编码单元尺寸并且分配用于基本编码单元尺寸的单个空间帧内划分类型来对帧内预测进行信令。该单个空间帧内划分类型是可从多个空间帧内划分类型中选择的。所述至少一个大块具有比基本编码单元的块尺寸大的大块尺寸。帧内预测是分层级的帧内预测并且通过以下操作中的至少一个而对至少一个大块执行:将大块尺寸拆分为基本编码单元尺寸以及从基本编码单元尺寸合并到大块尺寸。

[0020] 根据本原理的另一方面,提供了一种视频编码器中的方法。该方法包括通过对用于画面中的至少一个大块的帧内预测进行信令来编码所述至少一个大块的画面数据。通过选择基本编码单元尺寸并且分配用于基本编码单元尺寸的单个空间帧内划分类型来对帧内预测进行信令。该单个空间帧内划分类型是可从多个空间帧内划分类型中选择的。所述至少一个大块具有比基本编码单元的块尺寸大的大块尺寸。帧内预测是分层级的帧内预测并且通过以下操作中的至少一个而对至少一个大块执行:将大块尺寸拆分为基本编码单元

尺寸以及从基本编码单元尺寸合并到大块尺寸。

[0021] 根据本原理的又一方面,提供了一种装置。该装置包括视频解码器,所述视频解码器通过确定要为画面中的至少一个大块执行的帧内预测来解码所述至少一个大块的画面数据。通过确定基本编码单元尺寸并且确定用于基本编码单元尺寸的单个空间帧内划分类型来确定帧内预测。该单个空间帧内划分类型是可从多个空间帧内划分类型中确定的。所述至少一个大块具有比基本编码单元的块尺寸大的大块尺寸。帧内预测是分层级的帧内预测并且通过以下操作中的至少一个而对至少一个大块执行:将大块尺寸拆分为基本编码单元尺寸以及从基本编码单元尺寸合并到大块尺寸。

[0022] 根据本原理的再一方面,提供了一种视频解码器中的方法。该方法包括通过确定要为画面中的至少一个大块执行的帧内预测来解码所述至少一个大块的画面数据。通过确定基本编码单元尺寸并且确定用于基本编码单元尺寸的单个空间帧内划分类型来确定帧内预测。该单个空间帧内划分类型是可从多个空间帧内划分类型中确定的。所述至少一个大块具有比基本编码单元的块尺寸大的大块尺寸。帧内预测是分层级的帧内预测并且通过以下操作中的至少一个而对至少一个大块执行:将大块尺寸拆分为基本编码单元尺寸以及从基本编码单元尺寸合并到大块尺寸。

[0023] 本原理的这些和其它方面、特征和优点将从示例实施例的以下具体描述中变得明显,将结合附图阅读以下具体描述。

附图说明

- [0024] 依据以下示例性图将更好地理解本原理,其中:
- [0025] 图1是示出可以应用本原理的INTRA4×4和INTRA8×8预测模式的图;
- [0026] 图2是示出可以应用本原理的INTRA16×16预测模式的图;
- [0027] 图3是示出可以应用本原理的用于32×32块的运动划分的图;
- [0028] 图4是依据本原理实施例的可以应用本原理的示例性视频编码器的框图;
- [0029] 图5是依据本原理实施例的可以应用本原理的示例性视频解码器的框图;
- [0030] 图6是依据本原理实施例的可以应用本原理的示例性分级划分的框图;
- [0031] 图7A和7B表示依据本原理实施例的通过对用于大块的帧内预测进行信令来编码所述大块的画面数据的示例性方法的流程图;以及
- [0032] 图8A和8B表示依据本原理实施例的通过确定要被应用到大块的帧内预测来解码所述大块的画面数据的示例性方法的流程图。

具体实施方式

[0033] 本原理针对用于视频编码器和解码器的对大块的帧内预测进行信令的方法和装置。

[0034] 本描述例示了本原理。因此,将理解,本领域技术人员将能够开发未在这里明确描述或示出但是体现本原理并且被包括在本原理的精神和范围之内的各种布置。

[0035] 在此叙述的所有示例和条件性语言意欲用于教导的目的以便帮助读者理解本原理以及由(多个)发明人贡献以推动本领域发展的构思,并且应该被解释为不局限于这样具体叙述的示例和条件。

[0036] 另外,在这里叙述本原理的原理、方面和实施例的所有陈述,及其具体示例意欲包括其结构和功能上的等同物。另外,意图是:这样的等同物包括当前已知的等同物以及将来开发的等同物二者,即所开发的执行相同功能的任何元件,而不论其结构如何。

[0037] 因此,例如,本领域技术人员将认识到:在此呈现的框图表示体现本原理的示意性电路的概念性视图。类似地,将认识到:任何流程图示(flow chart)、流程图(flow diagram)、状态转换图、伪代码等表示实质上可以表示在计算机可读介质中并因此由计算机或处理器执行的各种处理,而不管是否明确地示出这样的计算机或处理器。

[0038] 可以通过使用专用硬件、以及与适当的软件相关联的能够执行软件的硬件来提供图中示出的各种元件的功能。当利用处理器来提供所述功能时,可以利用单个专用处理器、利用单个共享处理器、或者利用其中一些可被共享的多个独立处理器来提供所述功能。另外,术语“处理器”或“控制器”的明确使用不应该被解释为排他性地指代能够执行软件的硬件,而是可以隐含地不受限制地包括数字信号处理器(DSP)硬件、用于存储软件的只读存储器(“ROM”)、随机存取存储器(RAM)、和非易失性存储装置。

[0039] 还可以包括其它传统的和/或定制的硬件。类似地,图中示出的任何开关只是概念性的。它们的功能可以通过程序逻辑的运行、通过专用逻辑、通过程序控制和专用逻辑的交互、或者甚至手动地来执行,如从上下文更具体地理解的,实施者可选择具体技术。

[0040] 在其权利要求中,被表示为用于执行指定功能的部件的任何元件意欲包含执行那个功能的任何方式,例如包括:a)执行那个功能的电路元件的组合或者b)与适当电路相组合的任何形式的软件,所述软件因此包括固件或微代码等,所述适当电路用于执行该软件以执行所述功能。由这种权利要求限定的本发明在于如下事实,即,以权利要求所要求的方式将由各种所叙述的部件提供的功能组合和集合到一起。因此认为可以提供那些功能的任何部件与在此示出的那些部件等同。

[0041] 在本说明书中提到的本原理的“一个实施例”或“实施例”及其其它变型意味着:结合所述实施例描述的具体特征、结构、特性等被包括在本原理的至少一个实施例中。因此,在说明书各处出现的短语“在一个实施例中”和“在实施例中”、以及任何其它变型不一定都指代相同的实施例。

[0042] 应当认识到,“/”、“和/或”以及“至少一个”任何一个的使用,例如在“A/B”、“A和/或B”和“A和B中的至少一个”的情况中,意欲包括仅仅对第一个列出的选项(A)的选择、或仅仅对第二个列出的选项(B)的选择、或者对于两个选项(A和B)的选择。作为另一示例,在“A、B和/或C”以及“A、B和C中的至少一个”的情况中,这种措辞意欲包括仅仅对第一个列出的选项(A)的选择、或仅仅对第二个列出的选项(B)的选择、或仅仅对第三个列出的选项(C)的选择、或仅仅对第一个和第二个列出的选项(A和B)的选择、或仅仅对第一个和第三个列出的选项(A和C)的选择、或仅仅对第二个和第三个列出的选项(B和C)的选择、或者对于全部三个选项(A和B和C)的选择。如本领域和相关领域普通技术人员容易认识到的,这可以被扩展用于很多列出的条目。

[0043] 此外,应理解,尽管这里关于MPEG-4 AVC标准的扩展来描述本原理的一个或多个实施例,但本原理不仅仅限于该扩展或者该标准,并且因此可以关于其它视频编码标准、推荐、及其扩展而被利用,同时保持本原理的精神。

[0044] 如这里所使用的,“高级语法”指代在分级上高于宏块层驻留的比特流中存在的语

法。例如,如这里所使用的,高级语法可以指代但不限于:码片首标级、补充增强信息(SEI)级、画面参数集(PPS)级、序列参数集(SPS)级、和网络抽象层(NAL)单元首标级处的语法。

[0045] 而且,如这里所使用的,词语“画面”和“图像”被可互换地使用,并且指代静止图像或者来自视频序列的画面。如已知的,画面可以是帧或场。

[0046] 此外,如这里所使用的,词语“信令”指代向对应解码器指示某些内容(something)。例如,编码器可以对被指定用于特定的大块(如在此定义的)的帧内预测进行信令以便使得解码器得知在编码器侧使用了特定的预测类型(例如,帧内或者帧间)。以此方式,可以在编码器侧和解码器侧使用相同的预测类型。由此,例如,编码器可以对特定的大块传送关于在该大块上要执行帧内预测的指示(例如,信号),以便简单地使得解码器知道并且选择用于该大块的相同的预测类型。应理解,可以以多种方式来实现该信令。例如,可以使用一个或多个语法元素、标志等来向对应解码器对信息进行信令。

[0047] 转到图4,由参考标号400总地指示依据本原理实施例的可以应用本原理的示例性视频编码器。

[0048] 视频编码器400包括帧排序缓冲器410,该帧排序缓冲器410具有与组合器485的非反相输入端进行信号通信的输出端。组合器485的输出端以信号通信方式与变换器和量化器425的第一输入端连接。变换器和量化器425的输出端以信号通信方式与熵编码器445的第一输入端以及逆变换器和逆量化器450的第一输入端连接。熵编码器445的输出端以信号通信方式与组合器490的第一非反相输入端连接。组合器490的输出端以信号通信方式与输出缓冲器435的第一输入端连接。

[0049] 编码器控制器405的第一输出端以信号通信方式与帧排序缓冲器410的第二输入端、逆变换器和逆量化器450的第二输入端、画面类型判定模块415的输入端、宏块类型(MB类型)判定模块420的输入端、超级帧内预测模块460的第二输入端、去块滤波器465的第二输入端、运动补偿器470的第一输入端、运动估计器475的第一输入端、和参考画面缓冲器480的第二输入端连接。

[0050] 编码器控制器405的第二输出端以信号通信方式与补充增强信息(SEI)插入器430的第一输入端、变换器和量化器425的第二输入端、熵编码器445的第二输入端、输出缓冲器435的第二输入端、以及序列参数集(SPS)和画面参数集(PPS)插入器440的输入端连接。

[0051] 画面类型判定模块415的第一输出端以信号通信方式与帧排序缓冲器410的第三输入端连接。画面类型判定模块415的第二输出端以信号通信方式与宏块类型判定模块420的第二输入端连接。

[0052] 序列参数集(SPS)和画面参数集(PPS)插入器440的输出端以信号通信方式与组合器490的第三非反相输入端连接。

[0053] 逆量化器和逆变换器450的输出端以信号通信方式与组合器419的第一非反相输入端连接。组合器419的输出端以信号通信方式与超级帧内预测模块460的第一输入端和去块滤波器465的第一输入端连接。去块滤波器465的输出端以信号通信方式与参考画面缓冲器480的第一输入端连接。参考画面缓冲器480的输出端以信号通信方式与运动估计器475的第二输入端连接。运动估计器475的第一输出端以信号通信方式与运动补偿器470的第二输入端连接。运动估计器475的第二输出端以信号通信方式与熵编码器445的第三输入端连接。

[0054] 运动补偿器470的输出端以信号通信方式与开关497的第一输入端连接。超级帧内预测模块460的输出端以信号通信方式与开关497的第二输入端连接。宏块类型判定模块420的输出端以信号通信方式与开关497的第三输入端连接。开关497的第三输入端确定开关的“数据”输入(与控制输入(即,第三输入端)相对)是由运动补偿器470提供还是由超级帧内预测模块460提供。开关497的输出端以信号通信方式与组合器419的第二非反相输入端和组合器485的反向输入端连接。

[0055] 帧排序缓冲器410和编码器控制器405的输入端可用作编码器400的用于接收输入画面401的输入端。此外,补充增强信息(SEI)插入器430的输入端可用作编码器400的用于接收元数据的输入端。输出缓冲器435的输出端可用作编码器400的用于输出比特流的输出端。

[0056] 转到图5,通过参考标号500总地指示依据本原理实施例的可以应用本原理的示例性视频解码器。

[0057] 视频解码器500包括输入缓冲器510,该输入缓冲器510具有以信号通信方式与熵解码器545的第一输入端连接的输出端。熵解码器545的第一输出端以信号通信方式与逆变换器和逆量化器550的第一输入端连接。逆变换器和逆量化器550的输出端以信号通信方式与组合器525的第二非反相输入端连接。组合器525的输出端以信号通信方式与去块滤波器565的第二输入端和超级帧内预测模块560的第一输入端连接。去块滤波器565的第二输出端以信号通信方式与参考画面缓冲器580的第一输入端连接。参考画面缓冲器580的输出端以信号通信方式与运动补偿器570的第二输入端连接。

[0058] 熵解码器545的第二输出端以信号通信方式与运动补偿器570的第三输入端和去块滤波器565的第一输入端连接。熵解码器545的第三输出端以信号通信方式与解码器控制器505的第一输入端连接。解码器控制器505的第一输出端以信号通信方式与熵解码器545的第二输入端连接。解码器控制器505的第二输出端以信号通信方式与逆变换器和逆量化器550的第二输入端连接。解码器控制器505的第三输出端以信号通信方式与去块滤波器565的第三输入端连接。解码器控制器505的第四输出端以信号通信方式与超级帧内预测模块560的第二输入端、运动补偿器570的第一输入端、以及参考画面缓冲器580的第二输入端连接。

[0059] 运动补偿器570的输出端以信号通信方式与开关597的第一输入端连接。超级帧内预测模块560的输出端以信号通信方式与开关597的第二输入端连接。开关597的输出端以信号通信方式与组合器525的第一非反相输入端连接。

[0060] 输入缓冲器510的输入端可用作解码器500的用于接收输入比特流的输入端。去块滤波器565的第一输出端可用作解码器500的用于对输出画面进行输出的输出端。

[0061] 如上注意的,本原理针对用于视频编码器和解码器的对大块的帧内预测进行信令的方法和装置。此外,如上注意的,可以应用本原理的大块被定义为意指具有等于或者大于 32×32 的尺寸的块。

[0062] 在实施例中,为了易于标注,把帧内预测的信令拆分为以下两部分:sip_type(空间帧内划分类型,其可以是INTRA4×4、INTRA8×8、INTRA16×16等等);以及每个sip_type内的intra_pred_mode(诸如,例如INTRA4×4和INTRA8×8内的9种帧内预测模式)。关于特定实施例进一步的详述中,提出了用于本原理的以下三个规则:(1)选择基本编码单元;(2)通过从最大的帧内预测类型中拆分或者从基本编码单元中合并而允许分层级的帧内预测;

以及(3)对于每个sip_type,向最频繁使用的intra_pred_mode分配较高的优先级。关于规则(1),允许若干sip_type用于基本的编码单元。

[0063] 一实施例

[0064] 在一实施例中,将基本编码单元设置为 16×16 。在该编码单元中,允许sip_type为INTRA4 $\times 4$ 、INTRA8 $\times 8$ 、INTRA16 $\times 16$ 。还允许分层级的帧内预测,如图6所示。

[0065] 转到图6,通过参考标号600总地指示可以应用本原理的示例性的分级划分。在该实施例中,如果最大的块尺寸被设置为 64×64 ,则使用“拆分信令”来允许分层级的帧内预测。也就是说,在实施例中,添加了intra64_flag。如果intra64_flag等于1,则使用INTRA64 $\times 64$ 。否则,如果intra64_flag等于0,则将 64×64 块611拆分为四个 32×32 块621。对于每个 32×32 块621,添加intra32_flag。如果intra32_flag等于1,则使用INTRA32 $\times 32$ 。否则,如果intra32_flag等于0,则在此(例如对于 32×32 块621)同样允许在 16×16 基本编码单元中所允许的所有sip_type。对于INTRA16 $\times 16$ 中的intra_pred_mode,具有DC模式和定向模式(directional mode),后者通过发送模式信息而允许不同类型的定向预测。由此, 32×32 帧内预测块621可以被进一步拆分为4个 16×16 帧内预测块631。4个 16×16 帧内预测块631中的一个或多个可以进一步被拆分为DC模式(未示出)、 16×16 模式641、 8×8 模式651,和 4×4 模式661。在该实施例中,假设具有以下四个 16×16 帧内预测模式:DC;水平的(HOR);垂直的(VER);和多定向的(Multi_DIR)。通过考虑每个模式的优先级来对intra_pred_mode进行信令。在INTRA16 $\times 16$ 中,由于DC模式比其它模式更经常地使用,所以在INTRA16 $\times 16$ 之前在sip_type表中添加INTRA16 $\times 16$ DC。然后,移除用于INTRA16 $\times 16$ 的intra_pred_mode中的most_probable_mode指示。作为替代,绝对地(absolutely)指示其它3个模式(16×16 、 8×8 和 4×4)。

[0066] 语法

[0067] 在表2和表3中例示用于该实施例的语法示例。具体地,表2示出了依据本原理实施例的用于 16×16 编码单元的sip_type的示例性规范,表3示出了依据本原理实施例的示例性的INTRA16 $\times 16$ 预测模式。对于INTRA32 $\times 32$ /INTRA64 $\times 64$,使用与INTRA16 $\times 16$ 相同的模式。对于信令,利用intra32_DC_flag和intra64_DC_flag来替换most_probable_mode指示,这是由于更频繁地使用DC。然后,绝对地编码其它intra_pred_mode。

[0068] 可以与在MPEG-4 AVC标准中完全相同地执行对于INTRA4 $\times 4$ 和INTRA8 $\times 8$ 的intra_pred_mode信令,因此在任何表中不列出这些模式。

[0069] 表2

[0070]

Sip_type	索引	二进制比特
SIP8 $\times 8$	0	0
SIP16 $\times 16$ DC	1	10
SIP16 $\times 16$	2	110
SIP4 $\times 4$	3	1110

[0071] 表3

[0072]

帧内预测模式	索引	二进制比特
--------	----	-------

VER	0	0
HOR	1	10
Multi-DIR	2	11

[0073] 表4示出了依据本原理实施例的示例性的宏块层语法。

[0074] 表4

macroblock_layer() {	C	描述符
...		
intra64_flag	2	u(1)
if (intra64_flag==1) {		
intra64_DC_flag	2	u(1)
if (intra64_DC_flag==0) {		
intra_pred_mode_64	2	ue(v)/se(v)
if (intra_pred_mode_64==Multi-DIR) {		
intra64_multidir_index	2	ue(v)/se(v)
}		
}		
}		
else {		
for (i32=0; i32<4; i32++) {		
intra32_flag [i32]	2	u(1)
if (intra32_flag[i32]==1) {		
intra32_DC_flag [i32]	2	u(1)
if (intra32_DC_flag[i32]==0) {		
intra_pred_mode_32 [i32]	2	ue(v)/se(v)
if (intra_pred_mode_32[i32]==Multi-DIR) {		
intra32_multidir_index [i32]	2	ue(v)/se(v)
}		
}		
}		
else {		
for (i16=0; i16<4; i16++) {		
sip_type [i16]	2	ue(v)/se(v)
if (sip_type[i16]==SIP16x16) {		
intra_pred_mode_16 [i16]	2	ue(v)/se(v)
if (intra_pred_mode_16[i16]==Multi-DIR) {		
intra16_multidir_index [i16]	2	ue(v)/se(v)
}		
}		
else if (sip_type[i16]!=SIP16x16_DC) {		
mb_intra_prediction_syntax(); /* 与 H.264 相同 */		
}		
}		
}		
...		
}		

[0076] 表4的一些语法元素的语义如下：

[0077] Intra64_flag等于1规定使用 $INTRA64 \times 64$ 。Intra64_flag等于0规定 64×64 的大块被进一步拆分为 32×32 的划分。

[0078] Intra64_DC_flag等于1规定intra_pred_mode是用于 $INTRA64 \times 64$ 的DC模式。Intra64_DC_flag等于0规定intra_pred_mode不是用于 $INTRA64 \times 64$ 的DC模式。

[0079] intra_pred_mode_64规定用于 $INTRA64 \times 64$ 的帧内预测模式(不包括DC模式)。

[0080] intra_multidir_index规定用于 $INTRA64 \times 64$ 中的Multi_Dir模式的角度的索引。

[0081] Intra32_flag[i]等于1规定对于第i个 32×32 的大块使用 $INTRA32 \times 32$ 。intra32_flag[i]等于0规定第i个 32×32 的大块被进一步拆分为 16×16 的划分。

[0082] Intra32_DC_flag[i]等于1规定对于第i个 32×32 块intra_pred_mode是用于 $INTRA32 \times 32$ 的DC模式。Intra32_DC_flag[i]等于0规定对于第i个 32×32 块intra_pred_mode不是用于 $INTRA32 \times 32$ 的DC模式。

[0083] intra_pred_mode_32[i]规定用于第i个 32×32 大块的 $INTRA32 \times 32$ 的帧内预测模式(不包括DC模式)。

[0084] intra_multidir_index规定用于 $INTRA32 \times 32$ 中的Multi_Dir模式的角度的索引。

[0085] sip_type[i]规定在第i个 16×16 块中用于基本块编码单元的空间帧内划分类型。

[0086] intra_pred_mode_16[i]规定用于第i个 16×16 块的 $INTRA16 \times 16$ 的帧内预测模式(不包括DC模式)。

[0087] intra_multidir_index规定用于第i个 16×16 块的 $INTRA16 \times 16$ 中的Multi_Dir模式的角度的索引。

[0088] 另一实施例

[0089] 在另一实施例中,自适应地选择大块单元为 32×32 或者 64×64 。可以使用一个或多个高级语法元素来对该选择进行信令。在实施例中,如果选择 32×32 ,则仅仅移除与 64×64 有关的所有语法。

[0090] 在另一实施例中,分层级的帧内预测可以包含从基本编码单元合并。例如,如果最大的块单元是 64×64 并且基本编码单元是 16×16 ,则使用一个标志(is_all_16×16_coding)来指示一个 64×64 块内部的所有 16×16 块是否都属于 16×16 编码类型。如果is_all_16×16_coding等于1,则这指示使用 16×16 编码类型并且停止信令。否则,使用一个标志(is_all_32×32_coding)来指示一个 64×64 块内部的所有 32×32 块是否都属于 32×32 编码类型。如果is_all_32×32_coding等于1,则这指示一个 64×64 块内部的所有 32×32 块都属于 32×32 编码类型。否则,如果is_all_32×32_coding和is_all_16×16_coding等于0,则这指示使用 $INTRA64 \times 64$ 。

[0091] 在另一实施例中,引入了用于具有不小于 16×16 尺寸的块单元的SIP类型(large_sip_type)。三种类型如下被称为:large_intra_16×16;large_intra_32×32;和large_intra_64×64。large_intra_16×16意味着一个大块内部的所有 16×16 块都属于 16×16 编码类型。large_intra_32×32意味着一个大块内部的所有 32×32 块都属于 32×32 编码类型。在一实施例中,large_intra_32×32可以与以上利用intra32_flag描述的实施例进行组合以允许分层级的帧内预测。large_intra_64×64意味着一个大块内部的所有 64×64 块都被编码为 $INTRA64 \times 64$ 。

[0092] 在另一实施例中,可以引入若干sip/mode表。可以将这些表预先存储在编码器和

解码器二者中,或者这些表可以是用户指定的并且使用一个或多个高级语法元素进行传送。表5示出了依据本原理实施例的示例性的宏块层语法。

[0093] 表5

	C	描述符
macroblock_layer() {		
...		
is_all_16x16_coding	2	u(1)
if (is_all_16x16_coding == 0) {		
is_all_32x32_coding	2	u(1)
if (is_all_32x32_coding == 0) {		
decode_with_64x64_coding_type()		
}		
else{		
decode_with_32x32_coding_type()		
}		
}		
else {		
decode_with_16x16_coding_type()		
}		
...		
}		

[0095] 表5的一些语法元素的语义如下:

[0096] **is_all_16×16_coding**等于1规定大块内部的所有 16×16 块都通过 16×16 编码类型编码。**is_all_16×16_coding**等于0规定所述大块不通过 16×16 编码类型进行编码。

[0097] **is_all_32×32_coding**等于1规定大块内部的所有 32×32 块都通过 32×32 编码类型编码。**is_all_32×32_coding**等于0规定所述大块不通过 32×32 编码类型进行编码。

[0098] 转到图7A和7B,其一起表示由参考标号700总地指示的、通过对用于大块的帧内预测进行信令而编码所述大块的画面数据的示例性方法。方法700包括开始块705,其将控制传递到功能块710。功能块710执行初始化,并且将控制传递到循环限制块715。循环限制块715对 64×64 块(即,具有 64×64 块尺寸的块)执行循环(下文也称为循环1),并且将控制传递到功能块785和循环限制块720。

[0099] 功能块785执行帧内 64×64 模式判定,基于RD64(即,从Intra 64×64 模式判定产生的率失真)设置Intra 64_DC_flag ,并且将控制传递到判定块770。

[0100] 循环限制块720对四个 32×32 块(即,具有 32×32 块尺寸并且从由循环1处理的当前 64×64 块中获得的四个块)执行循环(下文也称为循环2),并且将控制传递到功能块790和循环限制块725。

[0101] 功能块790执行帧内 32×32 模式判定,基于RD32(即,从Intra 32×32 模式判定产生的率失真)设置Intra 32_DC_flag ,并且将控制传递到判定块750。

[0102] 循环限制块725对四个 16×16 块(即,具有 16×16 块尺寸并且从由循环2处理的当前 32×32 块中获得的四个块)执行循环(下文也称为循环3),并且将控制传递到功能块730和功能块735。

[0103] 功能块730评估Intra $16\times 16_DC$ 模式,并且将控制传递到功能块740。功能块735评估其它的 16×16 模式(即,除了Intra $16\times 16_DC$ 以外的)和以下的模式(例如, $8\times 8, 4\times 4$ 等

等),并且将控制传递到功能块740。

[0104] 功能块740基于RD16(即,从Intra16×16模式判定产生的率失真)执行16×16模式判定,然后累计每个16×16块的RD16以获得TotRD16(其指示当由四个16×16块编码时整个32×32块的总的率失真),并且将控制传递到循环限制块745。循环限制块745结束对16×16块的循环(即,循环3),并且将控制传递给判定块750。

[0105] 判定块750确定RD32是否小于TotRD16(即,当前32×32块的率失真成本是否小于从当前32×32块中获得的四个16×16块的总的率失真成本)。如果是这样,则控制被传递给功能块755。否则,控制被传递给功能块742。

[0106] 功能块755将Intra32_flag设置为等于1,并且将控制传递给功能块760。功能块742将Intra32_flag设置为等于0,并且将控制传递给功能块760。

[0107] 功能块760将每个32×32块的RD32的累计设置至TotRD32以便指示在通过四个32×32块进行编码时整个64×64块的总的率失真,并且将控制传递给循环限制块765。循环限制块765结束对32×32块的循环(即,循环2),并且将控制传递给判定块770。

[0108] 判定块770确定RD64是否小于TotRD32(即,当前64×64块的率失真成本是否小于从当前64×64块中获得的四个32×32块的总的率失真成本)。如果是这样,则控制被传递给功能块775。否则,控制被传递给功能块780。

[0109] 功能块775将Intra64_flag设置为等于1,并且将控制传递给循环限制块795。功能块780将Intra64_flag设置为等于0,并且将控制传递给循环限制块795。

[0110] 循环限制块795结束对64×64块的循环(即,循环1),并且将控制传递给功能块797。功能块797熵编码标志、intra_pred_mode和残差,并且将控制传递给结束块799。

[0111] 转到图8A和8B,其一起表示由参考标号800总地指示的、通过确定要被应用于大块的帧内预测而解码所述大块的画面数据的示例性方法。方法800包括开始块805,其将控制传递到功能块808。功能块808初始化解码器并且然后将控制传递到功能块810。功能块810解析比特流,并且将控制传递给循环限制块815。循环限制块815对64×64块执行循环(下文也称为循环1),并且将控制传递给判定块820。判定块820确定Intra64_flag是否被设置为等于1。如果是这样,则控制被传递给功能块885。否则,控制被传递给循环限制块825。

[0112] 功能块885确定Intra64_DC_flag是否被设置为等于1。如果是这样,则控制被传递给功能块887。否则,控制被传递给功能块888。功能块887执行Intra64×64DC预测,并且然后将控制传递到功能块890。功能块888执行除了Intra64×64DC模式以外的Intra 64×64预测,并且然后将控制传递给功能块890。功能块890解码当前64×64块,并且将控制传递给循环限制块880。循环限制块880结束对64×64块的循环(即,循环1),并且将控制传递给结束块899。

[0113] 循环限制块825对四个32×32块执行循环(下文也称为循环2),并且将控制传递给判定块830。判定块830确定Intra32_flag是否等于1。如果是这样,则控制被传递给功能块835。否则,控制被传递给循环限制块845。

[0114] 功能块835确定Intra32_DC_flag是否等于1。如果是这样,则控制被传递给功能块837。否则,控制被传递给功能块838。功能块837执行Intra32×32DC预测,并且将控制传递到功能块840。功能块838执行除了Intra32×32DC模式以外的帧内预测模式,并且然后将控制传递给功能块840。功能块840解码32×32块,并且将控制传递给循环限制块875。

[0115] 循环限制块875结束对32×32块的循环(即,循环2),并且将控制传递给循环限制块880。

[0116] 循环限制块845对四个16×16块执行循环(下文也称为循环3),并且将控制传递判定块850。判定块850确定是否sip_type=Intra16_DC。如果是这样,则控制被传递给功能块855。否则,控制被传递给功能块860。

[0117] 功能块855执行Intra16×16_DC模式预测,并且将控制传递到功能块865。功能块860使用其它帧内预测模式(即,除了Intra16×16_DC模式以外的)执行模式预测,并且将控制传递给功能块865。

[0118] 功能块865解码16×16块,并且将控制传递给循环限制块870。循环限制块870结束对16×16块的循环(即,循环3),并且将控制传递给循环限制块875。

[0119] 现在将描述本发明的许多伴随优点/特征中的一些,其中的一些已经在上面提及。例如,一个优点/特征是一种具有视频编码器的装置,所述视频编码器通过对用于画面中的至少一个大块的帧内预测进行信令来编码用于所述至少一个大块的画面数据。通过选择基本编码单元尺寸并且分配用于基本编码单元尺寸的单个空间帧内划分类型来对帧内预测进行信令。该单个空间帧内划分类型是可从多个空间帧内划分类型中选择的。所述至少一个大块具有比基本编码单元的块尺寸大的大块尺寸。帧内预测是分层级的帧内预测并且通过以下操作中的至少一个而对至少一个大块执行:将大块尺寸拆分为基本编码单元尺寸以及从基本编码单元尺寸合并到大块尺寸。

[0120] 另一优点/特征是具有如上所述的视频编码器的装置,其中对于多个空间帧内划分类型的每一个,向多个可用的帧内预测模式中最频繁使用的特定的帧内预测模式分配较高的优先级。

[0121] 又一优点/特征是具有如上所述的视频编码器的装置,其中自适应地选择大块尺寸。

[0122] 再一优点/特征是具有如上所述的视频编码器的装置,其中,使用一个或多个高级语法元素来执行信令。

[0123] 此外,另一优点/特征是具有如上所述的视频编码器的装置,其中空间帧内划分类型表和帧内预测模式表中的至少一个被视频编码器预先存储并且使用以便编码至少一个大块。该空间帧内划分类型表和帧内预测模式表中的至少一个被安排为被对应的视频解码器预先存储并且使用以便解码该至少一个大块。

[0124] 此外,另一优点/特征是具有如上所述的视频编码器的装置,其中空间帧内划分类型表和帧内预测模式表中的至少一个被视频编码器用于编码至少一个大块,并且被视频编码器使用一个或多个高级语法元素来传送。

[0125] 基于这里的教导,本领域普通技术人员可以容易确定本原理的这些和其它特征和优点。应理解,本原理的教导可以以硬件、软件、固件、专用处理器、或其组合的各种形式来实现。

[0126] 更优选地,本原理的教导被实现为硬件与软件的组合。此外,软件可以实现为有形地体现在程序存储单元上的应用程序。应用程序可以被上载到包括任何适当架构的机器并由该机器执行。优选地,在具有诸如一个或多个中央处理单元(“CPU”)、随机存取存储器(“RAM”)、以及输入/输出(“I/O”)接口等的硬件的计算机平台上实现该机器。计算机平台还

可以包括操作系统和微指令代码。这里描述的各种处理与功能可以是可能由CPU执行的微指令代码的一部分或是应用程序的一部分、或者是其任何组合。另外，各种其它的诸如附加数据存储单元以及打印单元之类的外设单元可以连接到计算机平台。

[0127] 还应理解，由于在附图中描绘的一些组成系统组件和方法优选地以软件实现，因此这些系统组件或处理功能块之间的实际连接可能取决于本原理被编程的方式而有所不同。给出这里的教导，本领域普通技术人员将能够预期本原理的这些和类似的实现方式或配置。

[0128] 尽管这里已经参考附图描述了示例实施例，应理解本原理不限于那些确切的实施例，并且本领域普通技术人员可以在其中进行各种改变和修改，而不偏离本原理的范围和精神。所有这些改变和修改意在被包括在所附权利要求阐述的本原理的范围之内。

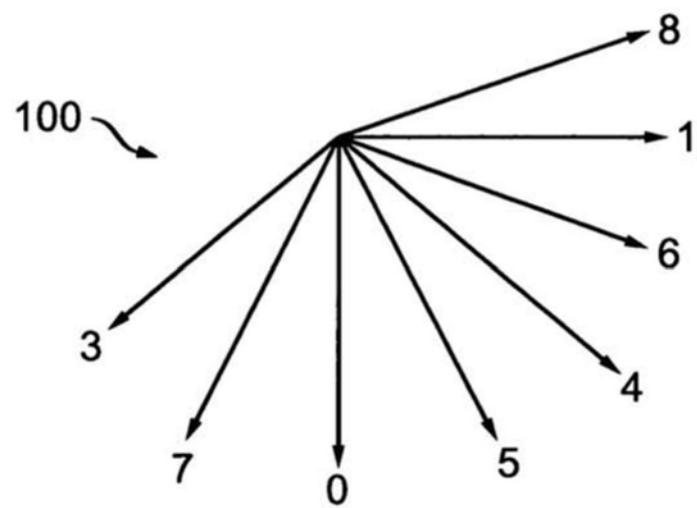


图1

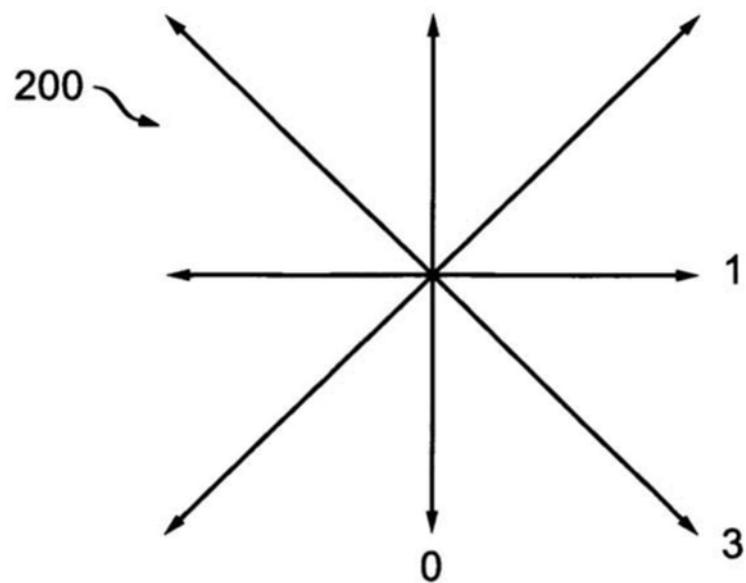


图2

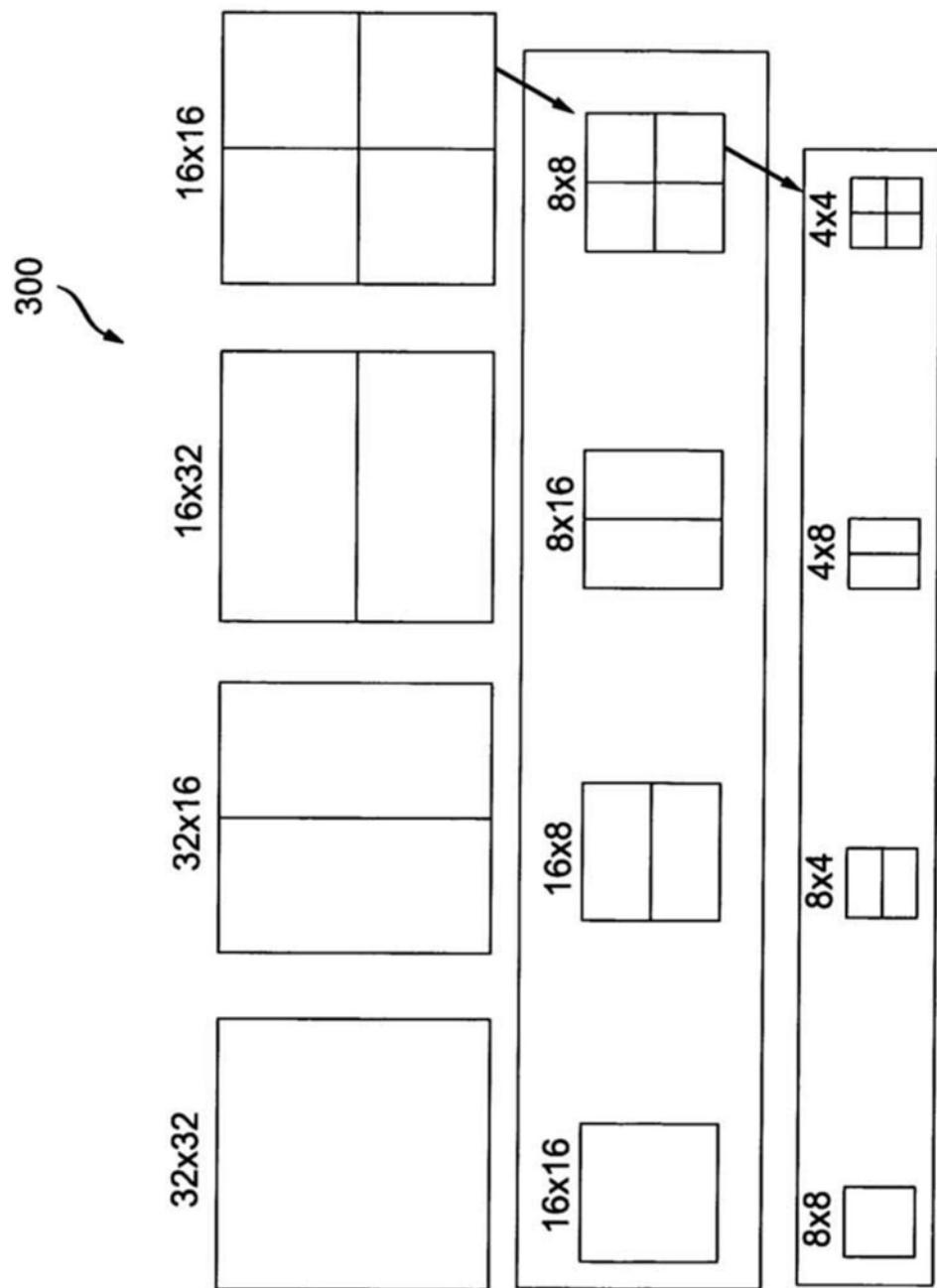


图3

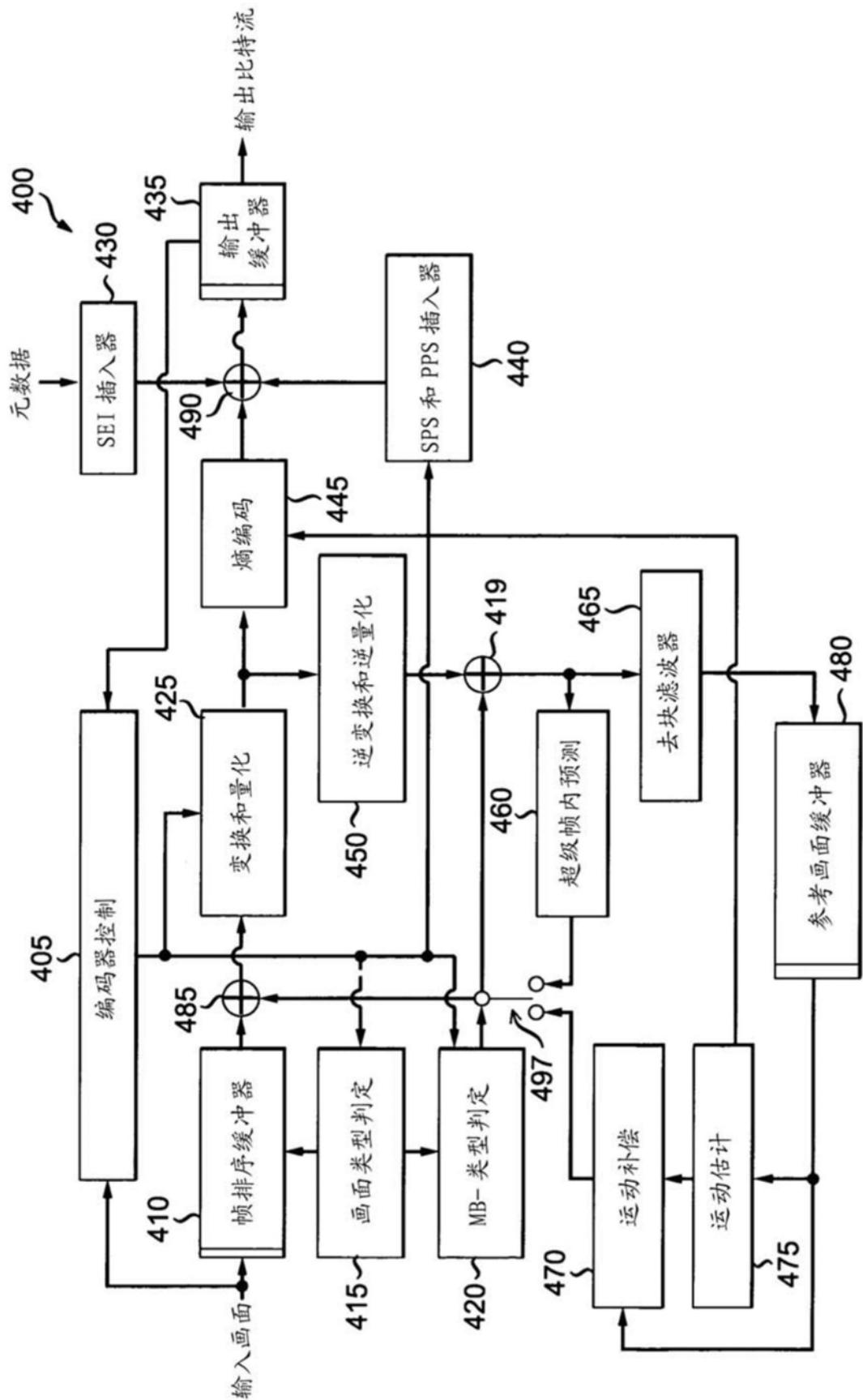


图4

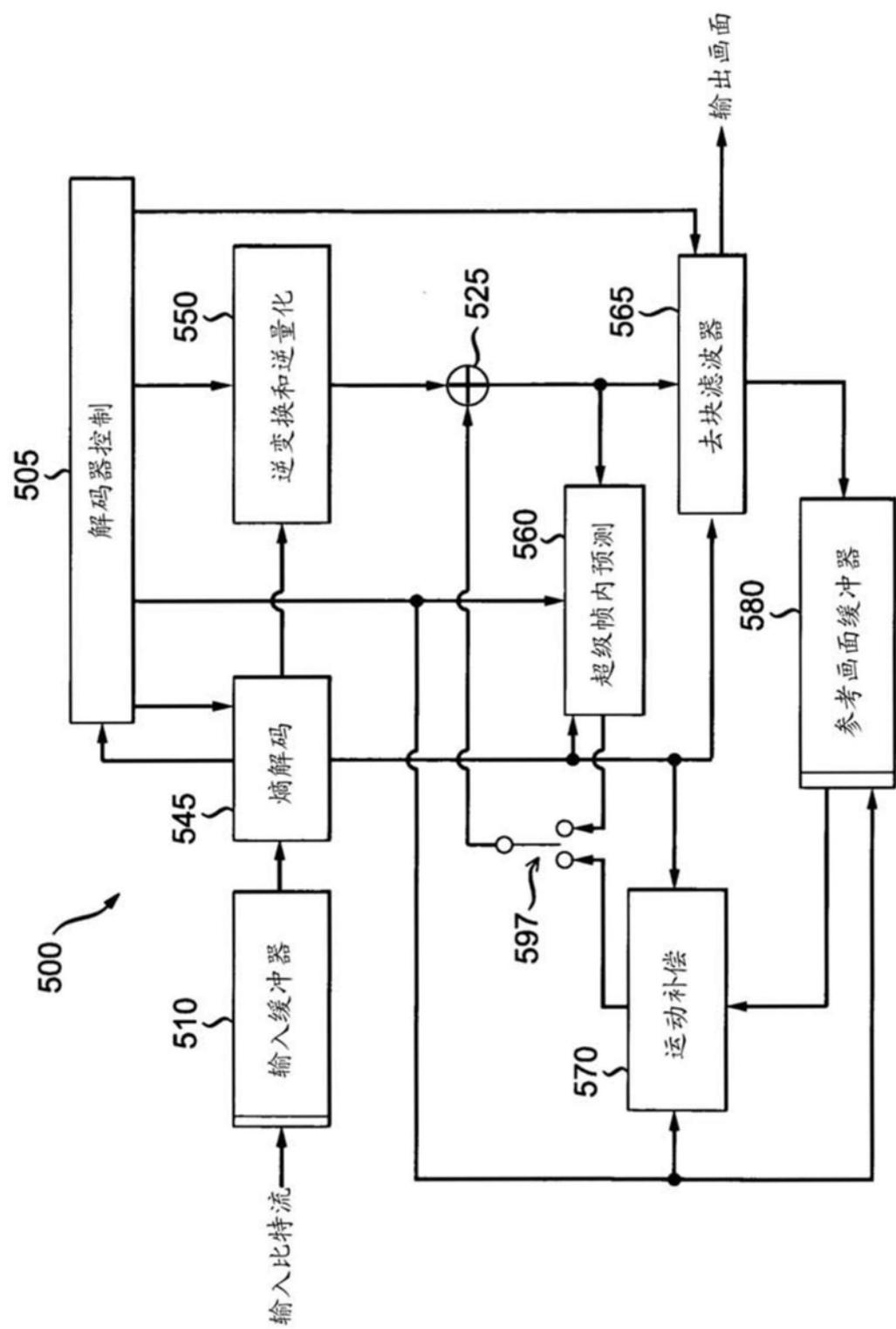


图5

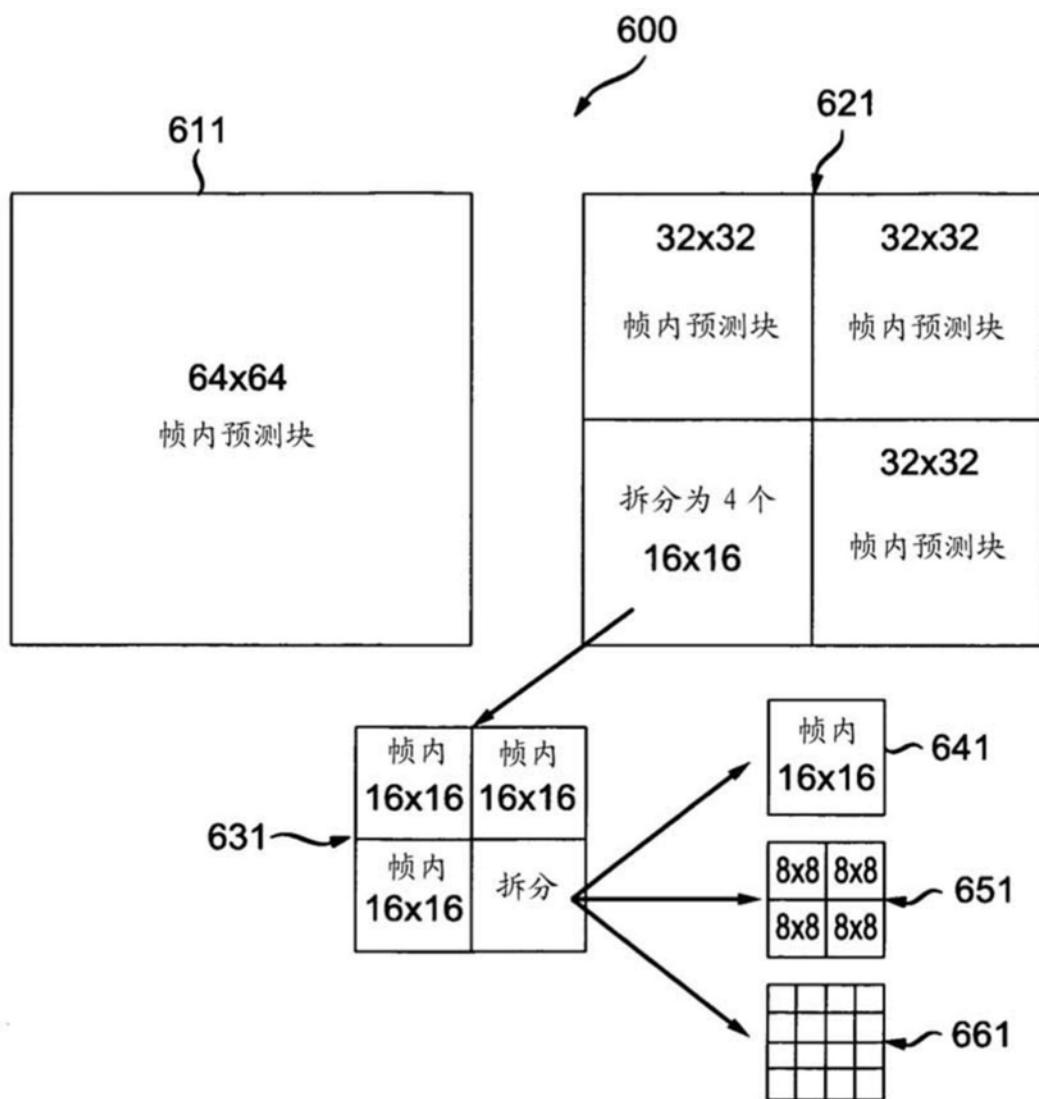


图6

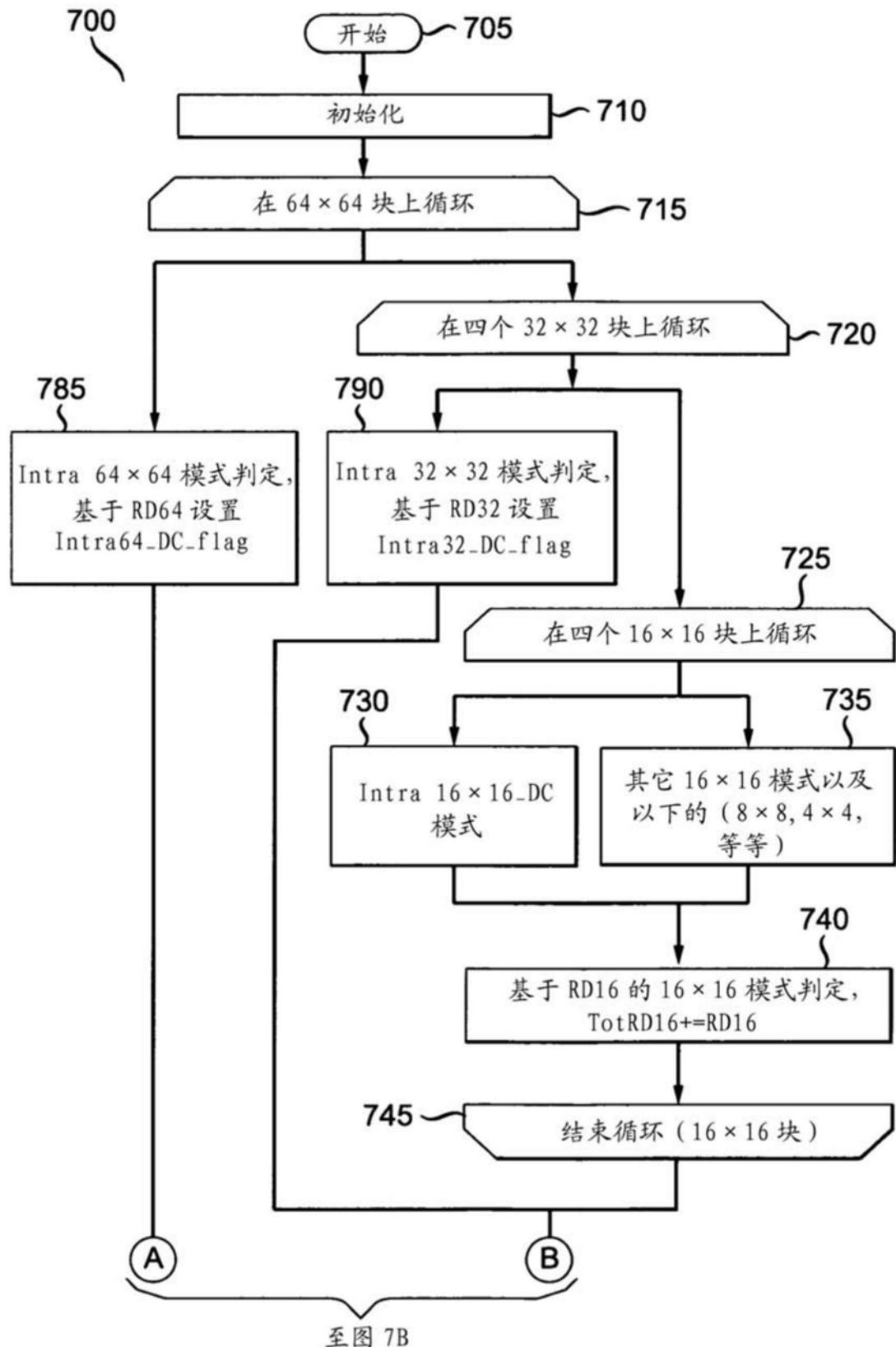


图7A

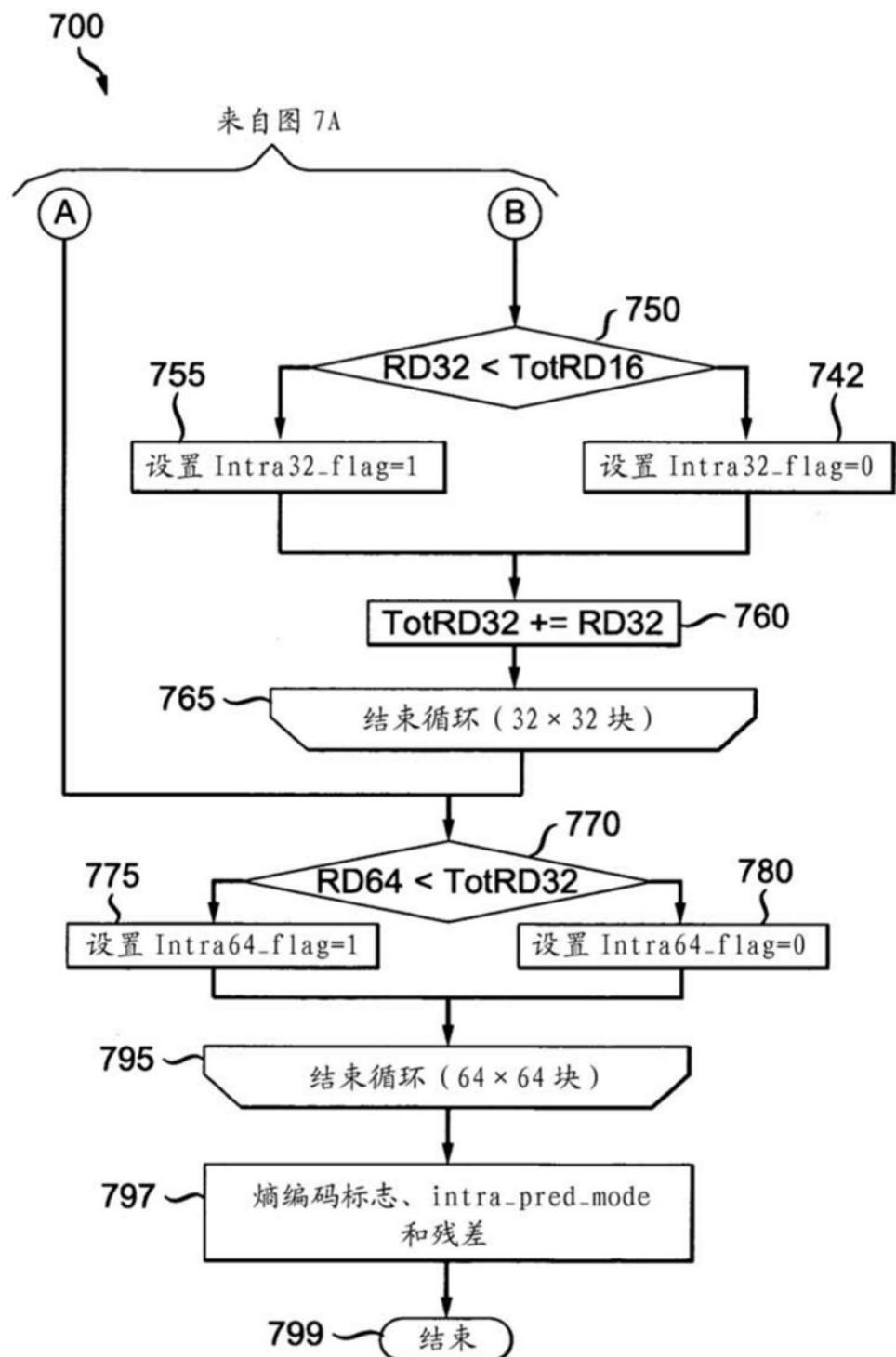


图7B

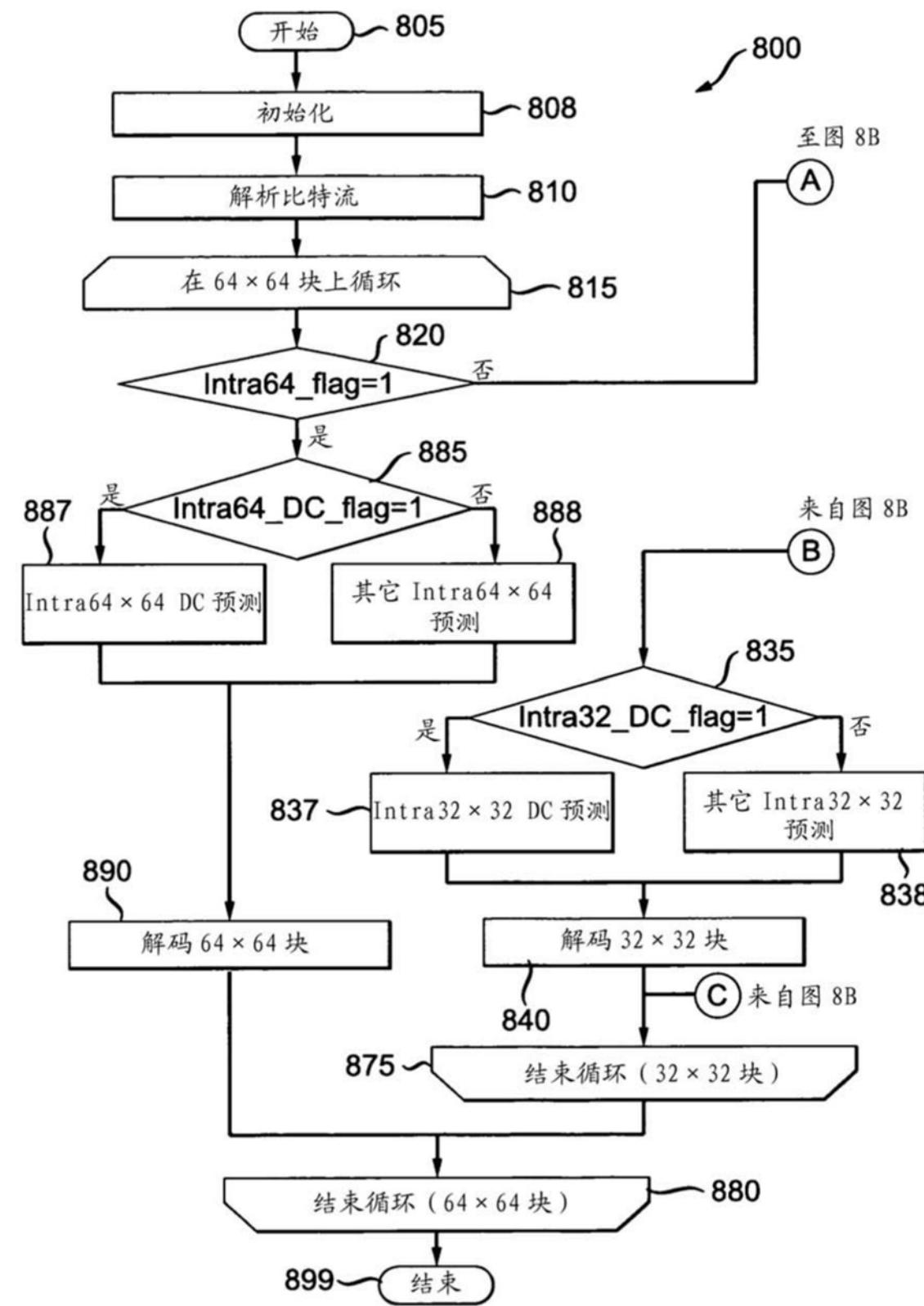


图8A

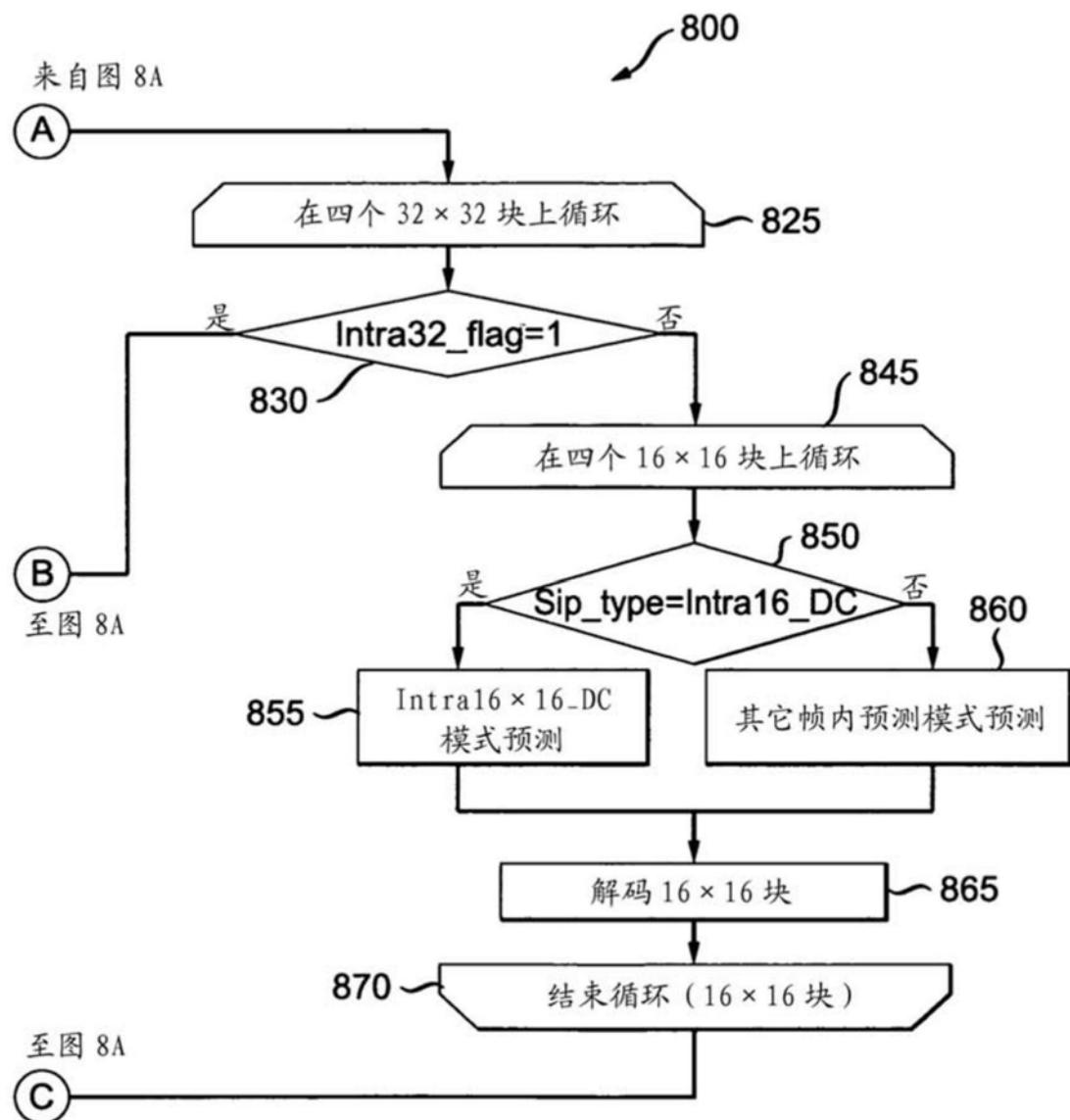


图8B