



(12) 发明专利

(10) 授权公告号 CN 101188757 B

(45) 授权公告日 2010.06.16

(21) 申请号 200710169376.5

(22) 申请日 2004.12.01

(30) 优先权数据

10-2004-0002076 2004.01.12 KR

60/525,827 2003.12.01 US

(62) 分案原申请数据

200410096448.4 2004.12.01

(73) 专利权人 三星电子株式会社

地址 韩国京畿道水原市灵通区梅滩洞 416

(72) 发明人 韩宇镇

(74) 专利代理机构 北京铭硕知识产权代理有限公司

公司 11286

代理人 郭鸿禧 李友佳

(51) Int. Cl.

H04N 7/26 (2006.01)

H04N 7/46 (2006.01)

H04N 7/50 (2006.01)

(56) 对比文件

WO 03/094526 A2, 2003.11.13, 全文.

US 2002/0150164 A1, 2002.10.17, 全文.

US 5742343 A, 1998.04.21, 全文.

US 2003/0202599 A1, 2003.10.30, 全文.

CN 1366777 A, 2002.08.28, 全文.

审查员 张春

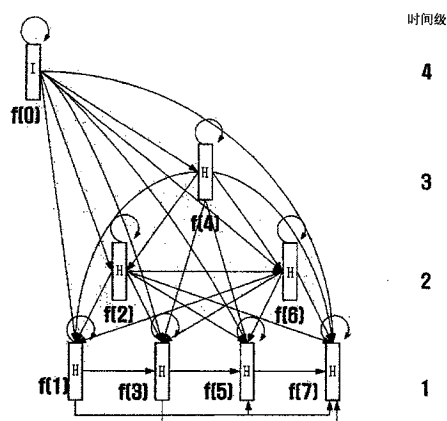
权利要求书 5 页 说明书 20 页 附图 21 页

(54) 发明名称

用于可伸缩视频编码和解码的方法和设备

(57) 摘要

提供一种用于可伸缩视频编码和解码的方法和设备。一种用于视频编码的方法,包括:从构成输入的视频序列的多个帧中以限定的时间级顺序消除时间冗余,以及通过量化从其时间冗余已被消除的帧中获得的变换系数来产生比特流。一种用于执行编码方法视频编码器,包括:时间变换单元、空间变换单元、量化单元和比特流产生单元。一种视频解码方法大体上对视频编码序列反转执行,其中,通过通过接收输入的比特流来提取关于编码的帧的信息并解释它们来执行解码。



1. 一种用于视频编码的方法,包括:
 - (a) 基于编码模式确定视频序列的多个帧的限定的时间级顺序,并且以所述限定的时间级顺序对所述帧进行时间变换以消除时间冗余;以及
 - (b) 通过量化从其时间冗余已被消除的帧中获得的变换系数来产生比特流,其中,所述限定的时间级顺序是从最高时间级到最低时间级的所述帧的顺序。
2. 如权利要求1所述的方法,其中,在步骤(a)中的帧是在通过小波变换之后其空间冗余已被消除的帧。
3. 如权利要求1所述的方法,其中,在步骤(b)中的变换系数通过执行其时间冗余已被消除的帧的空间变换而被获得。
4. 如权利要求3所述的方法,其中,空间变换基于小波变换而被执行。
5. 如权利要求1所述的方法,其中,帧的时间级具有二元体系结构。
6. 如权利要求1所述的方法,其中,限定的时间级顺序是从最高时间级到最低的时间级的帧的顺序和从相同的时间级上的最低帧指数到最高帧指数的帧的顺序。
7. 如权利要求6所述的方法,其中,限定的时间级顺序基于图像组(GOP)而被周期性重复。
8. 如权利要求7所述的方法,其中,最高时间级上的帧具有在构成GOP的帧中的GOP的最低帧指数。
9. 如权利要求8所述的方法,在步骤(a)中,基于GOP执行时间冗余的消除,在GOP中的最高时间级上的第一帧被编码为I帧,随后根据限定的时间级顺序从各个剩余帧消除时间冗余,并且在等于剩余帧的时间级的时间级上的帧中,基于在高于剩余帧的时间级的时间级上的至少一个参考帧或具有低于剩余帧的帧指数的帧指数的至少一个参考帧来执行从剩余帧中消除时间冗余。
10. 如权利要求9所述的方法,其中,参考帧包括在具有高于剩余帧的时间级的帧中的帧指数差最小的帧。
11. 如权利要求9所述的方法,其中,从剩余帧对时间冗余的消除基于剩余帧而被执行。
12. 如权利要求11所述的方法,其中,帧被编码为其中帧在对时间冗余进行的消除中涉及它们自身的比率大于预定值的I帧。
13. 如权利要求9所述的方法,其中,从剩余帧进行的对时间冗余的消除基于下一GOP的至少一帧而被执行,所述帧的时间级高于当前在步骤(a)中被处理的帧的每一个。
14. 如权利要求1所述的方法,其中,基于编码模式而被确定的限定的时间级顺序在相同的编码模式下基于图像组GOP而周期性地重复。
15. 如权利要求14所述的方法,其中,在构成GOP的帧中的最高时间级上的帧具有最低帧指数。
16. 如权利要求15所述的方法,在步骤(b)中,关于编码模式的信息被加入比特流。
17. 如权利要求15所述的方法,在步骤(b)中,关于空间消除和时间消除的顺序的信息被加入比特流。
18. 如权利要求14所述的方法,其中,编码模式根据根据端到端延迟参数D而被确定,其中限定的时间级顺序以这样的方式进行,即,从在具有与还未被时间冗余消除的最低时

间级上的帧相比不超过 D 的帧指数的帧之中的最高时间级上的帧向最低时间级上的帧,并且从相同时间级上的最低指数的帧向最高指数的帧进行。

19. 如权利要求 18 所述的方法,在基于 GOP 执行对时间冗余的消除的步骤 (a) 中,在 GOP 中的最高时间级上的第一帧被编码为 I 帧,随后在等于剩余帧的时间级的时间级上的帧中,根据限定的时间级顺序从各个剩余帧消除时间冗余,并且基于至少一个在高于剩余帧的时间级上的参考帧或至少一个具有低于剩余帧的帧指数的帧指数的参考帧来执行从剩余帧对时间冗余的消除。

20. 如权利要求 19 所述的方法,其中,参考帧包括在高于剩余帧的时间级时间级上的帧之中其帧指数差最小的帧。

21. 如权利要求 19 所述的方法,其中,在 GOP 内的最高时间级上的帧具有最低帧指数。

22. 如权利要求 19 所述的方法,其中,从剩余帧对时间冗余的消除基于剩余帧而被执行。

23. 如权利要求 22 所述的方法,其中,帧被编码为其中帧在对时间冗余的消除中涉及它们自身的比率超过预定值的 I 帧。

24. 如权利要求 19 所述的方法,其中,基于下一 GOP 的至少一个帧执行从剩余帧对时间冗余的消除,所述帧的时间级高于当前在步骤 (a) 中被处理的帧的每一个的时间级,并且所述帧到当前在步骤 (a) 中被处理的帧的每一个的时间距离小于或等于 D。

25. 一种视频编码器,包括:

时间变换单元,基于编码模式确定视频序列的多个帧的限定的时间级顺序,并且以所述限定的时间级顺序对所述帧进行时间变换以消除时间冗余;

空间变换单元,用于从所述帧中消除空间冗余;

量化单元,用于量化从在时间变换单元中消除时间冗余和在空间变换单元中消除空间冗余而获得的变换系数;以及

比特流产生单元,基于由量化单元产生的量化的变换系数产生比特流,

其中,所述限定的时间级顺序是从最高时间级到最低时间级的所述帧的顺序。

26. 如权利要求 25 所述的视频编码器,其中,时间变换单元消除帧的时间冗余并且将其时间冗余已被消除的帧传输到空间变换单元,并且空间变换单元消除其时间冗余已被消除的帧的空间冗余以产生变换系数。

27. 如权利要求 26 所述的视频编码器,其中,空间变换单元通过小波变换消除帧的空间冗余。

28. 如权利要求 25 所述的视频编码器,其中,空间变换编码器通过小波变换消除帧的空间冗余并且将其空间冗余已被消除的帧传输到时间变换单元,并且时间变换单元消除其空间冗余已被消除的帧的时间冗余以产生变换系数。

29. 如权利要求 25 所述的视频编码器,其中,时间变换单元包括:

运动估计单元,用于从帧中获得运动矢量;

时间滤波单元,基于由运动估计单元获得的运动矢量,以限定的时间级顺序进行时间滤波;以及

模式选择单元,用于确定限定的时间级顺序。

30. 如权利要求 29 所述的视频编码器,其中,由模式选择单元确定的限定的时间级顺

序基于图像组 GOP 的周期功能。

31. 如权利要求 29 所述的视频编码器,其中,模式选择单元从最高时间级到最低时间级并且从相同时间级中的最低帧指数到最高帧指数确定帧的限定的时间级顺序。

32. 如权利要求 31 所述的视频编码器,其中,由模式选择单元确定的限定的时间级顺序基于图像组 GOP 而被周期性地重复。

33. 如权利要求 29 所述的视频编码器,其中,模式选择单元基于延迟控制参数 D 而确定限定的时间级顺序,其中,确定的时间级顺序是从与其时间冗余未被消除的最低级上的帧相比不超过 D 的指数的帧之中的最高时间级到最低时间级的帧的顺序,和在相同的时间级中的最低帧指数到最高帧指数的帧的顺序。

34. 如权利要求 33 所述的视频编码器,其中,根据由模式选择单元选择的限定的时间级顺序,时间滤波单元基于图像组 GOP 消除时间冗余,其中,GOP 内的最高时间级上的帧被编码为 I 帧,随后从各个剩余帧消除时间冗余,并且在等于剩余帧的时间级的时间级上的帧中,基于在高于剩余帧的时间级的时间级上的至少一个参考帧或具有低于剩余帧的帧指数的帧指数的至少一个参考帧来执行从剩余帧中消除时间冗余。

35. 如权利要求 34 所述的视频编码器,其中,参考帧包括在高于剩余帧的时间级的时间级上的帧之中其帧指数差最小的帧。

36. 如权利要求 34 所述的视频编码器,其中,GOP 内的最高时间级上的帧具有最低时间帧指数。

37. 如权利要求 34 所述的视频编码器,其中,基于剩余帧执行从剩余帧对时间冗余的消除。

38. 如权利要求 37 所述的视频编码器,其中,时间滤波单元将当前滤波的帧编码为 I 帧,其中当前滤波的帧涉及其自身的比率超过预定值。

39. 如权利要求 25 所述的视频编码器,其中,比特流产生单元产生包括关于限定的时间级顺序的信息的比特流。

40. 如权利要求 25 所述的视频编码器,其中,比特流产生单元产生包括关于消除时间和空间冗余的顺序的信息的比特流以获得变换系数。

41. 一种视频解码方法,包括:

- (a) 通过接收并解释比特流来提取关于编码的帧的信息;
- (b) 通过反量化关于编码的帧的信息来获得变换系数;以及
- (c) 通过以限定的时间级顺序对变换系数进行反时间变换来恢复被编码的帧,其中,所述限定的时间级顺序是从最高时间级到最低时间级的所述帧的顺序。

42. 如权利要求 41 所述的方法,在步骤 (c) 中,通过对变换系数执行反时间变换并且通过对已被反时间变换的变换系数执行反小波变换来恢复被编码的帧。

43. 如权利要求 41 所述的方法,在步骤 (c) 中,通过对变换系数进行反空间变换并且通过对已被反空间变换的变换系数执行进行反时间变换恢复被编码的帧。

44. 如权利要求 43 所述的方法,其中,所述反空间变换采用反小波变换。

45. 如权利要求 41 所述的方法,其中,限定的时间级顺序是从最高时间级到最低时间级编码的帧的顺序,和从相同时间级中的最高帧指数到最低帧指数编码的帧的顺序。

46. 如权利要求 45 所述的方法,其中,限定的时间级顺序基于图像组 GOP 而被周期性地

重复。

47. 如权利要求 46 所述的方法,其中,反时间变换包括对编码的帧进行的反时间滤波,在 GOP 内,根据限定的时间级顺序,所述反时间滤波从在最高时间级上的编码的帧开始并且处理。

48. 如权利要求 41 所述的方法,其中,限定的时间级顺序根据从输入的比特流中提取的编码模式信息而被确定。

49. 如权利要求 48 所述的方法,其中,限定的时间级顺序在相同的编码模式下基于图像组 GOP 而被周期性地重复。

50. 如权利要求 48 所述的方法,其中,确定限定的时间级顺序的编码模式信息包括端到端延迟控制参数 D,其中,由编码模式信息确定的限定的时间级顺序以这样的方式进行,即,从在具有与还未被解码的最低时间级上的编码的帧相比不超过 D 的指数的编码的帧之中的最高时间级上的编码帧向最低时间级上的编码帧,并且从在相同的时间级中的最低指数的编码的帧向最高指数的帧进行。

51. 如权利要求 41 所述的方法,其中,冗余消除顺序从比特流中被提取。

52. 一种从比特流恢复帧的视频解码器,包括:

比特流解释单元,用于解释比特流以从其提取关于编码的帧的信息;

反量化单元,对关于编码的帧信息进行反量化以从其获得变换系数;

反空间变换单元,执行反空间变换处理;以及

反时间变换单元,以限定的时间级顺序执行反时间变换处理,

其中,通过对变换系数执行反空间处理和反时间变换处理来恢复比特流的编码的帧,并且所述限定的时间级顺序是从最高时间级到最低时间级的所述帧的顺序。

53. 如权利要求 52 所述的视频解码器,其中,反空间变换单元对已经被反时间变换单元反时间变换的变换系数执行反小波变换。

54. 如权利要求 52 所述的视频解码器,其中,反空间变换单元对变换系数执行反空间变换处理,并且反时间变换单元对已经被反空间变换单元反空间变换的变换系数执行反时间变换处理。

55. 如权利要求 54 所述的视频解码器,其中,反空间变换单元基于反小波变换执行反空间变换。

56. 如权利要求 52 所述的视频解码器,其中,限定的时间是从最高时间级到最低时间级的编码的帧的顺序,和从在相同的时间级中的最高帧指数到最低帧指数的编码的帧的顺序。

57. 如权利要求 56 所述的视频解码器,其中,限定的时间级顺序基于图像组 GOP 而周期性地重复。

58. 如权利要求 57 所述的视频解码器,其中,反时间变换单元基于 GOP 来执行反时间变换,并且编码的帧被反时间滤波,从 GOP 之内的最高时间级上的帧开始向最低时间级上的帧进行。

59. 如权利要求 52 所述的视频解码器,其中,比特流解释单元从输入的比特流提取编码模式信息,并且根据编码模式信息确定限定的时间级顺序。

60. 如权利要求 59 所述的视频解码器,其中,限定的时间级顺序基于图像组 GOP 而被周

期性地重复。

61. 如权利要求 59 所述的视频解码器,其中,确定限定的时间级顺序的编码模式信息包括端到端延迟控制参数 D,其中,由编码模式信息确定的限定的时间级顺序以这样的方式进行,即,从在具有与还未被解码的最低时间级上的编码的帧相比不超过 D 的帧指数的帧之中的最高时间级上的编码的帧向最低时间级上的解码的帧,并且从在相同时间级中的最低帧指数的帧向最高帧指数的帧进行。

62. 如权利要求 52 所述的视频解码器,其中,冗余消除顺序被从输入的流中提取。

用于可伸缩视频编码和解码的方法和设备

[0001] 本申请是申请日为 2004 年 12 月 1 日、申请号为 200410096448.4、题为“用于可伸缩视频编码和解码的方法和设备”的专利申请的分案申请。

技术领域

[0002] 本发明总的来说涉及视频压缩,尤其是,涉及根据限定的时间级顺序的具有通过运动补偿时间滤波的时间可伸缩性 (scalab) 的视频编码。

背景技术

[0003] 包括互联网的信息通信技术的发展导致视频通信的增长。然而,消费者对现有基于文本的通信方案并不满意。为了使消费者满意,包含多种包括文本、图像和音乐等的信息的多媒体数据被越来越多地提供。多媒体数据通常较庞大,从而其需要具有大容量的存储介质。而且,为了传输多媒体数据需要宽带。例如,具有 640×480 分辨率的 24 位真彩色图像需要每帧 $640 \times 480 \times 24$ 的容量,换句话说,大约 7.37Mbits 的数据。在这点上,需要大约 1200Gbits 的带宽以便以 30 帧 / 秒传输此数据,并且需要大约 1200Gbits 的存储空间以便存储具有 90 分钟长度的电影。考虑到这点,在传输包括文本、图像或声音的多媒体数据时使用压缩编码方案是必须的。

[0004] 数据压缩的基本原理是消除数据间的冗余。数据冗余是指三种冗余:空间冗余、时间冗余和感觉-视觉 (perceptual-visual) 冗余。空间冗余指的是在图像中相同的色彩或对象的重复,时间冗余指的是在运动图像帧中的相邻帧间没有或极少变化或音频中的相同声音的连续重复,并且感觉-视觉冗余指的是人的视觉和感觉对于高频率的迟钝。通过消除这些冗余,数据可被压缩。数据压缩的类型可分为取决于源数据是否丢失的有损 / 无损压缩、取决于数据是否相对于每一帧而被独立压缩的帧内 / 帧间压缩、月角取决于数据的压缩和恢复是否需要相同的时间周期的对称 / 非对称压缩。此外,当压缩和解压缩中的整个端到端延迟时间不超过 50ms 时,这被称作实时压缩。当帧具有多种分辨率时,这被称作可伸缩压缩。无损压缩主要用于压缩文本数据或媒体数据,并且有损压缩主要用于压缩多媒体数据。另一方面,帧内压缩通常用于消除空间冗余,并且帧间压缩用于消除时间冗余。

[0005] 用于传输多媒体数据的各种传输媒介依照介质具有不同的容量。当今使用的传输媒介具有多种传输速率,覆盖了能够每秒传输数十 Mbits 数据的超高速通信网络、具有每秒 384kbits 传输速率移动通信网络等。在传统视频编码算法,例如 MPEG-1、MPEG-2、H. 263 或 H. 264 中,通过基于运动补偿预测编码方案的运动补偿来消除时间冗余,并且通过变换编码方案来消除空间冗余。这些方案在压缩中具有较好性能,但是,因为方案的主算法采用递归方法,所以它们对于真正的可伸缩比特流不太具有灵活性。为此,最近的研究集中在基于小波的可伸缩视频编码上。可伸缩视频编码指的是具有可伸缩性的视频编码,可伸缩性的特性使部分压缩的比特流可被解码。由于该特性,可从比特流获得不同的视频。术语“可伸缩性”在此用于总地指可用于控制视频分辨率的特定可伸缩性、可用于控制视频质量的

信噪比 (SNR) 可伸缩性和可用于控制视频的帧速率的时间可伸缩性, 及其组合。

[0006] 在用于基于小波的可伸缩视频编码方案中的众多技术中, 由 Ohm(J. R. Ohm, “Three-dimensional subband coding with motion compensation,” IEEE Trans. Image Proc., Vol. 3, No. 5, Sept. 1994) 提出的并由 Choi 和 Wood(S. J. Choi and J. W. Woods, “Motion compensated 3-D subband coding of video,” IEEE Trans. Image Proc., Vol. 8, No. 2, Feb. 1999) 改进的运动补偿时间滤波 (MCTF) 是用于消除时间冗余, 并执行具有时间灵活性的可伸缩视频编码的核心技术。在 MCTF 中, 以图像组 (GOP) 为基础执行编码操作, 并且在运动方向上对成对的当前帧和参考帧进行时间滤波。参照图 1a, 此技术将被更详细地描述。

[0007] 图 1a 显示在采用 MCTF 方案的可伸缩视频编码和解码处理中的时间分解。

[0008] 在图 1a 中, L 帧表示低频或平均帧, 并且 H 帧表示高频或差别帧。如其中所示, 为了执行编码处理, 对最低时间级上的帧对进行第一时间滤波, 以因此将在低时间级的帧变换为在较高时间级的 L 帧和 H 帧, 并且对被变换的 L 帧对再次进行滤波并将其变换为在较高时间级的帧。编码器通过使用在最高级的 L 帧和经过小波变换的 H 帧来产生比特流。以图 1a 中的较暗颜色标记的帧表示它们被小波变换。时间级顺序从在较低级的帧向在较高级的帧进行。解码器通过操作通过以从在较高级的帧向在较低级的帧的顺序进行的逆小波变换而获得的较暗颜色的帧来恢复帧。通过使用在第三时间级的 L 帧和 H 帧来恢复两个在第二时间级的 L 帧并且通过使用在第二时间级的两个 L 帧和两个 H 帧来恢复在第一时间级的四个 L 帧。最后, 通过使用在第一时间级的四个 L 帧和四个 H 帧来恢复八个帧。采用原始 MCTF 方案的视频编码具有时间上灵活的可伸缩性, 但是其具有一些缺点, 诸如单方向运动高级性能较差和在低时间速率时质量较低等。已有一些尽力改进这些缺点的研究。它们中的一个是由 Turaga 和 mihaela(D. S. Turaga and Mihaela van der Schaar, “Unconstrained motion compensated temporal filtering,” ISO/IEC JTC1/SC29/WG11, MPEG03/M8388, 2002) 提出的非限定 MCTF (UMCTF)。将参照图 1b 描述 UMCTF。

[0009] 图 1b 显示在采用 UMCTF 方案的可伸缩视频编码和解码处理中的时间分解。

[0010] 在 UMCTF 方案中, 可用多个参考帧和双向滤波, 因此提供更通用的结构。另外, 通过使用未被滤波帧 (A 帧) 的适当的插入, 可在 UMCTF 方案方案中进行非二元时间滤波。因为由于缺乏准确的运动估计而导致 L 帧的视觉质量有时严重下降, 所以 A 帧的使用在较低的时间级上改善了视觉质量, 以代替被滤波的 L 帧。在过去的研究中, 很多实验结果显示没有更新步骤 (update-step) 的 UMCTF 具有比原始 MCTF 更好的性能。因此, 尽管 UMCTF 的最通用格式允许适应性选择低通滤波器, 但是通常使用没有更新步骤的 UMCTF 的特定格式。

[0011] 很多诸如视频会议的需要低端到端延迟的应用需要低编码侧延迟以及低解码侧延迟。由于 MCTF 和 UMCTF 二者都从最低时间级上的帧进行分析, 编码侧延迟应该和 GOP 的大小一样长。然而, 具有等于 GOP 大小的延迟的视频编码对于很多实时应用来说实际上是不可接受的。尽管我们知道 UMCTF 可通过限制未来参考帧的数量来降低延迟需求, 但是没有明确的用于控制端到端延迟的解决方案。另外, 没有提供编码侧时间可伸缩性, 为此, 视频编码不能在任一时间级上停止并且不能传输比特流。然而, 编码侧时间可伸缩性对实时双向视频流应用是很有用的。换句话说, 当在编码处理中操作不能进行时, 在当前时间级上的操作应该被挂起并且应该即时传输比特流。传统方案在这点上存在缺陷。

[0012] 考虑到上述问题,需要对视觉质量产生相对较小影响的并且能够控制端到端延迟以便提供低端到端延迟的视频编码算法。另外,需要用于提供从最高时间级到最低时间级的时间结构的视频编码算法,以便允许编码侧和解码侧二者具有时间可伸缩性。

发明内容

[0013] 因此,提出本发明以满足上述需要。本发明的一方面在于提供视频编码和解码方法和设备,其中,端到端延迟时间可被控制,并且编码器侧具有时间可伸缩性。

[0014] 根据本发明的示例性实施例,一种用于视频编码的方法包括:从构成输入的视频序列的多个帧中以限定的时间级顺序消除时间冗余,以及通过量化从其时间冗余已被消除的帧中获得的变换系数来产生比特流。

[0015] 在消除步骤中输入的帧可以是在通过小波变换之后其空间冗余已被消除的帧。

[0016] 在产生步骤中的变换系数可通过执行其时间冗余已被消除的帧的空间变换而被获得。空间变换基于小波变换而被执行。

[0017] 帧的时间级可具有二元体系结构。

[0018] 限定的时间级顺序可以是在最高到最低的时间级上的帧的顺序和具有在相同的时间级中最低到最高指数的帧的顺序。限定的时间级顺序基于 GOP 大小而被周期性重复。此刻,最高时间级上的帧具有在构成 GOP 的帧中的 GOP 的最低帧指数。

[0019] 基于 GOP 执行时间冗余的消除,其中,在 GOP 中的最高时间级上的第一帧被编码为 I 帧,随后根据限定的时间级顺序从各个剩余帧消除时间冗余,并且对于此消除,剩余帧的每一个均涉及在高于其自身的时间级上的一个或多个参考帧,或涉及在等于其自身的时间级上的帧中的其帧指数低于其自身的一个或多个参考帧。在具有高于它们自身的时间级的一个或多个帧之中,涉及从各个帧中消除时间冗余的参考帧可以包括其指数差最小的一个或两个帧。

[0020] 在时间冗余消除处理中每一帧涉及的参考帧还包括它们本身(当前在滤波处理中的帧)。当前在滤波处理中的帧被编码为其中当前在滤波处理中的帧涉及在时间冗余消除处理中的它们自身的比率大于预定值的 I 帧。

[0021] 在时间冗余消除处理中涉及的参考帧还可包括在下一 GOP 中的一个或多个帧,所述帧的时间级高于当前在滤波处理中的帧的每一个。

[0022] 限定的时间级顺序根据编码模式而被确定。根据编码模式而被确定的限定的时间级顺序在相同的编码模式下基于 GOP 而周期性地重复。在构成 GOP 的帧中的最高时间级上的帧可具有最低帧指数。

[0023] 在产生步骤中,关于编码模式的信息被加入比特流。

[0024] 在产生步骤中,关于空间消除和时间消除的顺序的信息被加入比特流。

[0025] 编码模式根据端到端延迟参数 D 而被确定,其中限定的时间级顺序以这样的方式进行,即,从在具有与还未被时间滤波的最低时间级上的帧相比不超过 D 的帧指数的帧之中的最高时间级上的帧向最低时间级上的帧,并且从相同时间级上的最低指数的帧向最高指数的帧进行。基于 GOP 执行对时间冗余的消除,其中,在 GOP 中的最高时间级上的第一帧被编码为 I 帧,随后根据限定的时间级顺序从各个剩余帧消除时间冗余,并且对于此消除,剩余帧的每一个涉及高于其自身的时间级上的一个或多个参考帧,或涉及在等于其自身的

时间级上的帧之中其指数低于其自身的一个或多个参考帧。涉及从各个帧中消除时间冗余的参考帧优选地包括在具有高于它们自身的时间级的一个或多个帧之中的一个或两个其指数差最小的帧。

[0026] 在 GOP 内的最高时间级上的帧具有最低帧指数。

[0027] 在时间冗余消除处理中,每一帧所涉及的一个或多个参考帧包括它们自身。当前在滤波处理中的帧被编码为其中当前在滤波处理中的帧在时间冗余处理中涉及它们自身的比率超过预定比率值的 I 帧。

[0028] 在时间冗余消除处理中涉及的参考帧还包括在下一 GOP 上的一个或多个帧,所述帧的时间级高于当前在滤波处理中的帧的每一个并且其到当前在滤波处理中的帧的每一个的时间距离在 D 之内。

[0029] 根据本发明的示例性实施例,一种视频编码器包括时间变换单元,用于从多个输入的帧中以限定的时间级顺序消除时间冗余;空间变换单元,用于从帧中消除空间冗余;量化单元,用于量化在消除时间冗余和空间冗余的处理中获得的变换系数;以及比特流产生单元,通过使用量化的变换系数产生比特流。

[0030] 视频时间变换单元消除帧的时间冗余并且将其时间冗余已被消除的帧传输到空间变换单元,随后空间变换单元消除其时间冗余已被消除的帧的空间冗余,并且因此获得变换系数。此时,空间变换单元通过小波变换消除帧的空间冗余。

[0031] 空间变换编码器通过小波变换消除帧的冗余并且将其空间冗余已被消除的帧传输到时间变换单元,随后时间变换单元消除其空间冗余已被消除的帧的时间冗余,并且因此获得变换系数。

[0032] 时间变换单元包括:运动估计单元,用于从多个输入的帧获得运动矢量;时间滤波单元,通过使用运动矢量,从多个输入的帧以限定的时间级顺序进行时间滤波;以及模式选择单元,用于确定限定的时间级顺序。模式选择单元选择的限定的时间级顺序基于 GOP 大小的周期功能。

[0033] 模式选择单元确定最高到最低时间级上的帧的和在相同时间级中的最小到最大帧指数的帧的限定的时间级顺序。由模式选择单元确定的限定的时间级顺序可基于 GOP 大小而被周期性地重复。

[0034] 模式选择单元通过参照延迟控制参数 D 而确定限定的时间级顺序,其中,限定的时间级顺序是在与其时间冗余未被消除的最低级上的帧相比不超过 D 的指数的帧之中的最高到最低时间级上的帧的顺序,和在相同的时间级中的最小到最大帧指数的帧的顺序。

[0035] 根据由模式选择单元选择的限定的时间级顺序,时间滤波单元基于 GOP 消除时间冗余,其中,GOP 内的最高时间级上的帧被编码为 I 帧,随后从各个剩余帧消除时间冗余,并且对于此消除,剩余帧的每一个均涉及比其自身高的时间级上的一个或多个参考帧,或涉及在等于其自身的时间级上的帧之中其指数低于其自身的一个或多个参考帧。涉及从帧中消除时间冗余的参考帧包括在高于当前滤波帧的时间级上的一个或多个帧之中具有与当前在滤波处理中的帧相比的最小指数差的一个或两个帧。

[0036] GOP 内的最高时间级上的帧具有最低时间帧指数。

[0037] 时间滤波单元还包括当从当前滤波的帧中消除时间冗余时涉及参考帧的当前在滤波处理中的帧。此时,时间滤波单元将当前滤波的帧编码为 I 帧,其中当前滤波的帧涉及

其自身的比率超过预定值。

[0038] 比特流产生单元产生包括关于限定的时间级顺序的信息的比特流。比特流产生单元可产生包括关于消除时间和空间冗余的顺序的（冗余消除顺序）信息的比特流以获得变换系数。

[0039] 根据本发明的示例性实施例，一种视频解码方法包括：通过接收输入的比特流并且解释它们来提取关于编码的帧的信息；通过反量化关于编码的帧的信息来获得变换系数；以及通过以限定的时间级顺序对变换系数进行反时间变换来恢复帧。

[0040] 在恢复步骤中，通过对变换系数进行反时间变换并且其后通过对结果进行反小波变换来获得帧。

[0041] 在恢复步骤中，通过在反时间变换之前对系数进行反空间变换并且其后通过对结果进行反时间变换来获得帧。反空间变换优选地采用反小波变换。

[0042] 限定的时间级顺序涉及从最高到最低时间级上的帧的顺序，并且涉及在相同时间级中的最大到最小指数的帧。限定的时间级顺序基于 GOP 大小而被周期性地重复。反时间变换涉及对帧进行的反时间滤波，在 GOP 内，根据限定的时间级顺序，所述时间滤波从最高时间级上的编码帧开始并且处理。

[0043] 限定的时间级顺序根据从输入的比特流中提取的编码模式信息而被确定。限定的时间级顺序在相同的编码模式下基于 GOP 大小而被周期性地重复。

[0044] 确定限定的时间级顺序的编码模式信息可包括端到端延迟控制参数 D，其中，由编码模式信息确定的限定的时间级顺序以这样的方式进行，即从在具有与还未被解码的最低时间级上的帧相比不超过 D 的指数的帧之中的最高时间级上的编码帧向最低时间级上的编码帧，并且从在相同的时间级中的最低指数向最高指数的帧进行。

[0045] 冗余消除顺序从输入的比特流中被提取。

[0046] 根据本发明的示例性实施例，一种从输入的比特流恢复帧的视频解码器包括：比特流解释单元，用于解释比特流以从其提取关于编码的帧的信息；反量化单元，对编码的帧信息进行反量化以从其获得变换系数；反空间变换单元，执行反空间变换处理；以及反时间变换单元，以限定的时间级顺序执行反时间变换处理，其中，通过变换系数的反时间变换处理来恢复帧。

[0047] 视频解码器还包括：反小波变换单元，对通过由反空间变换单元运算变换系数而获得的结果执行反小波变换。

[0048] 视频解码器还包括：反空间变换单元，对变换系数执行反空间变换，其中，通过变换系数的反空间变换获得的结果被反时间变换单元进行反时间变换。

[0049] 反空间变换单元基于反小波变换执行反空间变换，并且限定的时间级顺序从最高时间级上的编码帧开始向最低时间级上的编码帧进行。限定的时间级顺序基于 GOP 大小而周期性地重复。

[0050] 反时间变换单元基于 GOP 执行反时间变换，并且编码的帧被反时间滤波，从 GOP 之内的最高时间级上的帧开始向最低时间级上的帧进行。

[0051] 比特流解释单元从输入的比特流提取关于编码的信息，并且根据编码模式信息确定限定的时间级顺序。限定的时间级顺序基于 GOP 而被周期性地重复。

[0052] 确定限定的时间级顺序的编码模式信息包括端到端延迟控制参数 D，其中，由编码

模式信息确定的限定的时间级顺序以这样的方式进行,即,从在具有与还未被解码的最低时间级上的帧相比不超过 D 的指数的帧之中的最高时间级上的编码的帧向最低时间级上的编码的帧,并且从在相同时间级中的最低指数的帧向最高指数的帧进行。

[0053] 冗余消除顺序被从输入的流中提取。

附图说明

[0054] 通过下面结合附图进行的详细描述,本发明的上述和其他目的、特点和优点将会被理解得更清楚,其中:

[0055] 图 1a 显示在采用 MCTF 方案的可伸缩视频编码和解码处理中的时间分解;

[0056] 图 1b 显示在采用 UMCTF 方案的可伸缩视频编码和解码处理中的时间分解;

[0057] 图 2 是显示根据本发明的示例性实施例的可伸缩视频编码器的功能方框图;

[0058] 图 3 是显示根据本发明的示例性实施例的可伸缩视频编码器的功能方框图;

[0059] 图 4 是显示根据本发明的示例性实施例的可伸缩视频解码器的功能方框图;

[0060] 图 5 显示根据本发明的示例性实施例的连续时间逼近和参照(星)算法的基本概念;

[0061] 图 6 显示在星算法中的帧之间的众多可能的联系;

[0062] 图 7 显示根据本发明的示例性实施例的 GOP 之间的参照;

[0063] 图 8 显示根据本发明的示例性实施例的在非二元时间滤波的帧之间的可能的联系;

[0064] 图 9 显示根据本发明的示例性实施例的在端到端延迟控制参数为 0 的时间滤波中的帧之间的可能的联系;

[0065] 图 10 显示根据本发明的示例性实施例的在端到端延迟控制参数为 1 的时间滤波中的帧之间的可能的联系;

[0066] 图 11 显示根据本发明的示例性实施例的在端到端延迟控制参数为 3 的时间滤波中的帧之间的可能的联系;

[0067] 图 12 显示根据本发明的示例性实施例的在端到端延迟控制参数为 3 并且 GOP 大小为 16 的时间滤波中的帧之间的可能的联系;

[0068] 图 13 显示前向、后向、双向和内部预测模式;

[0069] 图 14 显示根据本发明的示例性实施例的在时间滤波中包括 4 种预测模式帧之间的可能的联系;

[0070] 图 15a 显示根据图 14 的示例性实施例的快速改变的视频序列中的视频编码的例子;

[0071] 图 15b 显示根据图 14 的示例性实施例的慢速改变的视频序列中的视频编码的例子;

[0072] 图 16 是示出在视频编码方案中的福曼通用中间格式(Foreman commonintermediate format(CIF))序列的峰值信噪比(PSNR)结果的图表;

[0073] 图 17 是示出在视频编码方案中的移动 CIF 序列的 PSNR 结果的图表;

[0074] 图 18 是示出在视频编码方案中的具有不同的端到端延迟设定的福曼 CIF 序列的 PSNR 结果的图表;

[0075] 图 19 是示出在视频编码方案中的具有不同的端到端延迟设定的移动 CIF 序列的 PSNR 结果的图表；

[0076] 图 20 是示出当通过使用四种预测模式对一部分从电影“Matrix 2”快速改变的场景进行编码以及不使用该四种预测模式进行编码时的 PSNR 结果的图表。

具体实施方式

[0077] 以下,参照附图来详细说明本发明的示例性实施例。

[0078] 图 2 是显示根据本发明的示例性实施例的可伸缩视频编码器的功能方框图。

[0079] 可伸缩视频编码器接收多个输入的帧以构成视频序列,并且将它们压缩以产生比特流。为了执行此工作,可伸缩视频编码器包括:时间变换单元 10,用于消除多个帧之间的时间冗余;空间变换单元 20,用于消除帧之间的空间冗余;量化单元 30,用于量化通过消除时间和空间冗余而产生的变换系数;以及比特流产生单元 40,用于产生将量化的变换系数和其它信息结合的比特流。

[0080] 为了补偿帧之间的运动并执行时间滤波,时间变换单元 10 包括运动估计单元 12、时间滤波单元 14 和模式选择单元 16。

[0081] 运动估计单元 12 获得用于在当前在时间滤波操作中的帧的每一宏块和与其对应的参考帧的每一宏块之间的运动补偿的运动矢量。关于运动矢量的信息被提供给时间滤波单元 14,并且时间滤波单元 14 通过使用关于运动矢量的信息执行对多个帧的时间滤波。在此示例性实施例中,基于 GOP 执行时间滤波。

[0082] 模式选择单元 16 确定用于时间滤波的顺序。在此示例性实施例中的时间滤波基本上以从 GOP 内的最高时间级上的帧开始向最低时间级上的帧进行的顺序来处理。当帧在相同的时间级上时,时间滤波以从最低指数的帧开始向最高指数的帧进行的顺序中被处理。帧指数是表示构成 GOP 的帧之间的时间顺序的指数。因此,在构成一个 GOP 的帧的数量是 n 时,通过定义为 GOP 的最早帧的 GOP 的第一帧为 0,在时间顺序中的最后帧具有指数 $n-1$ 。

[0083] 在此示例性实施例中,通过例子的方法,最低指数的帧被用作在构成 GOP 的帧之中具有最高时间级的帧。这应该被解释为 GOP 内的最高时间级上的其它帧的选择也被本发明的技术概念覆盖。

[0084] 模式选择单元 16 可以以限定延迟的模式执行视频编码处理以降低在视频编码处理中产生的端到端延迟。在此情况下,根据端到端延迟控制参数 D 的值,模式选择单元 16 可限制时间滤波以便作为如上所述的最高到最低时间级上的帧的顺序。另外,考虑到编码处理中的操作性能的限制,模式选择单元 16 可改变时间滤波的顺序或者通过删除一些帧来执行时间滤波。下面,术语“限定的时间级顺序”将被使用,其是指考虑到全部相关因素的时间滤波的顺序。限定的时间级顺序规定时间滤波从最高时间级上的帧开始。

[0085] 从其中消除了时间冗余的帧,即,被时间滤波的帧,通过空间变换单元 20 以从其消除空间冗余。通过使用空间变换,空间变换单元 20 从已被时间滤波的帧中消除空间冗余。在本示例性实施例中,使用小波变换。在当前公知的小波变换中,一帧被四分,四分之一帧被在区域中四分但是几乎等于该帧中的完整图像的缩小的图像(L 图像)所代替,并且该帧中的其它三个四分之一帧被可用于恢复完整图像的信息(H 图像)所代替。同样的,L

帧可再次被四分,并且被具有四分之一区域的 L 图像和用于恢复 L 图像的信息所代替。这种利用小波变换的压缩方案已被应用于 JPEG2000 压缩方案。可通过基于小波的变换消除该方案的空间冗余。不同于 DCT 变换,在小波变换中,原始图像信息被存储在大小被缩小的变换的图像中,因此通过使用缩小的图像使具有空间可伸缩性的视频编码能够进行。然而,小波变换的方案在此通过例子的方式而被使用,其中不需要完成空间可伸缩性,DCT 方案可被用作诸如 MPEG-2 的传统运动图像压缩方案。

[0086] 通过空间变换,时间滤波的帧被变换为变换系数。变换系数被传输到量化单元 30 并且随后被量化。量化单元 30 量化为浮点数类型系数的变换系数,并且将它们变换为整数类型的变换系数。即,代表图像数据的比特数可通过量化而被减少。在此示例性实施例中,通过嵌入式的量化方案,量化变换系数的处理被执行。由于变换系数的量化通过嵌入式的量化方案而被执行,所以视频序列所需的信息量可通过量化而被减少,并且 SNR 可伸缩性可通过嵌入式的量化而被获得。术语“嵌入式”被用于表示编码的比特流包含量化。换句话说,压缩的数据以视觉上重要的顺序而生成,并且按视觉重要性而被标记。实际上,量化(或视觉重要性)等级可用于解码器或传输信道。如果传输带宽、存储容量和显示资源被允许,则图像可被高质量地恢复。否则,图像仅被量化为被大多数限定的资源容许的限度。传统的嵌入式量化算法包括嵌入式零元树(zerotree)小波变换(EZW)、多级树集合分裂(set partitioning in hierarchical trees(SPIHT))、嵌入式零块编码(embedded zero block coding(EZBC))、具有最佳切断的嵌入式块编码(embedded block coding with optimal truncation(EBCOT))等。在本示例性实施例中,可使用任何已知的算法。

[0087] 比特流产生单元 40 通过在其上附上报头(header)来产生将关于编码的图像的信息和关于从运动估计单元 12 获得的运动矢量的信息结合的比特流。在此示例性实施例中,关于限定的时间级顺序的信息,比特流和延迟参数等被包括在关于比特流的信息中。

[0088] 在小波变换被用于消除空间冗余时,变换的帧的原始图像的形状保持不变。为此,不同于基于 DCT 的运动图像编码方案,图像可按顺序通过空间变换、时间变换和量化以产生比特流。此处理的另一示例性实施例将参照图 3 而被描述。

[0089] 图 3 是显示根据本发明的示例性实施例的可伸缩视频编码器的功能方框图。

[0090] 参照图 3,可伸缩视频编码器包括:空间变换单元 60,用于从多个帧中消除空间冗余;时间变换单元 70,用于从帧中消除时间冗余;量化单元 80,用于量化通过消除空间和时间冗余二者而获得的变换系数;以及比特流产生单元 90,用于产生将编码的图像信息和其它信息结合的比特流。

[0091] 在传统运动图像压缩中,因为空间变换在时间滤波之后已被应用,所以术语“变换系数”通常指的是由空间变换产生的值。此术语换句话说指的是其中值通过 DCT 变换而被产生的 DCT 系数。因此,术语“小波系数”被用于通过小波变换而产生值的情况。在本发明中,术语“变换系数”表示通过在它们被(嵌入式)量化之前从多个帧中消除空间和时间冗余二者而产生的值。在这一点上,应该注意术语“变换系数”是指通过图 2 中的空间变换而产生的系数,以及通过图 3 中的时间变换而产生的系数。

[0092] 空间变换单元 60 消除构成视频序列的多个帧之间的空间冗余。在此情况下,通过使用小波变换,空间变换单元 60 消除在帧内存在的空间冗余。从其消除了空间冗余的帧,即,空间变换的帧,被传输到时间变换单元 70。

[0093] 时间变换单元 70 消除空间变换的帧的时间冗余。为了执行此工作,时间变换单元 70 包括运动估计单元 72、时间滤波单元 74 和模式选择单元 76。在此示例性实施例中,不同于图 2,除输入的帧是空间变换的帧之外,时间变换单元 70 以与图 2 中相同的方式工作。而且,图 3 中的示例性实施例和图 2 中的示例性实施例之间的不同在于时间变换单元 70 产生用于量化其时间冗余在它们的空间冗余已被消除之后被消除的帧的变换系数。

[0094] 量化单元 80 量化变换系数以产生量化的图像信息(编码的图像信息),并且将产生的图像信息传输到比特流产生单元 40。类似于图 2,嵌入式量化被执行,并且与将被最终产生的比特流相关的 SNR 可伸缩性被获得。

[0095] 比特流产生单元 90 通过在其上附上报头来产生将编码的图像信息和关于运动矢量的信息结合的比特流。此时,类似于图 2,端到端延迟控制参数和时间级顺序可被包括。

[0096] 图 2 的比特流产生单元 40 和图 3 的比特流产生单元 90 可包括关于消除比特流中的时间冗余和空间冗余的顺序(此后被称作“冗余消除顺序”)的信息,以便允许解码器侧确定是根据图 2 对视频序列进行编码还是根据图 3 对视频序列进行编码。冗余消除顺序可被包括在使用不同的方案的比特流中。在比特流中可选择一个方案作为基本方案并且分别地表示其它方案。例如,如果图 2 中使用的方案被选为基本方案,则由图 2 的视频可伸缩编码器产生的比特流可被表示为不具有关于冗余消除顺序的信息,而是可被表示为仅具有关于由图 3 的可伸缩视频编码器产生的冗余消除顺序。与此相反,关于冗余消除顺序二者的信息可都被表示在图 2 和图 3 二者的方案中。

[0097] 通过对图 2 和 3 的方案中的视频序列进行编码和比较,还可通过创造具有分别根据图 2 和 3 的可伸缩视频编码器侧功能二者的可伸缩视频编码器来产生具有出色的编码效率的比特流。在此情况下,冗余消除顺序可被包括在比特流中。基于序列或 GOP,冗余消除顺序可被确定。在前面的情况下,冗余消除顺序应该被包括在视频序列的报头中;在后面的情况下,冗余消除顺序应该被包括在 GOP 的报头中。

[0098] 图 2 和 3 中显示的示例性实施例都可通过硬件来实现,但是它们也可通过具有软件模块和执行它们的计算能力的任一装置而被实现。

[0099] 图 4 是显示根据本发明的示例性实施例的可伸缩视频解码器的功能方框图。

[0100] 可伸缩视频解码器包括:比特流解释单元 100,用于解释输入的比特流并且提取构成比特流的每一分量;第一解码单元 200,用于恢复根据图 2 编码的图像;以及第二解码单元 300,用于恢复根据图 3 编码的图像。

[0101] 第一和第二解码单元 200 和 300 可通过硬件或软件模块而被实现。当它们通过硬件或软件模块而被实现时,它们可以被以分离的方式或综合的方式来实施。如果它们被以综合的方式来实施,则第一和第二解码单元 200 和 300 通过以相反的方式应用冗余消除顺序来从在比特流解释单元 100 中获得的比特流中恢复被消除的数据。

[0102] 视频可伸缩解码器可被实现以便恢复在不同的冗余消除顺序中编码的全部图像,但是它也可被实现从而仅在冗余消除顺序二者之一中编码的图像可被恢复。

[0103] 比特流解释单元 100 解释输入的比特流,提取编码的图像信息(编码的帧)并且确定用于冗余消除的顺序。如果冗余消除顺序被第一解码单元 200 解码,则视频序列通过第一解码单元 200 而被恢复。如果冗余消除顺序被第二解码单元 300 解码,则视频序列通过第二解码单元 300 而被恢复。另外,比特流解释单元 100 解释比特流,以因此确定限定的

时间级顺序,即,用于当时间冗余被消除时的帧的时间滤波的顺序。在此示例性实施例中,限定的时间级顺序可基于确定编码模式的延迟控制参数的值而被确定。在从编码的图像信息恢复视频序列的处理中,由第一解码单元 200 进行的解码将被首先描述,并且由第二解码单元 300 进行的解码将被随后描述。

[0104] 关于输入到第一解码单元 200 的解码的帧的信息被反量化单元 210 反量化并且被变换为变换系数。变换系数被反空间变换单元 220 反空间变换。反空间变换类型取决于编码的空间变换类型。在小波变换被用于空间变换的情况下,反小波变换被执行以用于反空间变换。在 DCT 变换被用于空间变换的情况下,反 DCT 变换被执行。通过反空间变换,变换系数被变换为全部被时间上反滤波的 I 帧和 H 帧。在这点上,反时间滤波单元 230 以限定的时间级顺序通过反时间变换恢复构成视频序列的帧。通过解释由比特流解释单元 100 输入的比特流,限定的时间级顺序可被获得。为了反时间变换,反时间滤波单元 230 使用通过解释比特流而获得的运动矢量。

[0105] 关于输入到第二解码单元 300 的编码的帧的信息被反量化单元 310 反量化并且被转换为变换系数。变换系数被反时间变换单元 320 反时间变换。用于反时间变换的运动矢量和限定的时间级顺序可从通过由比特流的比特流解释单元 100 解释而获得的信息中而被获得。关于已通过反时间变换的编码的图像的信息被变换为在已通过空间变换的状态下的帧。已通过空间变换的状态下的帧在反空间变换单元 330 中被反空间变换,并且被恢复为构成视频序列的帧。用在反空间变换单元 330 中的反空间变换采用反小波变换方案。

[0106] 对在具有保持时间可伸缩性并控制端到端延迟的限定的时间级顺序中的帧进行时间变换的处理将被更详细地描述。

[0107] 根据本发明,通过连续的时间逼近和参照算法,即星算法,帧可具有在编码侧和解码侧二者上的时间可伸缩性并且端到端延迟可被控制。

[0108] 图 5 显示星算法的基本概念。

[0109] 比特流描述星算法的基本概念,每一时间级上的每一帧被表示为节点,并且帧之间的参照关系被表示为弧。在每一时间级上,仅必要帧被定位,例如,在最高时间级上仅 GOP 的第一帧被需要。在此示例性实施例中, F(0) 帧在最高时间级上。在下一时间级上,时间分解被连续细化,并且具有高频率的损失 (missing) 帧被具有已被处理的帧指数的原始帧预测。如果 GOP 的大小是 8,则指数 0 的帧被编码为最高时间级上的 I 帧,并且通过使用指数 0 的原始帧,指数 4 的帧被编码为下一时间级上的中间帧 (inter-frame) (H 帧)。其后,通过使用指数 0 和 4 的原始帧,指数 2 和 6 的帧被编码为中间帧。最后,通过使用指数 0、2、4 和 6 的原始帧,指数 1、3、5 和 7 的帧被编码为中间帧。在此,术语“原始帧”主要指的是构成视频序列的帧,但是它也被用于指通过解码之前编码的帧而获得的帧。在解码处理中,指数 0 的帧被首先解码。其后,指数 4 的帧参照解码的指数 0 的帧而被解码。在相同的方式下,指数 2 和 6 的帧参照解码的指数 0 和 4 的帧而被解码。最后,指数 1、3、5 和 7 的帧参照解码的指数 0、2、4 和 6 的帧而被解码为中间帧。

[0110] 如图 5 中所示,编码侧和解码侧二者都可使用相同的可提供编码侧的时间可伸缩性的时间级处理流程。编码侧可在任一时间级停止,但是解码侧应该按照目标时间级对帧进行解码。即,由于最高时间级上的帧被首先编码,所以编码侧可获得时间可伸缩性。例如,如果当指数 0、4、2 和 6 的帧已被编码时编码处理停止,则解码侧解码指数 0 的帧,参照指数

0 的帧解码指数 4 的帧,并且参照指数 0 和 4 的帧解码指数 2 和 6 的帧。在此情况下,解码侧可输出帧 0、2、4 和 6。最高时间级上的帧(在此示例性实施例中的 F(0))被编码到 I 帧而不被编码到需要对其它帧操作的 L 帧是优选的,以便保持编码侧时间可伸缩性。

[0111] 为了比较,基于 MCTF 或 UMCTF 的传统的可伸缩视频编码算法可具有解码侧时间可伸缩性,但是编码侧在保证时间可伸缩性上有困难。参照图 1a 和 1b,时间级 3 上的 L 帧或 H 帧需要被执行解码处理。然而,在 MCTF 和 UMCTF 算法的情况下,最高时间级上的 L 帧或 H 帧可仅在编码处理完成后而被获得,但是解码处理可在任一时间级上停止。

[0112] 在编码和解码侧二者上保持时间可伸缩性的需求将在下面被描述。

[0113] 假设 F(k) 是第 k 个帧并且 T(k) 是 F(k) 的时间级。如果在某一时间级上的帧参照等于或小于该时间级的时间级上的帧而被编码,则时间可伸缩性被建立。例如,指数 4 的帧不能参照指数 2 的帧。这是因为如果这种参照被允许则编码处理不能在指数 0 和 4 的帧上停止。换句话说,指数 2 的帧不能在指数 4 的帧在之前被编码时被编码。可由帧 F(k) 参照的参考帧的集合 R_k 如等式 (1) 而被定义。

$$[0114] \quad R_k = \{F(1) \mid (T(1) > T(k)) \text{ or } ((T(1) = T(k)) \text{ and } (1 \leq k))\} \quad (1)$$

[0115] 其中,“1”表示帧指数。

[0116] 在等式 (1) 中, $((T(1) = T(k)) \text{ and } (1 = k))$ 表示在将在后面被描述的时间滤波处理(内模式)中帧(k)参照其自身执行时间滤波。

[0117] 利用星算法的编码和解码处理将在下面被描述。

[0118] 编码处理操作步骤:

[0119] (1) 将 GOP 的第一帧编码为 I 帧。

[0120] (2) 对于下一时间级上的帧,利用满足等式 (1) 的可能的参考帧进行运动预测并且对预测间残留(inter-prediction residuals)进行编码。在相同的时间级上,帧以从左到右的顺序(以从最低到最高帧指数的顺序)被编码。

[0121] (3) 重复操作步骤(2)直到全部帧被编码,并且其后编码下一 GOP 直到对全部帧的编码被完成。

[0122] 解码处理操作步骤:

[0123] (1) 解码 GOP 的第一帧

[0124] (2) 利用已被解码的帧中的合适的参考帧解码下一时间级上的帧。在相同的时间级上,帧以从左到右的顺序(以从最低到最高帧指数的顺序)被解码。

[0125] (3) 重复操作步骤(2)直到全部帧被解码,并且其后解码下一 GOP 直到对全部帧的解码被完成。

[0126] 如图 5 中所示,标记 I 的帧被内编码(intra-coded)(不参照其它帧),并且标记 H 的帧在高频子带内。高频子带表示帧参照一个或多个帧而被编码。

[0127] 在图 5 中,在 GOP 的大小为 8 时,帧的时间级是用于说明性目的的 0、4、(2、6)、(1、3、5、7) 的顺序。另外,即使在其以 1、5、(3、7)、(0、2、4、6) 的顺序的情况下,在编码侧时间可伸缩性和解码侧时间可伸缩性方面也几乎没有问题。同样地,2、6、(0、4)、(1、3、5、7) 的顺序可被容许。换句话说,可满足编码侧时间可伸缩性和解码侧时间可伸缩性的时间级上的任何帧可被容许。

[0128] 然而,当帧以 0、5、(2、6)、(1、3、4、7) 的时间级顺序的情况下,这能满足编码侧时

间可伸缩性和解码侧时间可伸缩性,但是因为帧间的间隔是奇数所以它不是优选的。

[0129] 图 6 显示在星算法中的帧之间的众多可能的联系。

[0130] 参照图 6,用于时间滤波的帧之间的可能的联系的例子将被描述。

[0131] 根据等式 (1),帧 $F(k)$ 可涉及很多帧。该特性允许算法适于使用多个参考帧。在此示例性实施例中, GOP 的大小为 8 时的帧之间的可能的联系被示出。从帧开始并且返回该帧(被称作自循环帧)的弧表示帧在内模式下被预测。具有包括在相同时间级中的 H 帧位置上的指数的之前编码的指数的全部原始帧可被用作参考帧,而仅仅在相同时间级中的帧之中的 H 帧位置上的原始帧之中的 L 或 A 帧可在大多数传统方法中被涉及。此特征构成本发明和传统方法之间的差别。例如,在传统方法中, $F(5)$ 可涉及 $F(3)$ 和 $F(1)$,但是 $F(5)$ 不能涉及 $F(0)$ 、 $F(2)$ 和 $F(4)$ 。

[0132] 尽管多个参考帧极大地增加了时间滤波和处理延迟所需的内存,支持此特征的可能性还是有价值的。

[0133] 如上所述,假设一个 GOP 内的最高时间级上的帧通过例子的方法具有最低帧指数。另外,应该注意最高时间级上的帧可具有另一帧指数。

[0134] 为了方便,在实验的结果中,用于对帧进行编码的参考帧的数量将被限制在用于双向预测的两个和用于单向预测的一个。

[0135] 图 7 显示根据本发明的示例性实施例的 GOP 之间的参照。在图 7 中,示出使用双向预测和交叉 GOP 优化的星编码算法。

[0136] 星算法可参照其它 GOP 的帧对帧进行编码,其被称作交叉 GOP 优化。因为 UMCTF 和星编码算法使用未被时间滤波的 A 或 I 帧,所以 UMCTF 也支持交叉 GOP 优化。参照图 5 和 6,指数 7 的帧中的预测误差是指数 0、4 和 6 的帧的预测误差之和。然而,如果指数 7 的帧涉及下一 GOP 的指数 0 的帧(如果以当前 GOP 计算则指的是指数 8 的帧),则此预测误差的漂移可被显著降低。而且,因为下一 GOP 的指数 0 的帧被帧内编码,所以指数 7 的帧的质量可被显著改善。

[0137] 图 8 显示根据本发明另一示例性实施例的非二元时间滤波中的帧之间的可能的联系。

[0138] 与 UMCTF 编码算法通过任意地插入 A 帧来容易地支持非二元时间分解一样,星算法也可通过以简单的方式改变图形结构来支持非二元时间分解。此示例性实施例示出支持 $1/3$ 和 $1/6$ 时间分解的星分解的例子。通过改变图形结构,全部帧速率的任何片段可被容易地获得。

[0139] 星算法规定编码侧和解码侧在时间级上具有相同的处理顺序,并且支持如上所述的多个参考帧和交叉 GOP 优化。通过传统方法,这些特征中的一些可以以限定的方式实现,但是很容易用传统方法控制端到端延迟。按照惯例,减小 GOP 的大小但是会导致性能显著下降的方法已被用于降低延迟。在使用星算法时,通过引入端到端延迟控制参数 (D) 的概念,直到帧在视频序列中再次通过编码和解码处理之后被恢复到视频序列的端到端延迟可被容易地控制。

[0140] 参照图 9 到 12,在延迟被限定时星算法将被考虑。

[0141] 等式 (1) 中的时间可伸缩性条件应该为延迟控制而被稍做改变,其将由等式 (2) 来定义。

[0142] $R_k^D = \{F(1) \mid (T(1) > T(k)) \text{ and } ((1-k) \leq D) \text{ or } ((T(1) = T(k)) \text{ and } (1 \leq k))\}$
(2)

[0143] 其中,当容许的延迟被定义为 D 时, R_k^D 代表当前编码的帧可涉及的参考帧的集合。为了解释等式 (2),最高时间级上的每一帧不能总是参考帧。尤其是,参考帧的指数和当前编码的帧的指数之间的差不超过 D。在这点上,应该注意 D 表示容许编码 F(k) 的最大端到端延迟。参照图 7,需要指数 4 的帧以编码指数 2 的帧,并且因此,如果 D 等于 2 则它是足够的。然而,应该注意需要指数 2 的帧编码指数 1 的帧,并且需要指数 4 的帧编码指数 2 的帧,并且因此,D 等于 3。如果指数 1 的帧不涉及指数 2 的帧,并且指数 5 的帧不涉及指数 6 的帧,如果 D 等于 2 则它是足够的。总而言之,为了对图 7 中所示的结构进行编码,则 D 应被设定为 3。

[0144] 还应该注意到,甚至在等式 (2) 的情况下,如上所述的多个参考帧或交叉 GOP 优化也可被应用。此特征对于以直接的和简单的方式实现延迟控制的情况是有利的。

[0145] 星算法方法的优点在于不需要牺牲时间可伸缩性。减小 GOP 大小的传统方法引起了最大时间级大小的减小,从而解码器侧的时间可伸缩性被严重限制。例如,在 GOP 的大小是 8 时,能够被解码器侧选择的帧速率被限制在 1、1/2、1/4 和 1/8。如果 GOP 的大小被设定为 4 以定义 D 为 3,则可选的帧速率仅为 1、1/2 和 1/4。当 GOP 的大小为 2 时,可选的帧速率仅为 1 和 1/2。如上所述,GOP 大小的减小具有视频编码效率被严重降低的缺点。相反的,即使当 D 通过使用星算法将 D 严格限制为 0 时,解码器侧的时间可伸缩性不会被影响,即,仅对编码侧的可伸缩性有损。即,当 GOP 的大小为 8 并且 D 为 0 时,如果在该数量的帧中基于 GOP 的编码器侧的处理能力被限制为 2,则指数 0 和 1 的帧应被编码并且随后被传输到解码器侧。在此情况下,解码器侧可恢复具有 1/4 帧速率的视频序列,但是被恢复的视频帧的时间间隔是奇数。

[0146] 参照图 9 到 12,不同的端到端延迟的例子将被描述。

[0147] 图 9 显示根据本发明的另一示例性实施例在延迟控制参数被设定为 0 的时间滤波中的帧之间的可能的联系。

[0148] 在图 9 中,支持双向预测和交叉 GOP 优化的星算法的时间结构,当 D 的值被设定为 0 时的延迟的限制被显示。因为延迟控制参数是 0,所以交叉 GOP 优化被自动减活,并且全部帧仅涉及时间上在后面的帧(即,较低指数的帧)。因此,帧以 0、1、2、3、4、5、6 和 7 的顺序来传输。即,一帧被处理,并且被立即参数到解码器侧。在此情况下,延迟仅对 I 帧缓冲存在。此特征也被保持在解码器侧,其解码器可在帧一到达时即开始解码。最终延迟仅为包括解码器侧操作延迟的两帧(在 30Hz 下 67ms)。但是在此情况下,与将 D 的值设定为大于 0 相比较,性能有所下降。

[0149] 图 10 显示根据本发明的示例性实施例的在延迟控制参数为 1 的时间滤波中的帧之间的可能的联系。

[0150] 在此示例性实施例中,交叉 GOP 优化的特性被自动激活。通过使用双向预测,最低时间级上的全部帧可被预测,并且 GOP 的最末帧可涉及下一 GOP 的第一帧。在此情况下,帧的编码顺序为 0、2、1、4、3、6、5、7 和 8(下一帧的 0)。仅需要在编码器侧缓冲两帧的延迟和在解码器侧操作的延迟。总延迟时间是 3 帧(在 30Hz 下 100ms),并且双向预测被应用于大多数帧并且交叉 GOP 优化被用于最末帧。

[0151] 图 11 显示根据本发明的示例性实施例的在延迟时间控制参数为 3 的时间滤波中的帧之间的可能的联系。

[0152] 如图 11 所示,如果 D 为 3,指数 2 的帧可涉及指数 4 的帧,并且指数 6 的帧可涉及下一 GOP 的第一帧。

[0153] D 为 3 而不为 2 的原因是因为两帧的延迟是足够的,这是由于指数 4 的帧对于编码指数 2 的帧是必需的,但是指数 2 的帧对于编码指数 1 的帧是必需的,并且指数 2 的帧需要两帧的延迟,因此需要三帧作为总的延迟时间。当延迟时间为 3 时,除了指数 4 的帧参照指数 8 的帧(下一帧的 0 帧),参照全部帧是可能的。此时的编码顺序为 0、4、2、1、3、8(下一 GOP 的 0 帧)、6、5 和 7。如果 D 为 4,则图 7 中示出的结构是可能的。图 12 中显示了 GOP 大小扩展到 16。

[0154] 图 12 显示根据本发明的另一示例性实施例的当 GOP 大小为 16 时在延迟时间控制参数为 3 的时间滤波中的帧之间的可能的联系。在此情况下,帧的编码顺序(与传输顺序相同)是 0、4、2、1、3、8、6、5、7、12、10、9、11、16(下一 GOP 的 0 帧)、14、13 和 15。应该注意星算法规定总的端到端延迟可仅被参数 D 控制。此特征使延迟控制简单化,因此根据总的端到端延迟导致编码效率的适度降低。因为根据没有重大改变而在编码系统上的任何应用的特点总的端到端延迟可被容易地控制,所以此一个架构中的灵活的延迟是很有用的。在单向视频流中,总的端到端延迟不重要。因此,D 的值可被设定以具有最大值(例如 GOP 大小的 1/2)。相反地,在双向视频会议中,总的端到端延迟被认为是非常重要的。在此情况下,如果总的端到端延迟被设定为小于 2,即使当编码效率稍稍降低,总的端到端延迟也可以很短。表 1 中显示总的端到端延迟和延迟参数 D 的值之间的关系。

[0155] 表 1

[0156]

GOP 大小 = 8	
D 值	总的端到端延迟

[0157]

0	2 帧 (30Hz 下 67ms)
1	3 帧 (30Hz 下 100ms)
2	5 帧 (30Hz 下 167ms)
4	9 帧 (30Hz 下 300ms)
GOP 大小 = 16	
D 值	总的端到端延迟
0	2 帧 (30Hz 下 67ms)
1	3 帧 (30Hz 下 100ms)
2	5 帧 (30Hz 下 167ms)
4	9 帧 (30Hz 下 300ms)
8	17 帧 (30Hz 下 567ms)

[0158] 总的端到端延迟 T 可被简单地表示为等式 (3)。

[0159] $T = \min(2, 2D+1)$ (3)

[0160] 其中, T 表示总的端到端延迟的值, 其单位为一帧时间。关于由于总的端到端延迟导致的 PSNR 降低的实验的结果将在下面描述。

[0161] 图 13 显示前向、后向、双向和宏块内预测模式。

[0162] 基本上, 星算法支持多模式时间预测。如图 13 中所示, 前向预测模式 (1)、后向预测模式 (2)、双向预测模式 (3) 和宏块内预测模式 (4) 中的预测被支持。在传统方法下, 前三种模式已经连同可伸缩视频编码一起而被支持, 但是星算法通过将内预测模式加入其中来改善快速改变的视频序列的编码效率。

[0163] 以下将首先描述由宏块内预测模式的确定。

[0164] 因为星算法允许双向预测和多参考帧, 所以前向、后向和双向预测可被容易地实现。尽管公知的 HVBSM 算法可被用于此设定, 但是本示例性实施例将受固定的块大小运动预测方案的限定。令 $E(k, -1)$ 为具有前向预测的第 k 个绝对差之和 (SAD), 并且 $B(k, -1)$ 为量化前向预测的运动矢量的总的运动比特。类似地, 假定 $E(k, +1)$ 为具有后向预测模式的第 k 个 SAD, $B(k, +1)$ 为被赋值以量化后向预测模式的运动矢量的总的运动比特, $E(k, *)$ 为具有双向预测模式的第 k 个 SAD, 并且 $B(k, *)$ 为被赋值以量化双向预测模式的运动矢量的总的运动比特。用于前向、后向和双向预测模式的开销可被描述为等式 (4)

[0165] $C_f = E(k, -1) + \lambda B(k, -1),$

[0166] $C_b = E(k, 1) + \lambda B(k, 1),$ 和

[0167] $C_{bi} = E(k, *) + \lambda \{B(k, -1) + B(k, 1)\}$ (4)

[0168] 其中, C_f 、 C_b 和 C_{bi} 分别指的是用于前向、后向和双向预测模式的开销。

[0169] λ 是用于控制运动和结构（图像）比特之间的平衡的拉格朗日（Lagrangian）系数。因为可伸缩视频编码器不能知道最终的比特率， λ 应该相对于主要在目的应用中使用的视频序列和比特率的性质而被优化。通过计算如在等式（4）中定义的最小开销，可以确定最佳优化的宏块内预测模式。

[0170] 接下来，下面将描述由宏块内预测模式的确定。

[0171] 在一些视频序列中，场景改变得很快。在极端情况下，可观察到绝对不具有与邻近帧间的时间冗余的一帧。为了克服此缺点，通过 MC-EZBC 实现的编码方法支持作为“适于 GOP 大小”的特征。当不相关联的像素的数量大于预定阈值（典型的是总像素的 30%）时，此对于 GOP 大小的适应停止时间分解并且将帧编码为 I 帧。此方法可被应用于星算法，但是从在标准混合编码器中使用的宏块内模式的概念借鉴的更灵活的方法被用在此示例性实施例中。通常，由于预测漂移，包括用星算法的编解码器的开环编解码器不能使用关于相邻宏块的信息，而混合编解码器可使用多宏块内预测模式。因此，在用于内预测模式的此示例性实施例中采用 DC 预测。在此模式下，宏块被用于 Y、U 和 V 分量的其 DC 值内预测。如果用于内预测模式的开销小于上述间预测模式下的最佳开销，则可选择内预测模式。在此情况下，原始像素和 DC 值之间的差和代替运动矢量的三个 DC 值之间的差被编码。用于内预测模式的开销可被定义为等式（5）。

$$[0172] \quad C_i = E(k, 0) + \lambda B(k, 0), \quad (5)$$

[0173] 其中， $E(k, 0)$ 为 SAD（原始亮度值和在第 k 内预测上的 DC 值之间的差）并且 $B(k, 0)$ 为用于编码三个 DC 值的总的比特。

[0174] 如果 C_i 小于由等式（5）计算的值，则编码在宏块内预测模式下被实现。一句话，如果帧中的全部宏块在仅使用一组 DC 值的宏块内预测模式下被编码，则该帧可被编码为 I 帧。同时，如果用户想要在视频序列中看任一点则具有视频序列中的很多 I 帧或者自动编辑视频是我们所期望的。在此情况下，用 I 帧代替 H 帧是好的解决方案。

[0175] 在不是帧中的全部宏块被以内预测模式编码时，如果在内预测模式下超过其预定百分比（例如，90%）被编码，则帧可被编码为 I 帧。通过这样做，视频中的 I 帧增加，并且作为结果，对用户来说看视频中的任一点或者编辑视频是较容易的。

[0176] 尽管星算法提供实现多模式时间预测的简单的方法，还是应该注意 MC-EZBC 使用的其它方法和其它编解码器也可被使用。除第一帧之外的全部剩余帧中的全部宏块可以以上述四种类型中的任一类型而被编码，所述四种类型是：前向预测、后向预测、双向预测和宏块内预测。本发明所属的本领域普通技术人员可以理解在星算法中的上述图的“H 帧”指的是间预测的和内预测的宏块的混合形式。而且，可以知道位于 H 帧的帧被改变为 I 帧并且随后被编码。这种灵活性在快速改变的视频序列和淡入 / 淡出的帧中是很有用的。

[0177] 图 14 显示根据本发明的另一示例性实施例的在时间滤波中的 4 种预测模式下的帧之间的可能的联系。

[0178] 在图 14 中，“I+H”意味着该帧包括间预测的宏块和内预测的宏块二者，并且“I”意味着完整的帧通过不用间预测的它们自身而被编码。尽管内预测可被用在 GOP 的开始帧（在最高时间级上）中，但是因为它不像原始帧的小波变换那样有效率，所以它未被用在图 14 中。

[0179] 图 15a 和 15b 显示具有多种模式的快速改变的视频序列和慢速或很少改变的视频

序列中的视频编码的例子,其中,“百分比”表示预测模式的速率。即,I表示帧(但是GOP的第一帧不使用内预测)中的内预测块的速率,BI表示双向预测的速率,F表示前向预测的速率并且B表示后向预测的速率。

[0180] 参照图15a,因为指数1的帧非常类似于指数0的帧,所以F的百分比构成压倒性的多数,75%,并且因为其接近指数0和4的帧的中间级,在指数2的帧(即,使指数0的帧明亮的图像)中,BI构成压倒性的多数87%。因为指数4的帧不同于其它帧,所以编码的I块为100%。但是,因为指数4的帧非常不同于指数6的帧但是类似于指数6的帧,所以编码的B块在指数6的帧中为94%。

[0181] 参照图15b,全部的帧看上去大体类似,其中BI实际上在非常相似的帧中示出最佳性能。因此,BI的比率在图15b中通常是最高的。

[0182] 用于确定星算法的性能的几个模拟被实施,其中,星算法被应用于时间滤波处理。对于运动估计,具有来自于从4乘4到16乘16的大小的子块的多模式分割的公知的菱形快速搜索的变量被使用。为了性能比较,MC-EZBC被使用,并且EZBC算法还被用于实现本发明的嵌入式量化。

[0183] 福曼和移动CIF序列的前64帧被作为实验材料。本发明关注的是改善时间传输。因此,不进行空间可伸缩性的实验。两种材料都在足够的比特率下被编码,并且其比特流被截断以具有用于传输的2048kbps、1024kbps、512kbps、256kbps和128kbps并且在其后被解码。

[0184] 为了性能测量,具有加权值的PSNR被使用。加权的PSNR在等式(6)中被定义。

$$[0185] \quad \text{PSNR} = (4\text{PSNR}_Y + \text{PSNR}_U + \text{PSNR}_V) / 6 \quad (6)$$

[0186] 为了检测星算法,为了简单除了多参考帧,上述的全部特征被包括。最后,基于GOP级的恒定的比特分配被用于星算法,而动态比特分配被用于MC-EZBC算法。如果其被应用于星算法,则动态比特分配可显示较好的性能。

[0187] 图16和17是示出在视频编码方案中的福曼CIF序列和移动CIF序列的PSNR(峰值信噪比)结果的图表。

[0188] 30Hz的帧速率被用于2048kbps和1024kbps。15Hz的帧速率被用于512kbps和256kbps。7.5Hz的帧速率被用于128kbps。对于星算法,双向预测和交叉GOP优化二者都被使用,其中,16的GOP大小和四分之一像素运动精确度被用于两种编码器。另外,当全部其它部分没有为不考虑其它部分的性能而仅比较时间分解效率而改变时,采用双向预测的MCTF算法被用在为星算法而实施的编解码器中。这在图中被标记为用MCTF。这些图示出星算法的性能在福曼CIF序列中比MC-EZBC和MCTF方案提高了1dB。MCTF的性能几乎类似于MC-EZBC的性能。然而,在移动序列中星算法在性能上几乎等于MC-EZBC,但是它显示了比MCTF更好的性能。能够想到用在MC-EZBC中的其它重要的编码技术,诸如动态比特分配和可变大小块匹配技术,有不同,并且如果它们二者都被应用于星算法,则能期望比MC-EZBC更好的结果。星算法清楚地超过MCTF 3.5dB,从其明显可知星算法比MCTF好。我们能够得出星算法明显比MCTF好,并且在时间分解效率的方面与MC-EZBC编码器兼容的结论。

[0189] 为了比较低延迟编码模式的性能,不同的端到端延迟设定的几个实验被进行。对于星算法,延迟控制参数D被从0变到8,这对应于与2到16的GOP大小对应的67ms到567ms的端到端延迟。然而,为了测定不同的端到端延迟条件,从2048kbps到256kbps的比

特率被使用,但是空间可伸缩性未被使用。宏块内预测模式也未被使用,以便仅比较时间分解结构之间的差别。

[0190] 图 18 示出了与用于改变端到端延迟条件的福曼 CIF 序列的 567ms 的最大延迟设定比较的 PSNR 下降。如图 18 所示,PSNR 值在具有减小的 GOP 大小的 MC-EZBC 中显著下降。尤其是,在 2 的 GOP 大小的情况下的性能明显低于其它 GOP 大小。即使具有 4 的 GOP 大小,端到端延迟也超过 150ms。相反地,PSNR 下降在星算法中不严重。即使在最短端到端延迟(67ms)中,仅大约 1.3dB 的 PSNR 下降和在可接受的低延迟模式(100ms)下的 PSNR 下降仅为 0.8dB。在两种算法之间的最大 PSNR 值的减小中的最大差达到 3.6dB。

[0191] 图 19 示出与移动 CIF 序列的最大延迟设定相比的 PSNR 下降。在图 19 中,MC-EZBC 的 PSNR 下降比已被描述的福曼 CIF 序列的情况更严重。对于星算法,最大和最小端到端延迟之间的 PSNR 下降大约达到 2.3dB,并且在 MC-EZBC 编码器中大约达到 6.9dB。具有 100ms 的端到端延迟的 PSNR 下降仅为 1.7dB 并且 MC-EZBC 的 PSNR 下降达到 6.9dB。具有 100ms 的端到端延迟的两种算法之间的 PSNR 下降的最大差达到 5.1dB。而且,星算法支持甚至具有最短延迟设定的完全时间可伸缩性,而传统方法仅支持具有 2 的 GOP 大小的一级时间可伸缩性。PSNR 值之间的差别在表 2 中被描述。

[0192] 表 2

[0193]

30Hz 下的福曼 CIF

[0194]

MC-EZBC					
比特率/延迟	67ms	100ms	167ms	300ms	567ms
256		31.66	33.43	34.61	35.19
512		34.75	36.68	37.73	38.09
1024		37.88	39.77	40.59	40.80
2048		41.62	43.12	43.64	43.72
星					
256	34.97	35.23	35.43	35.67	35.94
512	37.80	38.23	38.55	38.82	39.06
1024	40.36	40.89	41.22	41.45	41.63
2048	43.02	43.57	43.86	44.04	44.14
30Hz 下的移动 CIF					
MC-EZBC					
比特率/延迟	67ms	100ms	167ms	300ms	567ms
256		22.21	23.39	24.64	26.08
512		24.08	25.99	28.33	30.28
1024		26.80	29.51	32.20	33.70
2048		30.58	33.93	36.10	36.98
星					
256	25.61	25.66	25.80	26.15	26.72
512	38.42	28.70	29.03	29.62	30.27
1024	31.46	31.94	32.44	33.16	33.68
2048	34.96	35.63	36.23	36.89	37.27

[0195] 将参照图 20 描述快速改变的视频序列之间的比较。

[0196] 图 20 是示出当通过使用四种预测模式对一部分来自于电影“Matrix 2”的快速改变的场景进行编码以及不使用该四种预测模式对其进行编码时的 PSNR 结果的图表。

[0197] 对于实验, 仅一个 16 帧的 GOP 被使用。选择的帧包括很快速的运动、场景改变、空白帧和淡入 / 淡出。星算法被应用于具有宏块内预测和不具有宏块内预测的情况, 并且 MC-EZBC 的结果被包括以用于比较。为了检验合适的 GOP 大小特征, 当 MC-EZBC 配置字段中的 “adapt_flag” 被打开和关闭时的结果被包括。

[0198] 如图 20 所示, 内预测的结果很出色。在具有和不具有内预测特征的星算法之间 PSNR 差超过 5dB, 并且在具有和不具有合适的 GOP 大小特征的 MC-EZBC 之间超过 10dB。另

外,具有宏块内预测的星算法的性能超过具有合适的 GOP 大小特征的 MC-EZBC 主要由于更灵活的基于宏块的内预测方法导致的 1.5dB 的清楚边缘。

[0199] 根据本发明,可控制端到端延迟并执行引起短的端到端延迟中不严重的性能下降的视频编码。另外,快速改变的视频序列可被更有效率地压缩。而且,延迟控制不影响视频序列的时间可伸缩性。

[0200] 尽管已经为了示例性的目的而公开了本发明的优选实施例,本领域普通技术人员将会理解,在不脱离所附权利要求所公开的本发明的范围和实质的情况下,可进行各种改变、增加和代替。

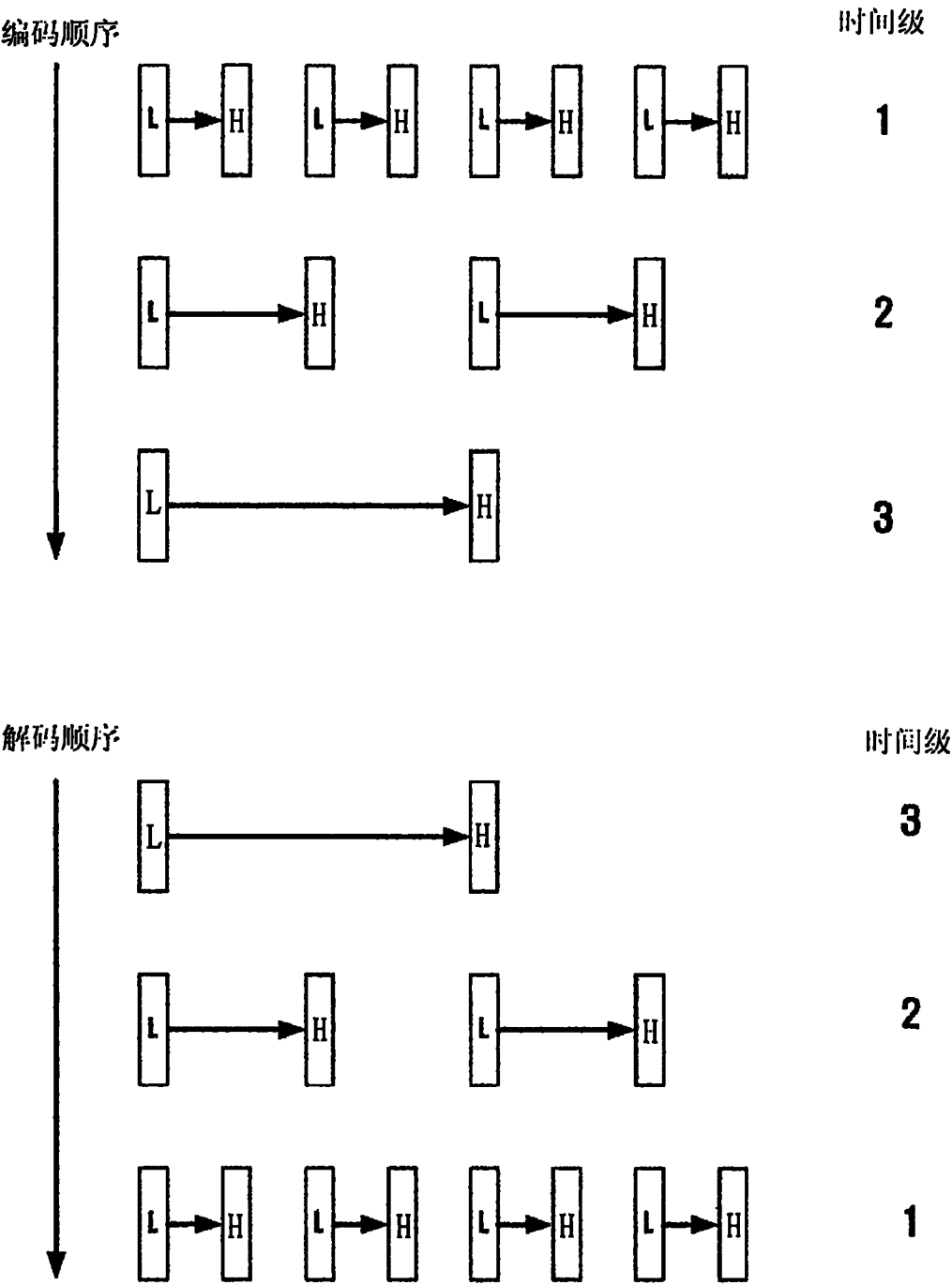


图 1a

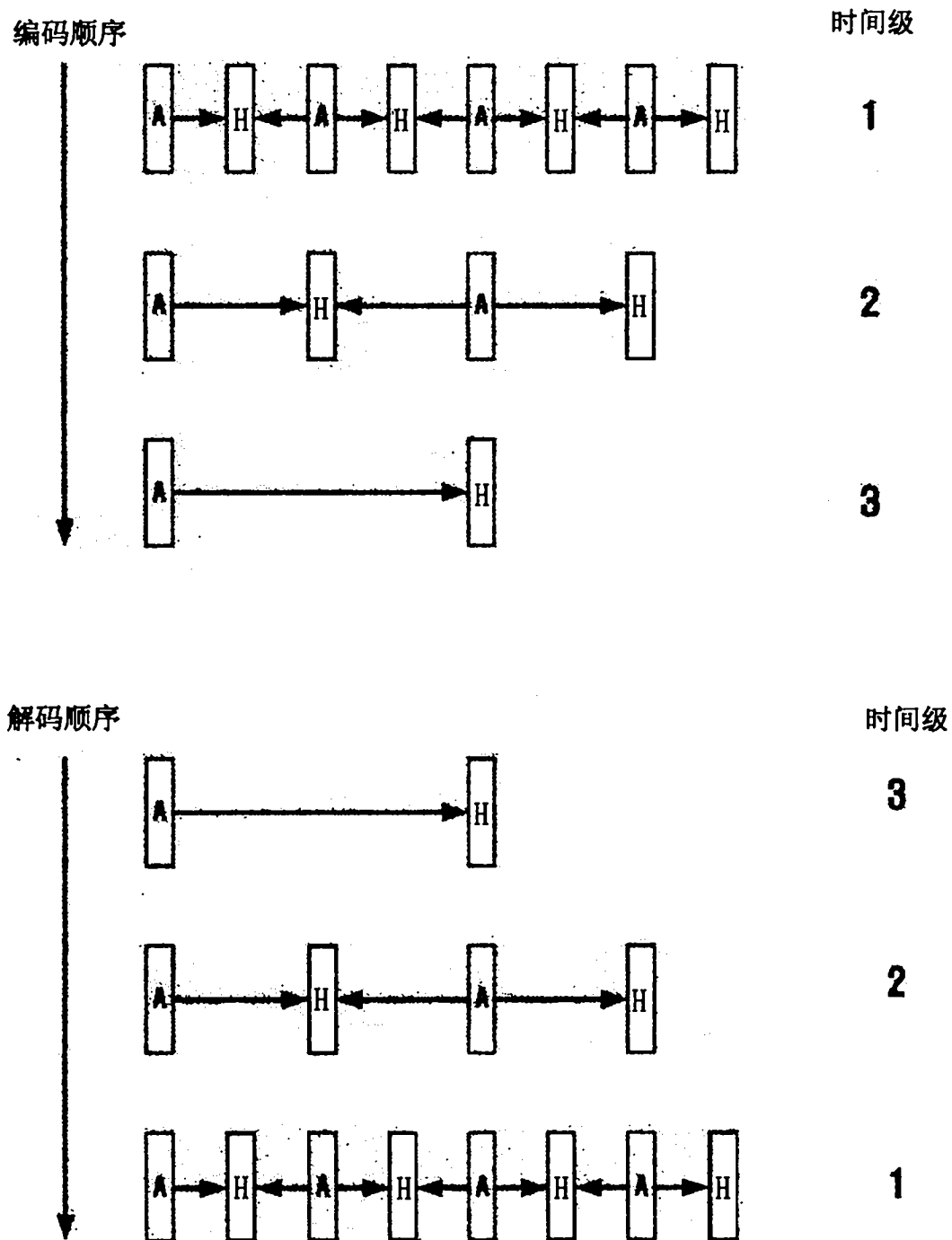


图 1b

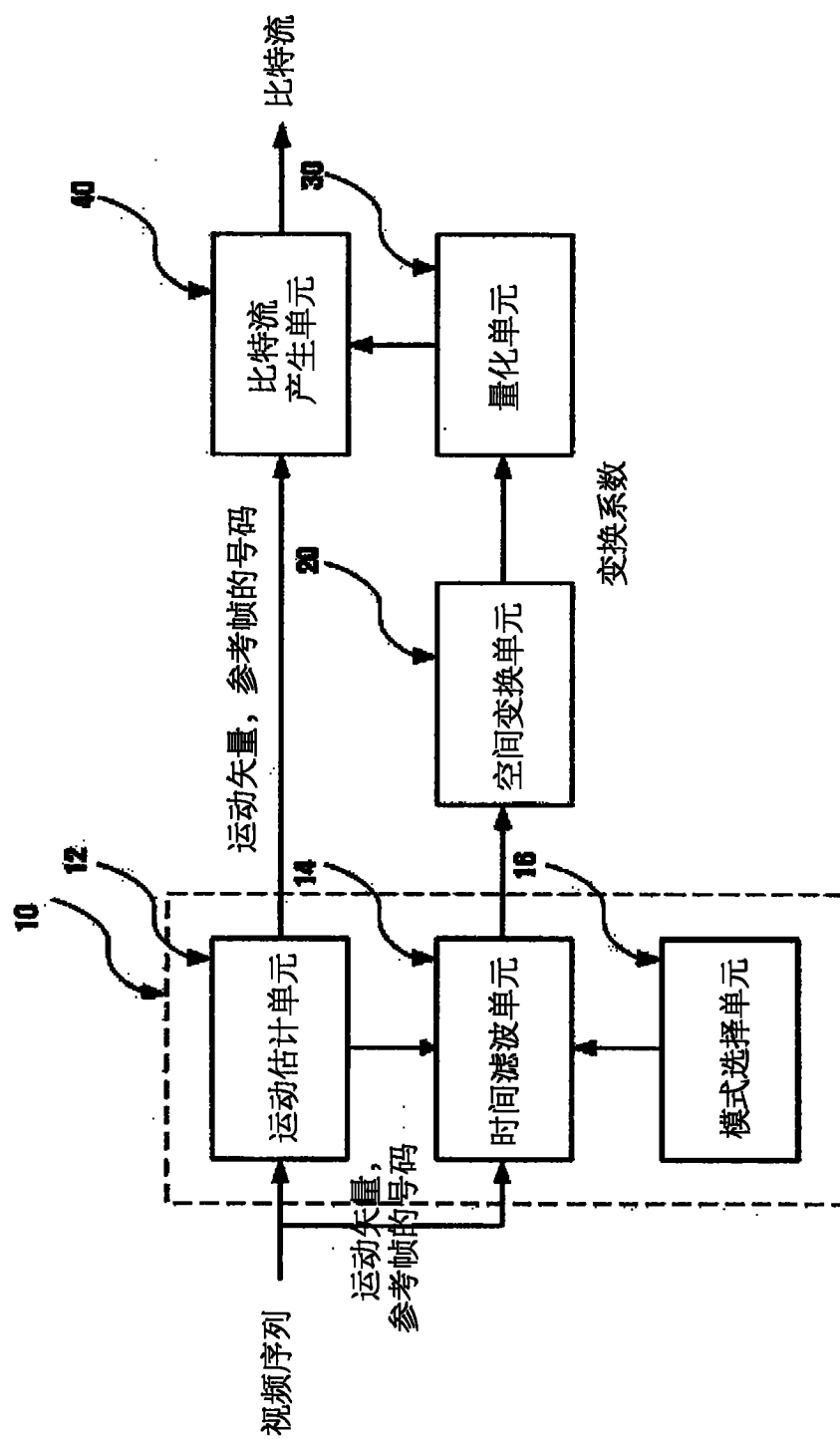


图 2

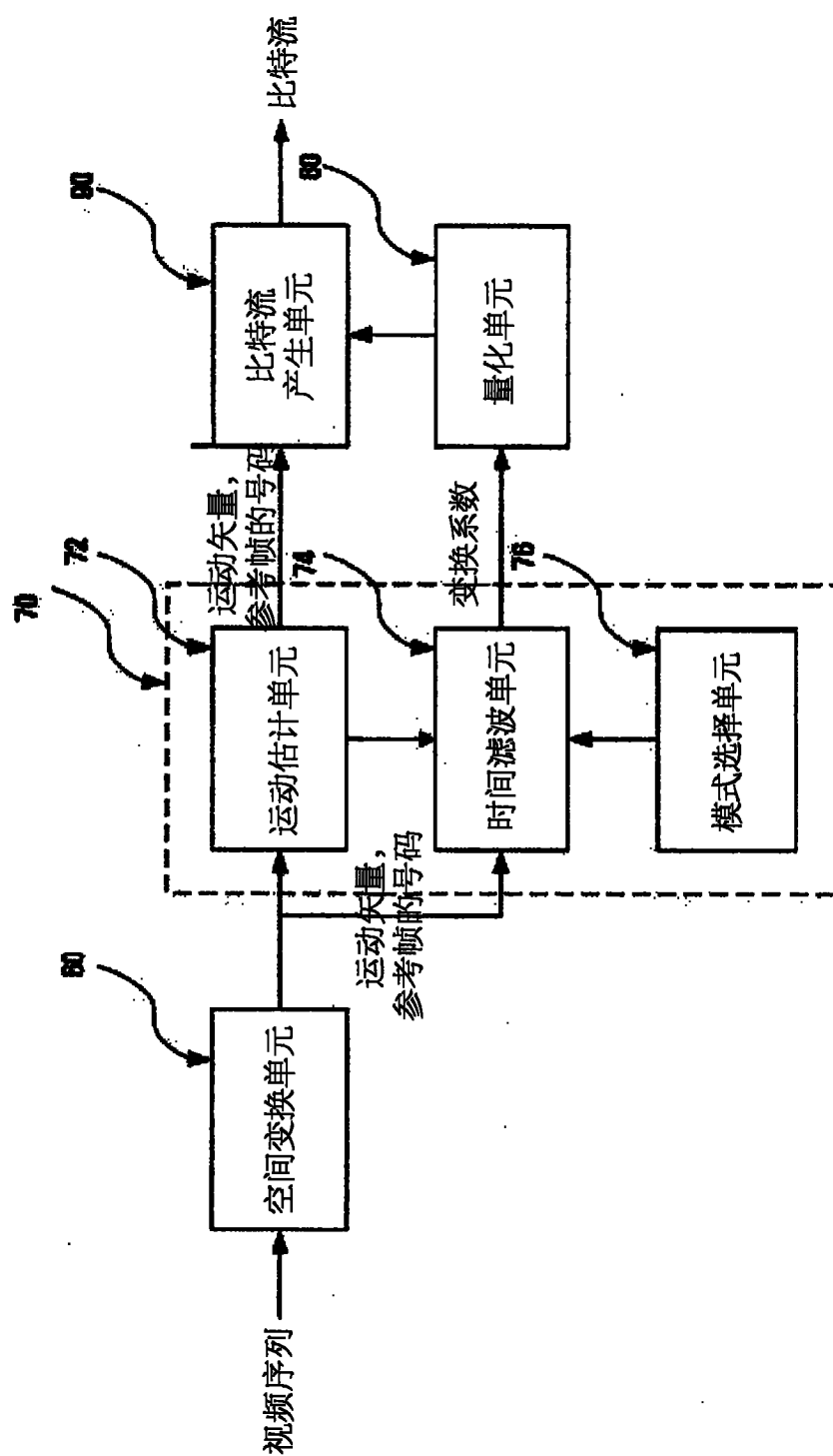


图 3

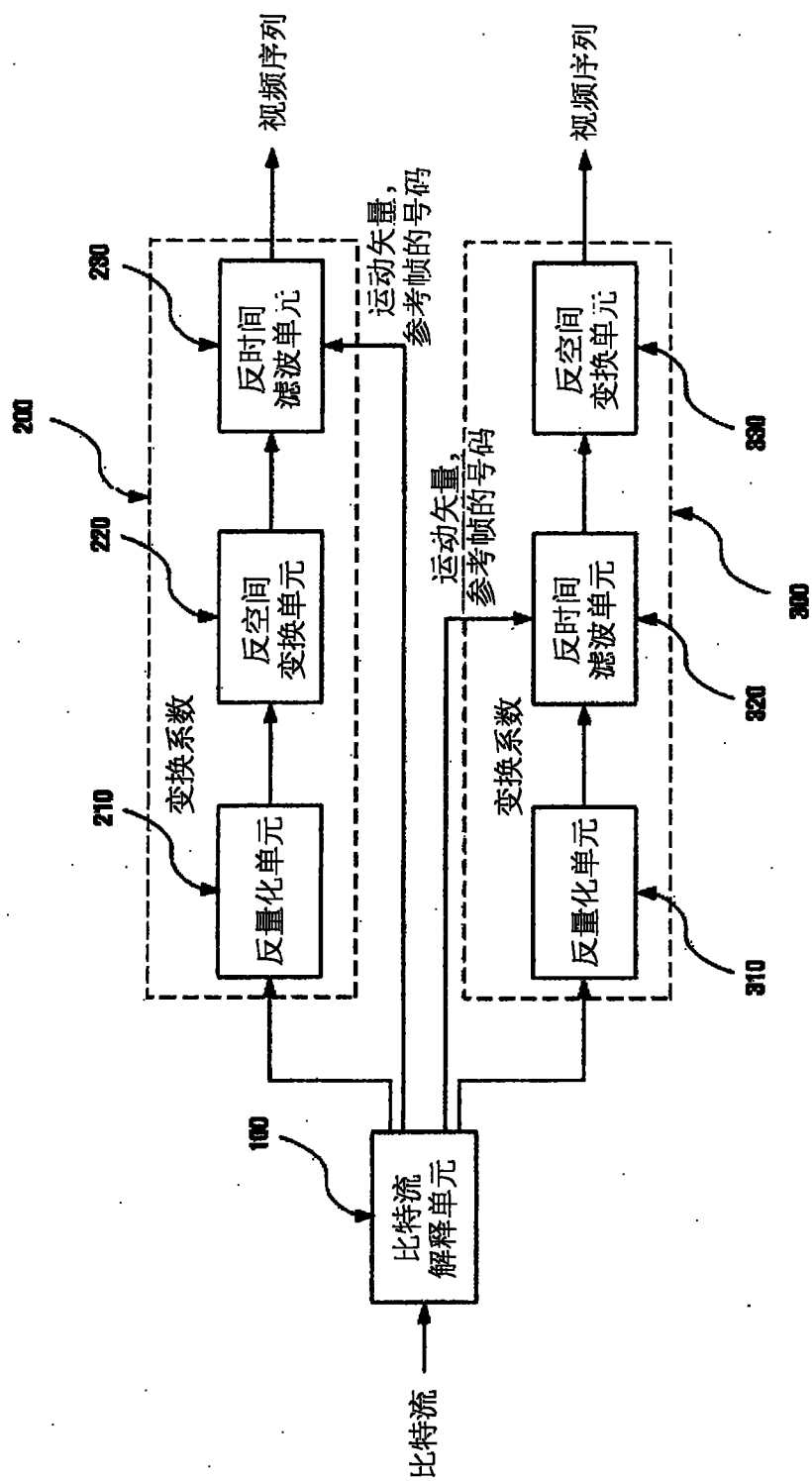


图 4

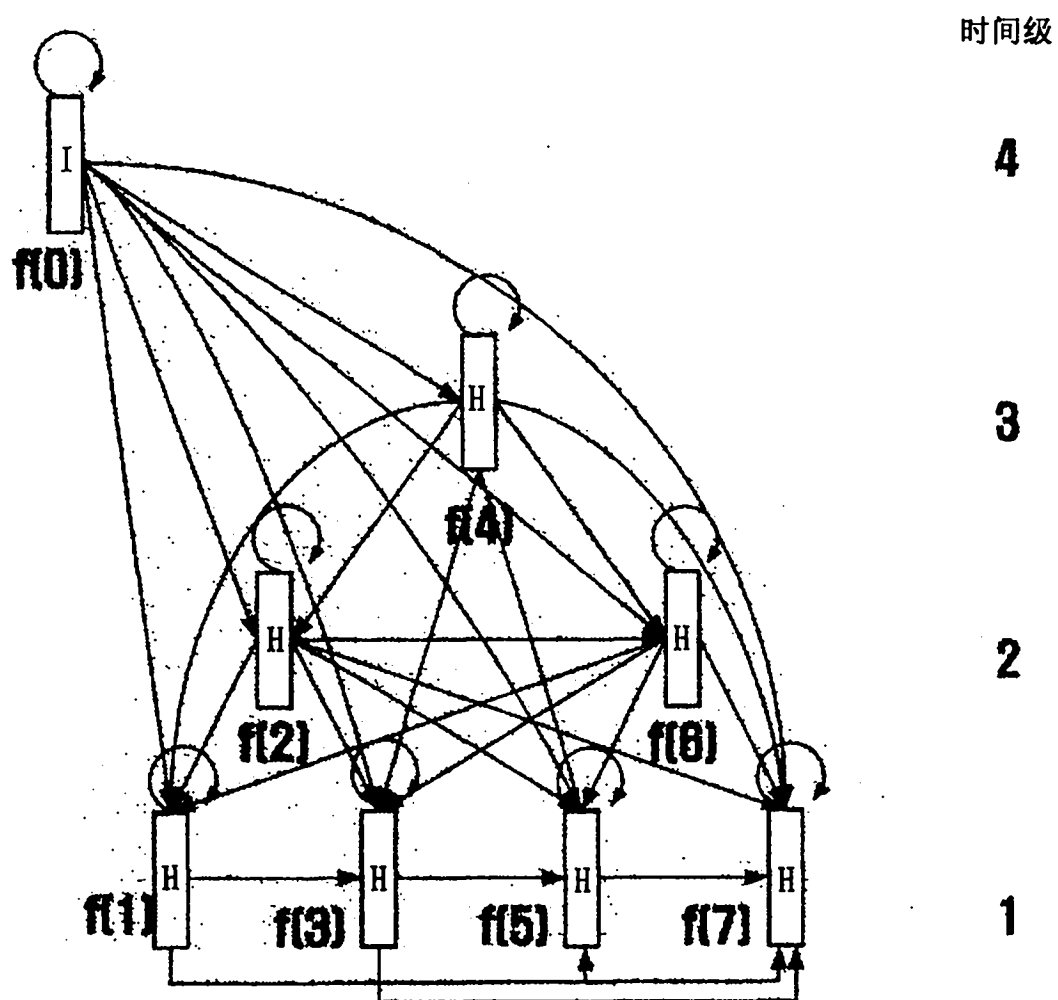


图 5

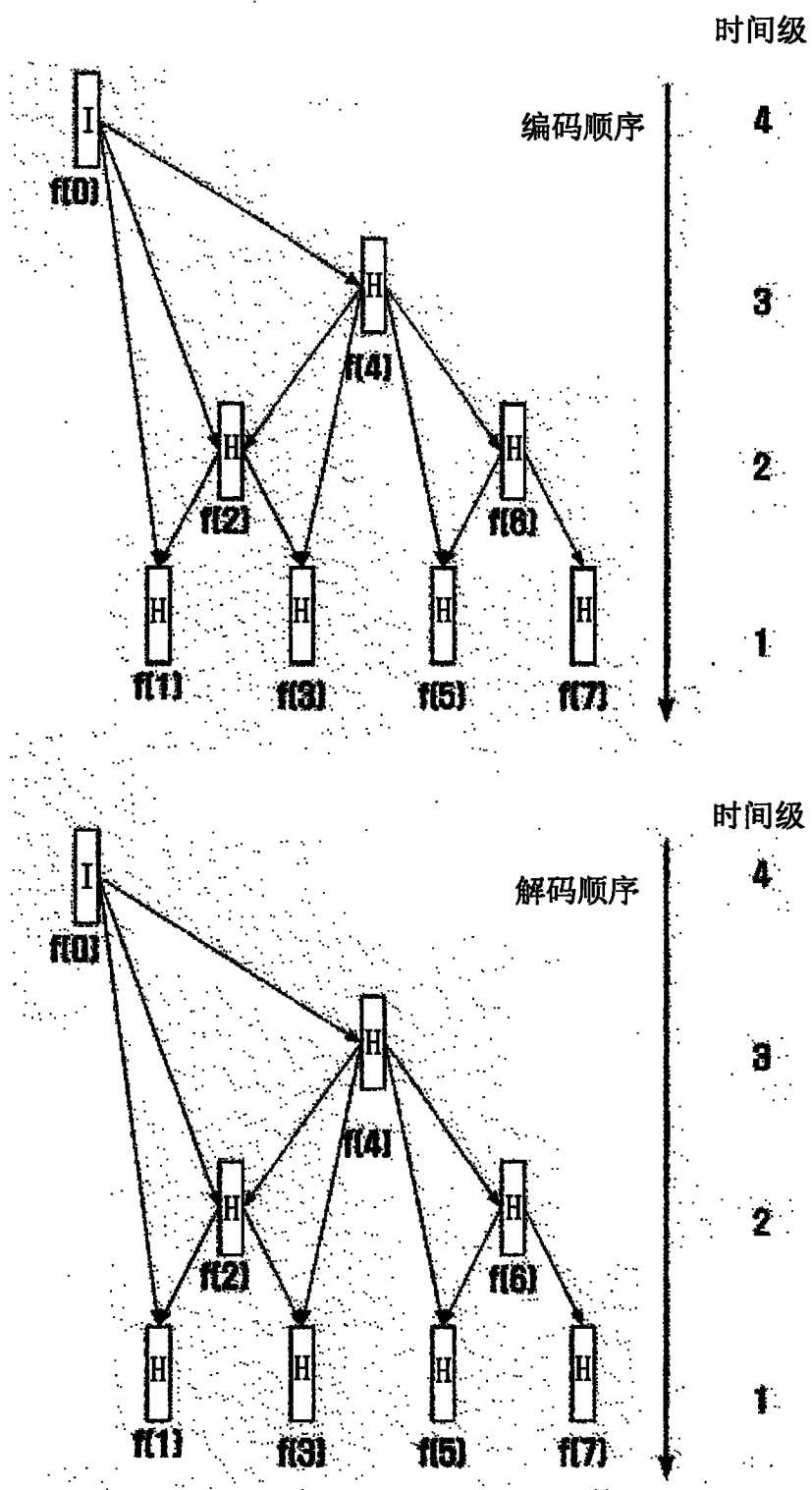


图 6

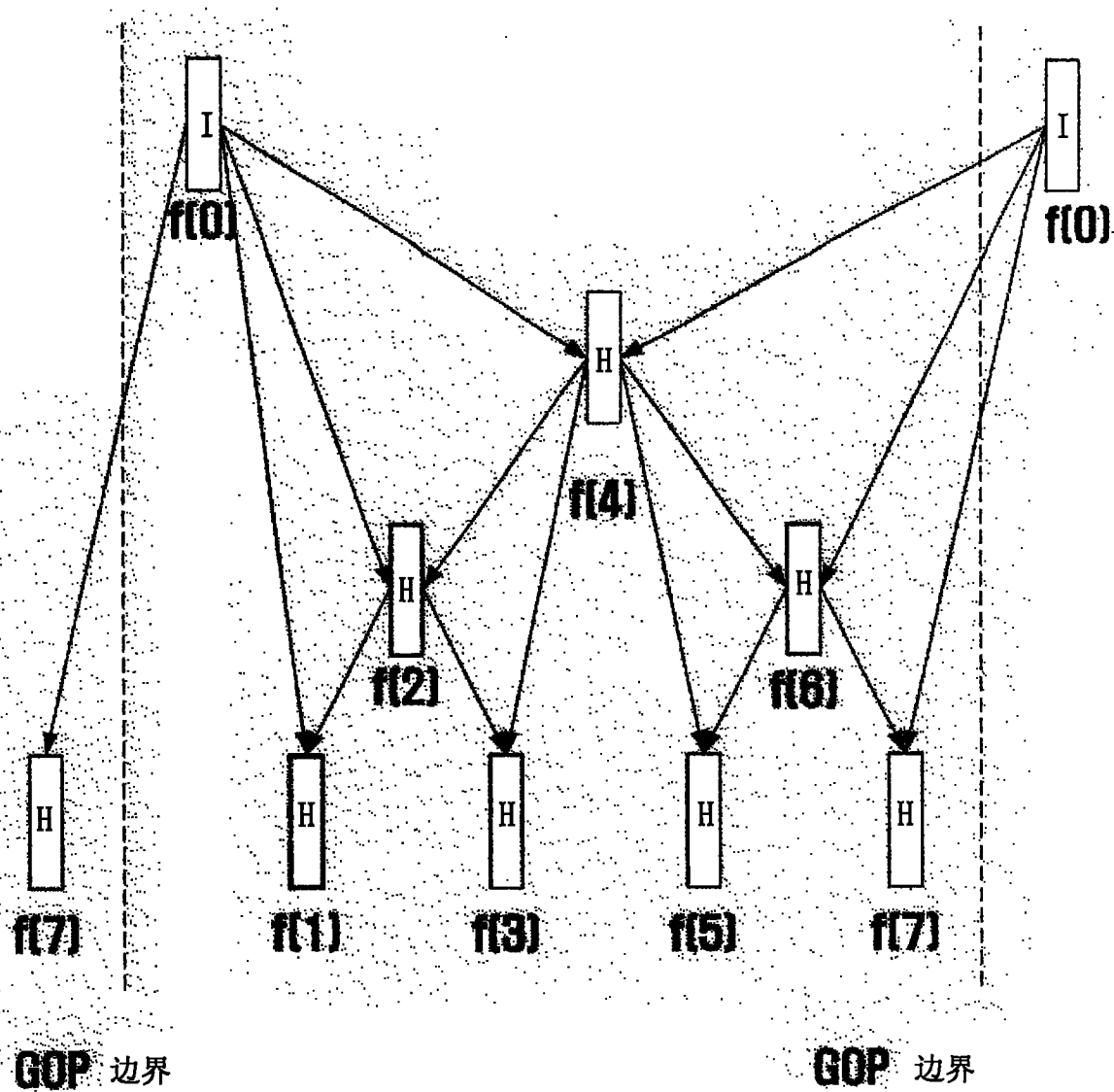


图 7

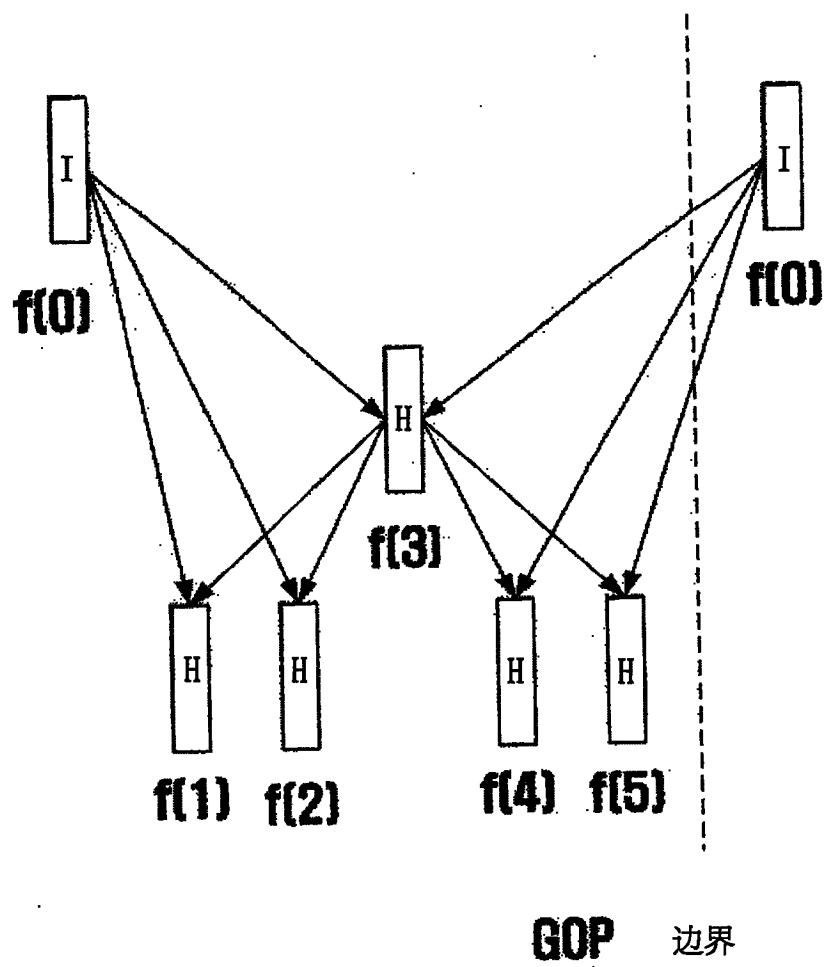


图 8

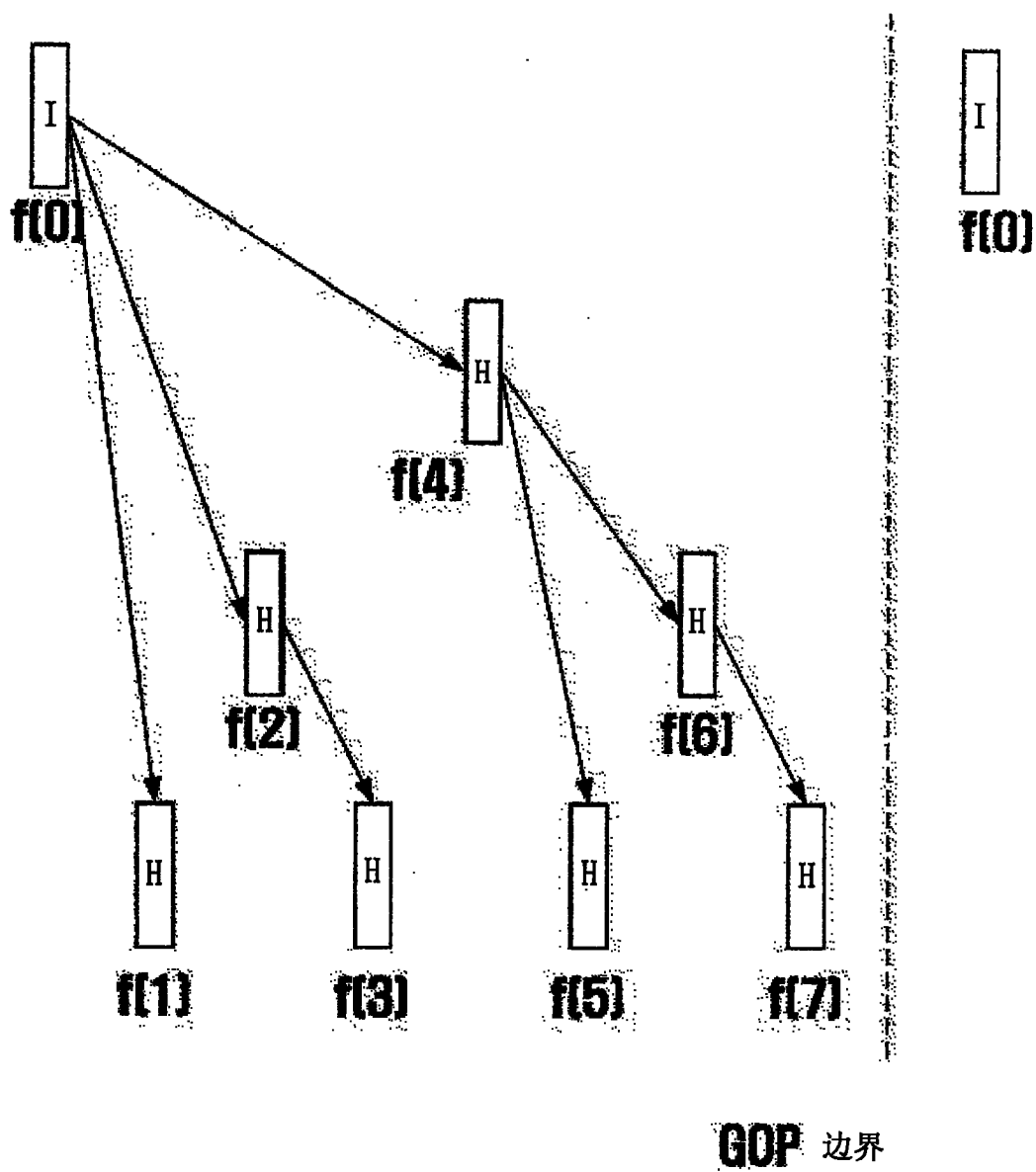


图 9

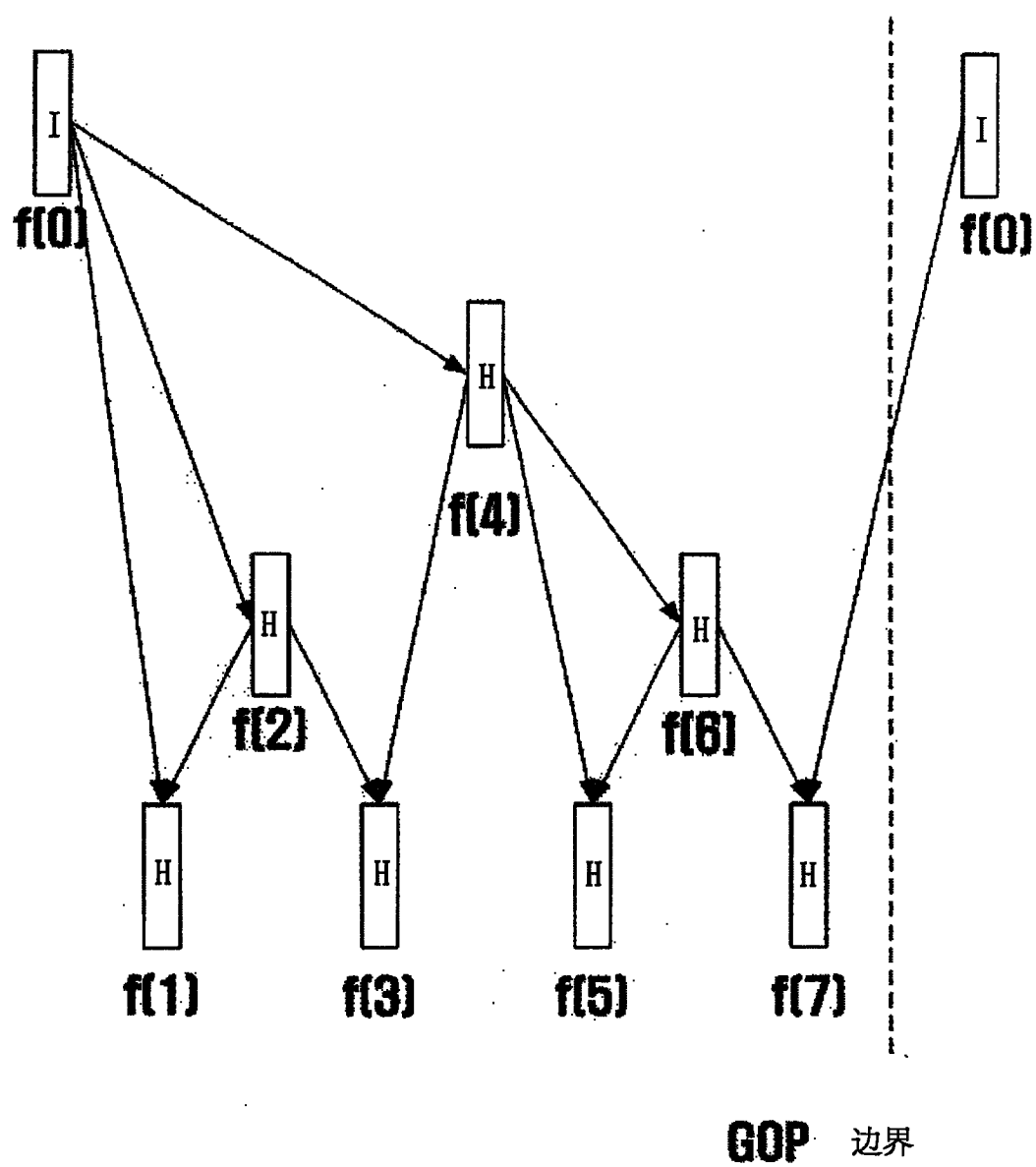


图 10

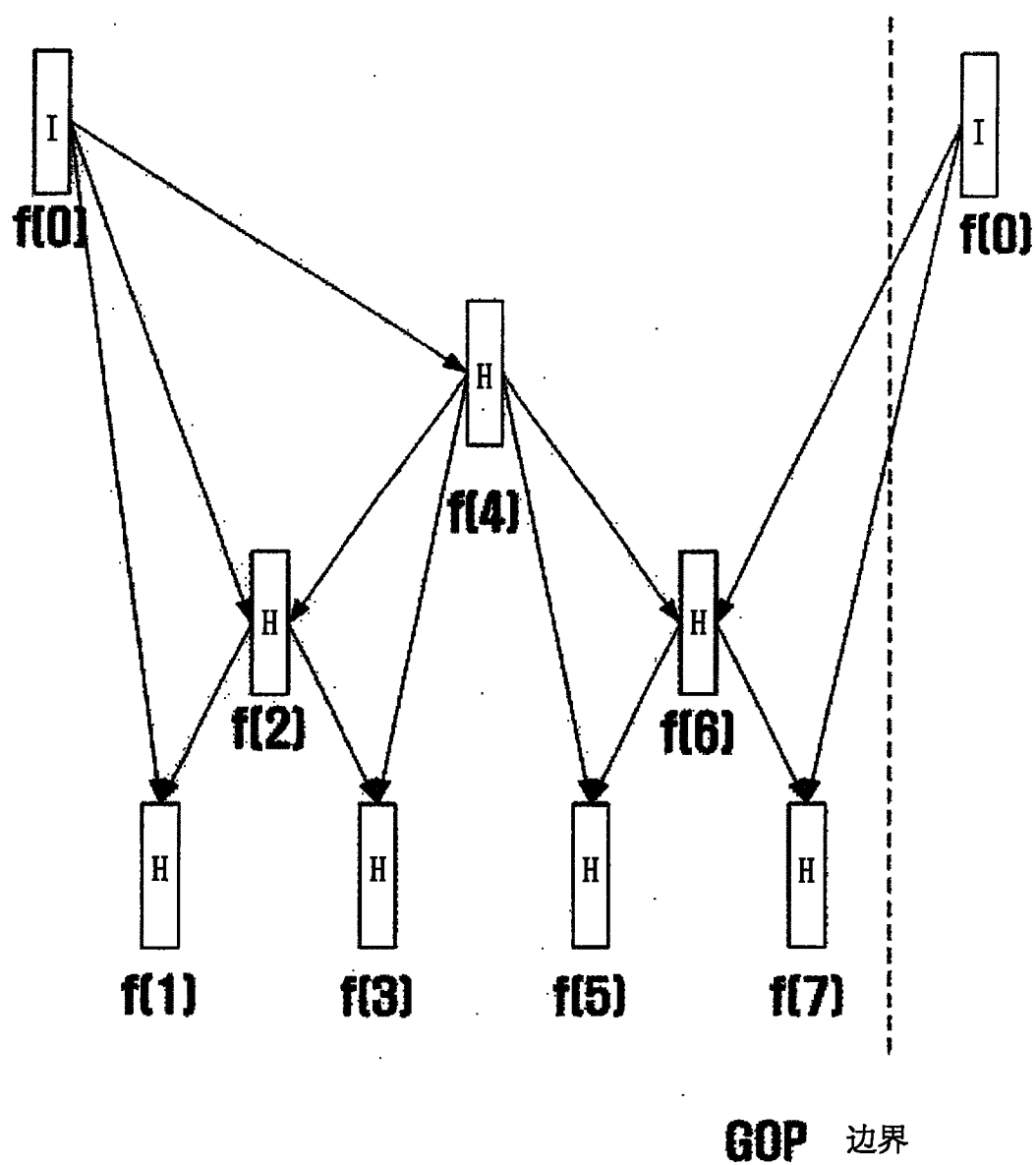


图 11

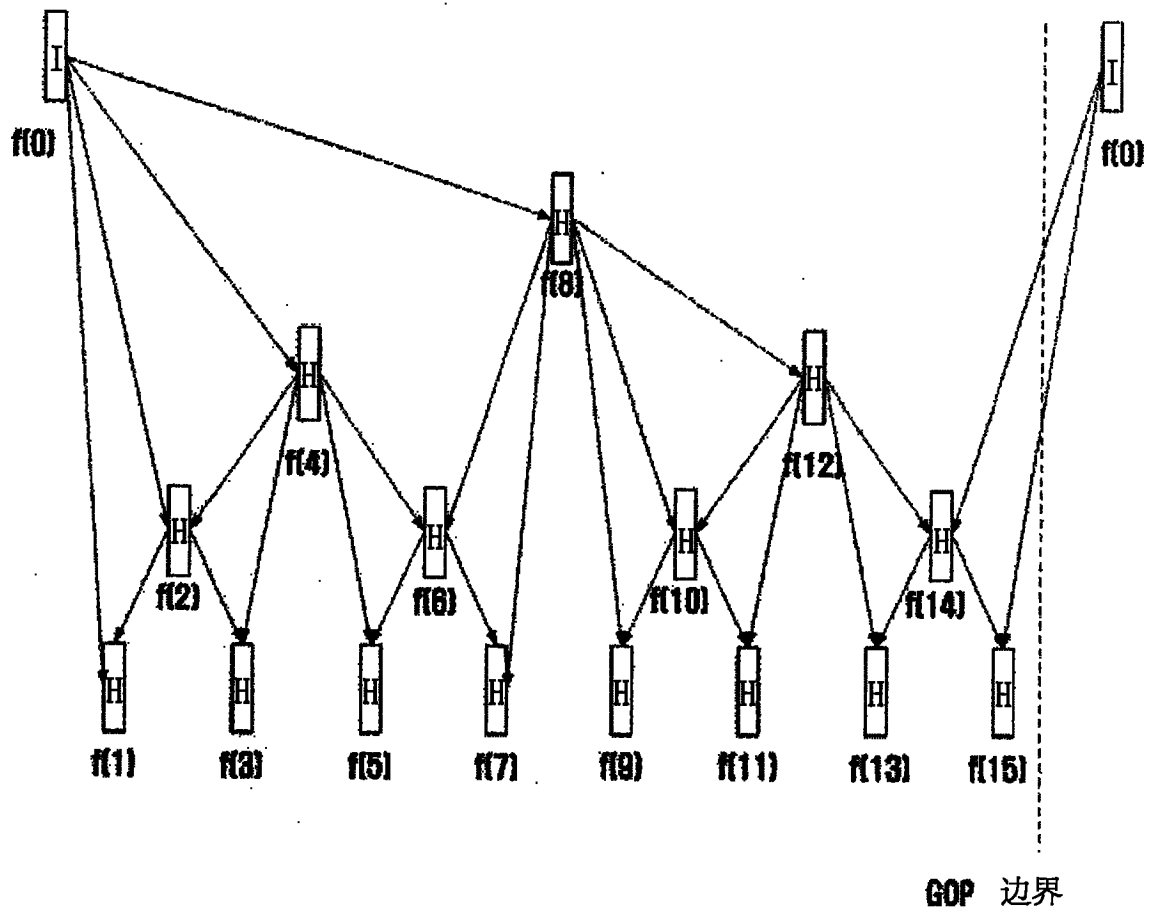


图 12

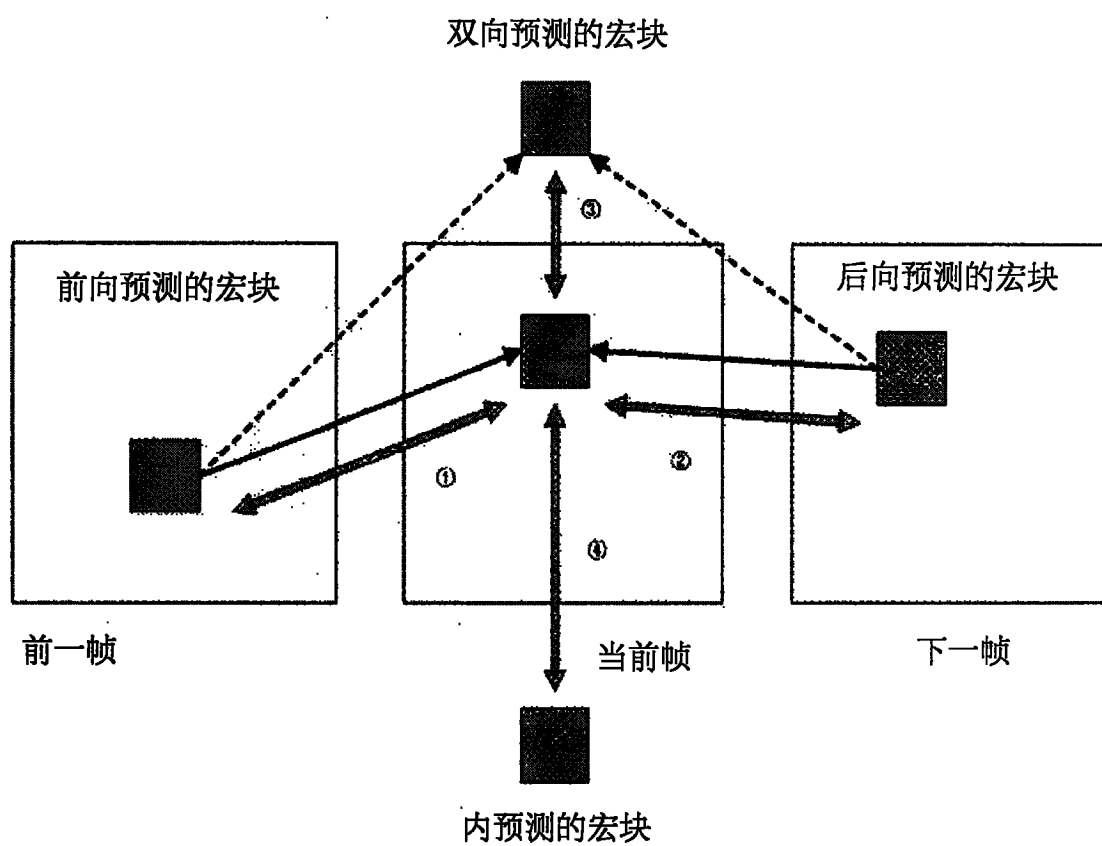


图 13

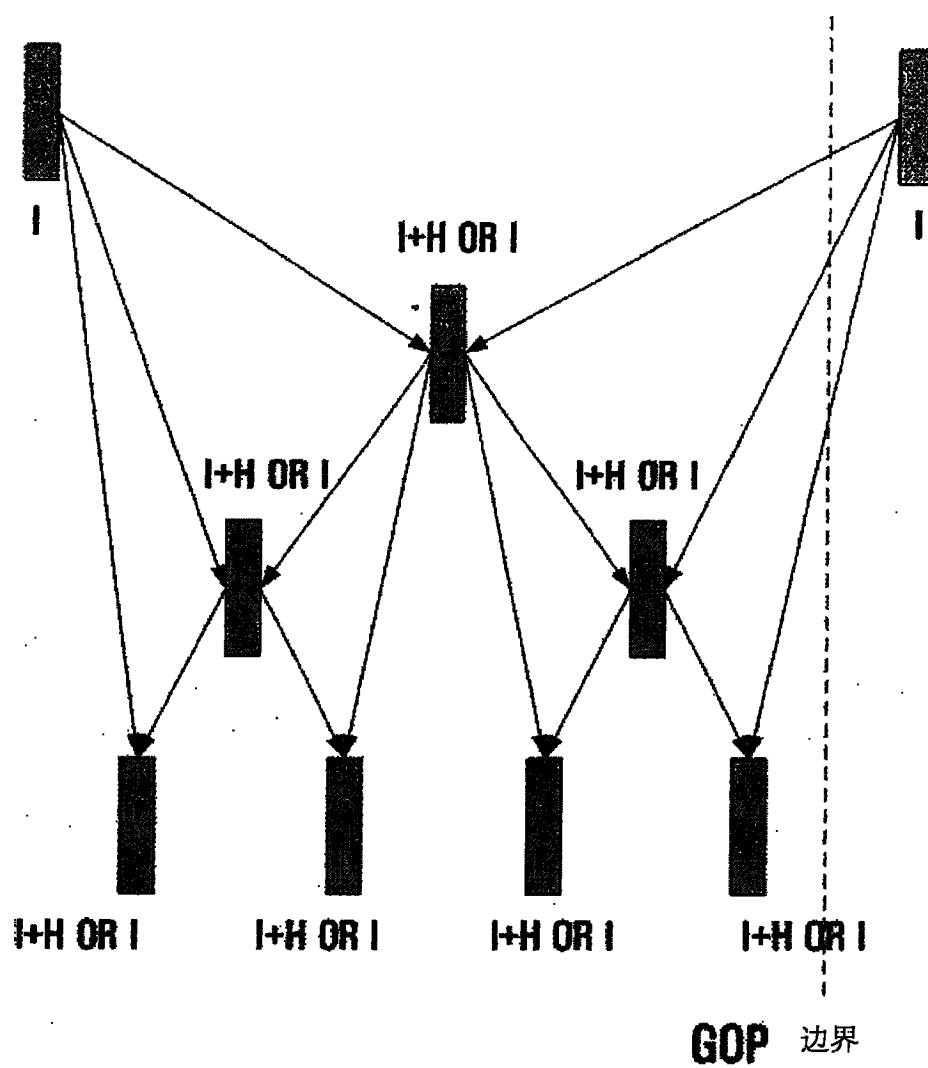


图 14

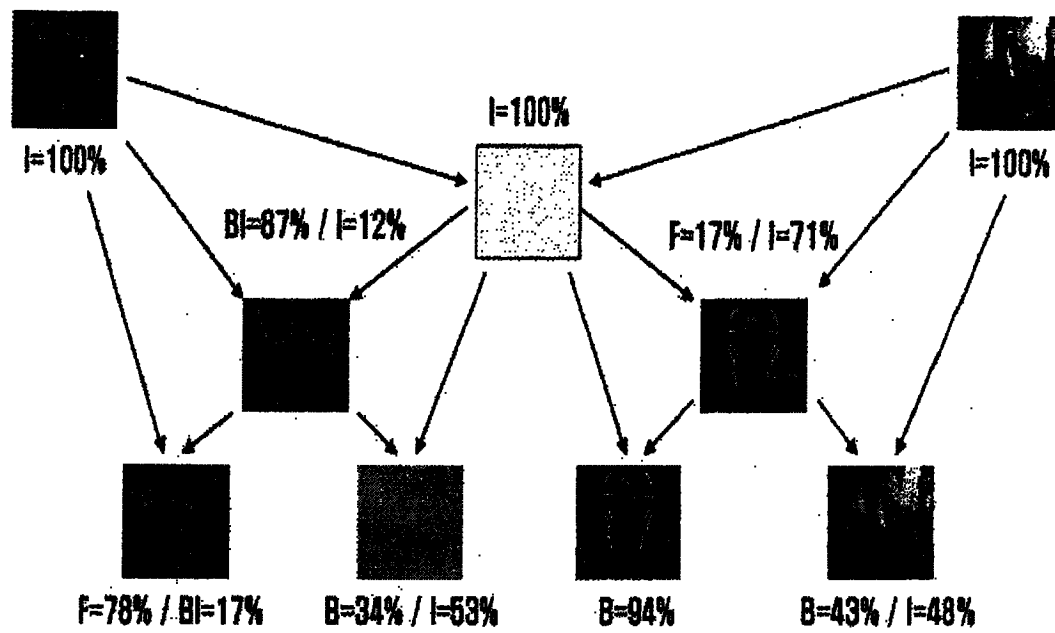


图 15a

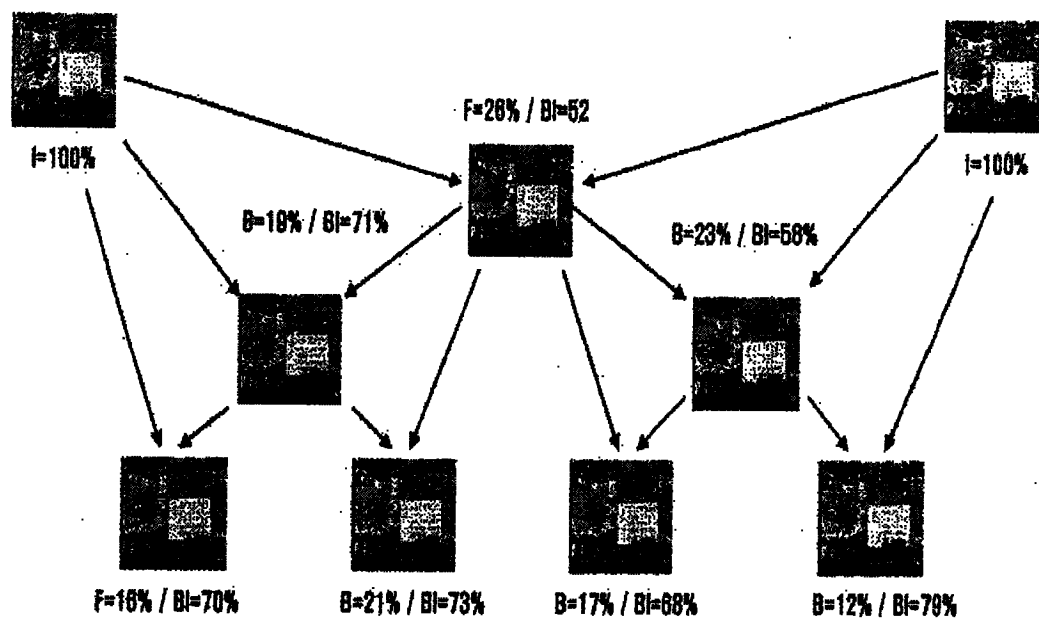


图 15b

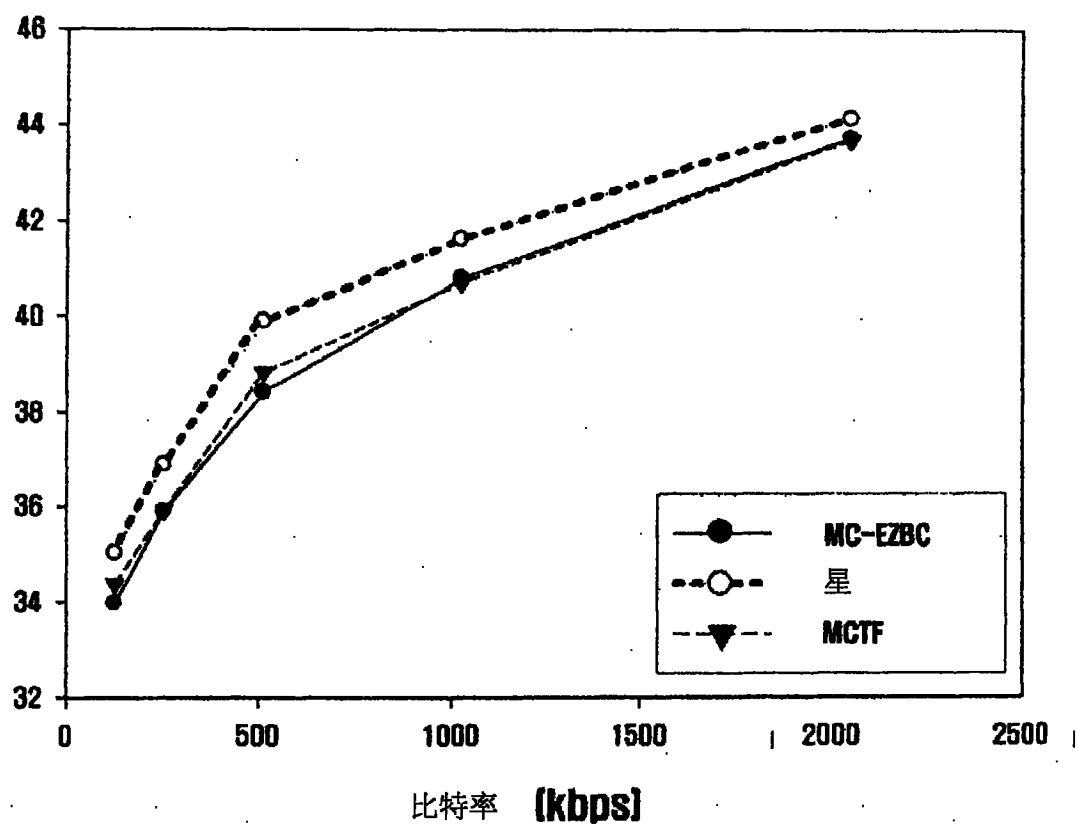
PSNR [dB]

图 16

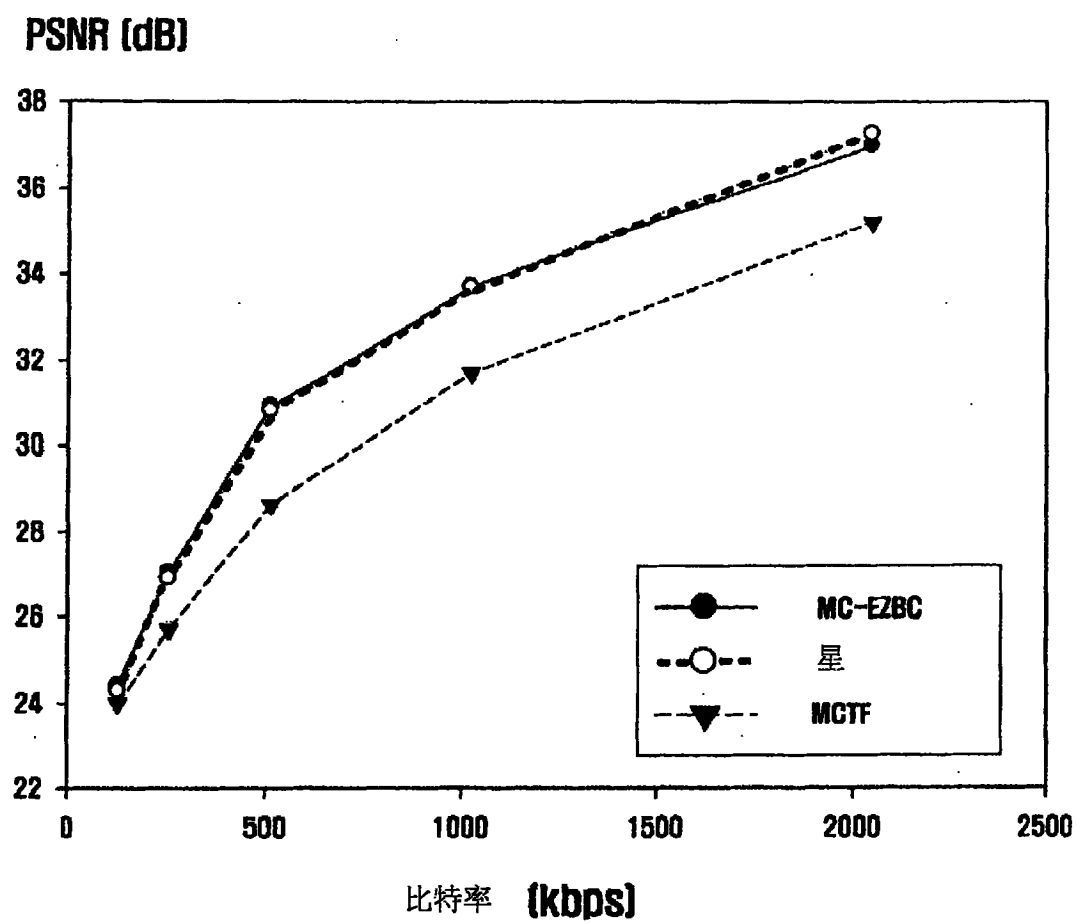


图 17

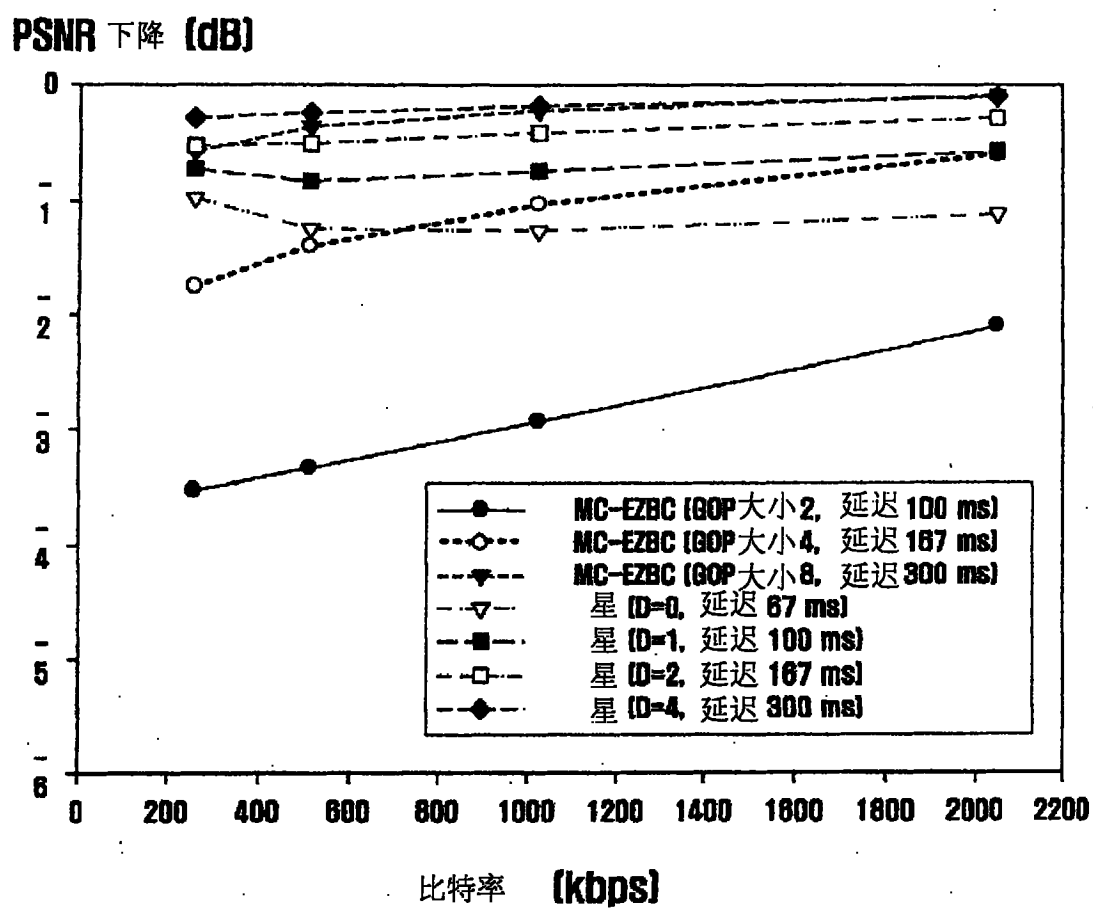


图 18

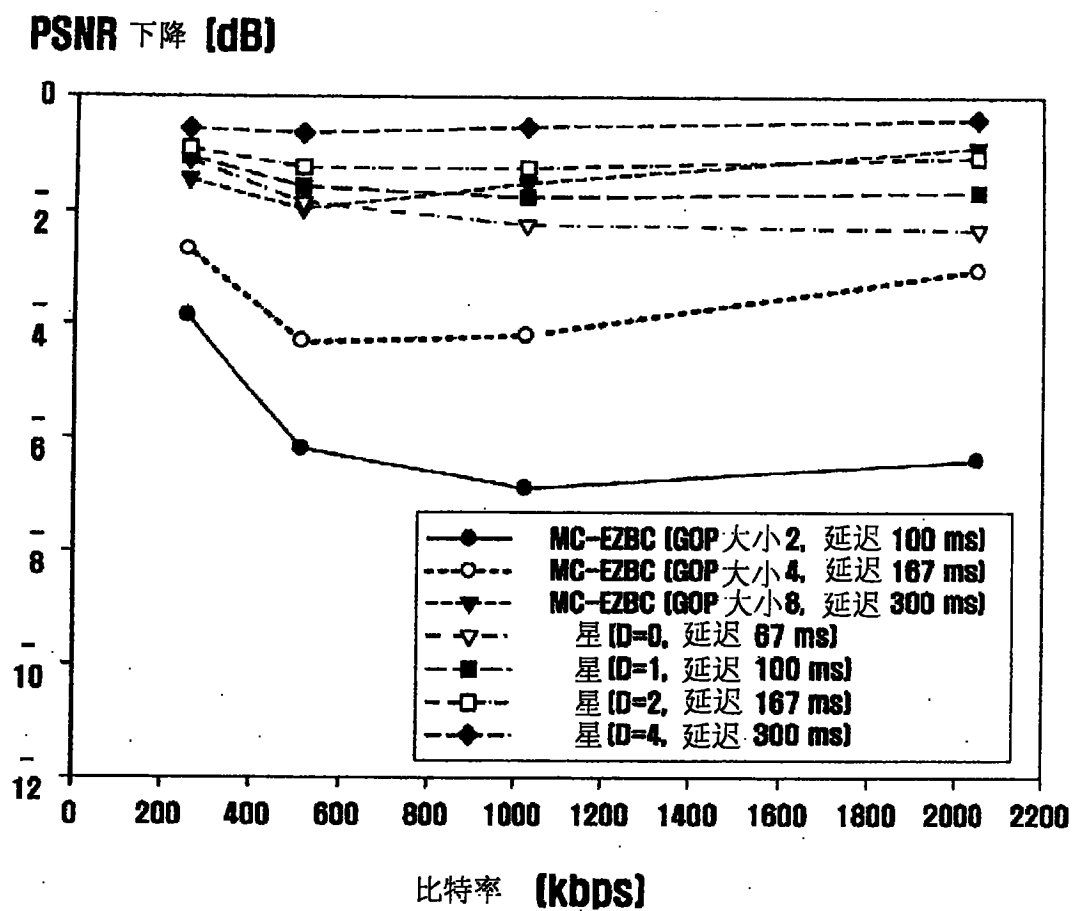


图 19

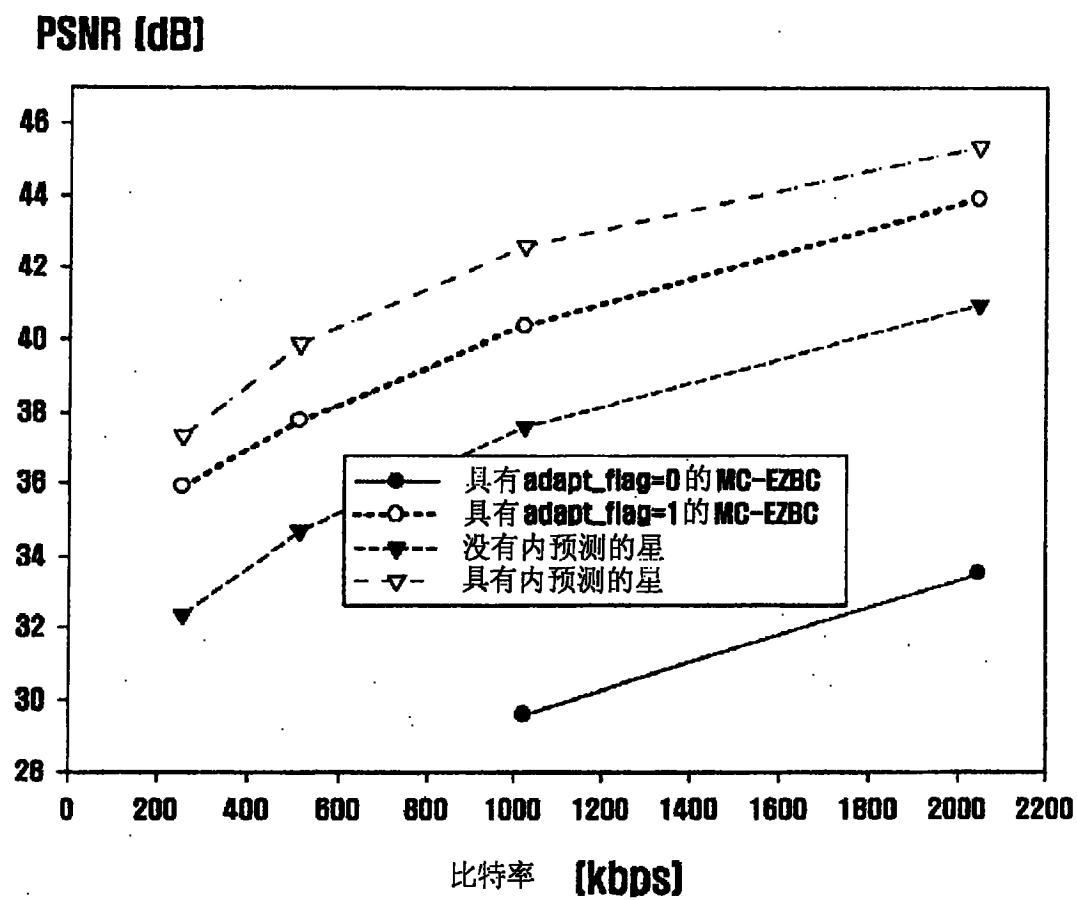


图 20