

一种基于 DCT 的磁盘阵列错误恢复算法

顾 健,周敬利,余胜生,向 东,陈加忠,郑俊浩

(华中科技大学外存储国家专业实验室/ 计算机科学与工程系,武汉 430074)

摘 要: 磁盘阵列可以提高磁盘存储系统的容量和数据吞吐量,而且具有容错能力,可以精确重建出错磁盘上的数据,但需要增加大量的磁盘访问开销,因而在恢复错误期间会影响存储系统的有效吞吐量.本文针对静止、活动图像压缩算法的特点,提出了一种客户端的近似图像错误恢复算法.该算法将图像分成若干子图存放在磁盘阵列的不同磁盘上,在单个磁盘失效时,利用 DCT 块之间的相关性近似恢复图像,从而不增加存储系统的开销.

关键词: 磁盘阵列; 图像压缩; DCT 变换; 错误恢复; 多媒体服务器

中图分类号: TP391 **文献标识码:** A **文章编号:** 0372-2112 (2001) 02-0208-03

DCT Based Algorithm of Image Recovery for Disk Array

GU Jian, ZHOU Jing-li, YU Sheng-sheng, XIANG Dong, CHEN Jia-zhong, ZHENG Jun-hao

(National Storage Lab/ Computer Sci. & Eng. Department, HUST, Wuhan 430074, China)

Abstract: The disk array can enhance the capacity and throughput of disk storage system, which can tolerate single disk failure and accurately reconstruct lost data. But the reconstruction may reduce the throughput in the reconstruction period because of the spending of increased disk access. An approximate image reconstruction algorithm at client is proposed according to the image compression properties, which divides image into a few subimages and stores them on different disks of disk array. The relativity of DCT blocks can be used to approximately reconstruct damaged images without increasing disk access.

Key words: disk array; image compression; discrete cosine transform; image recovery; multimedia server

1 引言

磁盘阵列 RAID (Redundant Array of Inexpensive (or Independent) Disks, RAID) 是按一定的存储和回取方式组织的一组磁盘, 实现大容量的数据存储和高的数据吞吐量. 多媒体服务器大多采用磁盘阵列存储设备, 由于 RAID 的数据分布在阵列中的各个磁盘上, 任一磁盘的错误会导致数据丢失. 因而 RAID 采取了冗余措施来提高 RAID 的可靠性: 增加冗余磁盘 (或块), 存放磁盘 (或块) 组的校验信息. 一旦某个磁盘出现错误, 可以通过剩余磁盘和校验信息恢复、重建出错数据, 以保持数据存取连续性和可靠性, 因而在大规模的数据存储中获得广泛应用. 现有的算法主要通过镜像或校验码重建数据^[1], 支持组数据恢复的分组 RAID^[2], 改进纠错编码算法^[3]和针对 RAID 的结构优化算法^[4~6]. 但是, 重建出错数据需要额外的磁盘访问时间, 由于多媒体流数据量大、具有时限要求, 而且同时请求服务的客户数多, 例如, 一个 MPEG1 流的数据率约为 1.5Mbps, 而 Microsoft 的 Tiger 视频文件服务器, 可以支持数万用户同时访问^[7], 重建出错数据引起可用磁盘带宽的下降会影响可支持的客户数和服务质量. 在此背景下, 本文提出一种基于 DCT 变换的客户端近似图像恢复算法, 以折中服务客户数目和服务质量的矛盾. 虽然图像质量有所下降,

但不增加额外的磁盘带宽需求. 而且由于人的视觉特性, 短时间、一定范围内的图像质量下降, 不会引起人的太大反感, 仍能满足视觉要求.

2 基于 DCT 的磁盘阵列错误恢复算法

JPEG、MPEG 等基于 DCT 的压缩图像数据占了多媒体存储容量的相当比例. 基于 DCT 的有失真 JPEG 算法把图像从时域经 8×8 DCT 变换到频域后, 每块得到 64 个 DCT 系数: DC0、AC1-63. 能量主要集中在低频部分, 特别是 DC 系数. 相邻 DCT 块的 DC 系数具有一定的相关性, 而 AC 系数之间相关性很小. 因此在图像缺失部分 DCT 块的情况下, 为了近似重建图像, 可以利用相邻块近似重建 DC 系数, 略去次要的 AC 系数. 要尽可能好地恢复出错图像, 应当使出错 DCT 块的相邻块保持完好, 以利于恢复出错块的 DC 系数. 由于 DC 系数是差分编码的, 所以要根据磁盘个数, 将 DCT 块重新分组, 每个磁盘一个分组, 组内仍采用差分编码. 这样, 既不影响 JPEG 的基本算法, 在单个磁盘发生错误时, 又可利用其它磁盘上的分组近似恢复图像. 再将一个 DCT 块的 AC 系数交叉分散到其它块, 使出错块尽可能保留一些系数, 均匀整幅图像. 因此, 可以在 JPEG 压缩算法的基础上, 增加一个容错模块, 实现容错 JPEG 压缩, 如图 1 所示.

收稿日期: 2000-01-21; 修回日期: 2000-08-10

基金项目: 国家 863 计划 (No. 863-317-01-10-99); 国防预研项目 (No. 15. 8. 4)

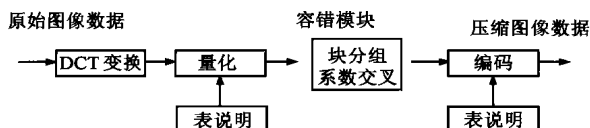


图1 基于DCT的容错JPEG编码过程

2.1 DCT块分组

DCT块分组是将一幅图像的 DCT 块分成若干组, 分别存放在磁盘阵列的不同磁盘组上。分组算法要使得在单个磁盘错误发生时, 客户端可以利用正常磁盘上的其它图像分组近似重建图像, 减轻服务器的错误处理负担, 保证一定的服务质量, 在轻载时再利用磁盘阵列本身的冗余精确重建出错数据。

假设一幅图像有 $IROW \times JCOL$ 个 DCT 块, $B(x, y)$ 为其中任意一个块, 其 DC 系数为 $DC_{B(x, y)}$, $0 \leq x \leq IROW-1, 0 \leq y \leq JCOL-1$, $IROW, JCOL$ 分别为图像每行、每列的 DCT 块数。则块 $B(x, y)$ 的 8 个相邻块可表示为:

$$NB_8(B(x, y)) = \{B(i, j) \mid |x - i| \leq 1, |y - j| \leq 1\} - \{B(x, y)\}$$

由于 DC 系数间的相关性, 可利用这 8 个相邻块的 DC 系数重建 $DC_{B(x, y)}$:

$$DC_{B(x, y)} = \frac{1}{8} \times \sum_{B(i, j) \in NB_8(B(x, y))} DC_{B(i, j)}$$

如果 8 个邻块中还有块缺失, 将会影响重建的效果, 因此分组应保证: 一个块与它的 8 个相邻块不在同一分组。在边界上, 一个块的邻块数小于 8, 计算时上式应相应调整。

2.2 交叉 AC 系数

由于 AC 系数之间不相关, 在重建子块时, 将缺失的 AC 系数置为 0。为减小快效应, 编码前在组间交叉 AC 系数, 把缺失的 AC 系数平均到整幅图像。

假设有 M 个 DCT 块: $\{B_0, B_1, \dots, B_{M-1}\}$, 交叉 AC 系数后: $\{B_0, B_1, \dots, B_{M-1}\}$, 第 l 个 AC 系数为:

$$AC_{B_l}^l = AC_{B_i}^l$$



图3 Lena (512 x 512, 8bit)

3 实验结果

为了评价本文所提出方案的效果, 用 Lena 图像 (图 3) 进行了不同出错比率 (4 组错 1 组、8 组错 1 组、16 组错 1 组、32

A								B							
A0	D1	C2	B3	A4	D5	C6	B7	B0	A1	D2	C3	B4	A5	D6	C7
A8	D9	C10	B11	A12	D13	C14	B15	B8	A9	D10	C11	B12	A13	D14	C15
A16	D17	C18	B19	A20	D21	C22	B23	B16	A17	D18	C19	B20	A21	D22	C23
A24	D25	C26	B27	A28	D29	C30	B31	B24	A25	D26	C27	B28	A29	D30	C31
A32	D33	C34	B35	A36	D37	C38	B39	B32	A33	D34	C35	B36	A37	D38	C39
A40	D41	C42	B43	A44	D45	C46	B47	B40	A41	D42	C43	B44	A45	D46	C47
A48	D49	C50	B51	A52	D53	C54	B55	B48	A49	D50	C51	B52	A53	D54	C55
A56	D57	C58	B59	A60	D61	C62	B63	B56	A57	D58	C59	B60	A61	D62	C63

C								D							
C0	B1	A2	D3	C4	B5	A6	D7	D0	C1	B2	A3	D4	C5	B6	A7
C8	B9	A10	D11	C12	B13	A14	D15	D8	C9	B10	A11	D12	C13	B14	A15
C16	B17	A18	D19	C20	B21	A22	D23	D16	C17	B18	A19	D20	C21	B22	A23
C24	B25	A26	D27	C28	B29	A30	D31	D24	C25	B26	A27	D28	C29	B30	A31
C32	B33	A34	D35	C36	B37	A38	D39	D32	C33	B34	A35	D36	C37	B38	A39
C40	B41	A42	D43	C44	B45	A46	D47	D40	C41	B42	A43	D44	C45	B46	A47
C48	B49	A50	D51	C52	B53	A54	D55	D48	C49	B50	A51	D52	C53	B54	A55
C56	B57	A58	D59	C60	B61	A62	D63	D56	C57	B58	A59	D60	C61	B62	A63

图2 分组数 $N=4, 8 \times 8$ DCT 块系数交叉后的分布情况

式中: $j = i + l \bmod M$, l 为 AC 系数序号, B_i 为交叉前 AC 系数所在的块, B_j 为交叉后 AC 系数所在的块。

若 DCT 块大小为 $n \times n$, $\{B_0, B_1, \dots, B_{M-1}\}$ 每个块的 $n \times n$ 个系数被分布到 $\{B_0, B_1, \dots, B_{M-1}\}$; $\{B_0, B_1, \dots, B_{M-1}\}$ 缺失一个块 ($n \times n$ 个系数) 也就是 $\{B_0, B_1, \dots, B_{M-1}\}$ 每个子块缺失 $n \times n / M$ 个系数, 重建的图像就比较均衡。 M 越大, 每块可能丢失的系数越少, 重建图像的质量就好, 但 M 不可能超过分组数 N , 即 $M \leq N$ 。本文以 $M = N = 4, n = 8$, 为例说明系数交叉算法。 A, B, C, D 4 个 DCT 块系数交叉后的 DCT 系数分布如图 2 所示。

A, B, C, D 4 个块交叉 AC 系数后得到 A, B, C, D 4 个新块。 A 的 DC 系数 $A0$ 在 A 块, AC 系数 $A1, A2, A3$ 分别交叉到了 B, C, D , 取代它们的是 $D1, C2, B3$ 三个 AC 系数。 $A4$ 仍在原来位置上, 如此重复, 直到第 63 个 AC 系数。从图中可以看出, 每个交叉后的 DCT 块中, AC 系数的序号不变, 保持了 JPEG 原有的优点, 对 AC 系数的编码效率影响小。

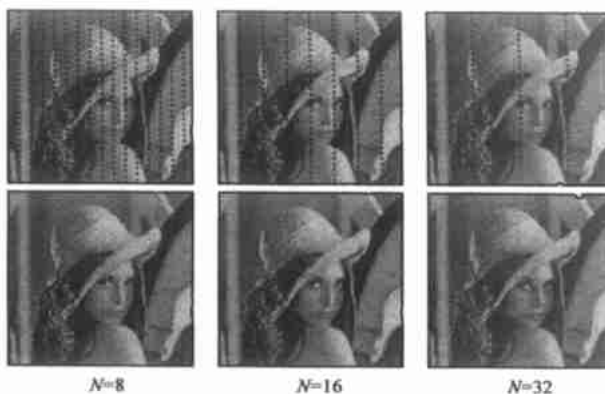


图4 缺失一组子图后的图像(上)与重建图像(下)

组错 1 组) 情况下的图像恢复实验 (见图 4), 并比较了它们的信噪比 ($SNR, PSNR$, 见表 1、图 5)。

从图 4 的缺损图像与重建图像的比较看出, 重建的图像明显改善了缺损图像的质量。

表 1 相对于正常解码图像的峰值信噪比(dB)

图像	PSNR	SNR	说明
$N=4$	26.6	21.0	丢失 1/4 的块(25 %)
$N=8$	29.9	24.2	丢失 1/8 的块(12.5 %)
$N=16$	32.1	26.4	丢失 1/16 的块(6.3 %)
$N=32$	34.8	29.2	丢失 1/32 的块(3.1 %)

从表 1 和图 5 可以看出,分组数从 4 提高到 8,信噪比提高较快;分组数从 8 增加到 16、32,信噪比的提高减缓。图 4 也体现了这一点,分组数 $N=8$ 时,重建图像的质量明显优于 $N=4$,而 $N=16,32$ 时,图像质量虽进一步得到改善,但幅度减小。因此,分组数 8 是一个较好的选择。

由于 DCT 系数在分组内才使用差分编码,而每个分组内的 DCT 块在空间上不连续,随着分组数的增加,分组内 DCT 块之间的相关性减小,图像压缩率会降低。表 2、图 6 是不同分组时的数据压缩率比较,分组数从 4 增加到 32,压缩后的数据长度增加了 1451 字节,比不分组增加了 3.8%。考虑到算法必要的冗余(备份量化表等头部信息 340 字节),分组数从 4 增加到 32,实际增加了 4.7%。

表 2 不同分组下的压缩率比较

分组数	不分组	4	8	16	32
数据段总长度(Byte)	37860	38563	38955	39287	39311
增加大小(Byte)	0	703	1095	1427	1451

4 结论

本文提出了一种可用于多媒体系统在客户端近似重建出错图像的基于 DCT 的磁盘阵列错误恢复算法,并进行了实验分析。该算法有如下特点:

- (1) 在几乎不增加服务器存储系统输入输出带宽和 CPU 开销的情况下,可以在客户端近似重建出错图像,而且有可接受的主观图像质量(见图 4);
- (2) 容错 JPEG 算法对压缩比和算法复杂性的影响较小;
- (3) MPEG 也是基于 DCT 的图像压缩算法,因此本文的错误恢复算法也适用于 MPEG。利用 MPEG 的辅助数据段说明分组情况,基本不会影响 MPEG 的数据流结构;

(4) 对多媒体服务器,特别是磁盘带宽负载较重时,采用该算法后,即使发生单个磁盘错误,系统仍然能够提供连续和稳定的服务,而不会由于资源不足暂停或中断部分用户。

参考文献:

- [1] M. S. Chen, H. I. Hsiao, C. S. Li, and P. S. Yu. Using rotational mirrored declustering for replica placement in a disk-array-based video server [A]. In Proceedings of the Third ACM Conference on Multimedia [C], San Francisco, California, November 1995, pages 121 - 130.
- [2] P. M. Chen and E. K. Lee. Striping in a RAID level 5 disk array [A]. In Proceedings of the 1995 ACM SIGMETRICS Conference on Measurement and Modeling of Computer Systems [C], Ottawa, May 1995: 136 - 144.
- [3] G. Gibson and D. Patterson. Designing disk arrays for high data reliability [J]. Journal of Parallel and Distributed Computing, January 1993: 4 - 27.
- [4] J. Menon and J. Cortney. The architecture of a fault-tolerant cached RAID controller [A]. In Proceedings of the 20th International Symposium on Computer Architecture [C], May 1993, 76 - 86.
- [5] A. Mourad. Reliable disk striping in video-on-demand servers [R]. Technical report, AT&T Bell Labs, 1995.
- [6] R. Tewari, D. Dias, R. Mukherjee, and H. M. Vin. High availability for clustered multimedia servers [A]. In Proceedings of International Conference on Data Engineering [C], New Orleans, February 1996: 645 - 654.
- [7] R. Haskin and F. Schmuck. Tiger shark file system [A]. Proceedings of COMPCON [C], Spring 1996: 226 - 231.

作者简介:



顾 健 1963 年生,1983 年毕业于江苏理工大学自动控制专业,1992 年获硕士学位,现为华中科技大学计算机系统结构专业博士生研究。研究方向:多媒体系统结构与设备、多媒体信息处理。



周敬利 教授,博导,1969 年毕业于华中工学院,1996~1997 年留学美国,现任华中科技大学计算机科学与技术学院副院长。研究方向:多媒体通信技术的应用,高性能网络存储技术。



余胜生 教授,博导,1966 年毕业于华中工学院,1985 年留学西德。研究方向:计算机系统结构与网络通讯技术。