



(12) 发明专利

(10) 授权公告号 CN 102404572 B

(45) 授权公告日 2013. 11. 27

(21) 申请号 201110372939. 7

(22) 申请日 2011. 11. 22

(73) 专利权人 西交利物浦大学

地址 215123 江苏省苏州市工业园区独墅湖
高等教育区仁爱路 111 号

(72) 发明人 林春雨 肖继民 罗天明

(74) 专利代理机构 苏州创元专利商标事务所有
限公司 32103

代理人 范晴

(51) Int. Cl.

H04N 7/26 (2006. 01)

H04N 7/14 (2006. 01)

审查员 张伟

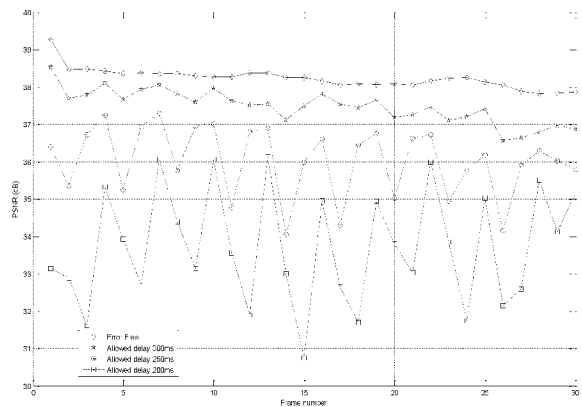
权利要求书2页 说明书8页 附图5页

(54) 发明名称

延迟约束条件下基于系统 RS 码的视频编解码系统及其方法

(57) 摘要

本发明公开了一种延迟约束条件下基于系统 RS 码的视频编解码系统, 编码端一方面编码并发送视频包数据, 同时将整个子图像组编码生成的视频包存入缓存区并作为系统 RS 码 (systematic Reed-Solomon code) 的信息位, 由于利用整个子图像组的视频包作为信息位, 提高了 RS 编码的效率; 解码端使用后端解码器以利用延迟到达的视频包; 解码端使用 RS 解码器以尽量恢复子图像组内丢失的视频包。



1. 一种延迟约束条件下基于系统 RS 码的视频编解码系统,包括处于网络传输信道两端的编码端和解码端,其特征在于所述编码端包括用于对图像组视频帧进行实时编码的主编码器和对图像组视频帧进行 RS (N, K) 编码的 RS 编码器,所述编码端将视频图像组划分成若干个子图像组,所述主编码器实时编码子图像组形成视频数据包放入传输信道传输,并将视频数据包存入视频缓冲区;所述 RS 编码器以子图像组的视频数据包为信息位生成子图像组的校验数据包,并放入传输信道传输;

所述解码端包括用于对图像组数据包进行实时解码和显示的主解码器、用于通过校验数据包进行 RS (Reed-Solomon) 解码恢复视频数据包的 RS 解码器和通过恢复视频数据包对图像组视频帧进行更新恢复的后端解码器,所述解码端接收到视频数据包,如果是延迟到来的数据包,先通过后端解码器进行解码然后更新前面的参考帧,否则先通过主解码器进行视频实时解码和显示视频帧;当主解码器不能解码时,根据校验数据包的情况调用 RS 解码器解码;当 RS 解码器恢复视频数据包,则启用后端解码器重新解码后更新视频参考帧;当 RS 解码器不能恢复视频数据包时,解码端通过错误掩盖处理;所述传输的带宽和丢包概率分别为 w 和 p ,每帧包含恒定的视频数据包 m 个,每个包的大小为恒定值 x ,那么理想情况下,帧率为 q fps 进行视频数据传输的同时具备的约束条件是:

$$N*(1-p) \geq S*m \quad (1);$$

$$N*x \leq w*S/q \quad (2);$$

其中, S 为子图像组的大小, $S*m$ 为所有视频包的数量,即 K 的值, N 为视频包和校验包的总长度。

2. 一种采用权利要求 1 所述的系统进行延迟约束条件下基于系统 RS 码的视频编解码方法,所述方法中编码端和解码端处于网络传输信道两端的,其特征在于所述方法包括以下步骤:

(1) 所述编码端将视频图像组划分成若干个子图像组,所述主编码器实时编码子图像组形成视频数据包放入传输信道传输,并将视频数据包存入视频缓冲区;所述 RS 编码器以子图像组的视频数据包为信息位进行 RS (N, K) 编码生成子图像组的校验数据包,并放入传输信道传输;

(2) 所述解码端接收到视频数据包,如果是延迟到来的视频数据包,先通过后端解码器进行解码,然后更新视频参考帧队列,否则先通过主解码器进行视频实时解码和显示视频帧;当主解码器不能解码时,根据校验数据包的情况调用 RS 解码器解码;当 RS 解码器恢复视频数据包,则启用后端解码器重新解码后更新视频参考帧;当 RS 解码器不能恢复视频数据包时,解码端通过错误掩盖处理;所述方法中所述传输的带宽和丢包概率分别为 w 和 p ,每帧包含恒定的视频数据包 m 个,每个包的大小为恒定值 x ,那么理想情况下,帧率为 q fps 进行视频数据传输需要同时具备的约束条件是:

$$N*(1-p) \geq S*m \quad (1);$$

$$N*x \leq w*S/q \quad (2);$$

其中, S 为子图像组的大小, $S*m$ 为所有视频包的数量,即 K 的值, N 为视频包和校验包的总长度。

3. 根据权利要求 2 所述的方法,其特征在于所述方法步骤(1)中编码端的具体按照如下步骤进行:

A1) 编码端先设置确定子图像组和 RS 码的长度,主编码器开始进行某一子图像组的编码;

A2) 主编码器编码按照子图像组的视频帧顺序进行,主编码器编码后将子图像组的视频帧数据包发送到传输信道上发送,并且将子图像组的视频帧数据包存入视频缓冲区,然后编码端根据子图像组的视频帧位置判断是否通过 RS 编码器进行产生校验码;

A3) 当子图像组的视频帧位置为该子图像组的视频结尾帧时,主编码器编码后将子图像组的视频帧数据包发送到传输信道上发送,并且将子图像组的视频帧数据包存入视频缓冲区,RS 编码器对整体的子图像组的视频帧数据包进行 RS (N,K) 编码产生校验码,发送到传输信道上发送;然后主编码器进行编码下一个子图像组,继续循环。

4. 根据权利要求 2 所述的方法,其特征在于所述方法步骤(2) 具体按照如下步骤进行:

B1) 当所述解码端接收到视频数据包后,如果是延迟到来的数据包,先要通过后端解码器进行解码然后更新前面的参考帧,否则主解码器尝试进行视频数据包解码;当主解码器能够进行视频实时解码时,主解码器输出显示视频帧并同时视频帧放入视频参考帧队列面;

B2) 当主解码器不能够进行视频实时解码时,解码端判断是否通过 RS 解码器 RS 解码已收到数据校验包恢复视频数据包;当 RS 解码器 RS 解码能够恢复视频数据包时,解码端通过后端解码器重新解码恢复后的视频数据包,并更新视频参考帧到恢复后的视频帧;

B3) 当在允许的延时阈值内未接收到视频数据包或丢失的数据包大于 N-K 时,解码端进行视频帧的错误掩盖处理。

5. 根据权利要求 2 所述的方法,其特征在于所述方法步骤(1) 中通过贪婪算法确定图像组中子图像组的数量和子图像组的分配位置。

6. 根据权利要求 5 所述的方法,其特征在于所述贪婪算法是根据校验包放在单幅影像画面帧的不同的位置,获得不同子图像组分配造成的误差期望;当形成最小误差期望时,所放置的单幅影像画面的位置为该校验包的位置;然后继续寻找下一个校验包的位置。

延迟约束条件下基于系统 RS 码的视频编解码系统及其方法

技术领域

[0001] 本发明属于视频网络传输中的编解码技术领域,具体涉及一种延迟约束条件下基于系统 RS 码的视频编解码系统及其方法。

背景技术

[0002] 在一些实时性要求较高的视频通信应用场景中,对视频延迟往往有一定的容忍范围,如视频会议的场景下,150 ~ 400ms 是启动延迟的容忍范围。当到某一视频包的传输超过了这一时间阈值,同时又轮到了这一视频包所在视频帧的显示时刻,那么该视频包将被认为是丢失的。另一方面,由于大量多媒体数据的传输,经常造成网络拥塞和网络丢包。而在 H. 264/AVC 和其它一些视频编码标准中,由于运动估计和运动补偿的使用,单个包的丢失往往会对随后多帧造成影响。

[0003] 对于第一个问题,常见的方法是加速编码器编码速度,同时减少传输过程中时间。加速编码器无疑给编码器的设计带来了较大负担,为了减少编码时间,编码性能经常会大大下降,如 <http://x264dev.multimedia.cx/archives/249> 中说明 X264 中为了进行并行加速同时减少编码时延,只能在帧内使用片级 (Slice) 并行,造成编码性能有较大损失。

[0004] 针对上面的第二个问题,目前常见的方法是使用一些容错编码技术。这些技术包括帧内宏块刷新 (Intra Macroblock Refreshment) [参见 R. Zhang, S. L. Regunathan, and K. Rose. Video coding with optimal inter/intra-mode switching for packet loss resilience. Selected Areas in Communications, IEEE Journal on, 18(6) : 966-976, June 2000. Yuan Zhang, Wen Gao, Yan Lu, Qingming Huang, and Debin Zhao. Joint source-channel rate-distortion optimization for h.264 video coding over error-prone networks. Multimedia, IEEE Transactions on, 9(3) :445-454, 2007.], 自动重传请求 (ARQ) [参见 S. Soltani, K. Misra, and H. Radha. Delay constraint error control protocol for real-time video communication. Multimedia, IEEE Transactions on, 11(4) :742-751, 2009.], 基于反馈的参考帧选择 (RPS) [参见 Shunan Lin, Shiwen Mao, Yao Wang, and S. Panwar. A reference picture selection scheme for video transmission over ad-hoc networks using multiple paths. In Multimedia and Expo, 2001. ICME 2001. IEEE International Conference on, pages 96-99, 2001.] 和多描述编码 (MDC) [参见 I. Radulovic, P. Frossard, Ye-Kui Wang, M. M. Hannuksela, and A. Hallapuro. Multiple description video coding with h.264/avc redundant pictures. Circuits and Systems for Video Technology, IEEE Transactions on, 20(1) : 144-148, 2010. 和 John G. Apostolopoulos Brian A. Heng and Jae S. Lim. End-to-end ratedistortion optimized md mode selection for multiple description video coding. EURASIP Journal on Applied Signal Processing, 2006.], 以及前向纠错 (FEC) 编码技术 [参见 E. Baccaglini, T. Tillo, and G. Olmo. Slice sorting for unequal loss

protection of video streams. Signal Processing Letters, IEEE, 2008. X. Yang, Ce Zhu, Zheng Guo Li, Xiao Lin, and Nam Ling. An unequal packet loss resilience scheme for video over the internet. Multimedia, IEEE Transactions on, 7(4) :753-765, 2005. N. Thomos, S. Argyropoulos, N. V. Boulgouris, and M. G. Strintzis. Robust transmission of h. 264/avc video using adaptive slice grouping and unequal error protection. In Multimedia and Expo, 2006 IEEE International Conference on, pages 593-596, 2006.]。在这些抗误码技术中,帧内宏块刷新和多描述编码不会产生额外延时,因此更适合实时视频通信。但是,就帧内宏块刷新技术而言,由于帧内编码效率远低于帧间编码的效率,导致整体编码效率显著降低。在多描述编码中,当冗余版本替代了原始版本或者当一些描述在传输过程中丢失时,就会造成失配误差。而失配误差将会传播至整个图像组 (GOP)。

[0005] 由于网络往返时延,自动重传请求和参考帧选择 (RPS) 技术都会产生较长延时,因此,这两种技术都无法应用在实时视频通信中。而在前向纠错码技术 (FEC) 中,时延的长短取决于 FEC 编码数据块的大小。如 Reed-Solomon (RS) 码中在 K 个原始数据包中引入 $N-K$ 个校验包来保护数据,其中的原始数据称为信息位,对应的 RS 码称为 (N, K) 码。就 RS 编码而言,对于相同的编码率 K/N ,增加 K 的值会提高 RS 编码的性能。但是, K 值过大意味着需要较大块的 RS 编码块。对于实时视频编码,需要等待采集多帧之后进行编码生成视频包,因此导致了较长的时延。Reed-Solomon (RS) 编码块包含了整个图像组,并且产生了一个图像组的延时。对于视频会议等实时应用场景,这样的时延是不可接受的。本发明因此而来。

发明内容

[0006] 本发明目的在于提供一种延迟约束条件下基于系统 RS 码的视频编解码系统,解决了现有技术中视频编解码传输时较长延时,无法应用在实时视频通信中等问题。

[0007] 为了解决现有技术中的这些问题,本发明提供的技术方案是:

[0008] 一种延迟约束条件下基于系统 RS 码的视频编解码系统,包括处于网络传输信道两端的编码端和解码端,其特征在于所述编码端包括用于对图像组视频帧进行实时编码的主编码器和对图像组视频帧进行 RS (N, K) 编码的 RS 编码器,所述编码端将视频图像组划分成若干个子图像组,所述主编码器实时编码子图像组形成视频数据包放入传输信道传输,并将视频数据包存入视频缓冲区;所述 RS 编码器以子图像组的视频数据包为信息位生成子图像组的校验数据包,并放入传输信道传输;

[0009] 所述解码端包括用于对图像组数据包进行实时解码和显示的主解码器、用于通过校验数据包进行 RS (Reed-Solomon) 解码恢复视频数据包的 RS 解码器和通过恢复视频数据包对图像组视频帧进行更新恢复的后端解码器,所述解码端接收到视频数据包,如果是延迟到来的数据包,先要通过后端解码器进行解码然后更新前面的参考帧,否则先通过主解码器进行视频实时解码和显示视频帧;当主解码器不能解码时,根据校验数据包的情况调用 RS 解码器解码;当 RS 解码器恢复视频数据包,则启用后端解码器重新解码后更新视频参考帧;当 RS 解码器不能恢复视频数据包时,解码端通过错误掩盖处理。

[0010] 优选的,所述传输的带宽和丢包概率分别为 w 和 p ,每帧包含恒定的视频数据包 m 个,每个包的大小为恒定值 x ,那么理想情况下,帧率为 q fps 进行视频数据传输的同时具备的约束条件是:

$$[0011] \quad N*(1-p) \geq S*m \quad (1);$$

$$[0012] \quad N*x \leq w*S/q \quad (2);$$

[0013] 其中, S 为子图像组的大小, S*m 为所有视频包的数量,即 K 的值, N 为视频包和校验包的总长度。

[0014] 本发明的另一目的在于提供一种采用所述的系统进行延迟约束条件下基于系统 RS 码的视频编解码方法,所述方法中编码端和解码端处于网络传输信道两端的,其特征在于所述方法包括以下步骤:

[0015] (1) 所述编码端将视频图像组划分成若干个子图像组,所述主编码器实时编码子图像组形成视频数据包放入传输信道传输,并将视频数据包存入视频缓冲区;所述 RS 编码器以子图像组的视频数据包为信息位进行 RS(N, K) 编码生成子图像组的校验数据包,并放入传输信道传输;

[0016] (2) 所述解码端接收到视频数据包,如果是延迟到来的数据包,先要通过后端解码器进行解码然后更新前面的参考帧,否则先通过主解码器进行视频实时解码和显示视频帧;当主解码器不能解码时,根据校验数据包的情况调用 RS 解码器解码;当 RS 解码器恢复视频数据包,则启用后端解码器重新解码后更新视频参考帧;当 RS 解码器不能恢复视频数据包时,解码端通过错误掩盖处理。

[0017] 优选的,所述方法中所述传输的带宽和丢包概率分别为 w 和 p,每帧包含恒定的视频数据包 m 个,每个包的大小为恒定值 x,那么理想情况下,帧率为 q fps 进行视频数据传输需要同时具备的约束条件是:

$$[0018] \quad N*(1-p) \geq S*m \quad (1);$$

$$[0019] \quad N*x \leq w*S/q \quad (2);$$

[0020] 其中, S 为子图像组的大小, S*m 为所有视频包的数量,即 K 的值, N 为视频包和校验包的总长度。

[0021] 优选的,所述方法步骤 (1) 中编码端的具体按照如下步骤进行:

[0022] A1) 编码端先设置确定子图像组和 RS 码的长度,主编码器开始进行某一子图像组的编码;

[0023] A2) 主编码器编码按照子图像组的视频帧顺序进行,主编码器编码后将子图像组的视频帧数据包发送到传输信道上发送,并且将子图像组的视频帧数据包存入视频缓冲区,然后编码端根据子图像组的视频帧位置判断是否通过 RS 编码器进行产生校验码;

[0024] A3) 当子图像组的视频帧位置为该子图像组的视频结尾帧时,主编码器编码后将子图像组的视频帧数据包发送到传输信道上发送,并且将子图像组的视频帧数据包存入视频缓冲区,RS 编码器对整体的子图像组的视频帧数据包进行 RS(N, K) 编码产生校验码,发送到传输信道上发送;然后主编码器进行编码下一个子图像组,继续循环。

[0025] 优选的,所述方法步骤 (2) 具体按照如下步骤进行:

[0026] B1) 当所述解码端接收到视频数据包后,如果是延迟到来的数据包,先要通过后端解码器进行解码然后更新前面的参考帧,否则主解码器尝试进行视频数据包解码;当主解码器能够进行视频实时解码时,主解码器输出显示视频帧并同时 will 视频帧放入视频参考帧队列面;

[0027] B2) 当主解码器不能够进行视频实时解码时,解码端判断是否通过 RS 解码器 RS 解

码已收到数据校验包恢复视频数据包;当 RS 解码器 RS 解码能够恢复视频数据包时,解码端通过后端解码器重新解码恢复后的视频数据包,并更新视频参考帧到恢复后的视频帧;

[0028] B4) 当在允许的延时阈值内未接收到视频数据包或丢失的数据包大于 $N-K$ 时,解码端进行视频帧的错误掩盖处理。

[0029] 优选的,所述方法步骤 (1) 中通过贪婪算法确定图像组中子图像组的数量和子图像组的分配位置。

[0030] 优选的,所述贪婪算法是根据校验包放在单幅影像画面帧的不同的位置,获得不同子图像组分配造成的误差期望;当形成最小误差期望时,所放置的单幅影像画面的位置为该校验包的位置;然后继续寻找下一个校验包的位置。

[0031] 优选的,所述方法步骤 (1) 中进行视频数据传输还满足以下约束条件:

$$[0032] \quad S * m \leq N \quad (3);$$

[0033] 其中, S 为子图像组的大小, m 为每帧包含恒定的视频数据包, $S * m$ 为所有视频包的数量,即 K 的值, N 为视频包和校验包的总长度。

[0034] 正常情况下主编码器在视频数据包解码成 1 个帧后会直接显示,并同时被放在参考帧队列里面供后面编码使用。主编码器在解码的同时放入参考帧队列。

[0035] 本发明提出了延迟约束条件下基于系统 RS 码的视频编解码系统,该系统具有以下特点:第一,编码端一方面编码并发送视频包数据,同时将整个子图像组编码生成的视频包存入缓存区并作为系统 RS 码(systematic Reed-Solomon code)的信息位,由于利用整个子图像组的视频包作为信息位,提高了 RS 编码的效率;第二,解码端使用后端解码器以利用延迟到达的视频包;第三,解码端使用 RS 解码器以尽量恢复子图像组内丢失的视频包。

[0036] 整个解码系统主要有三个优点,并且主要体现在后端解码器和 RS 解码上。

[0037] 第一,延迟到达的数据包并不完全丢掉,后端解码器将在后端对延迟到达的数据包和与该包相关的所有以前到达的视频包做更新解码操作,同时更新主解码器的视频参考帧,这样随后到达的视频包就有了正确的参考帧,由于延迟包造成的误差扩散被截断。

[0038] 第二,每多收到一个数据包,RS 解码器都尝试使用 RS 解码器解码,如果到达的总数据包数量大于 K 个,RS 解码器可以还原子图像组中的所有视频包,随后,后端解码器将会利用接收到的视频包和还原重建的视频包对这个子图像组进行重新解码并更新主解码器参考帧。

[0039] 第三,由于目前的 Internet 架构无法保证先发送的视频包在时间上一定优先到达,因此后面发送的视频包经常有先到的可能,在未到达当前帧的显示时刻前,如果整个子图像组的视频数据包超过了 K 个,那么尽管此时可能当前帧的视频包还没有完全接收到,仍然可以使用 RS 解码恢复整个子图像组的所有视频包,这样就实现了在当前帧视频包没有完全接收到的条件下,却可以显示该视频帧的目的。

附图说明

[0040] 下面结合附图及实施例对本发明作进一步描述:

[0041] 图 1 为本发明实施例编码端的架构示意图;

[0042] 图 2 为本发明实施例编码端的具体工作流程示意图;

- [0043] 图 3 为本发明实施例解码端的架构示意图；
- [0044] 图 4 为本发明实施例解码端的具体工作流程示意图；
- [0045] 图 5 为本发明实施例解码端数据包到达显示时序图；
- [0046] 图 6 为不同延迟允许条件下的 PSNR 实验结果；
- [0047] 图 7 为不同子图像组大小条件下的 PSNR 实验结果。

具体实施方式

[0048] 以下结合具体实施例对上述方案做进一步说明。应理解,这些实施例是用于说明本发明而并不限于限制本发明的范围。实施例中采用的实施条件可以根据具体厂家的条件做进一步调整,未注明的实施条件通常为常规实验中的条件。

[0049] 实施例延迟约束条件下基于系统 RS 码的视频编解码系统

[0050] 本实施例采用的延迟约束条件下基于系统 RS 码的视频编解码系统,包括处于网络传输信道两端的编码端和解码端,所述编码端包括用于对图像组视频帧进行实时编码的主编码器和对图像组视频帧进行 RS(N,K) 编码的 RS 编码器,所述编码端将视频图像组划分成若干个子图像组,所述主编码器实时编码子图像组形成视频数据包放入传输信道传输,并将视频数据包存入视频缓冲区;所述 RS 编码器以子图像组的视频数据包为信息位生成子图像组的校验数据包,并放入传输信道传输;所述解码端包括用于对图像组数据包进行实时解码和显示的主解码器、用于通过校验数据包进行 RS(Reed-Solomon) 解码恢复视频数据包的 RS 解码器和通过恢复视频数据包对图像组视频帧进行更新恢复的后端解码器,所述解码端接收视频数据包,如果是延迟到来的数据包,先要通过后端解码器进行解码然后更新前面的参考帧,否则先通过主解码器进行视频实时解码和显示视频帧;当主解码器不能解码时,根据校验数据包的情况调用 RS 解码器解码;当 RS 解码器恢复视频数据包,则启用后端解码器重新解码后更新视频参考帧;当 RS 解码器不能恢复视频数据包时,否则解码端通过错误掩盖处理。

[0051] 下面对编码端和解码端分别介绍如下:

[0052] 图 1 所示为编码端示意图,首先对采集到的视频帧进行实时编码。在传统编码器中,采集的视频进行编码然后将生成的视频包送入信道传输。在此基础上,本发明主编码器同时将编码生成的视频包保存在视频缓存区中,缓存区的作用相当于备份了信道中传输的视频包,以供 RS 编码的信息位使用。当主编码器采集的视频帧达到子图像组的数量时(本实施例中子图像组与图像组概念相近,但是所包含的视频帧数量不超过其对应的图像组,通常一个图像组包含多个子图像组,具体数量和子图像组的分组位置通过贪婪算法确定),将视频缓存区中备份的整个子图像组的视频包作为信息位进行 RS(N,K) 编码。由于这一编码技术在编码端使用系统 RS 码保护视频包,因此原始的视频包数据不会受到改变,而这些视频包(K)之前就已经无延时地发送到了接收端,因此只需要将校验包(N-K)再发送到接收端即可。这里使用了子图像组作为 RS 编码的信息位,相当于增加了 K 的长度,进而提升了 RS 编码的效率。此处将视频包和校验包统称为数据包。

[0053] 编码端的具体流程如图 2 所示,其中子图像组的大小和 RS 编码的长度可根据网络情况设定。假设信道的带宽和丢包概率分别为 w 和 p ,再假设每帧包含恒定的数据包 m 个,每个包的大小为恒定值 x ,那么理想情况下,帧率为 25fp/s 的视频传输应用场景中,需要保

证如下条件：

$$[0054] \quad N*(1-p) \geq S*m \quad (1);$$

$$[0055] \quad N*x \leq w*S/25 \quad (2);$$

$$[0056] \quad S*m \leq N \quad (3);$$

[0057] 其中, S 为子图像组的大小, S*m 为所有视频包的数量 (K), N 为视频包和校验包的总和, 即 RS 编码需要考虑的长度。

[0058] 对于 RS(N, K) 编码, 当传输 N 个数据包时, 只要接收到的数据包大于 K 个, RS 码就可以恢复丢失的数据包。约束条件 (1) 表示经过 RS 编码后, 以丢包率 p 传输 N 个数据包时, 接收到的数据包 $N*(1-p)$ 需要大于 S*m。约束条件 (2) 表示整个 RS 编码后包括视频包和校验包的总和 ($N*x$) 要小于信道的带宽, 其中 S/25 表示子图像组所占的时间长度, 即 N 个数据包需要在 S/25 这段时间内传送出去, $w*S/25$ 为这段时间所占用的带宽。约束条件 (3) 要求视频包的数量要小于视频包和校验包的总和 ($K \leq N$), 否则这个子图像组的大小需要增大。实际应用中, 信道的带宽和丢包率可以通过数据包的发送情况估计出来, 即使网络情况不稳定, 也可以估计出丢包率的上限值作为约束条件。

[0059] 编码端可以按照如下伪代码的方式实现：

[0060]

Set the sub GOP length as K, Set the length of RS as N

For all the frames in one GOP

For i=1 to K

Encode frame i and send the coded packets to the receiver and buffer as well

End for

Using the packets in the buffer as information to do the RS coding and send the checking packet to the receiver

End for

[0061] 图 3 所示为解码端示意图, 其中, 主解码器的作用与传统的解码器一致, 即完成实时解码和显示的功能。具体而言, 主解码器在到达每一帧的显示时刻将该帧解码并显示, 此时只需要与此帧相关的数据包。如果这一帧中的某些包在传输过程中丢失或超过允许的延时阈值, 尝试使用当前子图像组的 RS 解码恢复丢失或者超时的数据包, 如果恢复失败, 将使用误差掩盖技术来恢复丢失包。这样, 视频帧在解码端的显示不会产生额外时延。这里的误差掩盖技术常用的如 H. 264 里面正常的错误隐藏技术。

[0062] 解码端的具体流程图如图 4 所示, 其中 T 为启动延时, 即接收端等待 T 后开始显示第一帧数据; 更新标志 update_flag 设置为 1 时表示需要后端解码器或 RS 解码器重新解码并对参考帧进行更新操作。

[0063] 解码端可以按照如下伪代码的方式实现：

[0064]

```

Initialize the parameter T and update_flag=0
Receiving the packets from the channel
  If (Display time of frame i)Decode and reconstruct it with the received packets (the lost part will be
  reconstructed with error concealment technology). Store the packets in the buffer
Endif
If (Previous sub-GOP required update)
  Re-decode the previous sub-GOP,  update the reference frame and set update_flag=1
Else
  If(the number of received packet for the current sub-GOP $\geq$ K)
    Use RS decoding to recover lost packets
    If(lose existed in the previous frames before current sub-GOP or update_flag==1)
      Re-decode the previous frames
    Else
      Use all the received packets and corrected packets to decode current frame
  Else
    If(received packets for the previous frames before current sub-GOP or update_flag==1)
      Re-decode the previous frames with the newly received packets and update the reference frames
    Else
      Use all the received packets and corrected packets to decode current frame. Or use error
      concealment if there are still lost packets

  Finish decoding of the current frame.

```

[0065] 以下结合三种具体情况对上述方案做进一步说明,图5所示为解码端数据包到达显示时序示意图。

[0066] 第一种情况为延迟包发生的情况。其中红色包为延迟发生情况下的示意包,图中子图像组中的第一帧的第三个视频包在其显示时间 t_1 没有到达,此时解码端只能利用错误隐藏恢复该视频包,同时将重建的第一帧显示给用户。在第二帧的显示时刻 t_2 ,第一帧的第三个视频包延迟到达了,此时后端解码器将首先使用该视频包,同时结合之前接收到的第一帧的其它视频包进行解码恢复出第一帧,然后利用该第一帧更新主解码器参考帧序列中已经存储的第一帧数据,作为第二帧的参考。在这种情况下,延迟包带来的影响在第二帧之后即被截断了。

[0067] 第二种情况为发生丢包的情况。仍以第一个子图像组为例,其中采用的RS码为(15,12)。在 t_3 时刻,如果整个子图像组的视频包出现了丢包现象,那么只要丢包的数量少于3个,即只要接收到的视频包数量(包括延迟到达的视频包)大于等于12个,那么可以利用RS解码恢复丢失的视频包,这样整个子图像组中由于丢包造成的误差影响在该子图像组内得以截断。

[0068] 第三种情况为视频包随机到达的情况。以第一个子图像组为例,在第一帧的显示时刻 t_1 ,如果第一帧的某些视频包发生了错误或延迟,此时只能显示经过错误隐藏的视频数据给用户。但是如果此时第二帧和第三帧以及一些校验包已经到达了,且所有到达的包的总数大于等于 K (此处 $K = 12$),那么可以在后端做RS解码并将正确的第一帧显示给用户,这样没有给后面帧造成任何误差扩散。

[0069] 可见这种解码机制的基本原理是对延迟到来的视频包加以利用,同时提高RS编码中信息位 K 的大小,进而提高RS编码效率。值得指出的是,延迟包的利用又会进一步提

高 RS 编码性能,如在一个子图像组最后一帧的显示时刻,第一帧或第二帧中的延迟包将以很大的概率接收到,此时相当于增加了接收到的数据包的数量。

[0070] 本实施例采用 H. 264 JM 软件来生成视频流,在生成视频包的时候采用了固定包长度的方式,考虑到了无线网络的最大传输单元限制,将目标视频包大小设成了 400 字节,每个分片 (Slice) 放在一个网络包中传送。在 H. 264 预测编码过程中只使用一个编码参考帧。H. 264 图像组 (GOP) 的结构为 IPPP...,即在一个图像组中除了第一帧是 I 帧外,其它的帧都为 P 帧。I 帧是帧组 GOP 的基础帧 (第一帧),在一组中只有一个 I 帧。I 帧不需要参考其他画面而生成,I 帧描述了图像背景和运动主体的详情,I 帧是 P 帧的基础参考帧 (其质量直接影响到同组中以后各帧的质量)。P 帧是以 I 帧或前面的其它 P 帧为参考帧,在前面的 I 帧或 P 帧中找出当前 P 帧“某点”的预测值和运动矢量,取预测差值和运动矢量一起传送。在接收端根据运动矢量从 I 帧 I 帧或前面的其它 P 帧中找出 P 帧“某点”的预测值并与差值相加以得到 P 帧“某点”样值,从而可得到完整的 P 帧。P 帧可以是其后面 P 帧的参考帧,P 帧可以使用 I 帧做预测,也可以使用前一个 P 帧做预测。测试所用的视频序列为标准的 CIF 格式 Foreman 序列。

[0071] 图 6 给出了在丢包率为 10% 条件下,不同的延迟允许条件下的实验结果,其中子图像组的大小设为 3,所有延迟到达的数据包,后端解码器都会对其进行重新解码并更新参考帧。图 7 给出了丢包率为 10% 条件下,不同子图像组大小对 PSNR 的影响。两个测试结果中,RS 编码的冗余率均为 40%,即 $(N-K)/K = 0.4$ 。

[0072] 上述实例只为说明本发明的技术构思及特点,其目的在于让熟悉此项技术的人是能够了解本发明的内容并据以实施,并不能以此限制本发明的保护范围。凡根据本发明精神实质所做的等效变换或修饰,都应涵盖在本发明的保护范围之内。

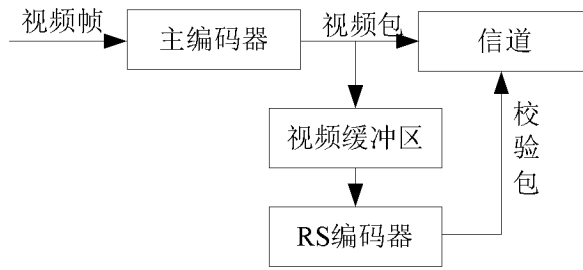


图 1

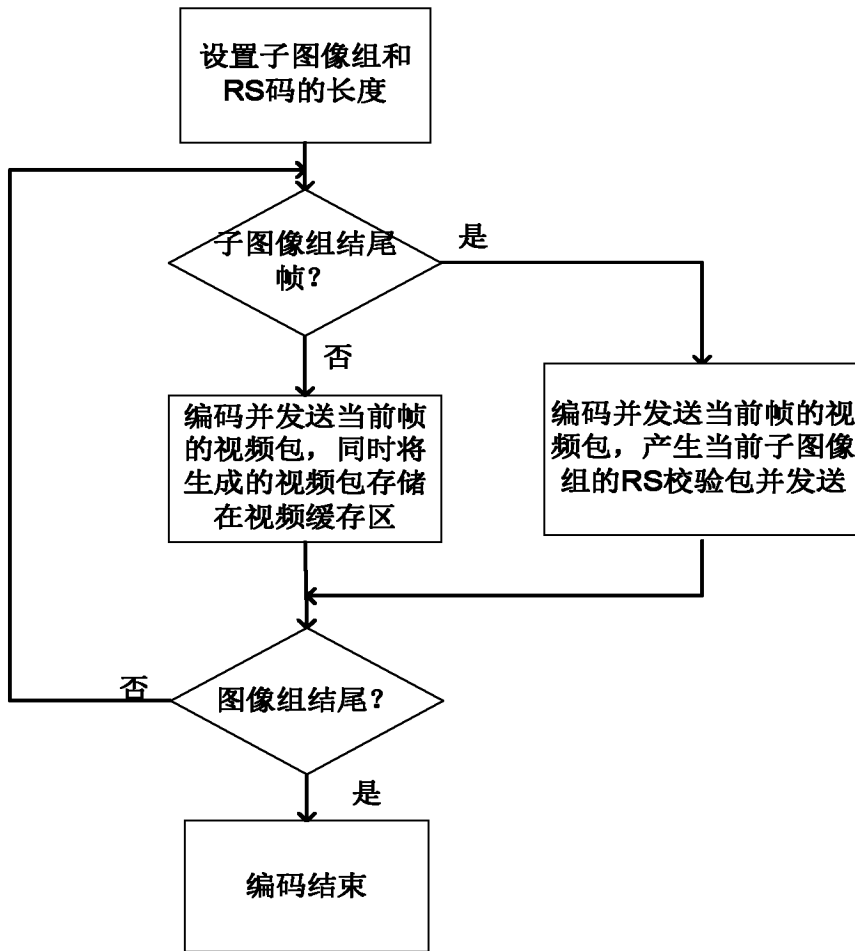


图 2

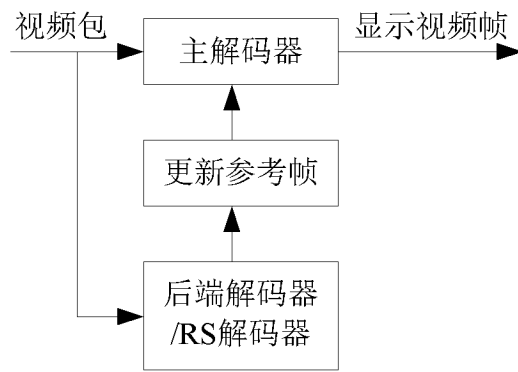


图 3

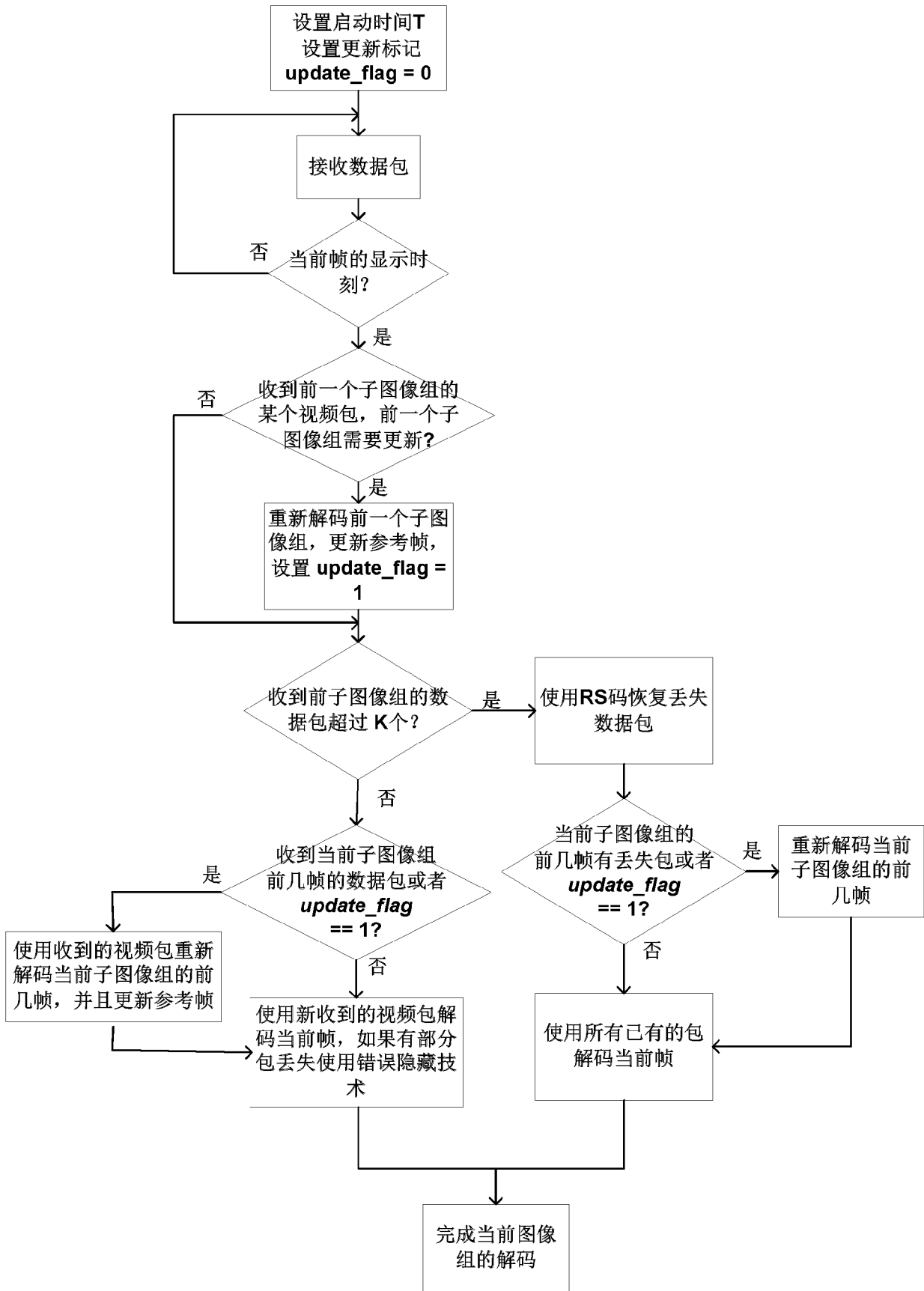


图 4

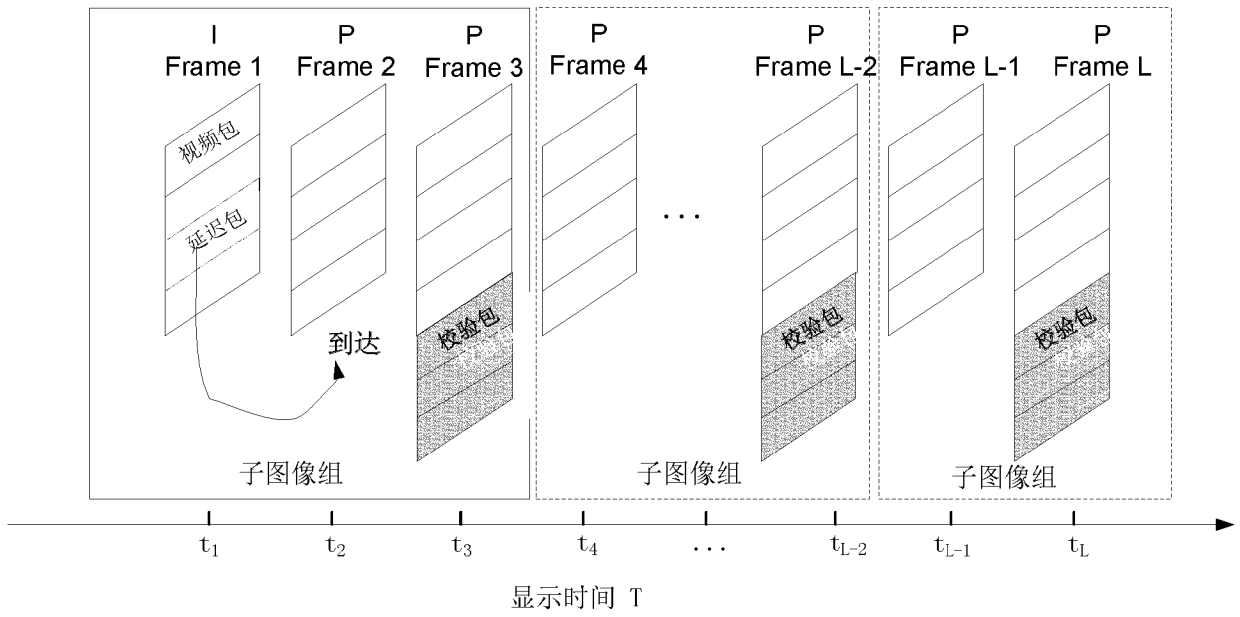


图 5

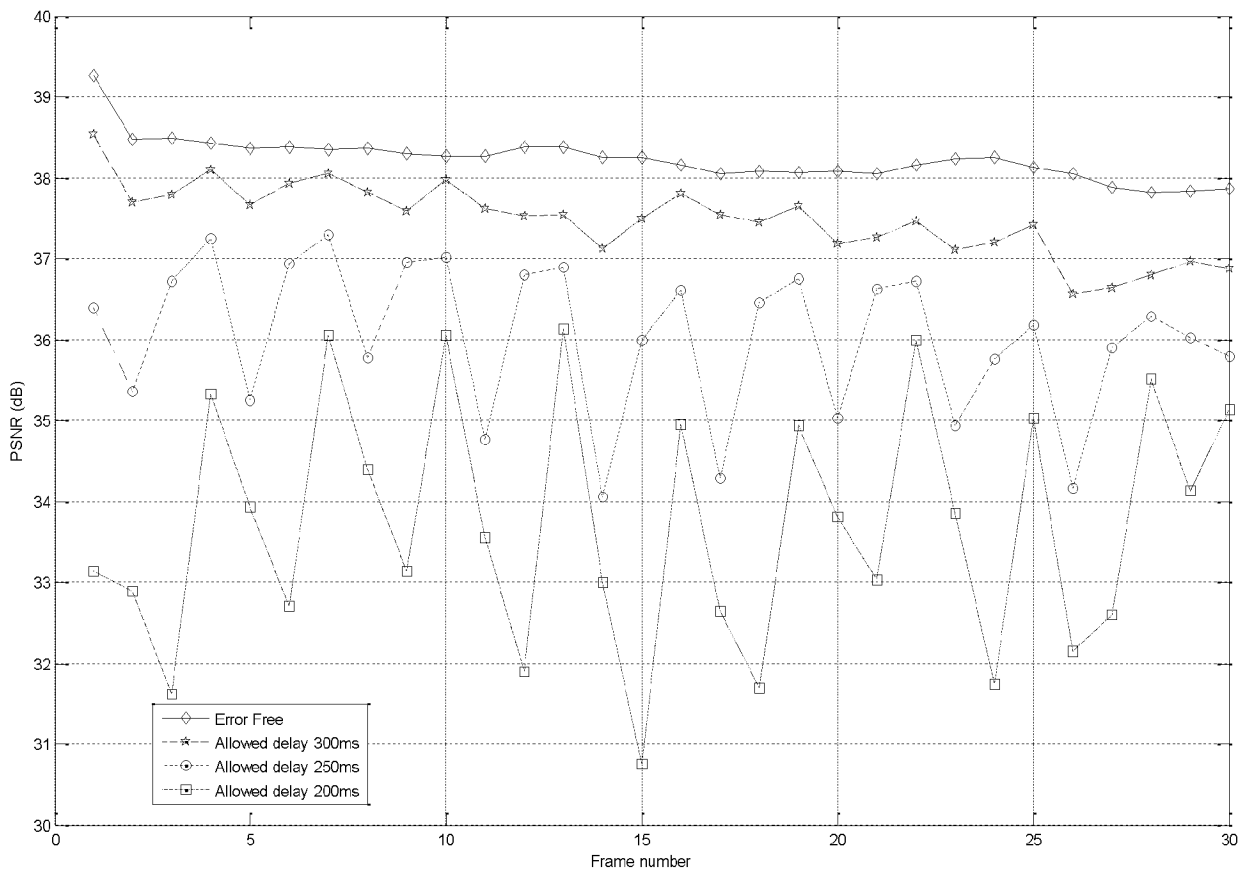


图 6

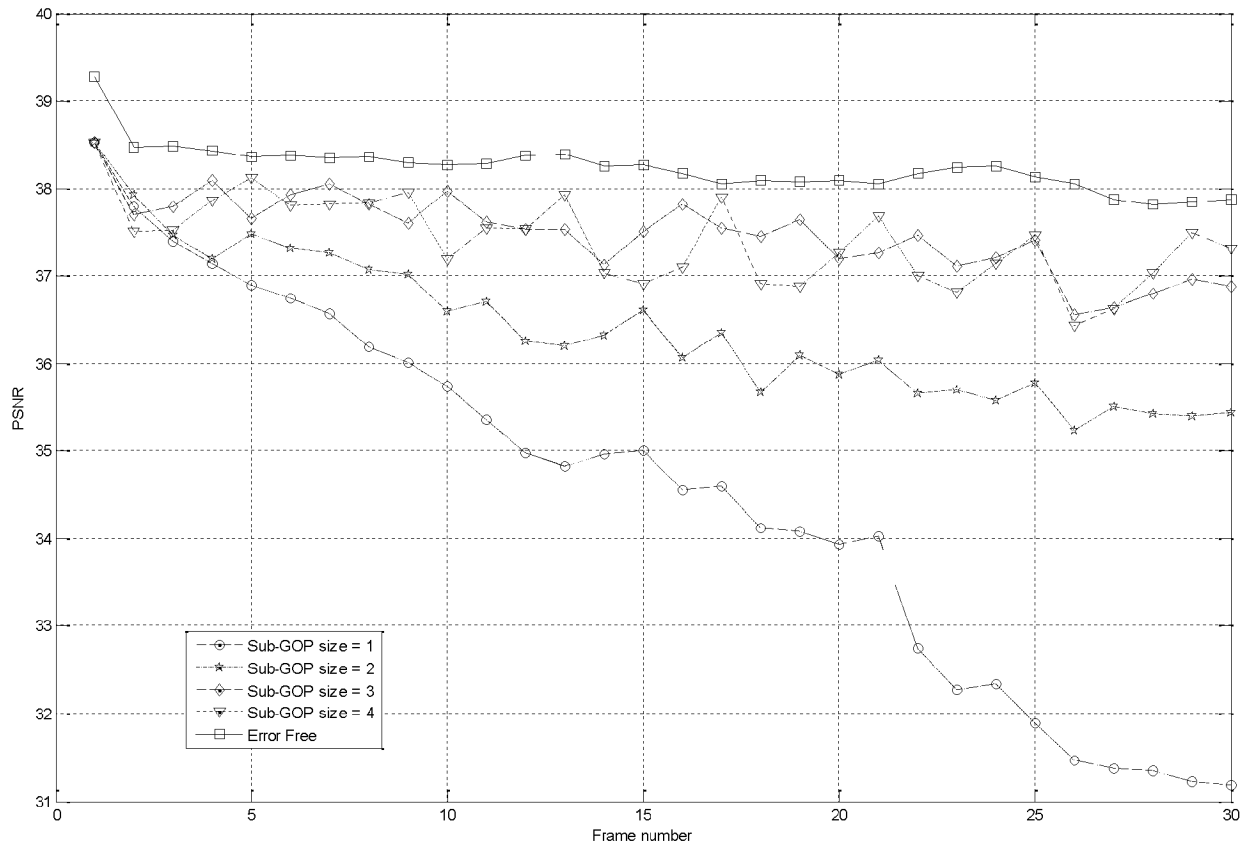


图 7