



(12)发明专利

(10)授权公告号 CN 105791824 B

(45)授权公告日 2018.08.31

(21)申请号 201610132621.4

(22)申请日 2016.03.09

(65)同一申请的已公布的文献号
申请公布号 CN 105791824 A

(43)申请公布日 2016.07.20

(73)专利权人 西安电子科技大学
地址 710071 陕西省西安市太白南路2号

(72)发明人 吴炜 吴倩 刘炯 冯磊

(74)专利代理机构 陕西电子工业专利中心
61205

代理人 王品华 朱红星

(51)Int.Cl.

H04N 19/107(2014.01)

H04N 19/122(2014.01)

H04N 19/176(2014.01)

(56)对比文件

CN 105120292 A,2015.12.02,

CN 103596003 A,2014.02.19,

CN 104219531 A,2014.12.17,

WO 2015180014 A1,2015.12.03,

US 2015264396 A1,2015.09.17,

CN 102611888 A,2012.07.25,

Guolin Jin 等.ON PREDICTION

TECHNIQUES FOR PALETTE CODING.《IEEE International Conference On Image Processing》.2014,

Do-Kyoung Kwon 等.Fast Intra Block Copy (IntraBC) search for HEVC screen content coding.《IEEE International Symposium on Circuits and Systems》.2014,

审查员 李维维

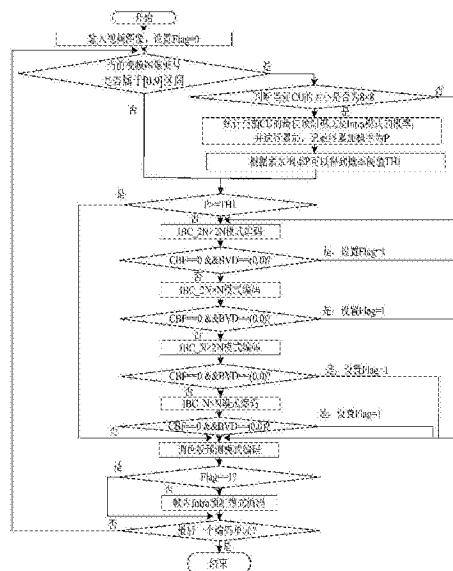
权利要求书2页 说明书10页 附图1页

(54)发明名称

基于边缘点密度的屏幕内容编码预测模式快速选择方法

(57)摘要

本发明公开了一种基于边缘点密度的屏幕内容编码预测模式快速选择方法,主要解决现有技术计算复杂度高的问题。其技术方案是:将屏幕内容编码SCC预测模式的编码顺序调整为IBC_{2N}×2N预测模式、IBC_{2N}×N预测模式、IBC_N×2N预测模式、IBC_N×N预测模式、帧内Intra预测模式、PLT预测模式;每进行完一种预测模式的编码,均对编码块标志CBF和块矢量差异BVD作检测,判断其是否分别为0和(0,0),如果是,则跳过后续预测模式的编码,否则按顺序进行后续预测模式编码,直到选择出最佳预测模式。本发明能快速区分出动画类型视频序列,并减少了对动画和非动画类型视频序列的编码时间,可用于视频处理。



1. 基于边缘点密度的屏幕内容编码预测模式快速选择方法,包括:

(1) 输入视频图像,设置标志位Flag为0;

(2) 判断当前编码单元所在的视频图像帧号是否属于[0,9]区间,如果是,则执行步骤(3),否则执行步骤(8);

(3) 判断当前编码单元CU的尺寸是否为 8×8 ,如果是,则用Sobel算子对当前编码单元进行卷积,并计算其边缘点密度,执行步骤(4),否则执行步骤(8);

(4) 判断当前编码单元的边缘点密度是否处于[0.9,1]区间,如果是,则统计当前编码单元的最佳预测模式是帧内Intra预测模式情况下的概率,并对当前编码单元的概率进行累加,得到当前帧视频图像的概率,执行步骤(5),否则执行步骤(8);

(5) 判断当前视频图像帧号是否等于9,如果是,则记前10帧视频图像的累加概率为P,执行步骤(6),否则将当前帧视频图像的概率累加,执行步骤(8);

(6) 根据步骤(5)的累加概率P,得到与量化参数QP相对应的不同概率阈值TH1;

(7) 判断步骤(5)的累加概率P是否小于步骤(6)的概率阈值TH1,如果是,则执行步骤(8),否则执行步骤(12);

(8) 调整屏幕内容编码SCC预测模式编码的顺序:首先进行帧内块拷贝IBC_ $2N \times 2N$ 预测模式的编码;再检测编码块标志CBF和块矢量差异BVD是否分别为0和(0,0),如果是,则将标志位Flag设置为1,执行步骤(12),否则执行步骤(9),其中, $2N \times 2N$ 是一种预测分割模式,N的值取4或8或16或32,IBC_ $2N \times 2N$ 预测模式表示在进行帧内块拷贝编码时,一个编码单元对应一个预测单元;

(9) 进行帧内块拷贝IBC_ $2N \times N$ 预测模式的编码,之后检测编码块标志CBF和块矢量差异BVD是否分别为0和(0,0),如果是,则将标志位Flag设置为1,执行步骤(12),否则执行步骤(10),其中, $2N \times N$ 是一种预测分割模式,IBC_ $2N \times N$ 预测模式表示在进行帧内块拷贝编码时,一个编码单元对应2个水平划分的预测单元;

(10) 进行帧内块拷贝IBC_ $N \times 2N$ 预测模式的编码,之后检测编码块标志CBF和块矢量差异BVD是否分别为0和(0,0),如果是,则将标志位Flag设置为1,执行步骤(12),否则执行步骤(11),其中, $N \times 2N$ 是一种预测分割模式,IBC_ $N \times 2N$ 预测模式表示在进行帧内块拷贝编码时,一个编码单元对应2个垂直划分的预测单元;

(11) 进行帧内块拷贝IBC_ $N \times N$ 预测模式的编码,之后检测编码块标志CBF和块矢量差异BVD是否分别为0和(0,0),如果是,则将标志位Flag设置为1,执行步骤(12),否则直接执行步骤(12),其中, $N \times N$ 是一种预测分割模式,IBC_ $N \times N$ 预测模式表示在进行帧内块拷贝编码时,一个编码单元对应4个预测单元;

(12) 进行调色板预测模式的编码,之后判断标志位Flag是否为1,如果是,执行步骤(13),否则进行帧内Intra预测模式的编码,执行步骤(13);

(13) 判断当前编码单元是否为视频图像的最后一个编码单元,如果是,则结束SCC预测模式的编码,否则返回步骤(2)。

2. 根据权利要求书1所述的方法,其中步骤(3)中用Sobel算子对当前编码单元进行卷积,并计算其边缘点密度,按如下步骤进行:

3a) 分别计算第i行第j列像素 $p_{i,j}$ 在垂直方向上的差异 $D_{x_{i,j}}$ 和在水平方向上的差异 $D_{y_{i,j}}$,即:

$$D_{x_{i,j}} = p_{i+1,j-1} + 2 * p_{i+1,j} + p_{i+1,j+1} - p_{i-1,j-1} - 2 * p_{i-1,j} - p_{i-1,j+1}$$

$$D_{y_{i,j}} = p_{i-1,j+1} + 2 * p_{i,j+1} + p_{i+1,j+1} - p_{i-1,j-1} - 2 * p_{i,j-1} - p_{i+1,j-1}$$

其中, $p_{i+1,j-1}$ 是编码单元的第 $i+1$ 行第 $j-1$ 列像素, $p_{i+1,j}$ 是编码单元的第 $i+1$ 行第 j 列像素, $p_{i+1,j+1}$ 是编码单元的第 $i+1$ 行第 $j+1$ 列像素, $p_{i-1,j-1}$ 是编码单元的第 $i-1$ 行第 $j-1$ 列像素, $p_{i-1,j}$ 是编码单元的第 $i-1$ 行第 j 列像素, $p_{i-1,j+1}$ 是编码单元的第 $i-1$ 行第 $j+1$ 列像素, $p_{i,j-1}$ 是编码单元的第 i 行第 $j-1$ 列像素, $p_{i,j+1}$ 是编码单元的第 i 行第 $j+1$ 列像素;

3b) 计算第 i 行第 j 列像素 $p_{i,j}$ 的梯度矢量 $D_{i,j}$:

$$D_{i,j} = \{D_{x_{i,j}}, D_{y_{i,j}}\};$$

3c) 计算第 i 行第 j 列像素 $p_{i,j}$ 的梯度矢量幅值 $A_{i,j}$:

$$A_{i,j} = |D_{x_{i,j}}| + |D_{y_{i,j}}|;$$

3d) 判断 $p_{i,j}$ 的梯度矢量幅值 $A_{i,j}$ 是否大于边缘阈值 $TH2$: 如果是, 则此像素是边缘点像素, 否则此像素不是边缘点像素, 边缘阈值 $TH2$ 与量化参数 QP 有关, 若 $QP=22$, $TH2=230$; 若 $QP=27$, $TH2=260$; 若 $QP=32$, $TH2=280$; 若 $QP=37$, $TH2=300$;

3e) 计算 8×8 大小的编码单元 CU 的边缘点密度 $EgDen$:

$$EdgeDen = EdgeNum / (Sum * 1.0)$$

其中, $EdgeNum$ 表示 8×8 大小的 CU 包含的边缘点像素数, Sum 表示当前 CU 像素总数, $EgDen$ 表示 8×8 大小的 CU 的边缘点密度。

3. 根据权利要求书1所述的方法, 其中步骤(4)中统计当前编码单元的最佳预测模式是帧内 $Intra$ 预测模式情况下的概率, 按如下公式进行:

$$pr = \frac{cnt_{intra}}{cnt_{intra} + cnt_{ibc}} \times 100\%$$

其中, pr 表示当前编码单元的最佳预测模式是帧内 $Intra$ 预测模式情况下的概率, cnt_{intra} 表示当前编码单元的最佳预测模式是帧内 $Intra$ 预测模式情况下的次数, cnt_{ibc} 表示当前编码单元的最佳预测模式是帧内块拷贝预测模式情况下的次数, 该帧内块拷贝预测模式包括 $IBC_2N \times 2N$ 预测模式、 $IBC_2N \times N$ 预测模式、 $IBC_N \times 2N$ 预测模式、 $IBC_N \times N$ 预测模式四种候选预测模式。

4. 根据权利要求书1所述的方法, 其中步骤(6)中的概率阈值 $TH1$, 其与量化参数 QP 有关, 若 $QP=22$, 则 $TH1=0.9$; 若 $QP=27$, 则 $TH1=0.83$; 若 $QP=32$, 则 $TH1=0.85$; 若 $QP=37$, 则 $TH1=0.85$ 。

基于边缘点密度的屏幕内容编码预测模式快速选择方法

技术领域

[0001] 本发明属于视频处理技术领域,特别涉及一种屏幕内容视频编码的预测模式快速选择方法,可用于远程教育、屏幕共享和无线局域网等应用中。

背景技术

[0002] 远程教育等应用中的内容可以分为两大类:屏幕内容图像和自然图像,其中屏幕内容图像是由计算机合成的,自然图像是由相机拍摄产生的。屏幕内容图像和自然图像之间的区别是屏幕内容图像包含对比度高的文字以及重复的预测模式等。物理特征的不同,也使两者的编码方法不同。如果依然采用高效率视频编码HEVC编码屏幕内容图像,则难免会出现严重的图像失真。因此,针对屏幕内容编码SCC技术应运而生。高效率视频编码HEVC标准的预测模式判决过程只包含传统的帧内编码技术,而屏幕内容编码SCC标准的预测模式判决过程不仅包含传统的帧内编码技术,还包含帧内块拷贝IBC编码技术和调色板PLT技术。其中帧内块拷贝预测模式包括IBC_{2N×2N}预测模式、IBC_{2N×N}预测模式和IBC_{N×2N}预测模式以及IBC_{N×N}预测模式等四种候选预测模式,因而增加了SCC预测模式判决的复杂度。又由于大多数的屏幕内容应用对实时效果有较高的要求,因此,屏幕内容编码标准不仅要同时对文字和图片等边缘处的高频成分进行无损编码,而且还要同时满足编码时间的要求,即针对屏幕内容编码的预测模式需要快速选择。

[0003] Do-Kyoung Kwon等人在“Fast encoding using early skipping of Intra block copy (IntraBC) search”中提出了提早终止帧内块拷贝搜索的快速方法。该方法包含两部分:一是通过传统帧内预测模式的率失真代价与最佳预测模式之间的关系从而提早终止预测模式判决过程;二是通过8×8大小的编码单元与其16×16大小的父编码单元最佳预测模式之间的关系从而提早终止预测模式判决过程。虽然Do-Kyoung Kwon等人的方法减少了计算复杂度,但是屏幕内容图像的编码时间依然是满足屏幕共享等应用实时性需求的一大障碍。Yong-Jo Ahn等人在“Fast intra coding mode decision for screen contents coding”中提出了一种针对屏幕内容编码的预测模式快速决定方法。为了减少编码时间,该方法将帧内块拷贝IBC_{2N×2N}预测模式的编码过程进行了调整,将IBC_{2N×2N}预测模式放在了传统帧内预测模式和其他帧内块拷贝预测分割预测模式的前面。目前该方法已被屏幕内容编码标准采纳,并作为提案JCTVC-S0067的一部分添加在屏幕内容编码标准中。然而,该方法减少的时间复杂度是有限的,因此需要提出更快的屏幕内容编码预测方法。

发明内容

[0004] 本发明的目的在于针对上述现有技术的不足,提出一种基于边缘点密度的屏幕内容编码预测模式快速选择方法,以在保持一定的视频编码效率的前提下减少编码时间,提高屏幕内容编码预测模式的选择速度。

[0005] 本发明的基本思想是:根据SCC最佳预测模式很有可能是IBC预测模式,故将SCC预测模式的编码顺序调整为IBC_{2N×2N}预测模式、IBC_{2N×N}预测模式、IBC_{N×2N}预测模式、

IBC_N×N预测模式、传统帧内预测模式、PLT预测模式,每进行完一种预测模式的编码过程,均对编码块标志CBF和块矢量差异BVD进行检测,判断其是否分别为0和(0,0),如果是,则跳过后续预测模式的编码过程,否则就顺序进行后续预测模式的编码过程,直到选择出最佳预测模式;根据在编码包含文本等高频率成分的屏幕内容视频时,预测模式编码顺序的调整确实可以减少计算复杂度,但在编码不包含文本等高频率成分的视频序列时,这种方法却会增加相应的计算复杂度的情况,提出基于边缘点密度的SCC预测模式快速选择方法区分出动画视频序列,并分别对动画视频序列和其他类型视频序列进行相应的快速编码处理,实现减少编码时间的目的。其技术方案包括如下:

[0006] (1) 输入视频图像,设置标志位Flag为0;

[0007] (2) 判断当前编码单元所在的视频图像帧号是否属于[0,9]区间,如果是,则执行步骤(3),否则执行步骤(8);

[0008] (3) 判断当前编码单元CU的尺寸是否为8×8,如果是,则用Sobel算子对当前编码单元进行卷积,并计算其边缘点密度,执行步骤(4),否则执行步骤(8);

[0009] (4) 判断当前编码单元的边缘点密度是否处于[0.9,1]区间,如果是,则统计当前编码单元的最佳预测模式是帧内Intra预测模式情况下的概率,并进行累加,执行步骤(5),否则执行步骤(8);

[0010] (5) 判断当前视频图像帧号是否等于9,如果是,则记前10帧视频图像的累加概率为P,执行步骤(6),否则将概率累加,执行步骤(8);

[0011] (6) 根据步骤(5)的累加概率P,得到与量化参数QP相对应的不同概率阈值TH1;

[0012] (7) 判断步骤(5)的累加概率P是否小于步骤(6)的概率阈值TH1,如果是,则执行步骤(8),否则执行步骤(12);

[0013] (8) 调整屏幕内容编码SCC预测模式编码的顺序:首先进行帧内块拷贝IBC_{2N}×2N预测模式的编码;再检测编码块标志CBF和块矢量差异BVD是否分别为0和(0,0),如果是,则将标志位Flag设置为1,执行步骤(12),否则执行步骤(9),其中,2N×2N是一种预测分割模式,N的值取4或8或16或32,IBC_{2N}×2N预测模式表示在进行帧内块拷贝编码时,一个编码单元对应一个预测单元;

[0014] (9) 进行帧内块拷贝IBC_{2N}×N预测模式的编码,之后检测编码块标志CBF和块矢量差异BVD是否分别为0和(0,0),如果是,则将标志位Flag设置为1,执行步骤(12),否则执行步骤(10),其中,2N×N是一种预测分割模式,IBC_{2N}×N预测模式表示在进行帧内块拷贝编码时,一个编码单元对应2个水平划分的预测单元;

[0015] (10) 进行帧内块拷贝IBC_N×2N预测模式的编码,之后检测编码块标志CBF和块矢量差异BVD是否分别为0和(0,0),如果是,则将标志位Flag设置为1,执行步骤(12),否则执行步骤(11),其中,N×2N是一种预测分割模式,IBC_N×2N预测模式表示在进行帧内块拷贝编码时,一个编码单元对应2个垂直划分的预测单元;

[0016] (11) 进行帧内块拷贝IBC_N×N预测模式的编码,之后检测编码块标志CBF和块矢量差异BVD是否分别为0和(0,0),如果是,则将标志位Flag设置为1,执行步骤(12),否则直接执行步骤(12),其中,N×N是一种预测分割模式,IBC_N×N预测模式表示在进行帧内块拷贝编码时,一个编码单元对应4个预测单元;

[0017] (12) 进行调色板预测模式的编码,之后判断标志位Flag是否为1,如果是,执行步

骤(13),否则进行帧内Intra预测模式的编码,执行步骤(13);

[0018] (13)判断当前编码单元是否为视频图像的最后一个编码单元,如果是,则结束SCC预测模式的编码,否则返回步骤(2)。

[0019] 本发明具有如下优点:

[0020] 本发明由于充分利用CU的边缘点密度与其最佳预测模式的关系,区分开动画视频序列和其他类型视频序列,分别对两者进行快速编码:对于动画视频序列,跳过帧内块拷贝预测模式的编码,进行帧内Intra预测模式和调色板预测模式的编码;对于其他类型视频序列,调整SCC预测模式的编码顺序为IBC_2N×2N预测模式、IBC_2N×N预测模式、IBC_N×2N预测模式、IBC_N×N预测模式、调色板PLT预测模式、帧内Intra预测模式,因而与Do-Kyoung Kwon等人的方法相比具有良好的兼容性;相比Yong-Jo Ahn等人的方法,能在保证编码效率的前提下,减少更多编码时间,进一步加快编码速度。

附图说明

[0021] 图1是本发明的实现流程图。

具体实施方式

[0022] 下面结合附图对本发明的实施过程进行详细描述。实施过程以本发明的两大基本思想为核心进行实施,详细的实施方式和具体操作过程如下所述。

[0023] 步骤一:输入视频图像,设置标志位Flag为0。

[0024] 输入的视频图像是屏幕内容编码SCC标准视频图像,分为运动的文本图片TGM类型、动画A类型、混合M类型,其中包括10个测试视频图像,分别为FlyingGraphics,Desktop,Console,MissionControlClip3,Web_Browsing,Map,Programming,SlideShow,Robot,wordEditing。

[0025] 步骤二:判断当前编码单元所在的视频图像帧号是否属于[0,9]区间,如果是,则执行步骤三,否则,执行步骤八。

[0026] 步骤三:区分开8×8大小和其他大小的编码单元CU,并分别进行编码。

[0027] 判断当前编码单元CU的尺寸是否为8×8,如果是,则执行步骤四,否则,执行步骤九。

[0028] 步骤四:用Sobel算子对当前编码单元进行如下卷积和边缘点密度计算,之后执行步骤五。

[0029] 4.1) 分别计算第i行第j列像素 $p_{i,j}$ 在垂直方向上的差异 $D_{x_{i,j}}$ 和在水平方向上的差异 $D_{y_{i,j}}$,即:

$$[0030] \quad D_{x_{i,j}} = p_{i+1,j-1} + 2 * p_{i+1,j} + p_{i+1,j+1} - p_{i-1,j-1} - 2 * p_{i-1,j} - p_{i-1,j+1}$$

$$[0031] \quad D_{y_{i,j}} = p_{i-1,j+1} + 2 * p_{i,j+1} + p_{i+1,j+1} - p_{i-1,j-1} - 2 * p_{i,j-1} - p_{i+1,j-1}$$

[0032] 其中, $p_{i+1,j-1}$ 是编码单元的第i+1行第j-1列像素, $p_{i+1,j}$ 是编码单元的第i+1行第j列像素, $p_{i+1,j+1}$ 是编码单元的第i+1行第j+1列像素, $p_{i-1,j-1}$ 是编码单元的第i-1行第j-1列像素, $p_{i-1,j}$ 是编码单元的第i-1行第j列像素, $p_{i-1,j+1}$ 是编码单元的第i-1行第j+1列像素, $p_{i,j-1}$ 是编码单元的第i行第j-1列像素, $p_{i,j+1}$ 是编码单元的第i行第j+1列像素;

[0033] 4.2) 计算第*i*行第*j*列像素 $p_{i,j}$ 的梯度矢量 $D_{i,j}$:

$$[0034] \quad D_{i,j} = \{D_{x,j}, D_{y,i}\};$$

[0035] 4.3) 计算第*i*行第*j*列像素 $p_{i,j}$ 的梯度矢量幅值 $A_{i,j}$:

$$[0036] \quad A_{i,j} = |D_{x,j}| + |D_{y,i}|;$$

[0037] 4.4) 判断 $p_{i,j}$ 的梯度矢量幅值 $A_{i,j}$ 是否大于边缘阈值TH2:如果是,则此像素是边缘点像素,否则此像素不是边缘点像素,边缘阈值TH2与量化参数QP有关,若QP=22,TH2=230;若QP=27,TH2=260;若QP=32,TH2=280;若QP=37,TH2=300;

[0038] 4.5) 计算 8×8 大小的编码单元CU的边缘点密度EgDen:

$$[0039] \quad \text{EdgeDen} = \text{EdgeNum} / (\text{Sum} * 1.0)$$

[0040] 其中,EgNum表示 8×8 大小的CU包含的边缘点像素数,Sum表示当前CU像素总数,EgDen表示 8×8 大小的CU的边缘点密度。

[0041] 步骤五:判断当前编码单元的边缘点密度是否处于 $[0.9, 1]$ 区间,如果是,则统计当前编码单元的最佳预测模式是帧内Intra预测模式情况下的概率,并对当前编码单元的概率进行累加后,得到当前帧视频图像的概率,执行步骤六,否则,执行步骤八。

[0042] 统计当前编码单元的最佳预测模式是帧内Intra预测模式情况下的概率,按如下公式进行:

$$[0043] \quad \text{pr} = \frac{\text{cnt}_{\text{intra}}}{\text{cnt}_{\text{intra}} + \text{cnt}_{\text{ibc}}} \times 100\%$$

[0044] 其中,pr表示当前编码单元的最佳预测模式是帧内Intra预测模式情况下的概率,cnt_{intra}表示当前编码单元的最佳预测模式是帧内Intra预测模式情况下的次数,cnt_{ibc}表示当前编码单元的最佳预测模式是帧内块拷贝预测模式情况下的次数,该帧内块拷贝预测模式包括IBC_2N×2N预测模式、IBC_2N×N预测模式、IBC_N×2N预测模式、IBC_N×N预测模式四种候选预测模式。

[0045] 步骤六:判断当前视频图像帧号是否等于9,如果是,则记前10帧视频图像的累加概率为P,执行步骤七,否则将当前帧视频图像的概率累加,执行步骤八。

[0046] 步骤七:根据步骤六的累加概率P,得到与量化参数QP相对应的不同概率阈值TH1,并将该概率阈值TH1与步骤六的累加概率P进行比较:如果TH1大于P,则执行步骤八,否则执行步骤十二。

[0047] 其中,概率阈值TH1与量化参数QP有关,若QP=22,则TH1=0.9;若QP=27,则TH1=0.83;若QP=32,则TH1=0.85;若QP=37,则TH1=0.85。

[0048] 步骤八:进行帧内块拷贝IBC_2N×2N预测模式的编码;再检测编码块标志CBF和块矢量差异BVD是否分别为0和(0,0),如果是,则将标志位Flag设置为1,执行步骤十二,否则执行步骤九。

[0049] 其中,2N×2N是一种预测分割模式,N的值取4或8或16或32,IBC_2N×2N预测模式表示在进行帧内块拷贝编码时,一个编码单元对应一个预测单元。

[0050] 步骤九:进行帧内块拷贝IBC_2N×N预测模式的编码;再检测编码块标志CBF和块矢量差异BVD是否分别为0和(0,0),如果是,则将标志位Flag设置为1,执行步骤十二,否则执行步骤十。

[0051] 其中, $2N \times N$ 是一种预测分割模式, $IBC_{2N \times N}$ 预测模式表示在进行帧内块拷贝编码时, 一个编码单元对应2个水平划分的预测单元。

[0052] 步骤十: 进行帧内块拷贝 $IBC_{N \times 2N}$ 预测模式的编码; 再检测编码块标志 CBF 和块矢量差异 BVD 是否分别为 0 和 (0, 0), 如果是, 则将标志位 Flag 设置为 1, 执行步骤十二, 否则执行步骤十一。

[0053] 其中, $N \times 2N$ 是一种预测分割模式, $IBC_{N \times 2N}$ 预测模式表示在进行帧内块拷贝编码时, 一个编码单元对应2个垂直划分的预测单元。

[0054] 步骤十一: 进行帧内块拷贝 $IBC_{N \times N}$ 预测模式的编码; 再检测编码块标志 CBF 和块矢量差异 BVD 是否分别为 0 和 (0, 0), 如果是, 则将标志位 Flag 设置为 1, 执行步骤十二, 否则直接执行步骤十二。

[0055] 其中, $N \times N$ 是一种预测分割模式, $IBC_{N \times N}$ 预测模式表示在进行帧内块拷贝编码时, 一个编码单元对应4个预测单元。

[0056] 步骤十二: 进行调色板预测模式的编码; 再判断标志位 Flag 是否为 1, 如果是, 执行步骤十三, 否则进行帧内 Intra 预测模式的编码, 执行步骤十三。

[0057] 步骤十三: 判断当前编码单元是否为视频图像的最后一个编码单元, 如果是, 则结束屏幕内容编码 SCC 预测模式的编码, 否则返回步骤二。

[0058] 本发明通过上述步骤减少了编码时间, 其效果可通过以下实验进一步说明:

[0059] 1) 实验条件

[0060] 测试平台: HM-16.2+SCM-3.0;

[0061] 测试环境: CPU: Intel (R) Xeon (R) CPU E5-2650V2@2.60GHz 2.60GHz, RAM: 64.0GB

[0062] 实验数据: SCC 测试序列包括 10 个测试序列, 分别为 FlyingGraphics, Desktop, Console, MissionControlClip3, Web_Browsing, Map, Programming, SlideShow, Robot, WordEditing。有关视频序列的详细信息如表 1。

[0063] 时域预测结构: 全 I 帧结构。

[0064] 量化参数 QP: 22, 27, 32, 37。

[0065] 表 1 屏幕内容视频序列的详细信息

[0066]

分辨率	序列名称	类别	帧率 fps	待编码的 帧数
1920×1080	FlyingGraphics_1920×1080_60_8bit	TGM	60	0-299
	Desktop_1920×1080_60_8bit	TGM	60	0-599
	Console_1920×1080_60_8bit	TGM	60	0-599
	MissionControlClip3_1920×1080_60p_8b444	M	60	0-599
1280×720	Web_Browsing_1280×720_30_8bit	TGM	30	0-299
	Map_1280×720_60_8bit	TGM	60	0-599
	Programming_1280×720_60_8bit	TGM	60	0-599
	SlideShow_1280×720_20_8bit	TGM	20	0-499
	Robot_1280×720_30_8bit	A	30	0-299
	WordEditing_1280×720_8bit_444	TGM	60	0-599

[0067] 2) 实验内容以及结果

[0068] 2.1) 仿真1

[0069] Do-Kyoung Kwon等人的快速方法目前已被屏幕内容编码标准采纳。因此,在设计本发明快速方法时,必须要考虑到快速方法与Do-Kyoung Kwon等人方法的兼容性。在与Do-Kyoung Kwon等人方法保持一定兼容性的条件下,本发明快速方法与Do-Kyoung Kwon等人方法进行比较的实验结果如表2~表4。

[0070] 表2QP=22和QP=27时比较结果

[0071]

类别	视频序列名称	QP=22			QP=27		
		$\Delta Time$ (%)	$\Delta bits$ (%)	$\Delta PSNR$ (dB)	$\Delta Time$ (%)	$\Delta bits$ (%)	$\Delta PSNR$ (dB)
TGM 类型	FlyingGraphics	-26.82	-0.10	-0.0103	-7.79	-0.15	-0.0151
	Desktop	-24.77	-0.09	-0.0355	-26.79	-0.08	-0.0087
	Console	-17.95	-0.09	0.0005	-15.90	-0.21	0.0035
	Web_Browsing	-13.50	-0.21	-0.123	-26.42	-0.06	-0.0033
	Map	-3.94	-0.07	0.0024	-10.01	0.07	-0.0126
	Programming	-11.07	-0.03	-0.0143	-8.47	-0.10	-0.0182
	SlideShow	-10.05	0.11	-0.0092	-42.95	0.07	-0.0156
	WordEditing	-25.07	-0.04	-0.0012	-20.96	0.00	-0.0256
M 类型	MissionControlClip3	-13.96	-0.05	-0.0111	-24.27	-0.07	-0.01
A 类型	Robot	-36.68	0.66	-0.0052	-64.43	1.16	-0.0042
平均		-18.38	0.009	-0.0206	-24.80	0.063	-0.0109

[0072] 表3QP=32和QP=37时比较结果

[0073]

类别	视频序列名称	QP=32			QP=37		
		$\Delta Time$ (%)	$\Delta bits$ (%)	$\Delta PSNR$ (dB)	$\Delta Time$ (%)	$\Delta bits$ (%)	$\Delta PSNR$ (dB)
TGM 类型	FlyingGraphics	-24.84	-0.21	-0.0236	-29.48	-0.27	-0.0279
	Desktop	-22.96	-0.14	-0.0175	-18.06	-0.18	-0.0266
	Console	-30.51	-0.39	0.0127	-18.30	-0.28	-0.0174
	Web_Browsing	-22.33	0.36	0.1559	-34.65	0.05	0.0291
	Map	-12.22	0.55	-0.0198	-10.64	0.60	-0.0249
	Programming	-22.68	-0.06	-0.0176	-25.85	-0.06	-0.0133
	SlideShow	-44.99	0.14	-0.0112	-40.30	-0.35	-0.0072
	WordEditing	-24.60	-0.36	0.0394	-22.61	-0.38	-0.0188
M 类型	MissionControlClip3	-25.79	-0.04	-0.0079	-28.91	-0.21	-0.0135
A 类型	Robot	-25.59	2.18	-0.0006	-7.65	2.32	-0.0125
平均		-25.65	0.203	0.0109	-23.65	0.124	-0.0133

[0074] 表4比较的BD-PSNR和BD-BR结果

类别	视频序列名称	性能指标		
		$\Delta Time$ (%)	BD-PSNR (dB)	BD-BR (dB)
[0075] TGM 类型	FlyingGraphics	-22.23	0.007309	-0.0486
	Desktop	-23.15	0.01863	-0.06047
	Console	-20.66	0.089329	-0.29994
	Web_Browsing	-24.22	0.037741	-0.19131
	Map	-9.2	-0.04449	0.423525
	Programming	-17.02	-0.00923	0.08613
	SlideShow	-34.57	-0.01562	0.20096
	WordEditing	-23.31	0.030507	-0.24137
M 类型	MissionControlClip3	-23.23	-0.00264	0.026541
A 类型	Robot	-33.59	-0.06556	1.574937
平均		-23.12	0.004597	0.147041

[0076] 表2~表4实验数据中的“-”表示在时间方面提速了或在比特数方面减少了或在亮度像素Y-PSNR方面减少了, $\Delta Time$ 、 $\Delta bits$ 和 $\Delta PSNR$ 分别表示编码时间、比特数和亮度像素Y-PSNR的大小变化, BD-PSNR和BD-BR是用来衡量本发明快速方法编码效率的, 前者的单位为dB, 表示在保持编码码率相同的情况下亮度峰值信噪比Y-PSNR之间的差异, 后者是比特百分比。

[0077] 从表2~表4的实验数据可以看出, 对于屏幕内容视频序列, 本发明方法在不同的量化参数下均实现了较好的结果。当量化参数QP等于22时, FlyingGraphics、Desktop和Robot视频序列节省了更多的编码时间, 分别节省了26.82%、24.77%和36.68%的编码时间; 当量化参数QP等于27时, Desktop、SlideShow和Robot视频序列节省了更多的编码时间, 分别节省了26.79%、42.95和64.43的编码时间; 当量化参数QP等于32时, Console和SlideShow视频序列节省了更多的编码时间, 分别节省了30.51%和44.99%的编码时间; 当量化参数QP等于37时, Web_Browsing和SlideShow视频序列节省了更多的编码时间, 分别节省了34.65%和40.30%的编码时间。在平均情况下, 本发明减少了23.12%的编码时间, BD-PSNR增加0.004597。

[0078] 2.2) 仿真2

[0079] 在与Do-Kyoung Kwon等人方法保持一定兼容性的条件下, 将本发明与Yong-Jo Ahn等人方法进行比较, 其实验结果如表5~表7。

[0080] 表5 QP=22和QP=27时本发明与Yong-Jo Ahn方法比较结果

[0081]

类别	视频序列名称	QP=22			QP=27		
		Δ Time (%)	Δ bits (%)	Δ PSNR (dB)	Δ Time (%)	Δ bits (%)	Δ PSNR (dB)
TGM 类型	FlyingGraphics	-15.44	0.12	-0.0019	-8.13	0.15	0.0023
	Desktop	13.15	-0.11	0.0303	0.58	-0.12	0.0799
	Console	-5.88	0.05	0.0098	11.26	0.06	0.0506
	Web_Browsing	17.89	-0.38	-0.0768	31.99	-0.40	0.0193
	Map	-15.24	0.41	-0.009	-13.83	0.31	-0.0258
	Programming	-3.37	0.02	0.0501	-6.03	-0.02	0.119
	SlideShow	-5.63	-1.09	-0.003	-11.32	-1.53	0.0036
	WordEditing	-4.24	-0.06	0.1187	-1.51	-0.18	0.1007
M 类型	MissionControlClip3	-11.67	-0.51	0.0227	-7.78	-0.34	0.0363
A 类型	Robot	-56.05	0.23	-0.0098	-55.36	0.66	0.0012
平均		-8.648	-0.132	0.01311	-6.01	-0.141	0.03871

[0082] 表6 QP=32和QP=37时本发明与Yong-Jo Ahn方法比较结果

[0083]

类别	视频序列	QP=32			QP=37		
		Δ Time (%)	Δ bits (%)	Δ PSNR (dB)	Δ Time (%)	Δ bits (%)	Δ PSNR (dB)
TGM 类型	FlyingGraphics	-13.24	0.12	0.0141	-19.28	0.08	0.0139
	Desktop	8.16	-0.15	0.0541	-0.17	-0.15	-0.0015
	Console	-1.71	0.12	0.0537	1.34	0.45	-0.0048
	Web_Browsing	16.36	-0.55	-0.0366	33.49	-1.00	0.014
	Map	-2.86	-0.18	-0.0411	-8.02	-0.72	0.0007
	Programming	1.71	-0.26	0.0619	11.35	-0.60	0.0442
	SlideShow	18.37	-2.02	-0.0542	8.89	-4.32	0.0207
	WordEditing	-3.33	-0.44	0.1391	12.78	-0.43	0.0878
M 类型	MissionControlClip3	-8.61	-0.79	0.037	-14.18	-1.21	0.0236
A 类型	Robot	-63.58	1.03	0.0154	-50.29	-0.03	0.0278
平均		-4.87	-0.312	0.02434	-2.41	-0.793	0.02264

[0084] 表7本发明与Yong-Jo Ahn方法比较的平均结果

[0085]

类别	视频序列名称	性能指标		
		$\Delta Time$ (%)	BD-PSNR (dB)	BD-BR (dB)
TGM 类型	FlyingGraphics	-14.02	-0.01051	0.072574
	Desktop	5.43	0.092311	-0.30126
	Console	1.25	0.001717	0.01084
	Web_Browsing	24.93	0.075793	-0.45007
	Map	-9.99	-0.03031	0.292757
	Programming	0.91	0.098816	-0.91564
	SlideShow	2.55	0.147219	-1.82322
	WordEditing	0.93	0.158006	-1.15467
M 类型	MissionControlClip3	-10.56	0.098247	-0.98435
A 类型	Robot	-56.32	-0.01742	0.395667
平均		-5.49	0.061387	-0.4858

[0086] 从表5~表7的实验数据可以看出,相比Yong-Jo Ahn等人方法,本发明在不同量化参数条件下均实现了编码时间的减少,当量化参数QP等于22时,Robot、FlyingGraphics和Map视频序列节省了更多的编码时间,分别节省了56.05%和15.44%和15.24%的编码时间;当量化参数QP等于27时,Robot和Map视频序列节省了更多的编码时间,分别节省了55.36%和13.83%的编码时间;当量化参数QP等于32时,FlyingGraphics和Robot视频序列节省了更多的编码时间,分别节省了63.58%和13.24%的编码时间;当量化参数QP等于37时,FlyingGraphics和Robot视频序列节省了更多的编码时间,分别节省了19.28%和50.29%的编码时间,在平均情况下,相比Yong-Jo Ahn等人方法,本发明节省了5.49%的编码时间,同时BD-PSNR增加了0.061387,从而证明了本发明快速方法的优越性。

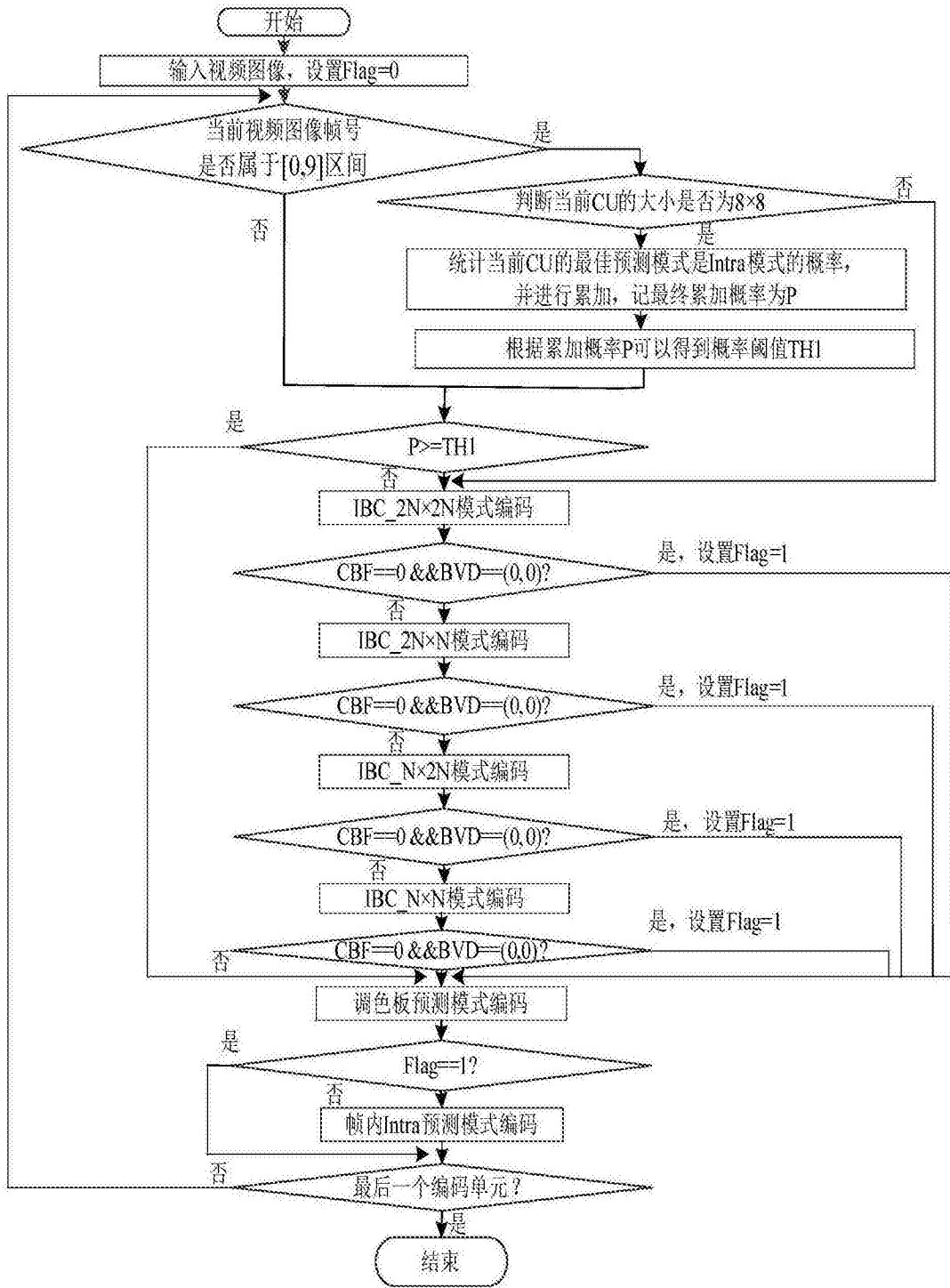


图1