



(12)发明专利申请

(10)申请公布号 CN 106851281 A

(43)申请公布日 2017.06.13

(21)申请号 201610853191.5

H04N 19/172(2014.01)

(22)申请日 2016.09.26

H04N 19/196(2014.01)

(30)优先权数据

14/867,143 2015.09.28 US

14/982,698 2015.12.29 US

15/155,868 2016.05.16 US

(71)申请人 苏州踪视通信息技术有限公司

地址 江苏省苏州工业园区星湖街328号创意产业园1栋-B702单元

(72)发明人 谷群山

(74)专利代理机构 广州三环专利代理有限公司

44202

代理人 郝传鑫 熊永强

(51)Int.Cl.

H04N 19/136(2014.01)

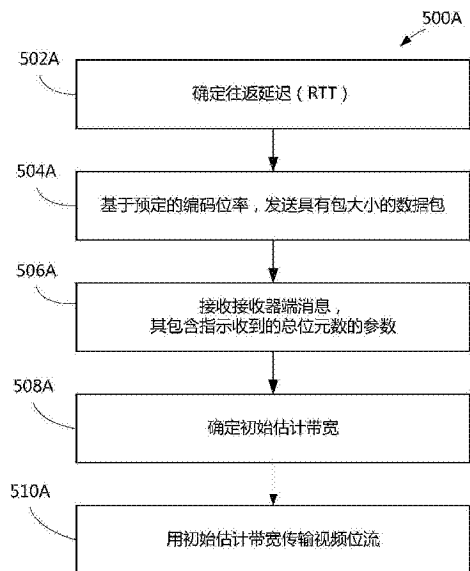
权利要求书3页 说明书26页 附图16页

(54)发明名称

实时视频传输的初始带宽估计

(57)摘要

本发明公开了一种为实时视频传输而估计初始带宽的方法。所述方法包括：由所述发送器，用处理器编码所述视频位流的第一部分，所述第一部分包括一个I-帧；从第一时间点开始，由所述发送器以系列数据包的形式传输所述视频位流的经编码的所述第一部分，所述数据包具有基于一个预定编码位率的数据包大小；在第二时间点，由所述发送器从所述接收器接收一条消息，所收到的消息包括指示所述接收器收到的位元总数的一个参数；基于所接到的参数、所述第一和第二时间点和一个在所述发送器和所述接收器之间的往返延迟，由所述发送器，用所述处理器确定一个初始估计带宽；用所述初始估计带宽，传输所述视频位流的第二部分至所述接收器，所述第二部分用所述初始估计带宽编码。



1. 一种估计初始带宽的方法,用于在一个发送器和一个接收器间传输一个视频位流,包括:

由所述发送器,用处理器编码所述视频位流的第一部分,所述第一部分包括一个I-帧;

从第一时间点开始,由所述发送器以系列数据包的形式传输所述视频位流的经编码的所述第一部分,所述数据包具有基于一个预定编码位率的数据包大小;

在第二时间点,由所述发送器从所述接收器接收一条消息,所收到的消息包括指示所述接收器收到的位元总数的一个参数;

基于所接到的参数、所述第一和第二时间点和一个在所述发送器和所述接收器之间的往返延迟,由所述发送器,用所述处理器确定一个初始估计带宽;

用所述初始估计带宽,传输所述视频位流的第二部分至所述接收器,所述第二部分用所述初始估计带宽编码。

2. 根据权利要求1所述的方法,其特征在于,所述视频位流的所述第一部分进一步包括一个或多个帧间预测的帧。

3. 根据权利要求1所述的方法,其特征在于,所述收到的消息进一步包括良劣基准数据,所述基准数据指示所述视频位流的经编码的所述第一部分所解码的至少一个帧是否被正确地从良基准帧解码。

4. 根据权利要求3所述的方法,其特征在于,进一步包括:

基于所述良劣基准数据,判断所述视频位流的经编码的所述第一部分是否包括至少一个良基准帧;

基于所述视频位流的经编码的所述第一部分包括至少一个良基准帧的判断,用所述至少一个良基准帧和所述初始估计带宽,编码所述视频位流的所述第二部分;

基于所述视频位流的经编码的所述第一部分不包括良基准帧的判断,用所述初始估计带宽,编码所述视频位流的所述第一和第二部分。

5. 根据权利要求1所述的方法,其特征在于,所述系列数据包的传输是在正在建立所述发送器和所述接收器间呼叫的一个流程期间。

6. 根据权利要求5所述的方法,其特征在于,从第一时间点开始,由所述发送器以系列数据包的形式传输所述视频位流的经编码的所述第一部分,所述数据包具有基于一个预定编码位率的数据包大小包括:

由所述发送器传输一条或多条为在所述发送器和所述接收器间建立所述呼叫的呼叫消息,所述一条或多条呼叫消息包括所述系列数据包;

所述在第二时间点,由所述发送器从所述接收器接收一条消息,所接到的消息包括指示所述接收器收到的位元总数的一个参数包括:

在第二时间点,由所述发送器接收一条应答消息,所述应答消息由所述接收器在接收到所述一条或多条为在所述发送器和所述接收器间建立所述呼叫的呼叫消息后所发送。

7. 根据权利要求1所述的方法,其特征在于,所述系列数据包的传输是在所述发送器和所述接收器间的呼叫已经建立之后。

8. 根据权利要求1所述的方法,其特征在于,所述系列数据包的各数据包包括一个序号和一个关联于所述序号的时标,在所述接收器收到具有大于或等于预定序号的序号的至少一个数据包后,由所述接收器确定指示收到的位元总数的所述参数,而无需考虑在收到具

有所述序号的所述至少一个数据包之后收到的任意数据包。

9. 根据权利要求1所述的方法,其特征在于,所述数据包大小被确定为所述预定编码位率的函数,如果所述预定编码位率增高超过一个预定阈值,所述数据包大小增大。

10. 一种估计初始带宽的方法,用于在一个发送器和一个接收器间传输一个视频位流,包括:

由所述接收器,接收所述视频位流的经编码的第一部分所关联的一个或多个数据包,而所述一个或多个数据包由所述发送器以系列数据包的形式发送且具有基于一个预定编码位率的数据包大小;

基于收到的所述一个或多个数据包,依照一个预定规则,由所述接收器用处理器确定指示所述接收器收到的位元总数的一个参数;

从所述一个或多个数据包,解码所述视频位流的经编码的所述第一部分;

在收到所述一个或多个数据包后,向所述发送器传输一条或多条消息,而每条消息包括指示收到的位元总数的所述参数;

从所述发送器接收所述视频位流的第二部分,所述视频位流用基于指示收到的位元总数的所述参数所确定的一个初始估计带宽编码。

11. 根据权利要求10所述的方法,其特征在于,所述视频位流的所述第一部分进一步包括一个I-帧以及零个或多个帧间预测的帧。

12. 根据权利要求10所述的方法,其特征在于,所述一条或多条消息的每一条进一步包括良劣基准数据,所述基准数据指示所述视频位流的经解码的所述第一部分的至少一个帧是否被正确地从一个良基准帧解码。

13. 根据权利要求12所述的方法,其特征在于,从所述发送器接收所述视频位流的第二部分,所述视频位流用基于指示收到的位元总数的所述参数所确定的一个初始估计带宽编码包括:

从所述发送器接收所述视频位流的第二部分,所述视频位流用基于所述良劣基准数据的所述良基准帧编码,所述良基准帧先前已从所述视频位流的经编码的所述第一部分解码。

14. 根据权利要求13所述的方法,其特征在于,进一步包括:

用从所述视频位流的经解码的所述第一部分的所述良基准帧,解码所述视频位流的经编码的所述第二部分。

15. 根据权利要求10所述的方法,其特征在于,所述基于收到的所述一个或多个数据包,依照一个预定规则,由所述接收器确定指示所述接收器收到的位元总数的一个参数包括:

在收到所述系列数据包所关联的一个数据包后,判定收到的所述数据包所关联的序号是否大于或等于一个预定序号;

在判定所述序号大于或等于所述预定序号后,确定指示收到的位元总数的所述参数,而无需考虑在收到具有大于或等于所述预定序号的所述序号的所述至少一个数据包后收到的任意数据包。

16. 根据权利要求15所述的方法,其特征在于,所述基于收到的所述一个或多个数据包,依照一个预定规则,由所述接收器确定指示所述接收器收到的位元总数的一个参数包

括：

自收到一个第一数据包起，判定一个预定时段是否已经流逝；

在判定所述预定时段已经流逝后，确定指示收到的位元总数的所述参数，而无需考虑在所述预定时段已经流逝后收到的任意数据包。

17. 根据权利要求10所述的方法，其特征在于，所述系列数据包的传输是在正在建立所述发送器和所述接收器间呼叫的一个流程期间。

18. 根据权利要求17所述的方法，其特征在于，由所述接收器，接收所述视频位流的经编码的第一部分所关联的一个或多个数据包，而所述一个或多个数据包由所述发送器以系列数据包的形式发送且具有基于一个预定编码位率的数据包大小包括：

接收一条或多条为在所述发送器和所述接收器间建立所述呼叫的呼叫消息，所述一条或多条呼叫消息包括所述系列数据包。

19. 根据权利要求18所述的方法，其特征在于，所述在收到所述一个或多个数据包后，向所述发送器传输一条或多条包括指示收到的位元总数的所述参数的消息包括：

在收到所述一条或多条为在所述发送器和所述接收器间建立所述呼叫的呼叫消息后，传输一条或多条应答消息，所述一条或多条应答消息的每一条都包括指示收到的位元总数的所述参数。

20. 根据权利要求10所述的方法，其特征在于，所述系列数据包的传输是在所述发送器和所述接收器间的呼叫已经建立之后。

实时视频传输的初始带宽估计

[0001] 相关申请的交叉引用

[0002] 本申请是美国专利申请《带反向信道消息管理的视频编码及解码》(申请号14/982698, 2015年12月29日提交)的部分延续案申请(continuation-in-part), 而前述专利申请本身又是美国专利申请《带反向信道消息管理的视频处理的方法和系统》(申请号14/867143, 2015年9月28日提交)的部分延续案申请。前述两份申请书的全部内容, 特此通过引用并入本文。

技术领域

[0003] 本发明涉及视频编码及解码, 更具体地, 涉及在实时视频传输中使用反向信道消息进行初始带宽估计和带宽调整的视频编码和解码。

背景技术

[0004] 数字视频位流可被编码后有效地把视频压缩成数字位流, 所述数字位流可存储于非瞬时数字媒介或通过有限带宽的通信信道进行流传输。然而, 在视频位流的传输或存储过程中, 可能会发生丢包或其他错误, 从而导致所述位流解码的错误。可用的信道带宽不时改变的情况也属常见, 导致实时视频传输的问题。

发明内容

[0005] 有鉴于此, 本发明公开了带反向信道消息管理的视频编码及解码的系统、方法和设备的各个方面。

[0006] 一方面, 本发明公开了一种估计初始带宽的方法, 用于在一个发送器和一个接收器间传输一个视频位流, 包括:

[0007] 由所述发送器, 用处理器编码所述视频位流的第一部分, 所述第一部分包括一个 I-帧;

[0008] 从第一时间点开始, 由所述发送器以系列数据包的形式传输所述视频位流的经编码的所述第一部分, 所述数据包具有基于一个预定编码位率的数据包大小;

[0009] 在第二时间点, 由所述发送器从所述接收器接收一条消息, 所收到的消息包括指示所述接收器收到的位元总数的一个参数;

[0010] 基于所接到的参数、所述第一和第二时间点和一个在所述发送器和所述接收器之间的往返延迟, 由所述发送器, 用所述处理器确定一个初始估计带宽;

[0011] 用所述初始估计带宽, 传输所述视频位流的第二部分至所述接收器, 所述第二部分用所述初始估计带宽编码。

[0012] 另一方面, 本发明公开了一种估计初始带宽的方法, 用于在一个发送器和一个接收器间传输一个视频位流, 包括:

[0013] 由所述接收器, 接收所述视频位流的经编码的第一部分所关联的一个或多个数据包, 而所述一个或多个数据包由所述发送器以系列数据包的形式发送且具有基于一个预定

编码位率的数据包大小；

[0014] 基于收到的所述一个或多个数据包，依照一个预定规则，由所述接收器用处理器确定指示所述接收器收到的位元总数的一个参数；

[0015] 从所述一个或多个数据包，解码所述视频位流的经编码的所述第一部分；

[0016] 在收到所述一个或多个数据包后，向所述发送器传输一条或多条消息，而每条消息包括指示收到的位元总数的所述参数；

[0017] 从所述发送器接收所述视频位流的第二部分，所述视频位流用基于指示收到的位元总数的所述参数所确定的一个初始估计带宽编码。

[0018] 本发明的上述及其他方面，将通过对以下实施例、所附权利要求书和附图的详细描述。

附图说明

[0019] 本发明在结合附图阅读时，可以从以下的详细描述中得到最好的理解。需要强调的是，根据通常的做法，附图的各种特征不是按比例。相反，为解释清楚，所述各种特征的尺寸被任意地放大或缩小。本发明的公开材料提到所附附图时，类似的参照数字在所有视图中指代类似的部分。在附图中：

[0020] 图1A是依本公开的一些实施例提供的一个视频编码及解码系统的示意图；

[0021] 图1B是可以依本公开的一些实施例使用的一个计算设备的示意图；

[0022] 图2是依本公开的一些实施例提供的一个视频位流的结构示意图；

[0023] 图3是依本公开的一些实施例提供的一个视频压缩系统的结构示意图；

[0024] 图4是依本公开的一些实施例提供的一个视频解压系统的结构示意图；

[0025] 图5A是依本公开的一些实施例提供的用于传输一个视频位流的初始带宽估计的示例性流程的流程图；

[0026] 图5B是依本公开的一些实施例提供的用于传输一个视频位流的初始带宽估计的另一示例性流程的流程图；

[0027] 图5C是依本公开的一些实施例提供的用于接收一个视频位流的初始带宽估计的示例性流程的流程图；

[0028] 图6A是依本公开的一些实施例提供的用于传输一个视频位流的调整带宽的示例性流程的流程图；

[0029] 图6B是依本公开的一些实施例提供的生成一个包括供发送器使用的接收器端参数的反向信道消息的示例性流程的流程图；

[0030] 图6C是依本公开的一些实施例提供的用于传输一个视频位流的调整带宽的示例性流程的流程图；

[0031] 图6D是依本公开的一些实施例提供的用于接收一个视频位流的调整带宽的示例性流程的流程图；

[0032] 图6E是依本公开的一些实施例提供的用于传输及接收一个视频位流的调整带宽的示例性流程的流程图；

[0033] 图7是依本公开的一些实施例提供的一个包括反向信道消息管理器的视频编码及解码系统的结构示意图；

[0034] 图8是依本公开的一些实施例提供的一个编码及解码基准帧选择的示意图；

[0035] 图9是依本公开的一些实施例提供的一个视频基准帧结构的示意图。

具体实施方式

[0036] 数字视频可以被用于娱乐、视频会议、广告和通常的信息共享。用户对数字视频质量的期望会很高,因为用户希望在带宽有限的共享互联网传播的视频与在专用有线频道传播的视频具有同样高的空间和时间质量。例如,数字视频编码可以压缩数字视频的位流,以允许高质量的数字视频在具有有限带宽的网络进行传输。例如,数字视频质量可以定义为输出的解压及解码的数字视频在何种程度上与输入的数字视频匹配。

[0037] 视频编码和解码统合各种技术用以压缩和解压数字视频流,得以在有限带宽能力的网络中传输高质量的数字视频流。这些技术可以将数字视频流处理为一系列的数字数据块,并处理这些数据块以压缩用于传输或存储,并且在收到数据块后,解压缩所述数据块以重建原始数字视频流。这种压缩及解压序列可能是“有损”的;“有损”的意义是指所述解压的数字视频可能不完全匹配所述输入的数字视频。例如,这可以通过测量所述输入视频流的像素数据及对应的所述经编码、传输和解码的视频流的像素数据之区别来测定。一个数字视频流通过编码及解码所述数字视频流所引入的失真程度,可被认为是压缩程度的一个函数,从而所述解码视频的质量可被认为是所述传输带宽的一个函数。

[0038] 本发明公开的实施例,可以通过调整所发送的视频位流的位率(bitrate)以匹配其传输所经的网络或信道的能力,允许经压缩的视频位流在“嘈杂”或易发潜在错误的网络中传输。一些实施例可以在传输压缩数字视频位流之前测试网络,通过发送一个或多个数据包到解码器并分析返回的数据包,以确定所述数字视频的最佳压缩比。例如,一个数据包可以包括一条或多条消息。数据包也可以包括视频或音频数据,无论是否携带消息。一些实施例可以通过分析由所述解码器发送到所述编码器的包含有关网络的信息的数据包,周期性地重新测试网络。调节所述位率可以提高或降低相较于所述输入的数字视频流的所述解码视频位流的所述空间和时间质量,其中,较高的位率可以支持更高质量的数字视频。

[0039] 本发明公开的实施例还可以通过向压缩视频位流中添加前向纠错(Forward Error Correction,FEC)数据包,以在嘈杂网络中传输所述压缩视频位流。FEC数据包对数字视频位流中的部分或全部信息,以包含在位流中的额外数据包的形式进行冗余编码。通过处理所述的额外数据包,解码器可检测数字视频流中丢失或损坏的信息,并可在某些情况下,用所述额外数据包中的所述冗余数据重建所述丢失或损坏的数据。根据如上所述的由所述编码器接收到的网络信息包,一些实施例可以调整与FEC有关的参数。动态调整所述FEC参数,可以在传输的数字视频数据和FEC数据之间分配可用的网络带宽,以允许在给定网络条件下每单位时间内传输最多有质量的图像。

[0040] 本发明公开的实施例可以改变编码器和FEC参数,以允许在数字视频位流传输所经的网络的给定条件下,传输可能的最高品质的数字视频。改变这些参数也可以影响解码的视频流的质量,因为它们可以导致播放中的解码视频的样子产生迅速的变化。通过分析参数变化的趋势和预测参数值的变化,一些实施例可以控制编码器和FEC参数的所述变化以避免视频质量的快速改变。

[0041] 图1A是视频编码和解码系统10的示意图,一些实施例可以在图中提供。在一个示

例中,计算设备12可以包括硬件的内部配置,包括一个处理器如中央处理单元(CPU) 18和一个数字数据存储设备如存储器20。例如,CPU 18可以是用于控制计算设备12的控制器,也可以是微处理器、数字信号处理器、场效可编程逻辑门阵列(FPGA)、布置于定制的专用集成电路(ASIC)上的离散电路元件或者任意其他数字数据处理器。例如,CPU 18可以通过存储器总线、电线、电缆、无线连接或任意其他连接而连接到存储器20。存储器20可以作为或包含只读存储器(ROM)、随机存取存储器(RAM)、光存储器、磁存储器诸如磁盘或磁带、非易失性存储卡、云存储,或者任意其他合适的数字数据存储设备或设备的方式或组合。存储器20可存储CPU 18使用的数据和程序指令。计算设备12也有其他合适的可能的实现方式。例如,计算设备12的处理可以分布在以多个网络16通信的多个设备上。图1中,计算设备12可以是一个编码计算设备,即包括一个编码器的计算设备。如下文详述,编码计算设备12可以整合编码器元件300和流程600A,而流程600A可整合硬件与软件元件以及相关的方法来实现编码设备12。

[0042] 在一个实例中,网络16可以连接计算设备12和计算设备14,用于编码和解码视频流。例如,所述视频流可以在计算设备12上进行编码,经编码的视频流可以在计算设备14上解码。网络16可以包括任意一个或多个适于即时应用的网络,诸如有线或无线的局域或广域网、虚拟专用网、蜂窝电话数据网络,或任意其他有线或无线的,对硬件、软件、通信协议的配置,以适用于图例中的从计算设备12向计算设备14传送视频位流,并从计算设备14传送关于所述网络的参数至计算设备12。

[0043] 计算设备14可以包括CPU 18和存储器20,类似于以上讨论的系统10的组件。如下文详述,计算设备14可以是一个解码计算设备14,其可以整合解码器元件400和流程500C,而流程500C可整合硬件与软件元件以及相关的方法来实现解码设备14。例如,计算设备14可配置用于显示视频流。连接到计算/解码设备14的显示器25可以有各种实现方式,包括液晶显示器(LCD)、阴极射线管(CRT)、有机或非有机发光二极管显示器(LED)、等离子体显示器、或任意其他用以向用户显示机器可读的视频信号的机制。例如,计算设备14可配置用于显示由计算设备14的解码器解码的所述视频位流的渲染。

[0044] 编码器和解码器系统10有其他可能的实施方式。除了计算设备12和计算设备14,图1还展示了额外的计算设备26,其分别具有一个或多个CPU 30及存储器32。这些计算设备可包括服务器,以及移动电话,移动电话也可以诸如创建、编码、解码、存储、转发或显示数字视频流。这些计算设备在处理能力和存储器可用性方面可具有不同的能力,包括用于创建视频的设备,如视频摄像机,和用于显示视频的设备。

[0045] 图1B是一个计算设备100(如图1A所示的设备12、14和26)的示例性内部配置的框图。如前所述,设备100可采用包括多个计算单元的计算系统的形式,或单一计算单元的形式,如移动电话、平板计算机、膝上型计算机、笔记本电脑,台式计算机,服务器计算机等。

[0046] 计算设备100可包括多个组件,如图1B所示。CPU(或处理器) 18可以是中央处理单元(如微处理器),且可包括一个或多个处理器,其中每个处理器具有一个或多个处理内核。又或者,CPU 18可包括另一类型的单个或多个设备,其能够操纵或处理现存的或以后开发的信息。当存在多个处理设备时,它们可以以任何方式进行互连,包括硬连线或联网(包括无线联网)的方式。因此,CPU 18的操作可以分布在多个机器上,所述多个机器可以直接对接或通过一个局域网或其他网络连接。CPU 18可以是通用处理器或专用处理器。

[0047] 随机存取存储器(Random Access Memory, RAM) 42是可用作内存储器的任意合适的非永久存储设备。RAM 42可包括可执行指令和数据,以供CPU 18即时访问。RAM 42通常包括一个或多个动态随机存取存储器(Dynamic RAM, DRAM) 模块,如双倍数据速率同步动态随机存取存储器(Double Date Rate Synchronous DRAM, DDR SDRAM)。可替代地, RAM 42可包括另一类型的单个或多个设备,其能够存储数据以供现存的或以后开发的CPU 18处理。CPU 18可以通过总线112访问和操作数据。CPU 18可以利用高速缓存(cache) 120作为局部快速内存,用以操作数据和指令。

[0048] 存储器44可以是只读存储器(Read Only Memory, ROM)、磁盘驱动器、固态驱动器、闪存、相变存储器(Phase-Change Memory, PCM) 或设计为保持数据一段时间的任何形式的非易失性存储器,尤其是断电的情况下。存储器44可包括可执行指令48和应用文件/数据52,以及其他数据。例如,可执行指令48可以包括一个操作系统以及一个或多个应用程序,其可全部或部分加载到RAM 42(包括基于RAM的可执行指令46和应用文件/数据50)并由CPU 18执行。可执行指令48可以组织成设计为执行本文所述的各种功能的可编程模块或算法、功能程序、代码及代码段。

[0049] 本文所用的术语“模块”,可以使用硬件、软件或两者组合来实现。模块既可以是一个较大实体的一部分,本身也可以细分成子实体。如果一个模块用软件实现,该软件可以实现为算法组件,包括存储在存储器中的程序指令,所述指令设计为在处理器上执行。术语“模块”不要求编码结构是任何具体的形式,并且不同模块的功能方面既可以是独立的,也可以重叠并由共同的程序指令来执行。例如,第一模块和第二模块可以使用一组公共的程序指令,而在实现所述第一和第二模块的相应和/或共同指令之间没有明显的边界。

[0050] 所述操作系统可以是一个小型设备(如智能手机或平板电脑)或大型设备(如大型计算机)的操作系统。例如,所述应用程序可以包括网络浏览、网络服务器和/或数据库服务器。例如,应用文件52可以包括用户档案、数据库目录和配置信息。在一些实施例中,存储器44包括指令以执行本文所述的发现技术。存储器44可包括一个或多个设备,并可以利用一种或多种类型的存储器(如固态存储器或磁存储器)。

[0051] 计算设备100还可以包括一个或多个输入/输出设备,如网络通信单元108和接口130,接口130可以具备有线通信组件或无线通信组件190,其可以经总线112连接到CPU 18。网络通信单元108可以利用任意种类的标准网络协议,如以太网、TCP/IP及不再举例的许多协议,在设备之间实现通信。接口130可以包括一个或多个利用以太网、电力线通讯(Power Line Communication, PLC)、WiFi无线网、红外线、GPRS/GSM、CDMA等的收发器。

[0052] 用户界面25可以包括显示器、位置输入装置(如鼠标、触摸板、触摸屏或类似物)、键盘或其他形式的用户输入和输出设备。用户界面25可经总线112对接到处理器18。特别地,图形用户界面(Graphical User Interface, GUI) 25 是允许人们与一个设备进行图形化交互的用户界面。它可以被分解为输入部、输出部以及可管理、处理及与输入和输出部交互的处理器。输入部可以接受部件创建的输入,如鼠标、触摸板、触摸屏或类似物。GUI的输出部可以生成在某种形式的显示器上可显示的输入,如阴极射线管(Cathode-Ray Tube, CRT)、液晶显示器(Liquid Crystal Display, LCD),以及发光二极管(Light Emitting Diode, LED) 显示器,如有机发光二极管(Organic Light Emitting Diode, OLED) 显示器。显示器一般形成网格化像素,每个像素都可以采用不同的明度及选用不同的颜色值,组合在

一起后在显示器上排列以形成各种更高级别的实体(以像素区域为单位)。这些像素区域可以形成图标、时窗、按钮、光标、控制元件、文本和其他显示的实体。显示器利用图形设备界面,其通常包括专门设计用于与显示器的硬件交互的图形处理器,并且可以接受来自其他处理器的高级指令,以减少对它们的依赖。图形设备界面通常拥有自己的存储器作为缓冲区,也允许图形处理器操纵所存储的数据。从而,显示器的操作通常包括由图形处理器访问存储在存储器的指令和数据,以修改用户显示器上的像素区域。

[0053] 客户端和服务器的内部配置或架构也可能有其他实现形式。例如,服务器可以省略显示器25。RAM 42或存储器44可以分布在多台机器上,如基于网络的存储器或在多台执行客户端或服务器的操作的机器内的存储器。虽然被本文描述为单个总线,总线112可以由多条总线构成,其可以通过各种桥、控制器和/或适配器彼此连接。计算设备100可包含任何数量的传感器和探测器用来监控设备100本身或其周围的环境;或可以包含一个位置识别单元160,如GPS或其他类型的定位装置。计算设备100也可以包含一个电源170(如电池),以使该单元可以独立运作。这些都可以通过总线112与CPU/处理器18进行通信。

[0054] 图2是待编码及随后解码的视频流200的框图。视频流200可以包括视频序列202。视频序列202是视频流在时间上连续的子集,也称为图像组(Group of Pictures, GOP)。视频序列202可以包括多个相邻的帧204。尽管图中帧204只描绘了四个帧,视频序列202可包括任意数目的相邻帧。帧204的单个实例表示为单帧206。进一步划分单帧206可以产生一系列的块208。在本例中,块208可以包含对应于单帧206内的一个 $N \times M$ 像素域的数据,比如对应像素的亮度和色度数据。块208可以是任意合适的尺寸,例如 128×128 像素组或所述像素组中的任意矩形子集。

[0055] 图3是依照公开的实施方式展示的编码器300的框图。编码器300可在计算设备中实现,诸如计算设备12。编码器300可以编码输入视频流200。编码器300包括数个级来执行前向路径的各个功能,以产生经编码和/或压缩的流322:帧内预测级302、模式决定级304、帧间预测级306、变换及量化级308、滤波级314和熵编码级310。编码器300还可以包括一个重建路径,以重建用于预测和编码未来块的帧。在图3中,编码器300包括逆量化及逆变换级312和可用于存储视频数据中多个帧的帧存储器316,以重建用于预测的块。编码器300的其他变体结构也可用于编码视频流200。

[0056] 视频流200用于编码时,其中每个帧(例如如图2中的单帧206)以块为单位进行处理。每个块可以从左上方的块开始,以光栅扫描的顺序单独进行处理。在帧内预测级302,针对视频流200的块,帧内预测的残差块可被确定。通过检查先前处理的相邻块以确定所述相邻块的像素值是否类似于当前块,帧内预测可以预测块的内容。因为视频流200以光栅扫描顺序进行处理,光栅扫描顺序中领先于当前块出现的块可用于处理所述当前块。光栅扫描顺序中出现于给定块之前的块之所以可用于帧内预测,是因为它们也可在解码器中使用,鉴于它们也将首先被重建。如果相邻块与当前块足够相似,则所述相邻块可被用作预测块,并在操作318中从当前块中减去该预测块,以形成残差块,并且表示当前块是帧内预测的信息,可以包含在所述视频流里。

[0057] 视频流200也可以在帧间预测级306进行帧间预测。帧间预测包含形成残差块,其通过从时间相邻帧转换像素以形成可从当前块中减去(操作318)的预测块。时间相邻帧可存储于帧存储器316并在帧间预测级306被访问,以形成可以传递给模式决定级304

的残差区块——在级304中,帧内预测产生的残差区块可以和帧间预测产生的残差区块进行比较。模式决定级304可确定用何种预测模式——帧间或帧内——来预测当前区块。例如,在某些实施方式中,可以使用率失真值(rate distortion value)来确定使用何种预测模式。

[0058] 率失真值的确定,可以通过计算用特定编码参数(例如预测模式)编码的视频位流中的每单位时间的位元的数目,即位率,并结合计算输入视频流的区块和解码视频流的时间上和空间上相同的区块之间的差异。因为编码器300是“有损”的,所述解码视频流区块的像素值可以不同于所述输入视频流区块的像素值。例如,为了确定最佳的参数值,可以改变编码参数以比较相应的率失真值。

[0059] 在相减操作318中,由模式决定级304确定的残差区块可从当前区块中减去,并传递到变换及量化级308。由于残差区块的值可以小于当前区块的值,变换及量化后的残差区块具有的值可以比变换及量化后的当前区块少,因而可以在视频位流中通过较少的变换系数来表示。基于区块的变换的实例包括Karhunen-Loève变换(Karhunen-Loève Transformation, KLT),离散余弦变换(Discrete Cosine Transform, DCT)和奇异值分解(Singular Value Decomposition, SVD),仅举几例。在一个实施例中,DCT将区块变换到频域。在DCT变换的例子中,变换系数的值是基于空间频率,直流系数(DC系数)或其他最低频率的系数在矩阵的左上角,最高频率的系数在矩阵的右下角。

[0060] 变换及量化级308转换所述变换系数为离散量子值,它可被称为量化变换系数。量化可以减少以变换系数代表的离散状态的数目,同时少量降低图像质量,如果量化是在空间域而不是变换域中进行的话。量化后的变换系数可以通过熵编码器级310进行熵编码。熵编码是一种可逆的、无损的算术编码方案,它可以减少视频位流中的位元的数目,并在解码时不会在所述位流中引起变化。熵编码后的系数与其他用于解码所述区块的信息一起,诸如使用的预测类型、运动矢量、量化器的值和滤波强度等,被输出为压缩的位流322。

[0061] 图3里的虚线所示的重建路径,可用于帮助确保编码器300和解码器400(参照下文图4)都使用相同的基准帧来形成帧内预测区块。重建路径执行的功能类似于下文详述的解码过程所执行的功能,包括:在逆量化及逆变换级312进行去量化和逆变换所述量化的变换系数,并在加法操作320中与模式决定级304产生的残差区块一起,创建重建区块。环路滤波级314可应用于重建的区块以减少失真,例如区块效应,因为解码器400可以在过滤所述重建的视频流之前对其采样以形成基准帧。例如,图3展示,环路滤波级314发送环路滤波参数至熵编码器310,与输出的视频位流322进行组合,以允许解码器400使用和编码器300相同的环路滤波参数。

[0062] 编码器300的其他变体可用于编码压缩的位流322。编码器300的各级可以以不同的顺序进行处理,或者可以无需改变其目的而组合成更少的级或分拆为更多的级。例如,不基于变换的编码器300可以无需变换级而直接量化残差区块信号。在另一实施例中,编码器300可以将变换及量化级308拆分为单一的操作。

[0063] 图4是依本公开的实施方式的各方面描述的解码器400的框图。在一个实施例中,解码器400可以在计算设备14上实现。解码器400包括以下级来执行各种功能,以从压缩位流322产生输出视频流418:熵解码级402,逆量化及逆变换级404,帧内预测级408,帧间预测级412,加法器410,模式决定级406和一个帧存储器414。解码器400的其他结构变体也可以

用于解码压缩位流322。例如,逆量化及逆变换级404可以被表示为两个独立的级。

[0064] 接收的视频位流322可以通过熵解码器402进行熵解码。熵解码器402执行编码器300在级310执行的熵编码的逆过程,以还原所述视频位流到熵编码前的原始状态。所述还原的视频位流可以进行逆量化和逆变换,类似于逆量化及逆变换级312的方式。逆量化及逆变换级404可以还原视频位流的残差区块322。注意,由于编码器300和解码器400能表示有损编码,所述还原的残差区块的像素值可不同于输入视频流200中相同的时间和空间位置的残差区块。

[0065] 在逆量化及逆变换级404还原残差区块之后,通过在加法器410中加入所述残差区块的预测区块,所述视频位流的残差区块可被大致还原到其预测前的状态。加法器410从模式决定级406处接收用于加到残差区块的所述预测区块。模式决定级406可经由编码器300解释所述输入视频位流322包含的参数,例如,确定是使用帧内还是帧间预测以还原视频位流322的区块。模式决定级406对输入视频位流322也可以执行计算来确定对特定的区块使用何种预测。作为解码器,通过对相同的数据执行相同的计算,模式决定级406可以针对预测模式做出与编码器300相同的决定,从而减少了为指示要用的预测模式而在视频位流中传输位元的需要。

[0066] 模式决定级406可以同时从帧内预测级408和帧间预测级412接收预测区块。因为帧内预测区块以光栅扫描顺序处理,帧内预测级408可以从加法器410输出的所述还原视频流接收区块以用作预测区块;并且由于帧内预测使用的区块是编码器300在还原残差区块之前以光栅扫描顺序选择的,帧内预测级408可以按需提供预测区块。如上面关于编码器300的讨论,帧间预测级412从储存在帧存储器414的帧中创建预测区块。帧存储器414从环路滤波器416接收经其滤波的还原区块。环路滤波可以去除基于区块的预测技术所引入的区块效应,如本文所述的编码器300和解码器400所用。

[0067] 帧间预测级412可以使用帧存储器414里经环路滤波器416滤波的帧,以使用编码器300用过的相同数据形成预测区块。尽管使用有损压缩,对预测使用相同的数据可使解码器400重建的区块具有接近对应输入区块的像素值。模式决定级406从帧间预测级412接收的预测区块可以传递到加法器410,以还原视频位流322的区块。经环路滤波器416滤波后,还原的视频流418可以从解码器400输出。解码器400的其他变体可用于解码压缩位流322。例如,解码器400可以不经环路滤波级416产生输出视频流418。

[0068] 在建立真正的视频或音频连接之前估计可用带宽是很有价值的,这样,编码器可以用适当的位率来编码。例如,初始带宽估计可以在接收器端使用数据包列(data package train)完成。然而,这种估算往往是不准确的。根据本公开的实施例,可以使用发送器和接收器两侧的数据共同估算带宽,以提高准确性。

[0069] 在从编码计算设备12向解码计算设备14传输视频位流322之前或期间,从解码计算设备14向编码计算设备12发送的消息,可被称为反向信道消息。本公开的实施方式可以通过消息(如反向信道消息)的传输和处理来确定网络带宽所关联的网络参数,用于优化编码参数。如下所述,图5A-6D将用更多细节说明带宽估计。

[0070] 根据本公开的实施方式,图5A是为传输一个视频位流而估计一个初始带宽的示例性流程500A的流程图。例如,流程500A可以由一个发送器,如编码计算设备12(如编码器300),来执行。图5A中的流程图展示了包括在流程500A里的若干操作。实现流程500A的操

作,可以包括在本文,或者多于或少于本文所述。例如,操作可以组合或划分而改变执行的操作的数目。流程500A的所述操作可以按本文包括的顺序或不同的顺序来执行,且仍然实现流程500A的意图。

[0071] 流程500A可以在发送器(如编码计算装置12/编码器300)和接收器(如解码运算装置14/解码器400)间的呼叫建立期间或建立之后发生,或者在任何其他合适的阶段(如在视频已中断一段时间后重新启动)。例如,呼叫可以包括一条或多条消息,用以在发送器和接收器之间建立视频传输连接。例如,所述一条或多条消息可以包括在编码流程和解码流程间交换的呼叫和应答消息,这将在下文结合操作一起进行详细描述。

[0072] 在步骤502A中,发送器可以确定发送器和接收器间的往返延迟/时间(Round Trip Time,RTT)。例如,发送器可以向接收器发送一系列数据包作为呼叫消息。收到呼叫消息时,接收器可以形成应答或确认(Acknowledge,ACK)消息并进一步形成数据包从接收器发送到发送器,而在其基础上可以如下文所述来确定往返延迟。

[0073] 例如,发送器可以向接收器发送一些数据包 $P(0)$ 、 $P(1)$ 、 $P(2)$ ……并将每个数据包的发送时间记录为 $T_s(0)$ 、 $T_s(1)$ 、 $T_s(2)$ ……例如,发送的数据包可以是小的数据包(如呼叫消息)。接收器可以接收所述数据包中的任何一个(如 $P(0)$)并向发送器确认收悉(如通过发送一条或多条确认消息)。发送器在收到确认(如确认消息)的任一项时,可以检查接收时间 $T_r(0)$ 、 $T_r(1)$ ……。发送器和接收器间的系统往返时间/延时可以计算为在发送和确认接收相同编号的数据包之间的时间差,如 $RTT = T_r(i) - T_s(i)$,其中 $i = 0, 1, \dots$ 。

[0074] 在步骤504A中,从第一时间点(T_0)开始,基于预定的编码位率,发送器可以发送具有包大小(以字节为单位的“Psize”)的系列数据包。

[0075] 例如,所述系列数据包可以包括数据(如编码的视频数据),或使用随机数据装包的伪数据包。例如,所述数据可以包括用于初始带宽估计的数据,并且可以被发送为在编码流程和解码流程间交换的呼叫和应答消息。任何设备都可以发送呼叫和应答消息两者。例如,在涉及将编码视频数据打包成系列数据包(如通过图5B中的流程500B)的实施例中,通过计算设备12,编码的视频位流可以使用编码器300进行编码,并经由网络16发送。例如,在接收器端,图5C的流程500C可以用解码器400来解码用于带宽估计的数据包。

[0076] 例如,发送器可以发送一系列的、总数为 $N+K$ (标号为 $0, 1, 2, \dots, N+K-1$)的、大小为Psize的数据包。每个数据包在一段等待时间(T_d)后发送。可估算的最大带宽取决于数据包大小Psize和发送速度($1/T_d$)。假设发送数据包0的时刻为 T_0 ,一旦数据包标号大于或等于 N ,接收器就基于收到的数据包总数及数据包大小计算收到的总位元(B_{total})。

[0077] 如上所述,例如,用于初始带宽估计的数据可以包括在编码流程和解码流程500间交换的呼叫和应答消息。“数据包”和“消息”在此可以互换使用。例如,呼叫和应答消息可以实现为一系列的数据包,其被“填充”用于带宽估计的数据(带宽估计可以在呼叫建立之前或之后进行)。例如,用于带宽估计的数据可以包括“哑”(dummy)数据,其可以是作为经填充的呼叫消息的形式的随机数据,或“真实的”编码视频数据——如第一个视频帧(通常被编码为“I帧”)、第一批视频帧、用户定义的帧集、以及能以一定时间间隔发送的、可用于估计带宽的“真实的”编码视频数据。

[0078] 例如,呼叫和应答消息可以是伴随编码视频位流的带外(out of band)数据包,独立数据的数据包,或作为编码视频位流的一部分而发送的数据包。相同或不同的消息类型,

可分别用于初始带宽估计和以后的带宽调整。

[0079] 在一些公开的实施例中,跟踪呼叫和应答消息可以通过给每个数据包(包括所述呼叫和应答消息)分配一个唯一的数据包号码(也称为“序号”),所述序号可以从某个数字(如“0”)开始、并对每个视频流增加1。每个数据包还可以包括一个时标(timestamp),其也可以从某个数字(如“0”)开始、并对用由计算设备12或14发送的一定的各数据包以一定时间间隔(如一或几毫秒)增加。所述消息可以作为系列数据包发送,其中每一个都具有所述序号和所述时标,且数据包尺寸都为Psize。所述时标可以是到达时标或发送时标,这取决于是发送器还是接收器在计算时标。Psize可以使用预定的(编码器)位率来确定,例如“Maxbitrate”。例如,“Maxbitrate”可以是一个预定的最大视频位率,其可被预存在流程500A(或其他实施例中的500B-D、600A-D)所关联的配置文件中、并在需要时取还。取决于网络条件,“Maxbitrate”可被调整,以指示用于传输一个视频的最大允许位率。

[0080] 在一些实施例中,Psize可以作为预定编码器位率的函数来确定,其中,当预定编码位率增高超过一个阈值时,Psize增加。例如,根据以下规则,Psize可以基于“Maxbitrate”来确定:

```
if (Maxbitrate <= 300Kbps)
```

```
    Psize = 400;
```

```
[0081] else if (Maxbitrate <=1Mbps)
```

```
    Psize = 600;
```

```
else
```

```
[0082] Psize = 1200;
```

[0083] 通过以这种方式设置Psize,网络带宽可以在发送任何呼叫和应答消息之前来估计,从而防止网络慢时因过快发送过多数据包而造成呼叫和应答消息拥塞网络。这样做的目的是无需长时间拥塞网络而估计带宽。例如,当网络很慢时,过快发送过多数据包是希望避免的。另一方面,足够快地发送数据包以确定真正的带宽又是很重要的。

[0084] 网络带宽可以在建立呼叫的过程中进行估计。当呼叫被连接时,视频编码器可以用估计的带宽初始编码所述视频位流,从而避免不合理占用可用的网络带宽。如果有足够数量的、包括呼叫及应答消息的数据包由编码计算设备12经网络16发送、并由解码计算设备14接收,呼叫及应答消息可被用来确定真实的网络带宽。对于任意带宽高于100Kbps的网络,流程500A(或其他实施例中的500B-D、600A-D)的设计可以应对单向三倍的预期位率,又不会长时间拥塞所述网络。

[0085] 在步骤506A中,在第二时间点(Tc),所述发送器从接收器接收一条消息,其包括一个指示接收器收到的总位元数(Btotal)的参数。例如,Btotal可以根据数据包的大小(Psize)和发送速度(1/Td)来确定。在一些实施例中,接收器可以发送多条(用于容错,消息可能会丢失)包含Btotal的消息至发送器。例如,所述消息可以发送为反向信道消息,这将在下面进一步讨论(如图6B和图7)。所述发送器接收任一含有参数Btotal的消息并检查当前时间Tc。

[0086] 在一些实施例中,在收到至少一个其序号大于或等于一个预定序号的数据包后,

Btotal可被接收器确定,而无需考虑任何在收到所述至少一个具有所述序号的数据包之后收到的数据包。这将在图5C中进一步说明。例如,一旦接收器收到任何其序号大于或等于N的数据包,它将确定收到的总位元数,而不考虑之后收到的任何其他数据包。数字N可以被设置为处于最小序号(如0)和所述系列数据包的最大序号(如,N+K-1)之间的任何数字。

[0087] 在步骤508A中,基于所接收的参数、所述第一和第二时间点以及所述往返延迟,发送器可以确定一个初始估计带宽。在一些实施例中,估计的带宽(“Best”)可以按照下列公式计算:

$$[0088] \quad \text{Best} = \text{Btotal} / ((\text{Tc} - \text{T0}) - \text{RTT})$$

[0089] 在步骤510A中,发送器用所述初始估计带宽发送视频位流到接收器。例如,所述视频位流可用所述初始估计带宽进行编码。

[0090] 在一些实施例中,可以基于所述视频数据来估计带宽,而且一个预定的音频带宽可被添加到视频信道。

[0091] 在一些实施例中,一旦带宽得以估计,在配置文件中的发送器的初始参数可以基于可用带宽及其他参数(如初始数据包丢失率以及往返时间)重新计算。例如,诸如自适应编码长度(AdaptiveCodingLength)、FEC率(FEC_ratio)、视频编码器位率、分辨率和帧率等参数都可以基于所述初始估计重新初始化。例如,初始估计可以包括以下三个参数中的一个或多个:(估计的)带宽、丢包率(PacketlossRatio)和往返时间(RTT)。

[0092] 在一些实施例中,初始带宽估计可以在呼叫应答期间使用呼叫-应答消息来完成(例如,在呼叫“振铃”时且通话建立前)。呼叫-应答消息可用填充数据,通过预定大小和定时信息来数据包化,以使接收器收到这些消息时可以估计带宽。例如,填充数据可以通过随机数生成器产生,以避免网络协议压缩。

[0093] 对于数据包结构的设计,所述包含呼叫和应答消息的数据包可以用一个序号和时标作为开端,然后以填充数据填充到预定尺寸。例如,所述填充数据可以是精确的Psize字节(呼叫/应答消息数据随后的所有数据)。例如,填充数据的前两个字可用于所述序号和时标(如,以无符号整型的格式)。

[0094] 在一个说明性示例中,序号开始于“0”,并针对每一个发送的数据包增加1。时标也可从零开始,且数据包可在它们各自的发送时间被标记时标。类似于以上描述,可以有两组呼叫消息和两组应答消息。第一组可以由25个相同的数据包构成,而第二组可以由10个数据包构成。

[0095] 在所述说明性示例中,可以由发送器产生两组呼叫和应答消息。例如,发送器可以发送N+k个尺寸为Psize的数据包序列(序号为0、1、2、...、N+K-1),N=25和K=10。

[0096] 第一组25个呼叫信息(如25个相同的数据包)可由编码计算设备12创建并以(大致)相同的时间间隔每100毫秒进行发送。在带宽比Maxbitrate更高的网络中,网络带宽可被估计为Maxbitrate。第一组的25个数据包之后,例如,编码计算设备12能够延迟一段时间——如约400毫秒(大于发送所述数据包的时间差)——再在大约100毫秒内(10毫秒一个)发送第二组的10个数据包。如果网络带宽不足,传输所有数据包(此例中共35个)会花费更长的时间。例如,一个100Kbps的信道需要约一秒钟来传输每个400字节的所述35个数据包,而相同的信道需要约三秒钟来传输每个1200字节的所述35个数据包。选择正确的数据包大小可以避免更长的延迟。

[0097] 依据Psize(依前例所讨论,400、800或1200字节),一组25个以大约100毫秒的间隔送出的数据包可以代表一个最大位率:

[0098] $\text{Maxbitrate} = 25 \times 8 \times \text{Psize} / 0.1 = \{0.8\text{Mbps}, 1.6\text{Mbps}, 2.4\text{Mbps}\}$

[0099] 在此例中,能被确定的最大位率,可用Psize的值估计为0.8Mbps、1.6Mbps或2.4Mbps。带宽比Maxbitrate更高的任何网络将只被估计为Maxbitrate。

[0100] 发送和接收第一和第二组数据包所需的时间可以用来指示网络带宽。例如,假设每个数据包为400个字节(Psize=400),一个100Kbps的网络可能花费大约一秒钟来传输35个包括在第一和第二组的数据包。在1200字节(Psize=1200)时,相同网络可能需要花费约三秒钟。发送和接收呼叫及应答消息数据包可在视频流的开端进行,这意味着用户需要等待,直到呼叫和应答消息在视频开始之前被处理。

[0101] 在本例中,当建立呼叫或第一视频位流开始时,接收器可以开始接收和存储数据包,并持续到接收到序号为N——本例中N为25(或任意大于25的序号)——的数据包,或一个预定的时间窗口/预定时段(如三秒钟)结束/流逝。在本例中,在第25号数据包之前或在所述时间窗口内未收到的任意数据包,可被认为已丢失且不被计入Btotal。在本例中,可以用以下等式来计算估计的带宽:

[0102] $\text{Bandwidth} = (24 - N_{\text{loss}}) \times \text{Psize} / (\text{T}_{\text{last}} - \text{T}_{\text{first}})$

[0103] 在这里,带宽以Kbps计算,Nloss是第一组N(如25)个数据包中丢包的总数。这并不包括在第二组10个数据包中丢失的任何数据包。Tlast是紧邻序号25的数据包之前(不含丢失的包)最后一个数据包的到达时标,其可以毫秒测量;Tfirst是第一个收到的数据包的到达时间(其可以毫秒测量)。注意,被用来确定带宽的是所述第一个数据包和所述最后一个数据包的到达时间的相对差值,因为传输所述第一个数据包所需的时间无法得知。

[0104] 图5B是依本公开的一些实施例提供的使用真实视频数据进行初始带宽估计的另一示例性流程500B的流程图,流程500B可用于一个发送器传输一个视频位流。这些步骤类似于图5A中的步骤,应结合图5A中的描述来理解。

[0105] 在步骤502B中,发送器编码视频位流的第一部分;例如,其可包括一个第一帧(如一个I-帧)和零个或多个其他帧(如帧间预测帧,比如P-帧、B-帧或PB-帧)。

[0106] 例如,视频位流可由编码计算设备12进行编码并传输到解码计算设备14,且编码的位流可包含一个用一个基准帧编码的帧,所述基准帧选自按显示顺序先于所述帧的多个基准帧。如下所述,所述多个基准帧可以包括一个良基准帧。所述良基准帧是所述编码器已知无错误的一个基准帧。在一些实施例中,为使一个基准帧为良基准帧,其本身解码所需的多个基准帧也不含任何错误。

[0107] 在步骤504B中,从第一时间点起,发送器以系列视频数据包的形式传输视频位流的经编码的第一部分,所述数据包具有基于预定的编码位率的大小。例如,发送器可以在发送器和接收器之间传输用于建立呼叫的一条或多条呼叫消息。

[0108] 在步骤506B中,在第二时间点,发送器从接收器接收一条消息,其包括一个指示所述接收器接收的位元总数的参数(Btotal)。接收器从所述发送器接收视频位流数据包,其来自于以系列数据包的形式传输的所述视频位流的经编码的第一部分,并将该数据送入解码器400用于解码。然后,接收器发送消息(如反向信道的消息,比如应答消息)到发送器;例如,所述消息可以包括(多个)参数(如Btotal)。例如,收到由发送器发送的呼叫消息后,接

收器可以发送一条或多条应答消息。

[0109] 如图5A所讨论,由发送器或接收器发送的每个数据包(如呼叫和应答消息)可以包括一个序号、一个时标等。例如,一旦接收器收到其序号大于或等于预定序号的数据包,接收器可基于接收器收到的位元总数来确定参数(如 B_{total}),而不考虑任何之后收到的数据包。

[0110] 在一些实施例中,消息还可以包括良/劣基准数据,将如下讨论。例如,所述良/劣基准数据可以指示从所述视频位流的经编码的第一部分解码的至少一个帧是否被正确地从一个良基准帧解码。例如,所述消息可以是反向信道消息。发送器接受任何含有参数 B_{total} 消息,并检查当前时间 T_c 。

[0111] 在步骤508B中,基于所述收到的(多个)参数(如 B_{total})、所述第一和第二时间点以及发送器和接收器间的所述往返延迟(未在图5B中示出但在图5A的步骤502A中示出),发送器确定一个初始估计带宽。本步骤类似于如图5A中描述的步骤508A。

[0112] 在一些实施例中,所接收的参数可以包括所述良/劣基准数据以及其他数据。例如,根据所述良/劣基准数据,可以判断所述视频的解码的第一部分是否包括至少一个良基准帧。如果是,所述视频位流的第二部分可以由发送器使用所述至少一个良基准帧以及所述初始估计带宽来编码。如果没有良基准帧,编码器300(发送器)可以使用由发送器估计的初始带宽来编码并重新发送完整的视频位流,其包括所述视频位流的第一部分和第二部分。

[0113] 在一些实施例中,所选中的基准帧可以选自多个当前帧按显示顺序之前的基准帧。所述多个先前基准帧可以包括至少一个良基准帧,其定义为所述编码器已知的、可被无错误解码的一个基准帧。例如,选中基准帧可以是一个良基准帧,而这个良基准帧可用于编码当前帧。又例如,作为选中基准帧的良基准帧可用于编码包括当前帧的一定数目的连续帧,在这种情况下,用同一良基准帧编码的连续帧的所述数目,可以基于一个或多个以下数据自适应地选取:丢包率、带宽数据,和FEC强度。例如,FEC强度可以基于从解码计算设备14接收到的用于编码视频位流322的数据,由一个FEC编码器所确定;而基于所接收的数据(例如反馈信息),所述FEC编码器可以自适应地改变FEC强度和数据包大小。在一些实施例中,操作704中确定的编码参数可以基于一个或多个以下数据被更新:FEC强度、位率,和用同一良基准帧编码的连续帧的数目。

[0114] 在步骤510B中,发送器传输使用初始估计带宽编码的视频位流的第二部分。在一些实施例中,发送器可以基于所估计的带宽重启编码器。如果某些图像在解码器被正确地解码(如上所述,基于收自接收器的消息所包含的所述良/劣基准数据),发送器可以使用该正确的(“良”)基准帧进行预测。但是,如果没有好的图像作基准,发送器可以从关键帧开始重新编码。

[0115] 在一些实施例中,编码计算设备12(发送器)可以使用基于计算设备14(接收器)发送的应答消息所确定的编码参数来编码视频位流的第二部分,所述接收器在接收所述视频位流的第一部分或一些带外发送(如图5A例)的随机数据包后发送所述应答消息。编码参数可以包括多个参数,其可被输入到编码流程以在带宽和纠错方面调整所产生的输出位流。例如,编码参数可以包括但不仅限于位率、FEC率、基准帧选择和关键帧选择。又例如,编码参数可以包括基于带宽数据确定的估计带宽,所述带宽数据包括于上述接收的数据中。公

开的实施方式可以调整编码参数以匹配网络带宽、丢包率以及往返时间,从而优化编码流程,以在给定网络带宽、丢包率及往返时间下在解码计算设备14上提供最高质量的解码视频。

[0116] 如在上述图5A中所述,所述系列数据包的传输,可以发生在建立发送器和接收器间的呼叫流程之中、建立呼叫之后、或任何其他阶段、或在另一个时间。类似地,例如,下述的流程500C的执行,可以发生在建立发送器和接收器间的呼叫流程之中、或在发送器和接收器间的呼叫流程建立之后、或发送器和接收器间传输视频位流的流程之中。

[0117] 图5C是依本公开的一些实施例提供的初始带宽估计的示例性流程500C的流程图,其可用于一个接收器接收一个视频位流。例如,流程500C可以由解码计算设备14(如解码器400)来执行。图5C中的流程图示出了包括在流程500C的几个步骤。流程500C可以用本文中所包含的、或用多于或少于这里包含的步骤来完成。例如,多个步骤可被组合或分割以改变所执行的步骤的数目。流程500C的步骤可以以本文所包含的或不同的顺序来执行,并且仍然实现流程500C的意图。

[0118] 流程500C开始于步骤502C,其中,由接收器接收由发送器发送的系列数据包所关联的一个或多个数据包,其可用于初始带宽估计。所谓“接收”,我们可指输入、获取、读取、访问或以任何方式接收用于初始带宽估计的数据的行为。用于初始带宽估计的所接收的数据可以包括一个或多个数据包,其具有一个数据包大小Psize,而其又可以根据预定的编码位率(如最大位率“Maxbitrate”)来确定,如图5A所述。在一些实施例中(如图5A所述的实施例),所述一个或多个数据包可以由发送器以系列数据包形式发送的填充数据的数据包。在一些其他实施例中(如图5B),所述一个或多个数据包可以关联于视频位流的经编码的第一部分,而视频位流由发送器以系列数据包的形式发送。

[0119] 在步骤504C中,基于所接收的用于初始带宽估计的数据,接收器可以根据预定的规则确定多个参数。由接收器所确定的参数也被称为接收端(解码器端)参数。例如,所述接收端参数可以包括一个指示接收器收到的位元总数的参数(Btotal)。

[0120] 例如,流程500C可以利用接收的呼叫消息的时间和大小来确定接收端参数(如信道参数)。如上所述,每个呼叫消息可以在被创建时被标记时标。此外,流程500C可以给每个数据包标记一个额外的时标来表明它被收到的时间,并发回带有接收时标的应答消息。

[0121] 对于初始带宽估计,如之前在图5A所述,接收器在收到每个数据包时打上一个时标(接收时标是数据包到达端口时的时间,与数据包内含的时标不同——这是发送时标)。在相同的说明性示例中,收到25号(或任何大于25的数据包序号)数据包时,或最长时间窗口/预定时段(如预定的三秒时窗)达到/流逝时,未到达的数据包可被视为已丢失。平均带宽可由以下规则来计算:

[0122] $Bandwidth = ((25-1) - N_{loss}) \times Psize / (T_{last} - T_{first});$

[0123] (以Kbps为单位)

[0124] 其中:

[0125] N_{loss} = 前25个(0-24)数据包中丢包的总数。其不包括后

[0126] 10个数据包(25-34)中的丢包数。

[0127] T_{last} = 紧邻25号数据包之前(不含丢失的包)最后一个数

[0128] 据包的到达时标。单位:毫秒(ms)

[0129] T_{first} = 第一个收到的数据包的到达时标。单位：毫秒 (ms)

[0130] 以上，第一个数据包不用于计算带宽，因为时标是到达时间，这意味着数据包已经收到。

[0131] 任选地，在一些实施例中，如果步骤502C所述一个或多个数据包是用视频位流的经编码的第一部分所生成（例如，见图5B中例），在步骤505C中，接收器从所述一个或多个数据包解码所述视频位流的经编码的第一部分。

[0132] 在步骤506C中，流程500C可以将步骤504C中确定的接收端参数传输到发送器。所述参数可以以一条或多条消息的形式传输，其中每条都包含所述参数（如Btotal）。网络参数可包括前文或后文所述的带宽指标（bandwidth indicator）、累积时间差参数（Tdacc）、接收位率（Rbitrate）和丢包率（Packetlossratio）。在步骤504C中确定的网络参数可经由反向信道消息被发送到编码计算设备12上。例如，反向信道消息可由控制器708上的反向信道消息管理器722发送。关于控制器708和反向信道消息管理器722的进一步细节详见后文与图7相关的描述。

[0133] 在一些实施例中，包括指示收到的位元总数的参数（Btotal）的多条消息（为容错，消息可能会丢失）可被传输到发送器。例如，流程500C可以使用类似于传输呼叫消息数据包的技术、以数据包的形式传输应答消息。例如，接收器可以暂停，直到——例如——25个数据包已被发送或三秒钟结束。例如，此时解码计算设备可以暂停发送数据包；而编码计算设备12（发送器或编码器300）可以利用收到的应答消息数据包来确定网络带宽和其他参数（如丢包率）。在编码计算设备12基于应答消息确定编码参数（如初始估计带宽）期间，编码计算设备可以暂停向解码计算设备14传输视频位流数据。在这段时间内，解码计算设备可保持在就绪状态，准备接收并解码视频位流。

[0134] 例如，接收端参数可以包括收到的位元总数（Btotal）、丢包率、往返延迟、接收位率、带宽数据、指示基准帧是好（良）是坏（劣）的数据，或以上的任意组合。传输的接收端参数可以用于编码计算设备12确定编码参数。意图发挥这种作用的其他数据并不限于本文所述。

[0135] 例如，在允许编码计算设备得以确定网络带宽的暂停之后，在步骤506C中，解码计算设备14（接收器或解码器400）可以形成应答消息，并创建和以一定间隔（如10毫秒的时间间隔）发送包括应答消息的数据包。一旦编码计算设备收到应答消息数据包并估计出网络带宽和数据包损失，编码参数可被重新计算，以反映可用带宽（如初始估计带宽或经调整的带宽）、丢包率以及往返时间。例如，编码参数可以在发送器上被重新计算，计算可基于所估计的带宽、丢包率、往返时间、自适应编码长度、FEC率、视频编码位率、空间分辨率（帧大小）、时间分辨率（帧率）等中的一个或多个。这些用于计算估计带宽的参数中的某些可以由发送器确定，而某些可以从接收器接收。另外，也可以使用接收端参数而不用任何发送端参数。

[0136] 在步骤508C中，接收器从发送器接收一个用初始估计带宽编码的视频位流，所述初始估计带宽是基于指示收到的位元总数的参数所确定。例如，可用的其他参数包括以上描述的任何参数，如带宽、丢包率等等。在一些实施例中，如果步骤502C所述一个或多个数据包是用视频位流的经编码的第一部分所生成，接收器可以接收视频位流的第二部分，而其使用由发送器基于指示收到的位元总数的参数所确定的初始估计带宽编码。如在图5B中

所述,如果从所述视频位流的第一部分没有解码出良基准帧,所述第一和第二部分可被编码并传输。

[0137] 在步骤510C中,流程500C可以解码视频位流。任选地,流程500C可返回到步骤504C,以基于收到并解码的视频位流322的部分继续确定网络参数,如上所述。通过不时确定网络参数(例如,根据视频位流322的部分的数据包所打上的时标),接收视频位流322的部分时可能改变的网络带宽可以被检测到。例如,编码计算设备12可以是一个移动中的主叫移动电话,而解码计算设备14可以是也在移动中的接收端移动电话,其可能会受到不断变化的网络条件(包括网络带宽的变化)的影响。

[0138] 步骤510C之后,如果解码计算设备14仍在接收视频位流322的数据,流程500C可返回到步骤508C以接收下一个视频位流。如果流程500C确定没有在解码计算设备14上收到更多视频位流322的数据,流程500C可以结束。

[0139] 在一些实施例中,如上所述,用接收端参数编码的视频位流的第一部分(如“真实的”视频数据)可由解码器400从编码器300接收,然后解码所述视频位流的第一部分,并用控制器708确定所述视频位流的第一部分相关的接收端参数。接收端参数可以作为反馈信息从控制器708传输到编码器300,以控制编码器300。解码器400从编码器300接收并解码视频位流的第二部分,其中视频位流的第二部分是使用发送端(编码器端)参数进行编码。

[0140] 在一些实施例中,例如,初始带宽估计的执行可以发生在通话过程的不同阶段、使用不同类型的数据/信息,包括但不限于真实视频数据或真实视频数据之外的数据(“填充数据”),如上所述。

[0141] 在一些实施例中,可以创建分离的消息,以在视频会话期间传输初始带宽估计数据及带宽估计/调整数据。

[0142] 图6A-6E示出了依本公开的一些实施例提供在视频位流的传输中进行带宽调整的示例性流程。当编码计算设备12传输视频位流322时,如果视频位流322的编码位率确定自一个比网络实际带宽更高的估计带宽,网络延迟将会增加。这可以通过检测网络延迟来确定,而从网络延迟计算带宽是相当简单的。更困难的是检测到比所述估计带宽更高的实际带宽。如果没有可靠高效的方法来检测比估计带宽更高的实际带宽,解码器动态检测到的网络带宽就只能随着时间下降而永远不会上升。

[0143] 带宽检测可以基于以下假设:如果位率(如,基于估计带宽)高于可用带宽,网络延迟将成比例地增加;而如果估计带宽低于可用带宽,网络延迟将不会增加。例如,如果位率为200Kbps而可用带宽为100Kbps,传输一秒钟的视频将需要两秒钟,或者不得不丢弃一些数据包。如果预期带宽为200Kbps而可用带宽高于200Kbps,传输一秒钟的视频将需要一秒钟。这可以通过比较视频位流322的数据包中包含的时标和视频位流322在解码处理设备14接收到时创建的本地时标来确定。对应时标间的相对差异可以指示最高估计带宽是否已达到。

[0144] 通过不时检测网络带宽的变化,如下图6A-6E所述的实施例可以在视频位流的部分正在传送的期间,自适应地响应网络带宽上升或降低的变化,其传送的速率高到足以克服网络带宽的变化而维持视频质量,又不因发送过多消息而过度降低带宽。这些实施例可以在检测到网络带宽降低时降低位率,并在网络延迟符合估计时小量提高位率。以这种方式,通过上述手段反复采样网络带宽、并每次在网络表现符合估计时小量调整编码参数(如

编码位率),所述网络的最高带宽可以在相当短的时间内确定下来。

[0145] 图6A是依本公开的一些实施例提供的为传输一个视频位流到一个接收器而调整带宽的示例性流程的流程图。例如,流程600A可以由编码计算设备12(发送器)来执行。图6A中的流程图示出了包括在流程600A的几个步骤。流程600A可以用本文中所包含的、或用多于或少于这里包含的步骤来完成。例如,多个步骤可被组合或分割以改变所执行的步骤的数目。流程600A的步骤可以以本文所包含的或不同的顺序来执行,并且仍然实现流程600A的意图。

[0146] 在一些实施例中,带宽调整可以只用通过一个接收器(如解码器400)确定的接收端参数。在一些实施例中,带宽调整可以同时使用接收端和发送端参数。

[0147] 在步骤602A中,发送器传输用于带宽估计的信息到解码设备(接收器)。在一些实施例中,发送器能够传输视频位流322的第一部分,其使用一个当前位率进行编码并被打包成系列数据包。例如,呼叫信息可以作为视频位流322的一部分传输,并由解码计算设备14接收。解码计算设备可基于收到的呼叫消息来确定接收端参数,并经由反向信道将应答消息发回到编码计算设备12。

[0148] 在步骤604A中,发送器可以接收反向信道消息,包括由接收器确定的接收端参数。例如,收到的反向信道消息可包括由接收器在收到所述系列数据包后所确定的接收端参数。例如,接收端带宽参数可以包括累积时间差参数(Tdacc)、接收位率参数、丢失率参数,带宽指示参数、FEC率参数,和/或指示基准帧良/劣的数据,或以上的任何组合。收到的反向信道消息内含的数据可供编码计算设备12确定编码参数。意图发挥这种作用的其他数据并不限于本文所述。

[0149] 在一些实施例中,接收端参数可包括一个良基准帧或者任何可选用于编码的基准帧,视当时的编码效率和带宽条件而定。例如,编码计算设备12可以在不同的基准帧选项之间以及每组中使用同一基准帧的帧的不同数目之间切换,以便根据反馈信息更好地调整适应当前的网络状况。编码参数可以包括多个参数,其可被输入到编码流程以在带宽和纠错方面调整所产生的输出位流。例如,编码参数可以包括但不仅限于位率、FEC率、基准帧选择和关键帧选择。

[0150] 在一些实施例中,包括接收端参数的反向信道消息可以通过图6B中所示的流程来生成,如下所述。

[0151] 图6B是依本公开的一些实施例提供的生成一个反向信道消息的示例性流程600B的流程图,所述反向信道消息包括被发送器用于调整编码参数的接收器端参数。

[0152] 带宽估计可以通过使用一个基于解码计算设备14的本地时间的滑动时窗(sliding window)来动态地执行。所述时窗的长度可以是两秒或依程序提供给流程600A的任意其他预定的时窗长度。示例性流程600B开始于步骤602B,其中,当第一个数据包到达(或被其到达所触发)接收器端口时,时间刻度(time scale)基可被初始化。时间刻度基的初始化如下:

[0153] T_0 = 第一个数据包被接收时的本地时间(使用同一刻度)

[0154] $Trtp_0$ = 第一个视频数据包的实时传输协议(Real-Time

[0155] Protocol,RTP)时标

[0156] 在步骤604B中,接收器检查两秒时窗(Twindow)中的首个和末个数据包的同步源

(Synchronization Source,SSRC)标识。如果它们相同,则步骤继续至创建带宽估计消息;否则,接收器重置T0和Trtp0的值,以便与新的SSRC的首个数据包同步,且不发送消息(这样RTP时标的基也改变了)。

[0157] 在步骤606B中,接收器可以(以接收器本地时间)抓取所述两秒时窗中的首个数据包和末个数据包的RTP时标间隔(Trgap)。假设首个和末个数据包的时标分别为Tr0和Tr1,则 $Trgap = Tr1 - Tr0$ 。用90KHz时钟或更高精度的计时器, $Twindow = 2 * 90000$ (转换到RTP时标的同一时间刻度)。

[0158] 一个或多个参数,如网络带宽指标、累积时间差及接收位率(Rbitrate),可以由接收器在步骤6082B、6084B、6086B分别确定。其他参数也可以被接收器确定,并包括在发到发送器的反向信道消息中。

[0159] 网络带宽指标(“带宽指标”)可以由接收器在步骤6082B算出,作为实时传输协议(RTP)时间间隔(Tgap)和预定的时窗时间(Twindow)的函数。在一些实施例中,网络带宽指标可以计算为Twindow对Tgap的一个比值,而其可根据以下规则指示当前网络条件:

[0160] bandwidth indicator<1: 指标表明因网络带宽不足引起的网络延迟

[0161] 上升。

[0162] bandwidth indicator=1: 指标表明网络能够无碍传输所述视频。带

[0163] 宽有可能适应更高位率。

[0164] bandwidth indicator>1: 指标表明数据包比实时更快地突发到达。

[0165] 这可能表明网络拥塞正在缓解。例如,这

[0166] 可能是停止文件下载或解除带宽限制手段

[0167] 的结果。数据包的突发到达也可能表明网

[0168] 络过度跳动(excessively jittery)。绝大多数

[0169] 网络跳动的条件下,带宽指标将接近1。

[0170] 在操作6084B中,可以计算RTP时间和本地时间之间累积的时间差Tdacc,根据以下公式:

[0171] $Tdacc = (Tr1 - Trtp0) - (Tcurrent - T0)$

[0172] 其中:

[0173] Tr1=当前时窗中末个数据包的时标

[0174] Trtp0=与所述末个数据包有相同SSRC的整个序列的首

[0175] 个数据包的时标

[0176] Tcurrent= 当前本地时间

[0177] T0=所述首个数据包被接收时的本地时间

[0178] 累计时间差Tdacc的持续增长可以指示所述网络带宽不足以传输所述视频位流。例如,当探测不到延迟的小量增加时,这可以用来校正对所述两秒时窗的调整。

[0179] 在步骤6086B中,实际接收位率(Rbitrate)可计算为在当前时窗收到的数据包(包括FEC数据包)的总位元数除以当前时窗的本地时间的总时长(本例中为两秒)。

[0180] 此外,数据包的总数(Ptotal)和丢失的数据包的总数(Plost)可以通过校检数据包序号来检查。例如,可以通过用最末的RTP序号减去最初的RTP序号,并与接收到的包的计数作比较。Ptotal和Plost可以用来确定丢包率Packetlossratio。

[0181] 在步骤610B中,反向信道消息可以在同一消息中包括一个或多个如下参数:带宽指标、Tdacc、Rbitrate和Packetlossratio。然后,所述反向信道消息可以被发送到发送器/编码器并依照2015年9月28日提交的美国专利申请号14/867143(“143申请”)中所述的方式被用于设置发送器/编码器中的参数,143申请的全文通过引用在此并入本文。

[0182] 回到图6A,任选地,在步骤606A中,流程600A可确定发送端参数。例如,从解码计算设备14收到反向信道消息后,编码计算设备12可以分析所述反向信道消息,并可以结合其他消息及存储的参数(包括统计)来确定用于编码视频位流322的第二部分的发送端参数。在一些实施例中,基于如步骤502A所述的发送端发送和接收系列数据包的时标差,发送器能够确定发送端参数(如往返延迟)。

[0183] 当仅用接收端参数来调整带宽时,步骤606A可以省略。在这些实施例中,发送器将基于所述接收端参数来调整当前位率(带宽)。例如,接收端参数可以包括一个或多个如上所述的参数,如Tdacc、Btotal、Rbitrate、带宽指示、FEC率、丢包率等。

[0184] 在步骤608A中,发送器调整用于编码视频位流的当前位率。在一些实施例中,所述调整可以仅基于步骤604A中所述的收到的接收端参数。在一些实施例中,所述调整可以基于收到的接收端参数和由发送器确定的发送端参数,如接收端带宽指标和往返延迟数据,如在步骤606A中所述。仅用接收端参数调整编码参数(如用于编码视频位流的当前位率)的一个例子,可见图6C示出的流程600C。

[0185] 图6C是依本公开的一些实施例提供的为编码视频位流而调整(当前)位率的示例性流程600C的流程图。例如,当前位率的动态调整可以基于流程500A-500C及600A-B中所述的参数。例如,流程600C可以由编码计算设备12来执行。图6C所示的流程图展示了包括在流程600C中的若干步骤。实现流程600C的步骤,可以包括在本文,或者多于或少于本文所述。例如,多个步骤可被组合或分割以改变所执行的步骤的数目。流程600C的步骤可以以本文所包含的或不同的顺序来执行,并且仍然实现流程600C的意图。

[0186] 如上所述,前向纠错(FEC)是一种纠错技术,它在一个视频位流的数据包中增加额外的数据包,以允许接收器恢复丢失或损坏的数据包而无需重传所述数据包的数据。输出视频位流的每个数据包,可以通过零个或多个FEC包的数据来保护,例如:所述输出视频位流的一个数据包可能未被FEC包的数据保护,也可能被多个FEC包所保护,取决于所述数据包在解码视频位流中所预定的重要性。例如,相对于一个中间帧的代表像素数据的系数,包含运动矢量的数据包可被更多的FEC包数据保护。例如,使用FEC包保护视频位流的数据包的流程可被若干参数控制,如FEC比率参数(FEC_ratio),其描述的是视频位流的数据包和FEC包之间的比率。

[0187] 流程600C开始于步骤602C,其假设FEC_ratio被设定为用于保护当前视频位流322的当前值,当前的编码器位率被设置为Ebitrate,允许的预定最高位率为Maxbitrate。

[0188] 在步骤602C中,流程600C测试FEC_ratio是否为0;如果是,就在步骤604C中设定变量Sbitrate=Ebitrate。

[0189] 在步骤606C中,如果FEC_ratio不为0,设定Sbitrate=Ebitrate(1+1/FEC_ratio)。所达到的效果是,当前位率的增量正比于FEC保护的量。

[0190] 在步骤608C中,例如,收到的网络带宽指标(也称为“网络带宽”或“BWidthI”)被归一化到0,并测试其是否小于一个小值(如0.05),且同时测试当前累积时间差(Tdacc)是否

也小于一个小值(如200毫秒)。

[0191] 如果以上条件同时满足(如同时为“真”),则意味着网络可以正常处理当前位率,因此在步骤614C中流程600C可以小额(如5%)提高估计的位率,例如,通过设置变量 $Newbitrate = Sbitrate * BwidthI * 1.05$ 。

[0192] 如果在步骤608C的测试为假,则在步骤610C中进一步测试带宽指标 $BwidthI$ 是否大于1.1;如果是,则如前文所述,所述网络可能处在快速突发传输中,因此在步骤616C中流程600C可以通过设置变量 $Newbitrate = Sbitrate * 1.1$ ——即增高10%的位率——来探测网络带宽是否有所增加。

[0193] 如果在步骤610C中确定 $BwidthI < 1.1$,则意味着网络的延迟在上升,因此步骤612C通过设置 $Newbitrate = Sbitrate * BwidthI$ 来调低位率。

[0194] 在步骤618C中,估计位率被设定为 $Ebitrate = Newbitrate / (1 + 1/FEC_ratio)$,以补偿前向纠错要添加到位流中的附加位元。

[0195] 在步骤620C中,累积延迟被测试是否大于或等于200毫秒的预期值。

[0196] 如果是,则意味着网络延迟正在上升,因此在步骤622C中估计位率 $Ebitrate$ 被设定为其值的90%。

[0197] 如果在步骤620C中所述网络延迟小于其估计值,则在步骤624C中测试 $Ebitrate$ 是否大于允许的最大值 $Maxbitrate$ 。如果是,则其在步骤626C中被降低到等于 $Maxbitrate$ 。

[0198] 继上述步骤后,流程600C可以返回到图6A的步骤608A以完成流程600A。

[0199] 用经调整的编码参数——如调整后位率(基于带宽估计),流程600A可以编码视频位流322的第二部分(未示出)。在一些实施例中,编码计算设备12确定用于编码视频位流322的当前帧的一个选中基准帧。在一些实施例中,所选中的基准帧可以选自多个当前帧按显示顺序之前的基准帧。所述多个先前基准帧可以包括至少一个良基准帧,其定义为所述编码器已知的、可被无错误解码的一个基准帧。例如,选中基准帧可以是一个良基准帧,而这个良基准帧可用于编码当前帧。又例如,作为选中基准帧的良基准帧可用于编码包括当前帧的一定数目的连续帧,在这种情况下,用同一良基准帧编码的连续帧的所述数目,可以基于一个或多个以下数据自适应地选取:丢包率、带宽数据,和FEC强度。例如,FEC强度可以基于从解码计算设备14接收到的用于编码视频位流322的数据,由一个FEC编码器所确定;而基于所接收的数据(例如反馈信息),所述FEC编码器可以自适应地改变FEC强度和数据包大小。在一些实施例中,操作704中确定的编码参数可以基于一个或多个以下数据被更新:FEC强度、位率,和用同一良基准帧编码的连续帧的数目。在一些实施例中,视频流322的当前帧使用所选择的基准帧和编码参数进行编码。在一些实施例中,编码过程可在以下描述中阐述。

[0200] 例如,视频位流322的第一部分可被编码并作为视频位流322的部分传输,再由解码计算设备14接收。解码计算设备14能够基于收到的消息确定接收端参数,并通过反向信道(如反向信道消息)将消息发回编码计算设备12。例如,编码的计算设备12可以接收接收端参数并计算调整后的编码参数,然后用所确定的下一批编码参数编码视频位流322的第二部分。视频位流322的第二部分的编码可以基于接收端参数以及——任选地——发送端参数。被编码后,视频位流322的第二部分可以由编码计算设备12经由网络16传输到解码计算设备14。例如,解码的计算设备可以确定接收端参数并通过反向信道信息向编码计算设

备12发回所确定的接收端参数。

[0201] 在步骤610A中,发送器将用调整后的编码参数(如调整后的当前位率)编码的视频位流的第二部分传输到接收器。在一些实施例中,流程600A可以通过回到步骤604A而继续,步骤604A可以接收下一批用于下次带宽调整的反向信道消息,直到停止传输视频。

[0202] 仍在图6A中,通过回到步骤604A,基于收到的反向信道消息(包括接收端参数),发送器可以确定是否需要额外调整带宽。如果为真,流程600A可以继续到可选步骤606A以确定下一批发送端参数,如同带宽调整需要接收端和发送端参数的情形;或者流程600A可以继续到步骤608A再一次调整带宽,如同带宽调整只需要接收端参数的情形。如上所述,确定编码参数的频率将决定流程600A可以多快、多顺地响应网络带宽的变化,且不因增加反向信道消息而显著降低网络带宽。如果流程600A确定没有余存的视频流数据,则流程600A可以就此结束。

[0203] 在一些实施例中,编码计算设备12(发送器)可以在使用已知的良基准帧和使用任一基准帧(例如当前帧的前一帧)之间切换。例如,所述选择可以基于编码效率和质量之间的权衡。例如,选择任一基准帧(例如当前帧的前一帧)时,编码效率更好,但由于传输过程中发生的错误,经解码的视频质量可能更低。

[0204] 图6D是依本公开的一些实施例提供的调整带宽的示例性流程的流程图,所述流程可由一个接收器用于接收一个视频位流。带宽调整流程类似于图6A-C的说明,而我们的描述集中在由接收器执行的操作。流程600D包括步骤602D-610D,其对应于流程600A中的步骤602A-610A。

[0205] 在步骤602D中,接收器接收视频位流的第一部分所关联的一个或多个数据包,所述视频位流用当前位率编码并传输为系列数据包。在一些实施例中,所述一个或多个数据包可由发送器在步骤602A中发送。

[0206] 在步骤604D中,接收器基于收到的一个或多个数据包确定接收端带宽参数。在一些实施例中,例如,如果接收端参数是在带宽调整中用到的唯一信息,接收端带宽参数可以包括累积时间差参数、接收位率参数、丢包率参数、带宽指示参数及FEC比率参数。

[0207] 在步骤606D中,接收器从所述一个或多个数据包中解码视频位流的经编码的第一部分。

[0208] 在步骤608D中,在收到一个或多个数据包后,接收器向发送器传输一条或多条反向信道消息,其中每条都包括接收端带宽参数。

[0209] 在步骤610D中,接收器从发送器接收用调整后的当前位率进行编码的视频位流的第二部分。在一些实施例中(如,见图6A),可由发送器在收到所述一条或多条反向信道消息后、基于接收端带宽参数来确定调整后的当前位率。在另一些实施例中(如,见图6E),可由发送器在收到所述一条或多条反向信道消息后、基于接收端带宽参数和发送端数据来确定调整后的当前位率,在所述发送端数据在收到所述一条或多条反向信道消息后被确定。

[0210] 在一些实施例中,在步骤610D后,流程600D可以回到步骤604D,以根据收到的视频的第二部分的数据包来确定下一批接收端带宽参数,直到视频停止传输。

[0211] 在一些实施例中,由于音频和视频占据同一带宽的事实,动态带宽估计可以仅基于视频信道,而音频信道带宽 $A_{bandwidth}$ 可以预先固定在一个速率上(如100Kbps)。因此,用于带宽调整的反向信道消息可以仅仅参考视频信道带宽(如果带宽小于音频信道带宽,

其被设置为零)。

[0212] $V_{bandwidth} = Bandwidth - A_{bandwidth}$;

[0213] $V_{bandwidth}$ 是反向信道消息中用以控制编码器/接收器参数设定的参数。

[0214] 图6E是依本公开的一些实施例提供的为传输及接收一个视频位流而调整带宽的示范性流程600E的流程图。流程600E示出了同时涉及发送器(如编码计算设备12)和接收器(如解码运算设备14)的动态带宽调整流程600E。

[0215] 在视频会话期间,网络条件(包括带宽)可以改变,而发送器需要动态调整编码位率。这个例子同时使用了发送器端和接收器端的信息来作出调整。

[0216] 例如,使用如上所述的数据包的发送时间和应答时间,发送器可以在步骤602E中计算当前往返时间/延时(CurrentRTT),在步骤604E中计算平均往返时间/延迟(AverageRTT),并在步骤606E中计算本地最小往返时间/延迟(LocalMinimalRTT)。LocalMinimalRTT是在呼叫会话的某段时间内最小的RTT,直到因满足某些条件(如达到一个最小位率等)而被复位。

[0217] 例如,接收器在步骤608E中计算带宽指标BandwidthIndicator,并如上所述在步骤610E中计算累计时间差指标AccumulatedIndicator(Tdacc)。BandwidthIndicator和AccumulatedIndicator可以在步骤612E中以消息的形式通过反向信道消息从接收器端发送到发送器端。由发送器和接收器两者算出的值在步骤614E中可用于动态修改发送参数。然后在步骤616E中,基于修改后的参数,视频位流从发送器发送到接收器,而视频信号最后在步骤618E中由接收器处理。

[0218] 根据公开的实施例,图7是包括反向信道消息管理器的示范性编解码器700的框图。编解码器700可以实现上文图5A-5C和图6A-6D所述的流程500C-500D和600A-600D。编解码器700可以用计算设备12、14实现。编解码器700可以编码视频流200或解码视频位流322,取决于它在运行时受到的指示。编解码器700可以用录制器702采集视频流200的数据。通过实时数据采集——例如用摄像机,或通过读取视频流的数据——例如从存储设备或网络,录制器702可以采集未压缩的视频流数据。

[0219] 当编解码器700作为编码器工作时,录制器702可以将未压缩的视频流200传递到编码器封装器704。编码器封装器704可以检查所述输入的未压缩的视频流200,从反向信道控制器708接收参数(如反向信道消息),并从非易失性存储设备读取储存的参数和统计信息以确定编码参数,再将编码参数与视频流200一起发送到编码器706。编码器706可以是类似于图3中编码器300的编码器。编码器706可以用所述接收到的编码器参数来编码视频流200,以产生一个编码视频位流322,而其具有由反向信道控制器708选择的估计位率。编码器可以将包含在所述编码视频位流中的所述数据包传递到前向纠错(FEC)编码器716,716可以创建,并依据包括诸如FEC比率的FEC编码参数添加FEC数据包到输出的视频位流中。随后,FEC编码器可以将包括在输出视频位流中的数据包经网络718传递到数据传出模块720进行传输。

[0220] 当编解码器700作为解码器工作时,可以通过数据传入模块712,从网络718接收含在经编码的视频位流322中的数据包,并传递到前向纠错(FEC)解码器726。FEC解码器可以将FEC数据包从传入的视频位流中去除,并在需要并可行时,还原丢失或损坏的数据包。例如,FEC解码器可以将有关丢失或无法恢复数据包的信息发送到良/劣信息提供器714。然

后,FEC解码器可以将视频位流322与解码器参数一起发送到解码器封装器732。解码器封装器可以检查视频位流,并将参数信息——例如时标和数据包序号——返回至解码器状态回调724。解码器730可以是类似于图4中解码器400的解码器。解码器730可以根据所述传递的解码器参数解码所述视频位流322,并将解码后的视频流输出到渲染器728;而在728中,所述视频流在渲染后,例如,可被显示在连接至解码计算设备14的一个显示设备上,或存储到一个非易失性存储设备中。

[0221] 除了编码和解码视频数据,编解码器700可以包括反向信道消息管理器722,而其可以是控制器(也可称“反向信道控制器”)708的一部分。如上所述,反向信道消息管理器722负责创建、传输和接收消息(如呼叫及应答消息)。运行于编码模式时,反向信道消息管理器722可以通过数据传出模块720将呼叫消息传送到网络718,并通过数据传入模块718从网络718接收应答消息。所接收到的应答消息,可以通过带宽估计模块710进行分析,以确定网络参数。例如,在一些上述实施例中,所述网络参数可以包括以下一项或多项: B_{total} 、带宽指标、 T_{dacc} 、R位率、数据包 $lossratio$ 以及其他参数;上述参数可以用于带宽估计(如初始带宽估计)或调整。所述参数可以包含于反向信道消息(如呼叫和应答消息)。反向信道消息管理器722可以通过传入端口712和传出端口720接收和发送反向信道信息(如呼叫和应答消息),并通过解码器状态回调724和带宽估计模块710,针对用于设置编码器参数的所述网络参数,管理其计算和收集。运行于解码模式时,反向信道消息管理器724可以通过传入端口712从网络718接收呼叫消息,用带宽估计模块710确定网络参数,并创建应答消息,以通过传出端口720传输到网络718。

[0222] 基于所接收和计算出的网络参数,带宽估计模块710可以估计可用的网络带宽;所述网络参数包括往返延迟、解码器端接收位率、丢包率和解码器端带宽指标(包括带宽指标和累积指标)。带宽估计模块710的示例流程已经在图5A-5C、6A-6E中进行了讨论。例如,由控制器708确定的编码参数可以包括FEC强度、位率、基准帧的数目和供使用的基准帧。根据控制器708确定的编码参数,FEC编码器可以自适应地改变FEC强度和数据包大小。

[0223] 编解码器700的一个特点,是其有能力为适应变化的网络条件而动态地改变用于帧内预测的基准帧的数目。

[0224] 图8示出的是编码器802正在输入视频流200以待编码成视频位流322。视频编码器802可使用某个数目818个的基准帧 R_1, R_2, \dots, R_n ,即812、814、816,来编码视频位流322。使用更多基准帧可以提高所传输视频位流的质量,但也可能需要更大的网络带宽。调整要用的基准帧的数目为818,可以将传输所需基准帧的数目匹配到可用的网络带宽。视频解码器804可以调节解码的基准帧 R_1, R_2, \dots, R_n 即820、822、824的数目826,用于解码视频位流322,以匹配编码器802用来编码所述视频位流的基准帧的所述数目;所述匹配可以根据从编码器802接收描述帧的数目的参数及关联于所述基准帧的其他数据,而所述接收可以直接经由所述视频位流或经由反向信道消息来进行。

[0225] 根据公开的实施例,图9示出了选择基准帧的一个例子。图9示出的视频流900包括帧群组 M_1, M_2 和 M_3 。群组 M_1 包括一个帧内编码的基准帧I及其若干预测帧P。预测帧P可用包含在I的信息和编码在所述视频位流中的预测信息重建。群组 M_2 包括首帧 P_I ,而 P_I 帧是用解码器缓存中已知的良基准帧编码而成。如果解码器(接收器)可以无错误地解码一个基准帧,所述基准帧是一个良基准帧。在一些实施例中,为使一个基准帧为良基准帧,其所依赖

的基准帧也必须没有任何错误。如果编码器已知所述良基准帧无错,则所述良基准帧是一个已知良基准帧。良基准帧不必须是一个I帧,且可以从先前(正确地)解码的帧被重建,例如从群组M1的帧I。这意味着,针对群组M2传输一个独立的I帧并无必要。例如,一旦解码器(接收器)确定 P_I 是在解码器缓存中的良好基准帧,就可以向编码器(发送器)指示 P_I 是编码器的良基准帧,而指示可以直接在所述位流中或通过反向信道消息来进行。因此,所述编码器(发送器)得知 P_I 是良基准帧,且可以用于预测后续帧。同样地,帧群组M3包含一个 P_I 帧,而其也可以从一个由运行时反向信道消息提示的已知良基准帧重建,因此无需传输独立的I帧以重建群组M3的预测帧P。如图9中的省略号所示,该方案可以继续用于更多群组的帧。

[0226] 通过反向信道消息控制器708中的反向信道消息管理器722,视频编码器可使用来自解码器的反馈信息来确定应当用哪个帧来编码。例如,一个良基准帧或任一基准帧可以针对编码而被选定,取决于其时的编码效率和带宽条件。例如,为了更好地适应当前网络条件,编码计算设备12可以基于反馈信息,在基准帧的不同选项中切换,也可以切换使用同一基准帧的各图像群组中帧的不同数目。

[0227] 例如,针对编码视频位流322的一个当前帧,编码器(发送器)可以在使用已知良基准帧和使用任一基准帧(例如当前帧的前一帧)之间切换。例如,所述选择可以基于编码效率和质量之间的权衡。例如,选择任一基准帧(例如当前帧的前一帧)时,编码效率更好,但由于传输过程中发生的错误,经解码的视频质量可能更低。

[0228] 当所选的基准帧是一个良基准帧时,同一良基准帧可用于编码,例如,一定数目的包括当前帧的连续帧。使用同一良基准帧编码的连续帧的数目(如图9中的M2、M3)可以基于以下因素自适应地选择:丢包率、带宽数据、FEC强度,或上述的任意组合。如在图9中,各群组中帧的数目,如 $M_1, M_2, M_3 \dots M_i$,可以在帧边界动态地变化,并且各群组 $M_1, M_2, M_3 \dots M_i$ 的值可以通过以下因素确定:丢包率、带宽、FEC强度,或上述的任意组合。例如,基于以下因素,编码参数可以被更新:FEC强度、位率、使用同一良基准帧编码的连续帧的数目,或上述的任意组合。

[0229] 在一些实施例中,FEC强度可以基于从解码计算设备接收到的用于编码视频位流的数据,由一个FEC编码器所确定;而基于接收自所述解码计算设备的用于编码所述视频位流的数据(例如反馈信息),所述FEC编码器可以自适应地改变FEC强度和数据包大小。例如,所述用于编码视频位流的数据(例如反馈信息),可以进一步包括丢包率、往返延迟、接收位率、带宽数据、指示一个基准帧属良或劣的数据,等等。例如,所述编码参数可以包括估计带宽,而其确定是基于在反馈信息中接收到的所述带宽数据。

[0230] 以上描述的编码和解码的实施例说明了一些示例性的编码和解码技术。然而,“编码”和“解码”,如同在权利要求中使用一样,可以意指压缩、解压缩、转换,或任意其他处理或变化数据。

[0231] 计算设备12和/或计算设备14的实现方式,以及存储其上和/或由其执行的算法、方法、指令等,可以实现为硬件、软件或两者的任意组合。例如,所述硬件可以包括计算机、知识产权(IP)核心、专用集成电路(ASIC)、可编程逻辑阵列、光学处理器、可编程逻辑控制器、微代码、微控制器、服务器、微处理器、数字信号处理器或任意其他适合的电路。在权利要求中,术语“处理器”或单独或组合地包括任意上述硬件。术语“信号”和“数据”可互换使用。此外,计算设备12和计算设备14的各部分无须以相同的方式实现。

[0232] 此外,例如,在一个实施例中,计算设备12或计算设备14可用通用计算机/处理器及计算机程序实现;且在执行时,所述计算机程序执行任意上述相应的方法、算法和/或指令。此外,例如,也可以另外使用专用计算机/处理器,而其可以包含专用硬件用于执行本文所述的任意所述方法、算法、或指令。

[0233] 计算设备12和计算设备14可以,例如,在一个屏幕录制(screencasting)系统的计算机上实现。另外,计算设备12可以在服务器上实现,而计算设备14可以在独立于服务器的设备上实现,诸如手机或其他手持式通信装置。在这个例子中,计算设备12可以使用编码器300将内容编码为一个编码的视频信号,并发送所述编码的视频信号到通信设备。反过来,所述通信设备可以使用解码器400解码所述编码的视频信号。另外,通信设备也可以解码本地存储在所述通信设备上的内容,例如非由计算设备12传送的内容。计算设备12和计算设备14也可以有其他合适的实施方案。例如,计算设备14可以是大致固定的个人计算机而非便携通信设备,和/或包括编码器300的设备还可以包括解码器400。

[0234] 本文所述的全部或部分方面可用带计算程序的通用计算机/处理器来实现;其在执行时,可实现任意在此描述的各方法、算法和/或指令。例如,可以额外地或可替代地使用专用计算机/处理器,其可包含专用硬件用于执行任意本文描述的方法、算法或指令。

[0235] 本文所述的计算设备(及存储其上和/或由其执行的算法、方法、指令等)可以在硬件、软件或两者组合上实现。例如,硬件可以包括知识产权(Intellectual Property, IP)核心、应用专用集成电路(Application-Specific Integrated Circuits, ASICs)、可编程逻辑阵列、光学处理器、可编程逻辑控制器、微代码、固件、微控制器、服务器、微处理器、数字信号处理器,或任何其他合适的电路。在权利要求中,术语“处理器”应被理解为包括单独或组合使用的任意前述装置。术语“信号”和“数据”可互换使用。此外,计算设备的各部分不必以相同的方式来实现。

[0236] 例如,一个或多个计算设备可以包括一个ASIC或可编程逻辑阵列(如场效可编程逻辑门阵列,Field-Programmable Gate Array, FPGA)等配置为用来执行一个或多个本文描述或要求保护的操作的专用处理器。一个示例性FPGA可包括逻辑模块和随机存取存储器(RAM)模块的集合,可单独配置和/或互连配置,以使FPGA执行某些功能。某些FPGA也可含有其他通用或专用的模块。一个示例性FPGA可以基于硬件定义语言(Hardware Definition Language, HDL)设计进行编程,如超高速集成电路(Very High Speed Integrated Circuit, VHSIC)硬件描述语言或者Verilog。

[0237] 本文的实施例可用功能块组件和各种处理步骤来描述。所公开的方法和顺序,可以单独或以任何组合来执行。功能块可以通过任何数量的执行指定功能的硬件和/或软件组件来实现。例如,所描述的实施例可以使用各种集成电路组件,如存储器元件、处理元件、逻辑元件、查找表等,其可以在一个或多个微处理器或其他控制设备的控制下执行多种功能。类似地,所描述的实施例的元件使用软件编程或软件元件实现时,本发明可以使用任何编程或脚本语言来实现,如C、C++、Java、汇编等;各种算法的实现可以结合任意组合的数据结构、对象、过程、例程或其他编程元素。功能方面可以用在一个或多个处理器上执行的算法来实现。此外,本发明的实施例可以采用任意数量的常规技术,用于电子学配置、信号处理和/或控制、数据处理等。术语“机制”和“元件”属广义使用,且不限于机械的或物理的实施例或方面,但可包括结合处理器等的软件例程。

[0238] 上述公开内容的各方面或方面的一些部分可以以计算机程序产品的形式存在,而其可以从比如计算机可用或计算机可读介质中读写。计算机可用或计算机可读介质可以是任何设备;例如,其可以有形地包含、存储、通信或经任何处理器或通过连接到任何处理器传输程序或数据结构以供使用。例如,所述介质可以是电子、磁、光、电磁或半导体器件。其他合适的介质也可以使用。这样的计算机可用或计算机可读介质可以被称为非易失性存储器或介质,并且可以包括随机存取存储器或其他可能会随时间而改变的易失性存储器或存储设备。除非另有说明,本文中所描述的装置的存储器不必在物理上包含于该装置,但可以由装置远程访问,且不必与可能被物理包含于该装置的其他存储器邻近。

[0239] 任何本文中描述的作为示例执行的单个或组合功能,可以用机器可读指令来实现,而其形式可以为任一或任意组合的前述计算硬件的操作代码。计算代码可以实现为一个或多个模块的形式,通过所述形式单个或组合功能可以作为计算工具来执行,而在本文所述的方法和系统的操作中,每个模块的输入和输出数据可被传递到/自一个或多个其他模块。

[0240] 信息、数据和信号可以使用多种不同的技术和工艺来表示。例如,任何数据、指令、命令、信息、信号、位元、符号和本文引用的芯片,可以通过电压、电流、电磁波、磁场或磁性粒子、光场或光学粒子、其他物品或前述的任意组合来表示。

[0241] 本说明书包含各种标题和副标题以增强可读性,并帮助在说明书中寻找和索引材料的过程。这些标题和副标题并非意图,且不应被用于,影响权利要求的解释或以任何方式限制权利要求的范围。本文所示和所述的具体实现方式是本公开的说明性实施例,且不意在以任何方式限制本公开的范围。

[0242] 本文引用的所有参考文献,包括出版物、专利申请和专利,在此通过引用以同等程度并入本文;每个参考文献视同被单独、明确地指明并通过引用并入本文且以其整体在此阐述。

[0243] 上述实施方式已经依序描述,以便容易理解、且不限本公开。相反,本公开旨在覆盖包含在所附权利要求书的范围内的各种修改和等同布置,所附权利要求书的范围在法律范围内应被赋予最宽泛的解释,以包含所有上述修改和等同结构。

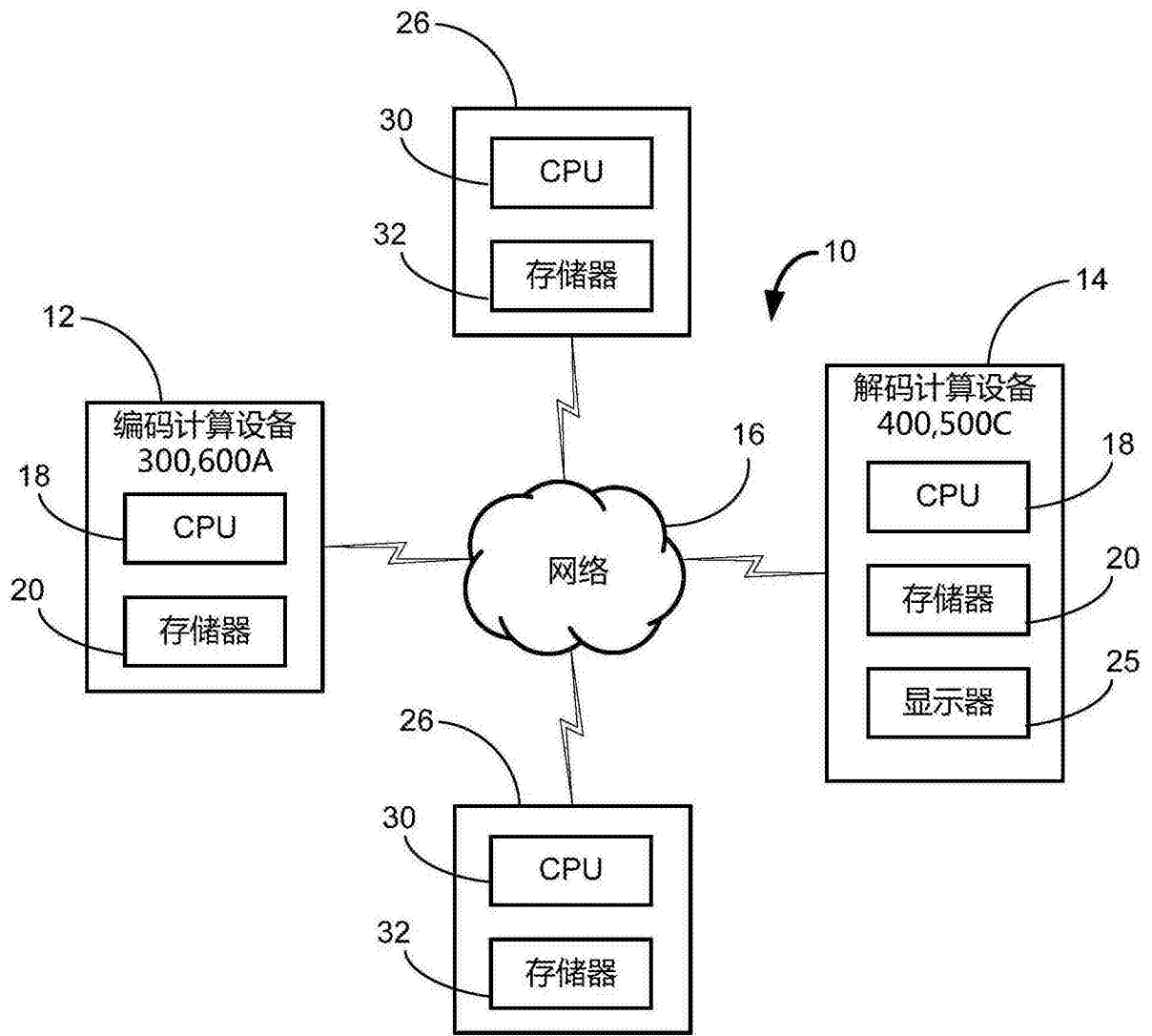


图1A

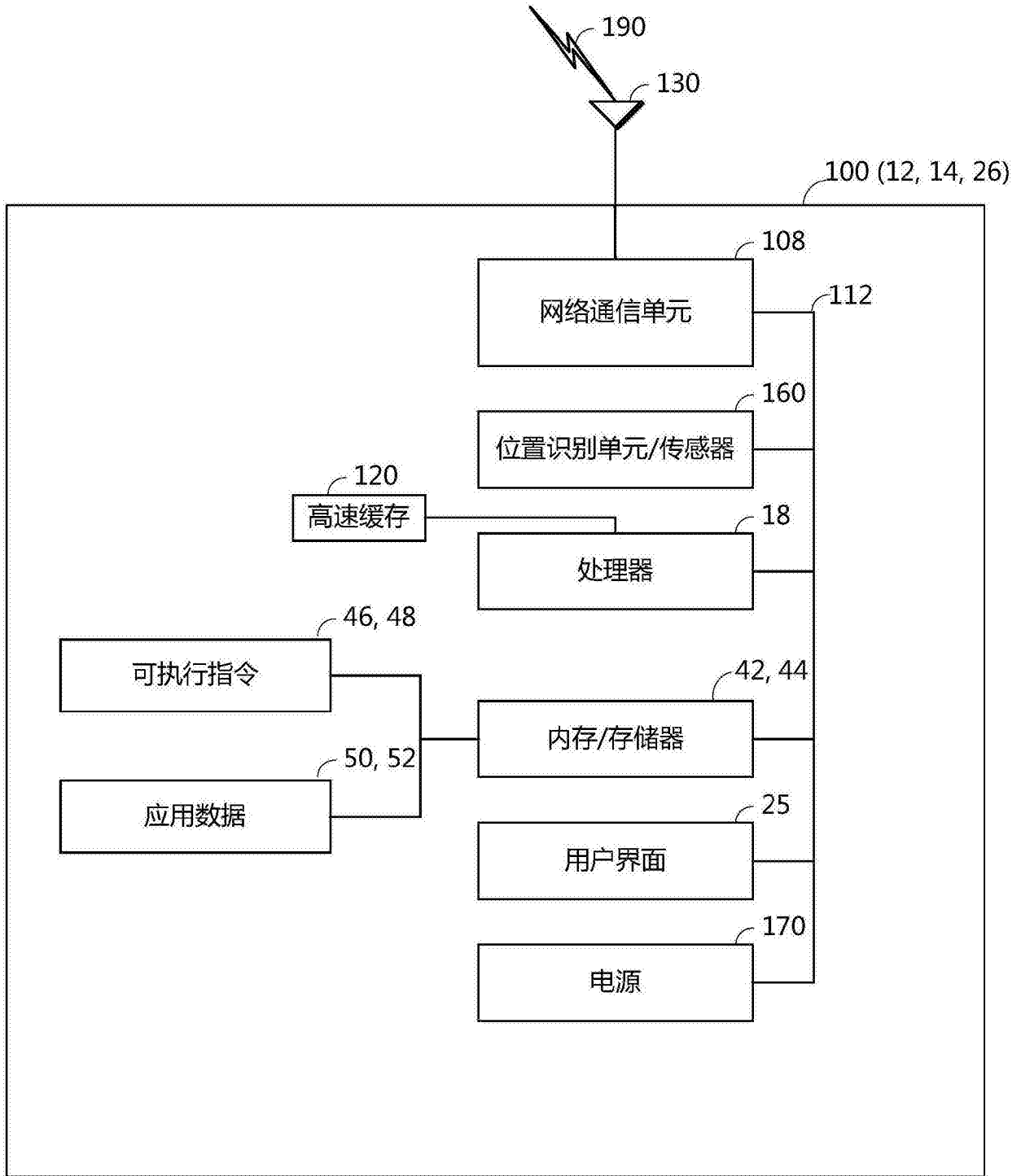


图1B

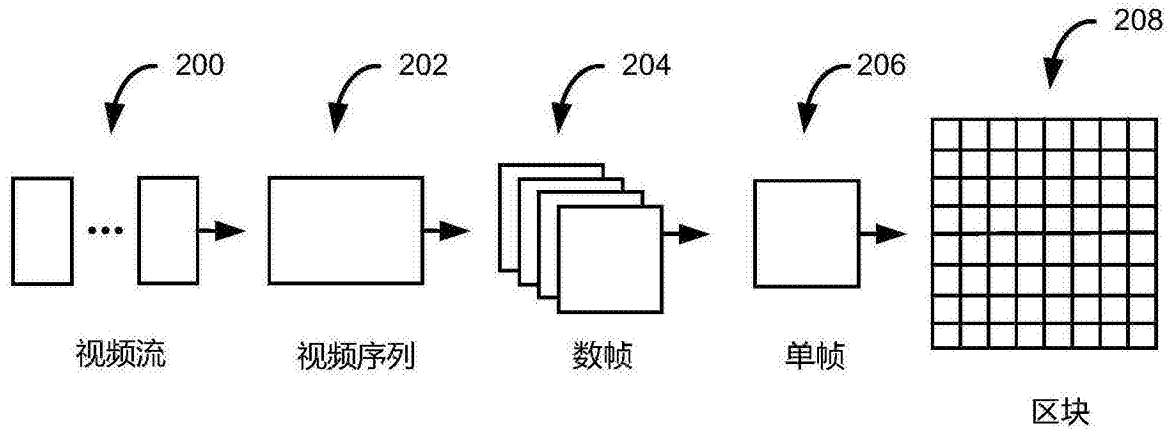


图2

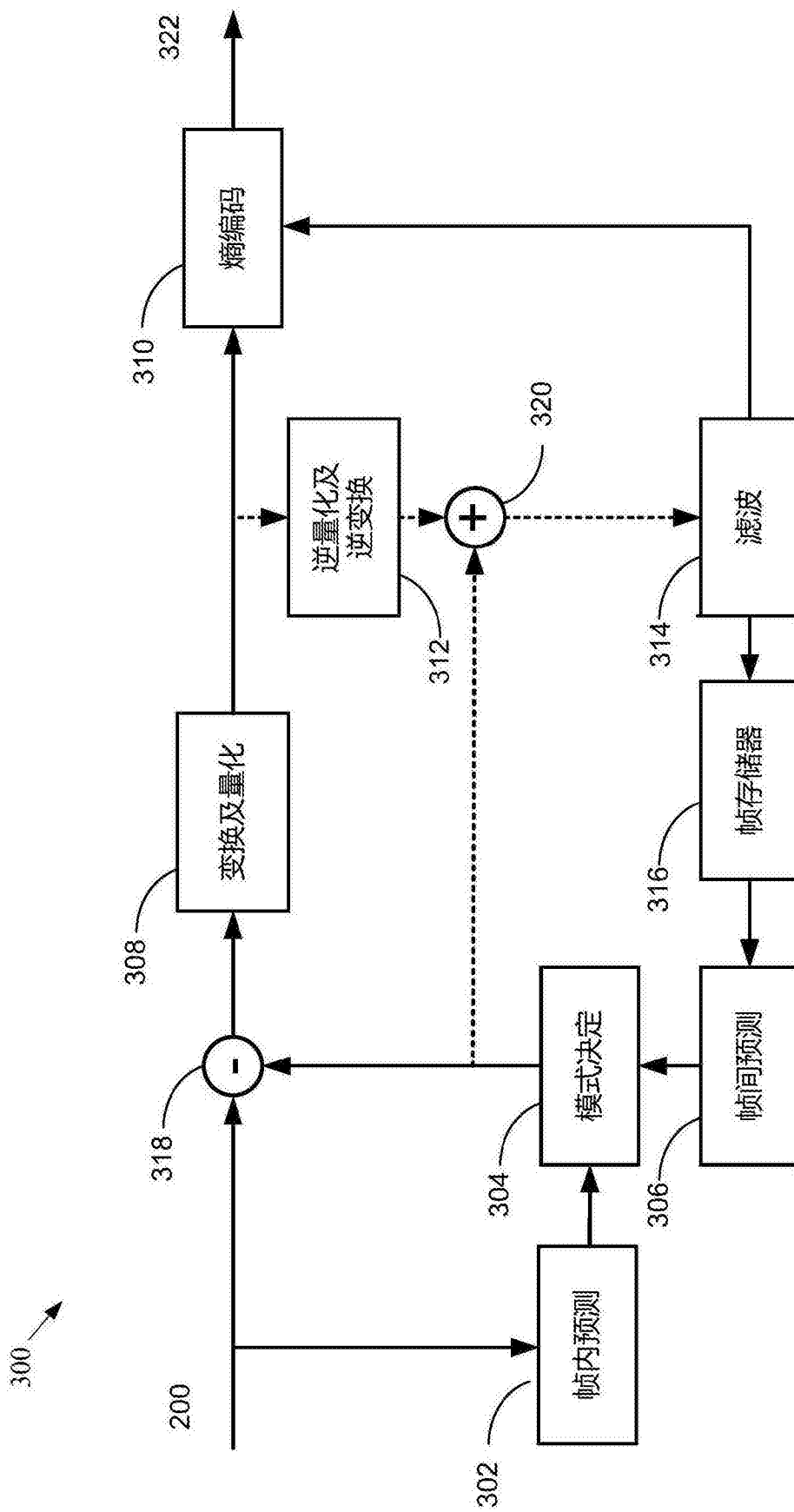


图3

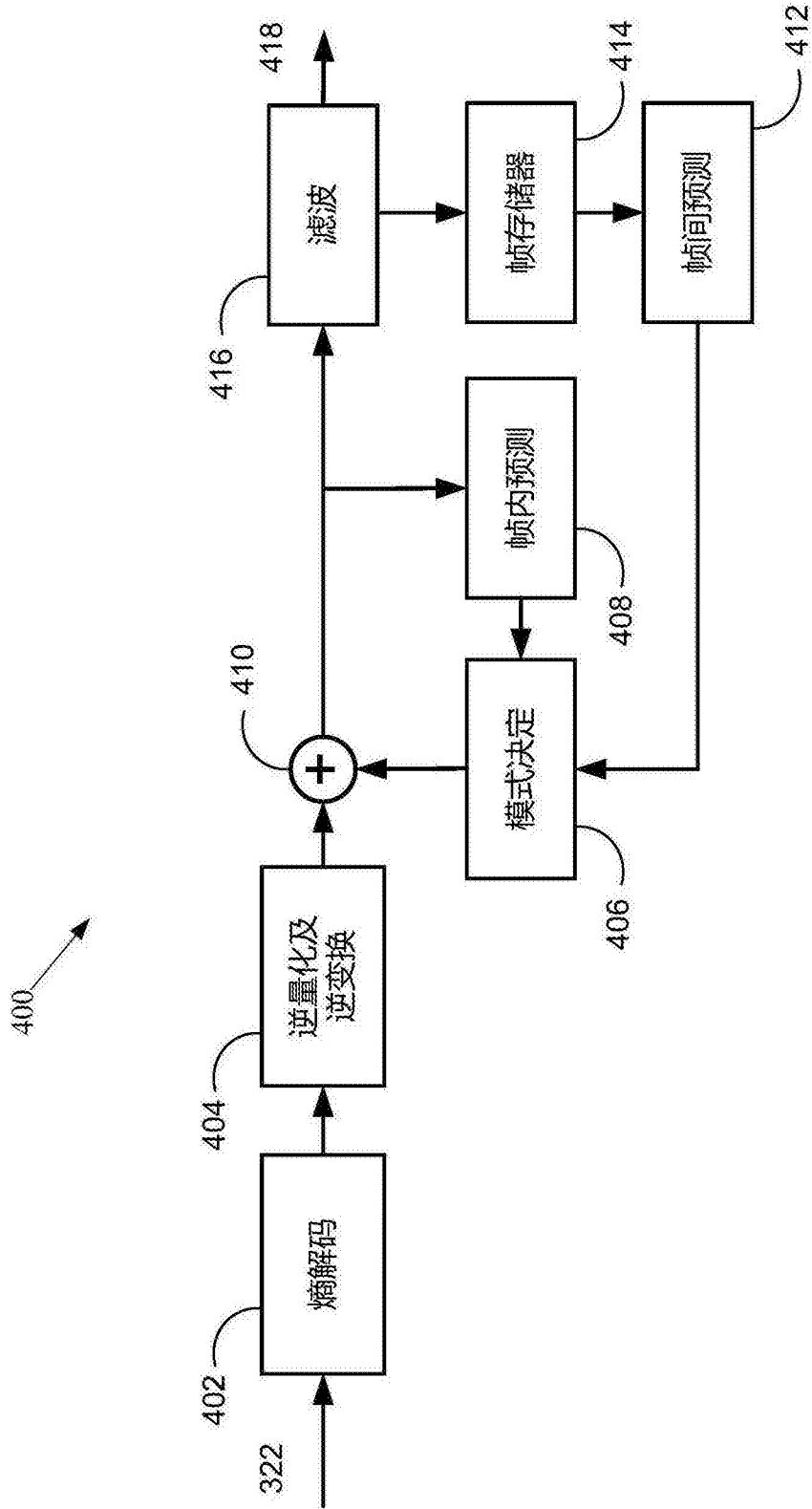


图4

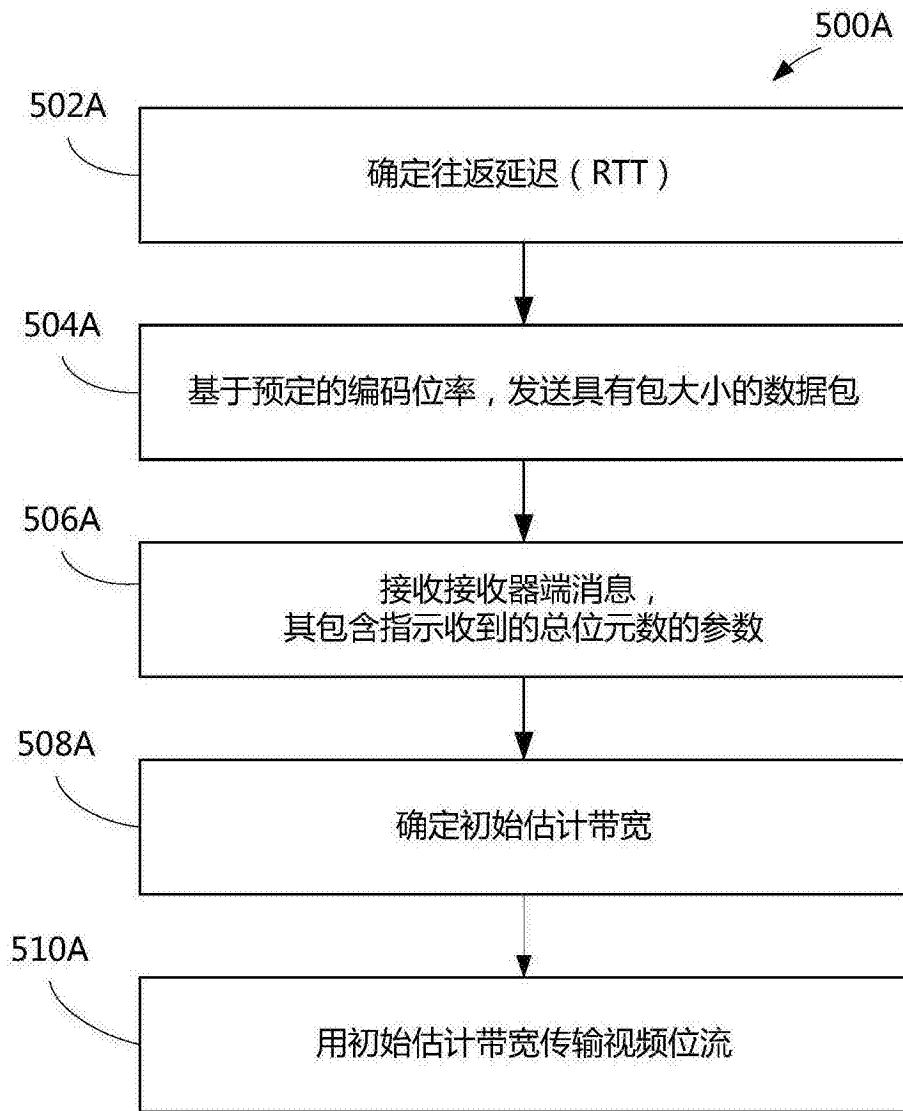


图5A

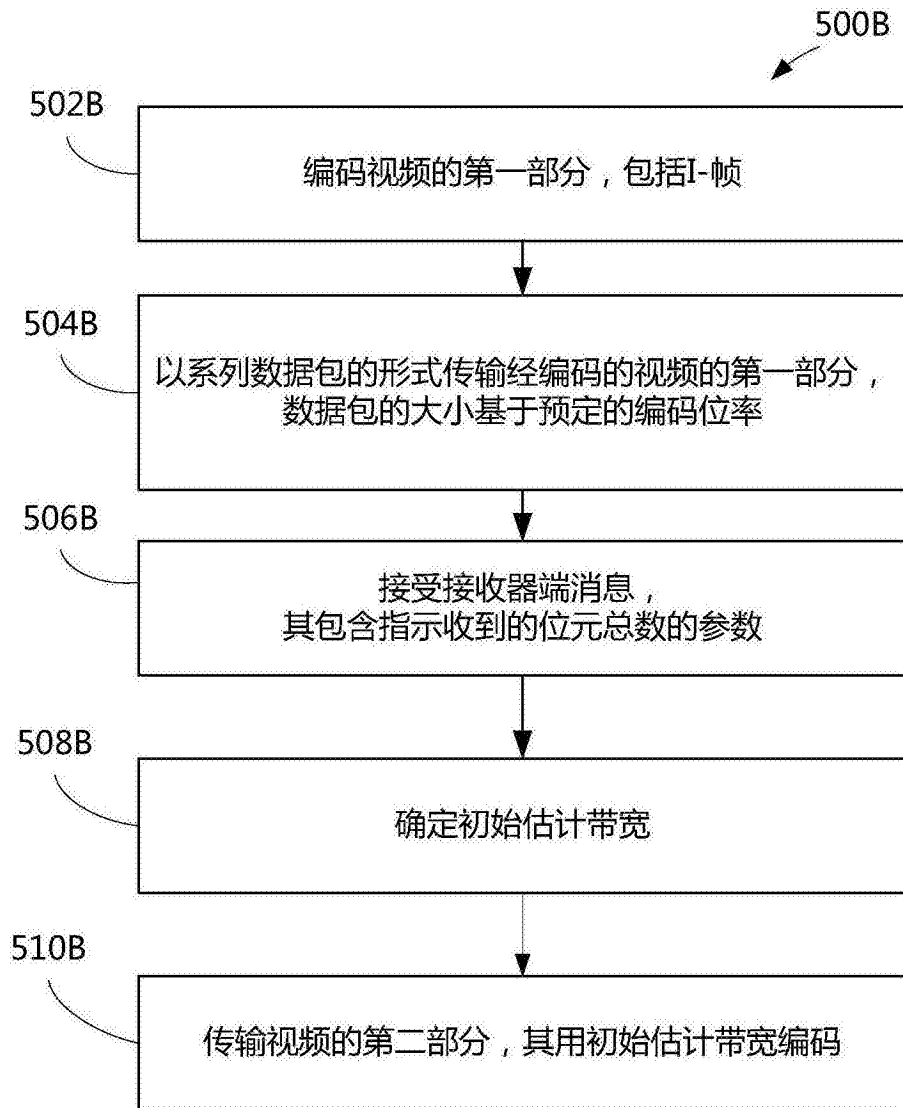


图5B

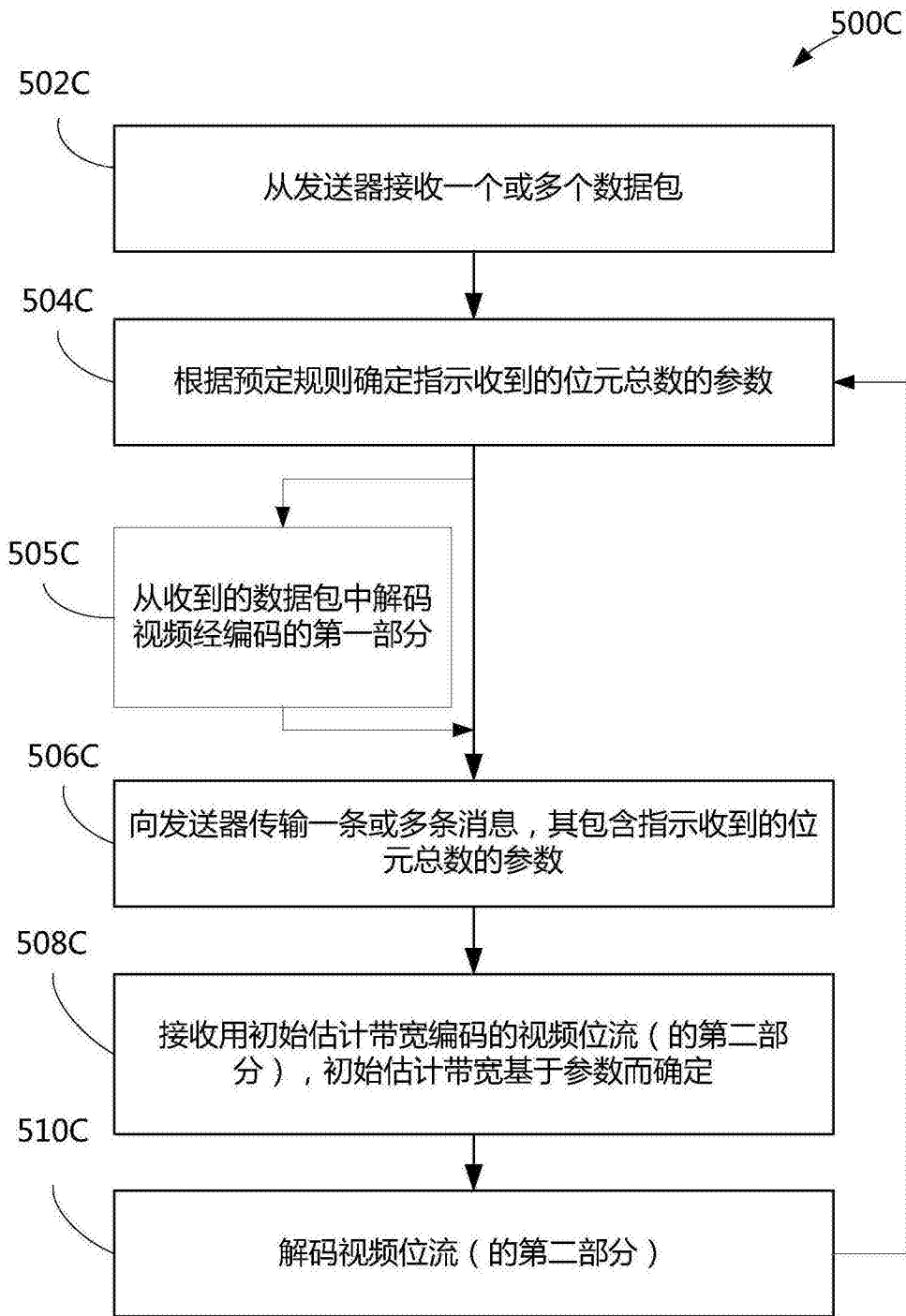


图5C

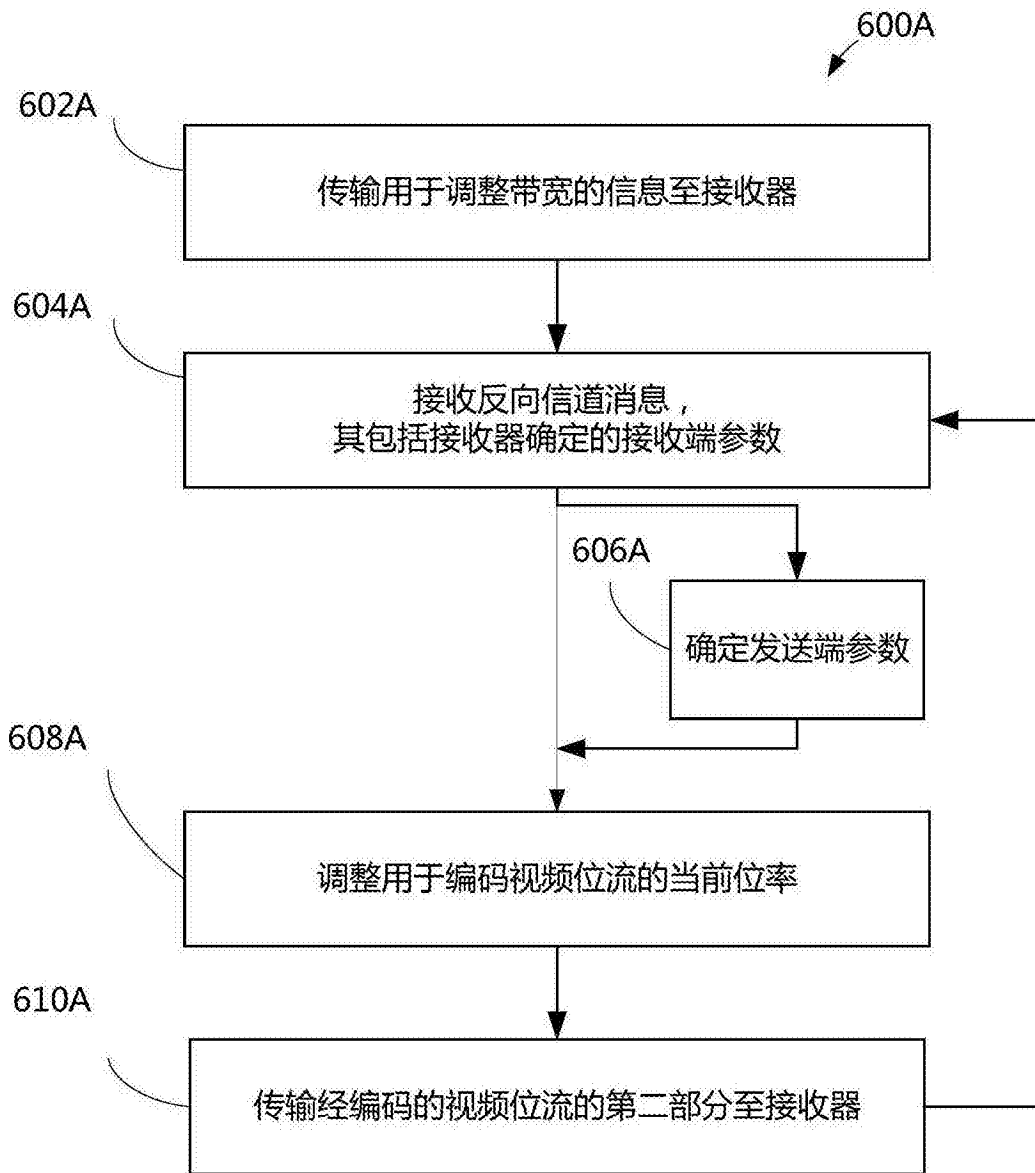


图6A

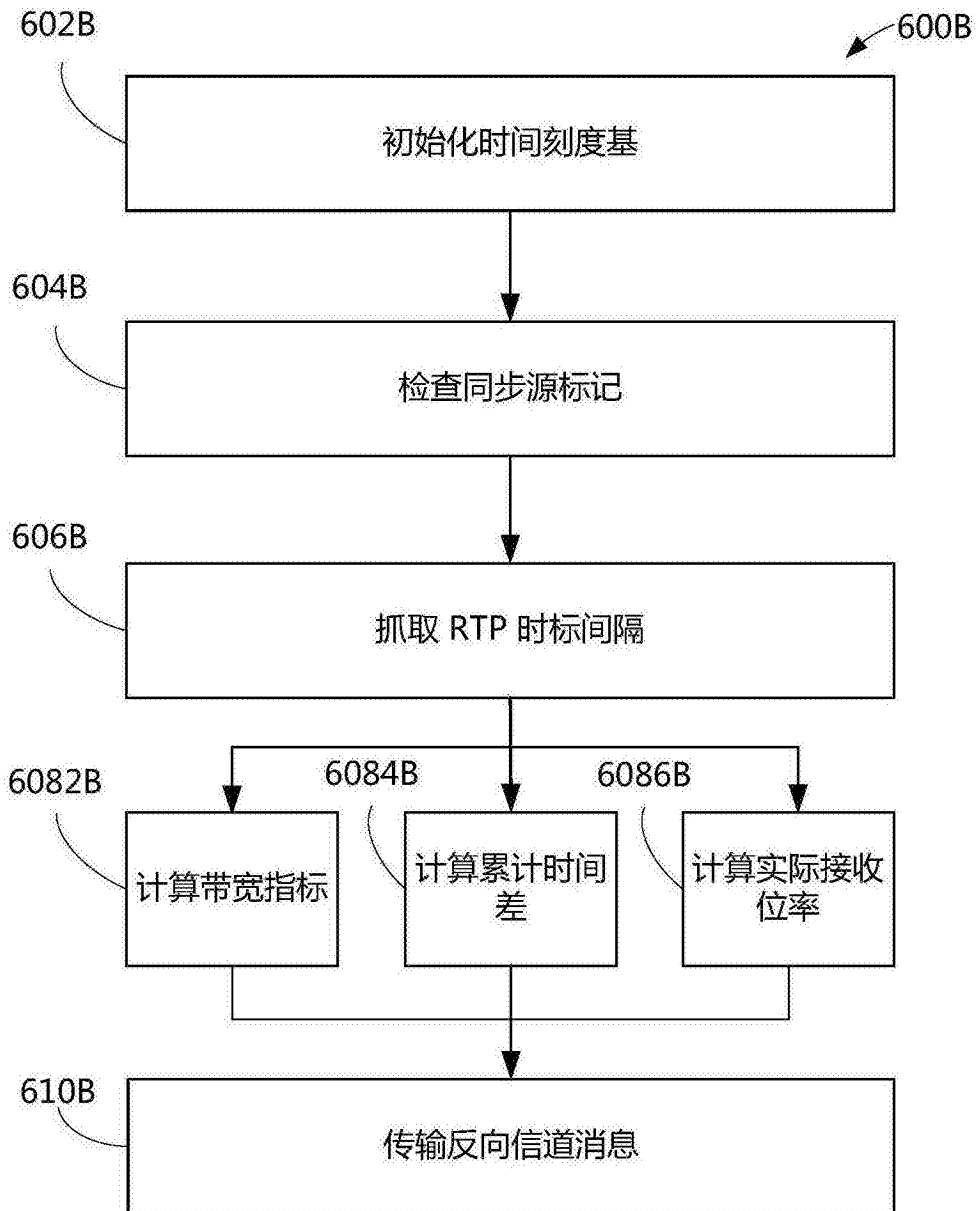


图6B

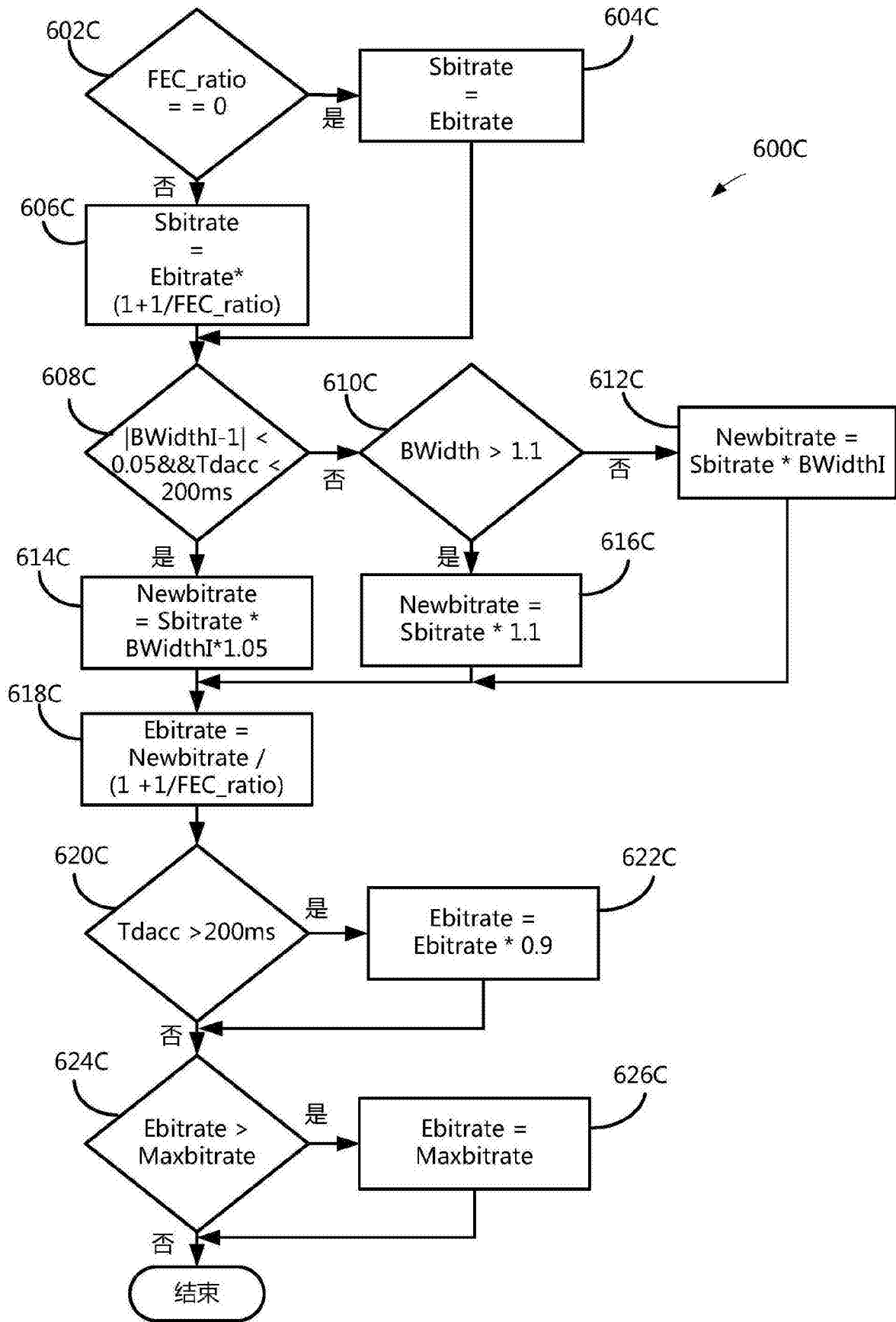


图6C

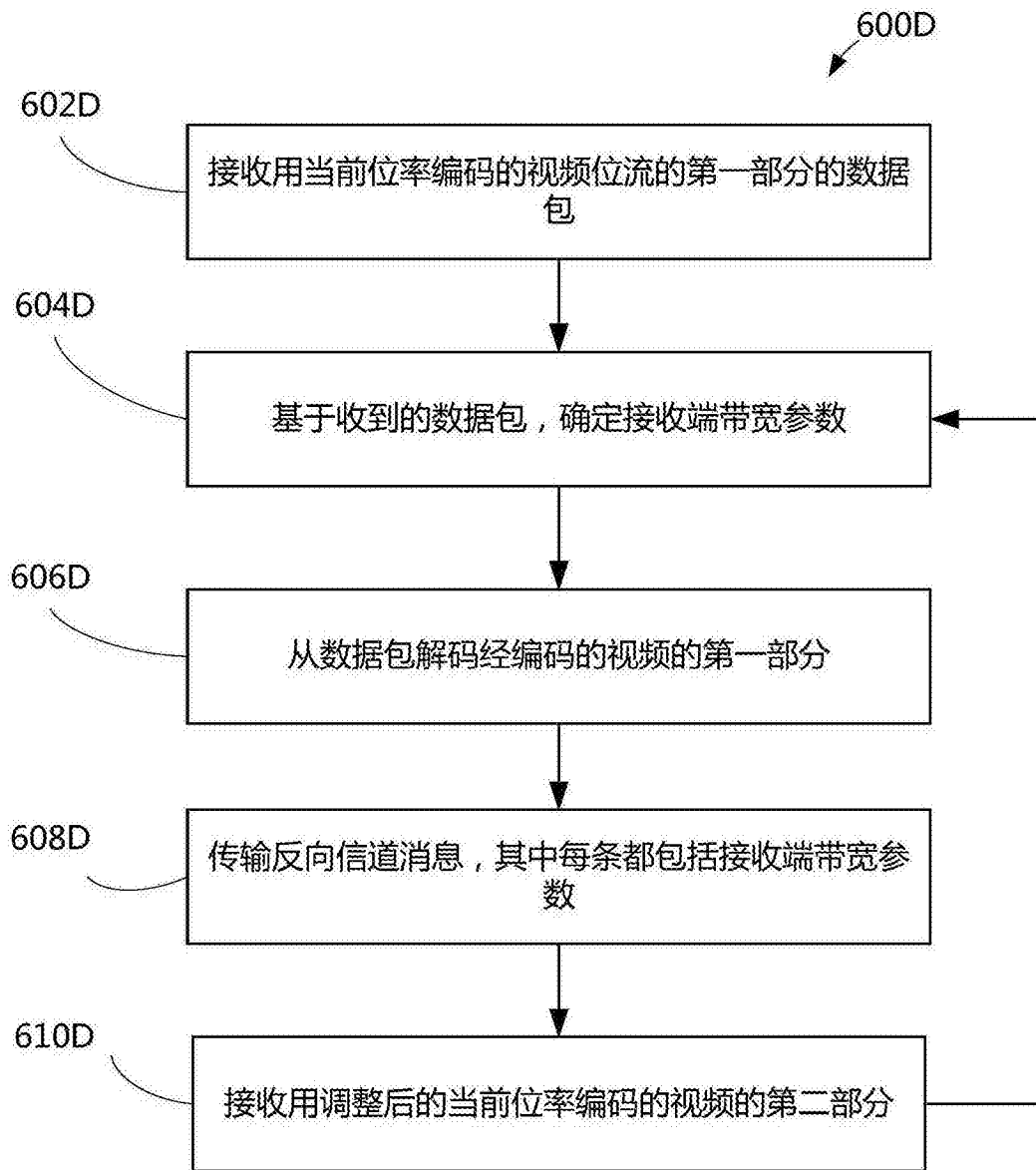


图6D

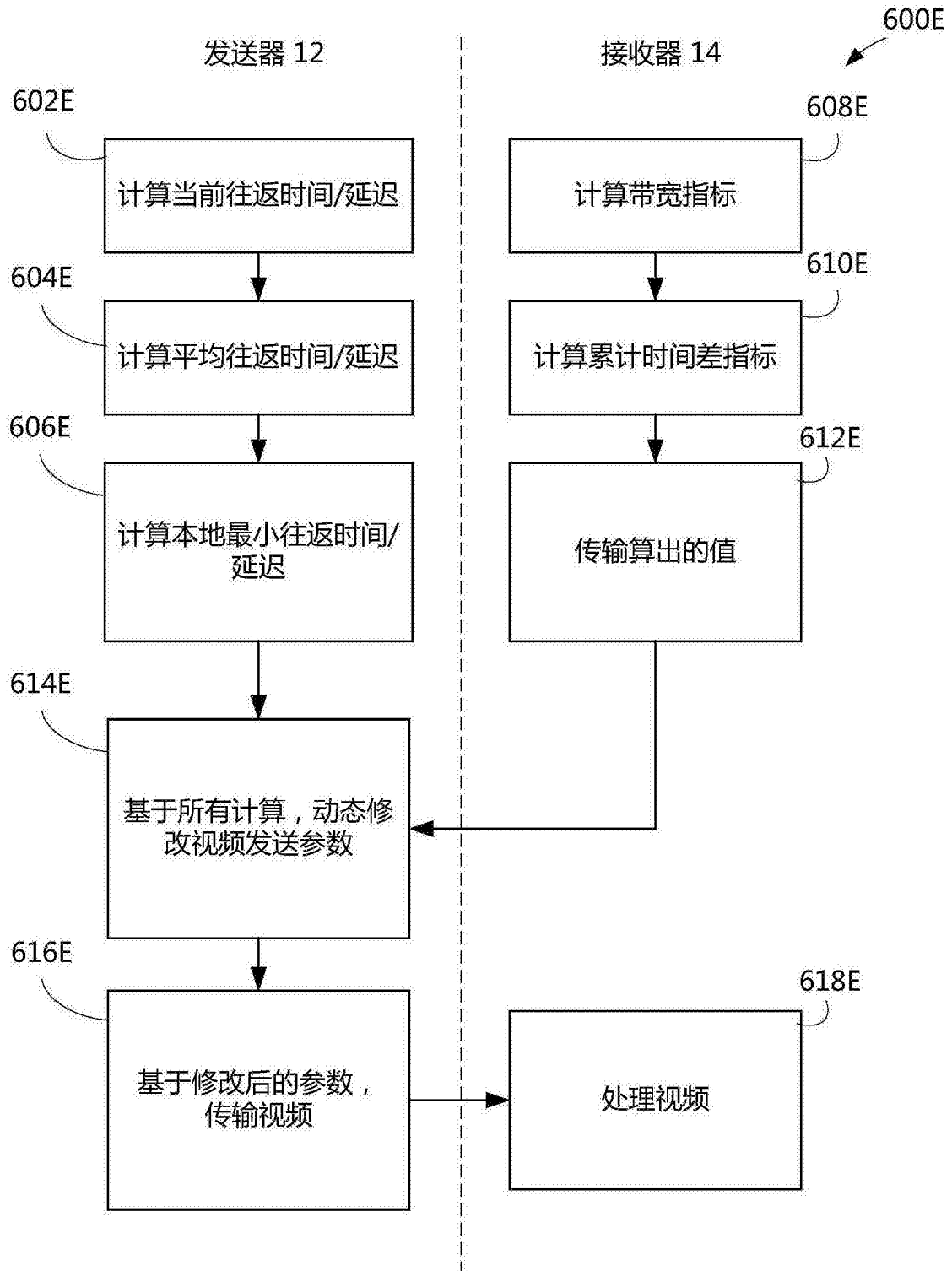


图6E

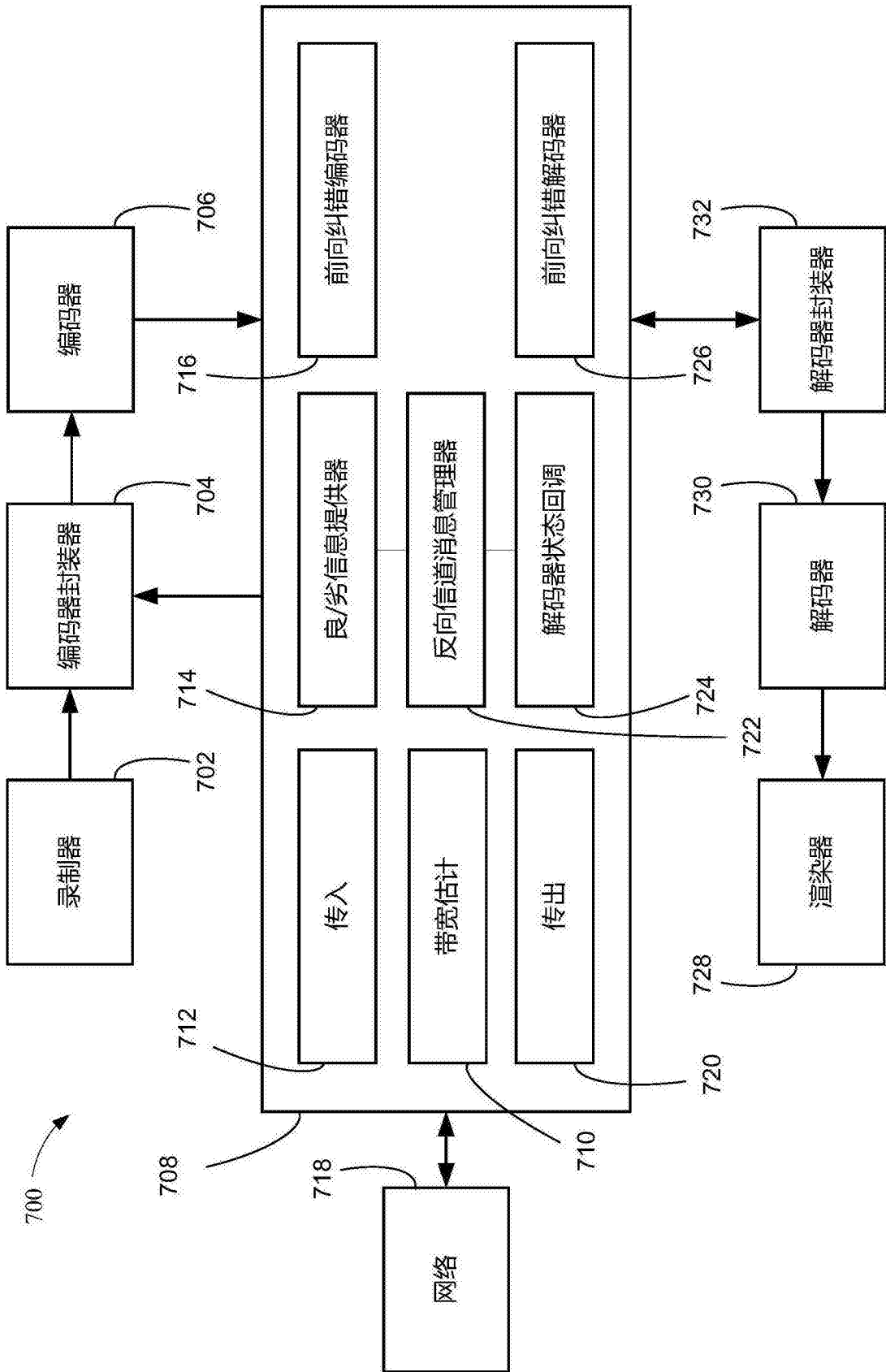


图7

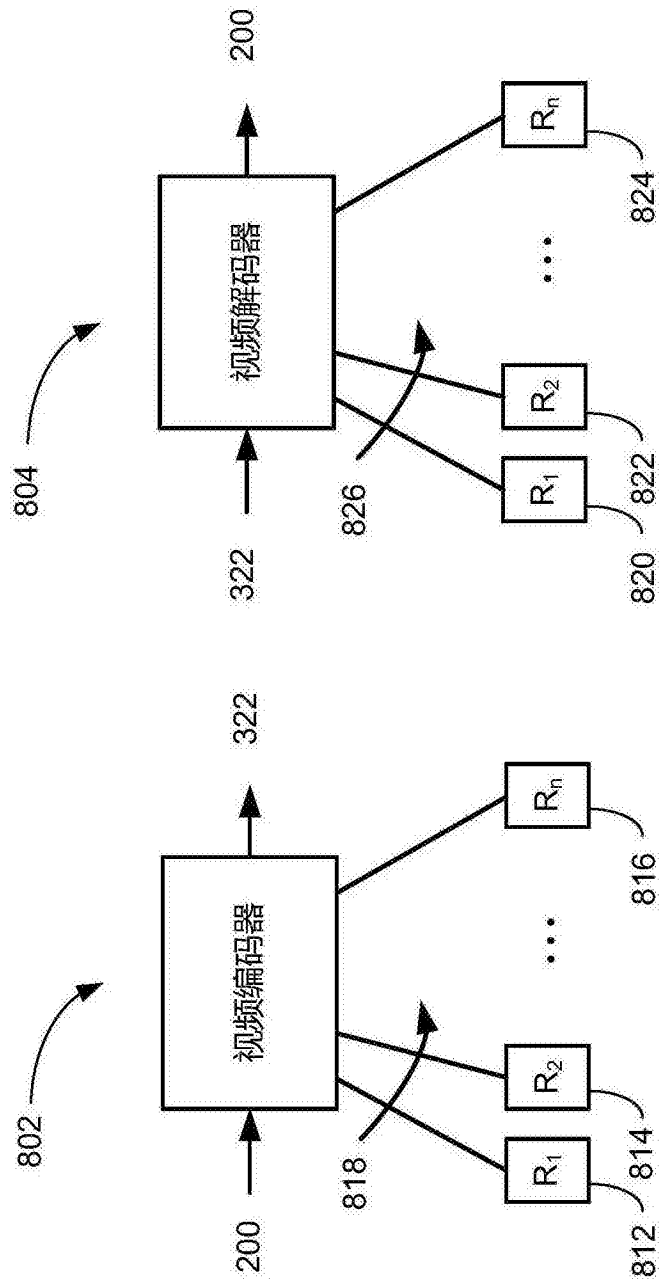


图8

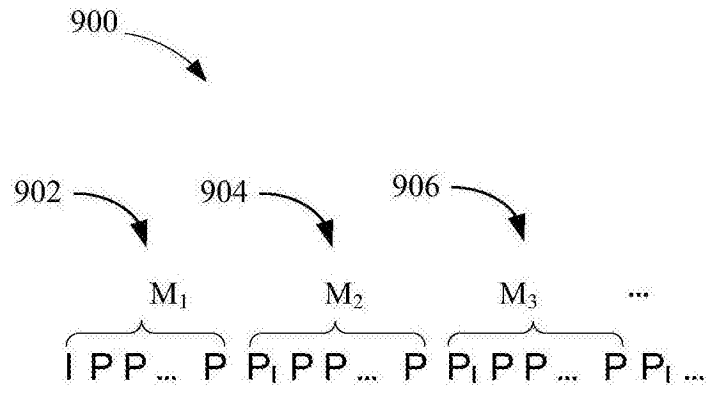


图9