



(12) 发明专利

(10) 授权公告号 CN 101389033 B

(45) 授权公告日 2011.09.28

(21) 申请号 200810305297.7

(22) 申请日 2008.10.30

(73) 专利权人 四川长虹电器股份有限公司
地址 621000 四川省绵阳市高新区绵兴东路
35号

(72) 发明人 展华益 刘明华

(74) 专利代理机构 成都虹桥专利事务所 51124
代理人 李顺德

(51) Int. Cl.
H04N 7/32 (2006.01)

审查员 王剑

权利要求书 1 页 说明书 3 页

(54) 发明名称

帧场自适应解码模式下解码器缓存控制方法

(57) 摘要

本发明涉及图像解码技术,特别涉及帧场自适应解码模式下解码器缓存控制方法。本发明针对现有技术对帧场自适应解码模式下,存储器利用率低,需要较大的存储空间,解码器成本高的缺点,公开了一种帧场自适应解码模式下解码器缓存控制方法,以克服上述缺陷。本发明的技术方案是,帧场自适应解码模式下解码器缓存控制方法,包括以下步骤:a、在解码器存储空间中划分一个图像缓存区;b、根据当前图像的编码模式,预设存取当前图像和参考图像数据的跨度;c、图像解码过程中,帧和场的空间共享所述图像缓存区。本发明用于解码器缓存控制,能够有效提高存储空间的利用率,节省缓存空间,提高解码器的执行效率。

1. 帧场自适应解码模式下解码器缓存控制方法,包括以下步骤:

a、在解码器存储空间中划分一个图像缓存区,所述图像缓存区与 $n+1$ 帧图像大小相当;其中 n 为解码器所需最多参考帧的数目;

b、根据当前图像的编码模式,预设存取当前图像和参考图像数据的跨度:如果当前图像编码模式为帧,所述跨度设为图像宽度;如果当前图像编码模式为场,所述跨度设为图像宽度的 2 倍;其中,当涉及到存取当前图像和参考图像数据跨行的情况时依据所述跨度来进行;

c、图像解码过程中,帧和场的空间共享所述图像缓存区:如果当前编码模式为帧,将当前图像的指针指向当前帧缓存的地址,将 n 帧参考帧的指针分别指向对应参考帧的缓存的地址;如果当前编码模式为场,且当前图像为顶场,将当前图像的指针指向当前帧缓存的地址,将 $2*n$ 帧参考场的指针分别指向对应参考场的缓存的地址;如果当前编码模式为场,且当前图像为底场,将当前图像的指针指向当前帧缓存的地址加上图像宽度,将 $2*n$ 帧参考场的指针分别指向对应参考场的缓存的地址。

帧场自适应解码模式下解码器缓存控制方法

技术领域

[0001] 本发明涉及图像解码技术,特别涉及帧场自适应解码模式下解码器缓存控制方法。

背景技术

[0002] 目前,一些流行的视频压缩标准,为了获得较高的压缩性能,都采用了帧间预测的方法。帧间预测就是用当前图像前后相邻的一帧或几帧参考图像来预测当前图像。在帧场自适应解码模式下,解码器需要为帧和场分别开辟一段独立的存储空间,用于存放当前帧、参考帧、当前场和参考场的数据。如果当前图像编码模式为帧且需要用作参考,在当前帧解码完成后,需要将当前帧分离为两场。如果当前图像编码模式为场且需要用作参考,在当前图像的两场解码完成后,需要将两场合并为帧。现有技术中存储器的这种分配模式,存储器的利用率低,解码器需要较大的存储空间,不利于解码器成本的降低。而且帧场的分离、合并操作也降低了解码效率。

发明内容

[0003] 本发明所要解决的技术问题,就是针对现有技术中在帧场自适应解码模式下,存储器利用率低,需要较大的存储空间,解码器成本高的缺点,提供一种帧场自适应解码模式下解码器缓存控制方法,以克服上述缺陷。

[0004] 本发明解决所述就是问题,采用的技术方案是,帧场自适应解码模式下解码器缓存控制方法,包括以下步骤:

[0005] a、在解码器存储空间中划分一个图像缓存区,所述图像缓存区与 $n+1$ 帧图像大小相当;其中 n 为解码器所需最多参考帧的数目;

[0006] b、根据当前图像的编码模式,预设存取当前图像和参考图像数据的跨度:如果当前图像编码模式为帧,所述跨度设为图像宽度;如果当前图像编码模式为场,所述跨度设为图像宽度的 2 倍;其中,当涉及到存取当前图像和参考图像数据跨行的情况时依据所述跨度来进行;

[0007] c、图像解码过程中,帧和场的空间共享所述图像缓存区:如果当前编码模式为帧,将当前图像的指针指向当前帧缓存的地址,将 n 帧参考帧的指针分别指向对应参考帧的缓存的地址;如果当前编码模式为场,且当前图像为顶场,将当前图像的指针指向当前帧缓存的地址,将 $2*n$ 帧参考场的指针分别指向对应参考场的缓存的地址;如果当前编码模式为场,且当前图像为底场,将当前图像的指针指向当前帧缓存的地址加上图像宽度,将 $2*n$ 帧参考场的指针分别指向对应参考场的缓存的地址。

[0008] 本发明的有益效果是,能够有效提高存储空间的利用率,节省缓存空间,不需要再为存储场的数据开辟独立的存储空间,即帧和场共享存储空间。在当前图像解码完成后,不需要做帧和场的合并或分离操作,能有效提高解码器的执行效率。

具体实施方式

[0009] 下面结合实施例,详细描述本发明的技术方案。

[0010] 本发明的技术方案是,帧场自适应解码模式下解码器缓存控制方法,包括以下步骤:

[0011] a. 在解码器存储空间中划分一个图像缓存区,其大小与 $n+1$ 帧图像大小相当, n 为解码器所需最多参考帧的数目;

[0012] b. 定义一全局变量,用于表示存取当前图像和参考图像数据时的跨度,假设为 s ;解码器在整个解码活动中,凡是涉及到存取当前图像和参考图像数据跨行的情况都依据跨度 s 进行;如果当前图像编码模式为帧,将跨度 s 设为图像宽度;如果当前图像编码模式为场,将跨度 s 设为图像宽度的 2 倍;

[0013] c. 图像解码过程中,帧和场的空间共享所述图像缓存区。

[0014] 下面描述存储器地址空间分配方法:

[0015] 设 $pReference[0], pReference[1], \dots, pReference[n-1]$ 分别表示 n 帧参考帧缓存的地址, $pImage$ 表示当前帧缓存的地址。

[0016] 用 $pCurImage$ 表示当前图像(帧或场)指针,用 $pRef[0], pRef[1], \dots, pRef[2*n-1]$ 分别表示参考图像(帧或场)的指针, $imgWidth$ 表示图像宽度 s 。

[0017] 如果当前编码模式为帧,将当前图像的指针指向当前帧缓存的地址,将 n 帧参考帧的指针分别指向对应参考帧的缓存的地址,即将 $pImage$ 赋值给 $pCurImage$,将 $pReference[0]$ 赋值给 $pRef[0]$,将 $pReference[1]$ 赋值给 $pRef[1]$,依此类推,直到将 $pReference[n-1]$ 赋值给 $pRef[n-1]$ 。

[0018] 如果当前编码模式为场,且当前图像为顶场,将当前图像的指针指向当前帧缓存的地址,将 $2*n$ 帧参考场的指针分别指向对应参考场的缓存的地址,即将 $pImage$ 赋值给 $pCurImage$,将 $pReference[0]$ 加上 $imgWidth$ 赋值给 $pRef[0]$,将 $pReference[0]$ 赋值给 $pRef[1]$,将 $pReference[1]$ 加上 $imgWidth$ 赋值给 $pRef[2]$,将 $pReference[1]$ 赋值给 $pRef[3]$,依此类推,直到将 $pReference[n-1]$ 加上 $imgWidth$ 赋值给 $pRef[2*n-2]$,直到将 $pReference[n-1]$ 赋值给 $pRef[2*n-1]$ 。

[0019] 如果当前编码模式为场,且当前图像为底场(假设允许用当前图像顶场作为参考),将当前图像的指针指向当前帧缓存的地址加上 $imgWidth$,将 $2*n$ 帧参考场的指针分别指向对应参考场的缓存的地址,即将 $pImage$ 加上 $imgWidth$ 赋值给 $pCurImage$,将 $pImage$ 赋值给 $pRef[0]$,将 $pReference[0]$ 加上 $imgWidth$ 赋值给 $pRef[1]$,将 $pReference[0]$ 赋值给 $pRef[2]$,将 $pReference[1]$ 加上 $imgWidth$ 赋值给 $pRef[3]$,依此类推,直到将 $pReference[n-2]$ 赋值给 $pRef[2*n-2]$,直到将 $pReference[n-1]$ 加上 $imgWidth$ 赋值给 $pRef[2*n-1]$ 。

[0020] 实施例

[0021] 假设解码器参考帧数目为 2,当前图像顶场允许作为参考。

[0022] a. 在解码器存储空间开辟三帧图像的缓存;

[0023] b. 定义一全局变量 s ;

[0024] 如果当前图像编码模式为帧,将 s 设为图像宽度;如果当前图像编码模式为场,将跨度 s 设为图像宽度的 2 倍;在整个解码活动中,凡是涉及到存取当前图像和参考图像数据

跨行的情况都依据跨度 s 进行；

[0025] 设置当前图像和参考图像的指针；

[0026] $pImage$ 表示当前帧缓存的地址, $pReference[0]$, $pReference[1]$ 分别表示 2 帧参考帧缓存的地址;用 $pCurImage$ 表示当前图像(帧或场)指针,用 $pRef[0]$, $pRef[1]$, $pRef[2]$, $pRef[3]$ 分别表示参考图像(帧或场)的指针, $imgWidth$ 表示图像宽度。

[0027] 如果当前编码模式为帧,

[0028] $pCurImage = pImage$;

[0029] $pRef[0] = pReference[0]$;

[0030] $pRef[1] = pReference[1]$;

[0031] 如果当前编码模式为场,且当前图像为顶场,

[0032] $pCurImage = pImage$;

[0033] $pRef[0] = pReference[0] + imgWidth$;

[0034] $pRef[1] = pReference[0]$;

[0035] $pRef[2] = pReference[1] + imgWidth$;

[0036] $pRef[3] = pReference[1]$;

[0037] 如果当前编码模式为场,且当前图像为底场,

[0038] $pCurImage = pImage + imgWidth$;

[0039] $pRef[0] = pImage$;

[0040] $pRef[1] = pReference[0] + imgWidth$;

[0041] $pRef[2] = pReference[0]$;

[0042] $pRef[3] = pReference[1] + imgWidth$ 。