

[19] 中华人民共和国国家知识产权局

[51] Int. Cl.

H04N 13/00 (2006.01)



[12] 发明专利说明书

专利号 ZL 200380101874.6

[45] 授权公告日 2010 年 1 月 20 日

[11] 授权公告号 CN 100584039C

[22] 申请日 2003.10.1

[21] 申请号 200380101874.6

[30] 优先权

[32] 2002.10.23 [33] EP [31] 02292623.2

[86] 国际申请 PCT/IB2003/004363 2003.10.1

[87] 国际公布 WO2004/039086 英 2004.5.6

[85] 进入国家阶段日期 2005.4.22

[73] 专利权人 皇家飞利浦电子股份有限公司

地址 荷兰艾恩德霍芬

[72] 发明人 S·奥伯格 Y·皮卡尔德

[56] 参考文献

US6084979A 2000.7.4

CN1158684A 1997.9.3

US2002/0061131A1 2002.5.23

US5982390A 1999.11.9

审查员 黄海云

[74] 专利代理机构 中国专利代理(香港)有限公司

代理人 程天正 王忠忠

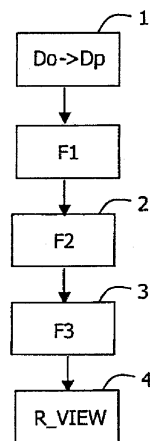
权利要求书 2 页 说明书 10 页 附图 4 页

[54] 发明名称

3D 数字视频信号的后处理方法

[57] 摘要

本发明涉及一种 3D 数字视频信号的后处理方法，所述数字视频信号具有多个带有相关联的视差图的视图。本发明的特征在于，所述方法包括从原始视差图生成投影视差图的第一步骤，以及在所述投影视差图内对孔进行填充的第二步骤，所述第一步骤包括通过对所述投影视差图上的孤立投影像素进行第一过滤来消除所述孤立投影像素的子步骤。



1. 一种数字视频信号的后处理方法, 所述数字视频信号具有多个带有相关联的视差图的视图, 其特征在于, 所述方法包括从原始视差图生成投影视差图的第一步骤, 以及在所述投影视差图内对孔进行填充的第二步骤, 所述第一步骤包括通过对所述投影视差图上的孤立投影像素进行第一过滤来消除所述孤立投影像素的子步骤。

2. 如权利要求 1 所述的数字视频信号的后处理方法, 其特征在于, 所述第一过滤是通过选择孤立投影像素来执行的, 所述孤立投影像素被孔所包围。

3. 如权利要求 1 所述的数字视频信号的后处理方法, 其特征在于, 所述第一过滤是通过选择孤立投影像素来执行的, 所述孤立投影像素不具有与周围的像素一致的特性。

4. 如权利要求 2 或 3 所述的数字视频信号的后处理方法, 其特征在于, 所述第一过滤应用于投影视差图的两个投影方向。

5. 如权利要求 2 或 3 所述的数字视频信号的后处理方法, 其特征在于, 所述第一过滤包括一个改变视差图投影大小的子步骤。

6. 如权利要求 2 或者 3 所述的数字视频信号的后处理方法, 其特征在于, 所选择的孤立投影像素被设置为孔, 并且导致这些被选择的孤立投影像素的原始视差图上的对应的像素被标记为错误, 从而导致一个修改后的原始视差图。

7. 如权利要求 1 所述的数字视频信号的后处理方法, 其特征在于, 所述第二步骤包括以下步骤:

- 选择一个比一个阈值 (T_2) 大的孔,
- 对所述选择的孔的一个所选择的边界像素值进行中值过滤, 过滤时考虑所选择的孔的所选择的边界像素值的周围投影像素值, 以及
- 用所述中值过滤的结果填充所选择的孔。

8. 如权利要求 7 所述的数字视频信号的后处理方法, 其特征在于, 当所选择的边界像素通过过滤后得到一个新值时, 导致所述选择的边界像素的原始视差图上的对应的原始视差图上的像素被标记为错误, 并且与具有新值的边界像素相对应的像素被分派予所述新值, 从而导致一个修改后的原始视差图。

9. 如权利要求 6 所述的数字视频信号的后处理方法, 其特征在于,

所述方法还包括用修改后的原始视差图的错误像素的边界像素值之一来填充所述错误像素的第三步骤，所述边界像素值具有最接近于原始视差图上的原始像素的值。

10. 如权利要求9所述的数字视频信号的后处理方法，其特征在于，从修改后的原始视差图来重建视图。

3D 数字视频信号的后处理方法

发明领域

本发明涉及一种数字视频信号的后处理方法，所述数字视频信号具有多个带有相关联的视差图（disparity map）的视图。

所述方法例如可以用于 MPEG 标准内的 3D 视频应用的视频通信系统。

发明背景

大多数 3D 视频信号表示（称为 3D 场景）依赖于深度图和视差图。通常，从给定的 3D 场景的一组图像开始（所述图像对应于不同的观察点），每一幅图像可能具有不同的相关联的特性，例如深度图和纹理。一个观察点的深度图是灰度图像，其每个像素包含到拍摄该场景的摄影机的距离。当要生成场景的一个新观察点（更简单地称为视图）时，如果给出另外的观察点、其深度图、固有的摄影机参数以及当摄影机从该观察点移动到新观察点时所经历的变化的参数（位移，旋转，镜头变化），就可能计算新观察点的某些区域。这个过程叫做“重建”，所创建的新观察点叫做重建视图（或重建图像）。如果场景中的一点 P 从两个观察点都可以看到，那么通过一个转换矢量（translation vector）可以从该点在原始观察点中的像素坐标得到该点在新观察点中的像素坐标。这些矢量叫做视差矢量。如文献“three dimensional computer vision（三维计算机视觉）- O.Faugeras MIT Press 1993”所公开的，投影几何结果在视差图的视差矢量与深度图的深度值之间建立一个简单关系。

在视频信号的传输过程中，本领域技术人员熟知的多视图或立体编码方案，一般通过压缩覆盖观察点特定范围所需要的纹理和深度图来进行编码。虽然纹理可以用标准方法进行编码（这在有损编码的情况下潜在地将导致公知的假象问题），深度（或视差）编码则更复杂一些：对于一个已经编码的深度图，要具有一个与原始深度图在视觉上相似的缩图（aspect）并不意味着其必须具有同样的重建属性。在新视图生成过程中，点或者区域可能被转换到错误的位置（由于错误的

视差矢量)。这将产生可能比已编码图所提出的“视觉”缩图更明显的纹理不连续。而且密集的深度图是非常大的文件,如果想把深度图的尺寸控制在合理的范围(也就是小于纹理比特率的20%),则有损压缩几乎是不可避免的。因此必须处理假象和不适当的深度/视差值,并且必须设计解码视频信号后的后处理和增强算法。

在MPEG-4标准内,深度图可以用多辅助部件工具(MAC)进行编码(参见文献《Amendment 1: Visual extensions, ISO/IEC JTC 1/SC29/WG 11 N 3056, December 1999》),其中在块的基础上对深度图进行DCT编码,这与本领域技术人员熟知的经典亮度图像编码相似。没有提到对潜在的假象的具体处理,但是如前面提到的,传统的MPEG工具有利于纹理,对于深度图则不必然如此。因此,例如在纹理重建图像中,这可能导致连同孤立的纹理像素一同出现的模糊边缘,此外,这两种效应在其后的观察点期间是在时间上不一致的。

发明目的和概要

因此,本发明的一个目的在于提供一种数字视频信号的后处理方法,所述数字视频信号具有多个带有相关联的视差图的视图,所述方法能够校正深度图(或视差图)的编码假象。

为达到上述目的,提供了一种如权利要求1所述的方法。

正如将会详细地看到的那样,通过去掉孤立的投影像素,投影的和原始的视差图中的不一致的值被消除了。

根据本发明的方法的一个实施例包括权利要求2所限定的步骤。

根据本发明的方法的另一实施例包括权利要求3所限定的步骤。

根据本发明的方法的另一实施例包括权利要求4所限定的步骤。

根据本发明的方法的另一实施例包括权利要求5所限定的步骤。

根据本发明的方法的另一实施例包括权利要求6所限定的步骤。

通过考虑孔边界周围的投影像素,有利于避免用来自由于压缩所导致的错误的孔边界值的错误值来填充所述孔。

根据本发明的方法的另一实施例包括权利要求7所限定的步骤。

这有利于避免对于空间上不一致的未定义值的插值。

附图简述

通过阅读下面参照附图进行的详细说明，本发明的上述和其他目的、特点和优点将会变得更加清楚，其中：

图 1 是根据本发明的视频信号后处理方法的示意图；

图 2 绘出了投影的视差图中的一些孔，这些孔是由图 1 的视频信号后处理方法的第一个步骤产生的；

图 3 绘出了在图 1 的视频信号后处理方法的第一个步骤中，过滤投影的视差图中的孔的过程；

图 4 示出了在图 1 的视频信号后处理方法的第二个步骤中，填充投影的视差图中的孔的过程；以及

图 5 示出了在图 1 的视频信号后处理方法的第三个步骤中，过滤未定义像素值的过程。

说明书全文中对于同样的单元将采用对应的附图标记。

发明的详细说明

在以下说明中，本领域技术人员熟知的功能或构造将不详细描述，以免让不必要的细节模糊本发明。

本发明涉及一种数字视频信号的后处理方法。

所述方法可以在用于 MPEG4 中的 3D 视频应用的视频通信系统内使用。

一个 3D 视频信号包括多个观察点，其具有不同的相关联的特性，例如形状、纹理、运动矢量、视差图、深度图、颜色等等。

当传输一个视频信号时，要对其进行编码。在编码过程中，其不同的特性被用压缩算法编码，尤其是视差图和深度图被采用压缩算法进行编码。这种压缩可能导致带有假象的受损的视差图和深度图。

视频信号传输后要对其进行解码，然后所有的视图在包括后处理解码视差的算法中被重建。当要生成场景的一个新观察点时，如果给出另外的观察点、其深度图、固有的摄影机参数以及当摄影机从该观察点移动到新观察点时所经历的变化的参数（位移，旋转，镜头变化），就可能计算新观察点的某些区域。

后处理方法是基于这样一个事实：当从另一个视图及其受损的深度图重建一个新视图时，如将进一步描述的样，深度图经历一些变换（例如投影），使得比使用原始解码图时更容易检测有问题的值。

首先, 由于深度和视差之间存在对应关系(投影几何)这一事实, 我们考虑视差图。由于总是有可能具有都沿着同样方向的视差矢量(例如本领域技术人员熟知的以及在文献“O.Faugeras, Three-dimensional computer vision (三维计算机视觉), MIT Press, 1993”中描述的, 通过根据共极性(epipolar)约束来调整原始立体对), 我们将用水平视差矢量(视频摄影机的“平行立体设置”的常见的情况)来举例说明这个方法。当然, 这绝不是对本发明的限制。这种情况下, 视差矢量由单个数值定义, 因此, 以下提到时称其为“视差值”。

在下文中, I_O 表示原始纹理图像, I_N 表示新的重建的视图, d 表示视差图, $d(x,y)$ 表示像素 (x,y) 处的视差值。新的视图位于原始视图的左侧或右侧以及具有特定基线(或多或少远离原始视图)这一事实, 将由系数 α 表示。对于视图中的每个像素 (x,y) , 有下列关系:

$$I_O(x,y)=I_N(x+\alpha.d(x,y),y)$$

重建过程可以以几种方式完成, 通常的一种方式(参见文献“Disparity field and depth map coding for multi-view 3D image generation”, by D.Tzovaras, N.Grammalidis, G.Strinzis, Signal Processing: Image Communication II (1998) 205-230 页)可以分为以下几个主要步骤, 下面将详细描述:

1. 投影视差图的生成
2. 投影视差图上的孔的填充
3. 基于孔填充后的投影视差图生成最终的 I_N 视图

在重建过程的基础上, 如下地完成对观察点的后处理并如图 1 所示。

在第一步骤 1), 从一个观察点的原始视差图生成投影视差图。视图上的 N 个像素被投影到未来视图的一个像素上, 其中 N 可以取任意值:

-取 0, 如果没有像素被投影的话, 在此情况下重建视图的对应像素位于新视图中被原始视图遮蔽的部分(遮蔽区域),

-取 1, 如果只有一个(或更多个)像素被投影到这个像素上的话, 这意味着原始视图的某些像素具有将它们投影到重建视图上的相同像素上的视差值。

因此, 视差值的列表与未来视图的该像素对应。这些与重建视图

上的每个像素对应的列表组就是投影视差图。

本发明的主要思想是，随着投影过程的继续，投影视差图应该具有或多或少与原始图相同的规则性。这也意味着如果原始视差图有一些不规则性，则投影视差图也具有同样的不规则性。

如果考虑投影到每个像素上的点数，也应该会有某种规则性。尤其是，在孔的中间，不应有孤立的投影像素。然而，由于用于视差图编码的 DCT 量化过程，有些像素值改变了，并且一旦被投影，它们不再接近于其原始的周围像素。这些像素叫做孤立的投影像素。如图 2 所示，原始视差图 Do 具有两个其值相同的像素区域 do1 和 do2。投影视差图 Dp 也有两个对应的像素区域 dp1 和 dp2，但是带有“孔”区域 Hp1 和 Hp2。

由于一些压缩假象，有些像素可能会被错误地投影到这些孔区域上。这些像素叫做孤立像素。

所述后处理方法通过使用第一过滤 F1 除去那些孤立的投影像素。在第一非限制性的实施例中，过滤被执行如下。对于投影视差图 Dp 上的每个像素 Pp，被定义为孔的那些周围像素 Pps 的数目被计算（所述概念“周围像素”是依赖于应用的：其可以是以该像素为中心的一个方形邻域、圆形邻域、矩形邻域……）。如果这个数目超过一个特定阈值 T1，例如超过包含在该邻域中的像素数目的一半，则所考虑的这个孤立像素被设为“孔”，并且原始视差图上任何导致这个孤立像素的像素被标记为“错误视差值”。

第一实施例如图 3 所示。原始视差图 Do 上的第一错误像素 Po2，在投影视差图 Dp 的区域 Dp2 上具有一个对应的孤立投影像素 Pp2。原始视差图 Do 上的第二像素 Po11 和第三像素 Po12 具有位于投影视差图 Dp 的区域 Dp1 上的同一个对应的孤立投影像素 Pp1。虚线窗限定了像素 Pp2 的周围像素 Pps（这里定义为孔），使得 Pp2 成为一个孤立像素。这一点对于 Pp1 也同样成立。结果，与 Pp1 和 Pp2 对应的原始视差图 Do 上的像素 Po11、Po12 和 Po2 分别被设为“错误值”。

应该注意的是，上述过滤非常适合量化对视差图的预期效果，所述量化是压缩的一个步骤，也是导致视差图上的假象的原因之一。要注意视差量化错误主要发生在视差图的尖锐的边缘上，这些边缘具有由一条直线（主要在对象边界处）分隔的两个相差很大的灰度级均匀

的区域。就是这些区域在投影视差图上生成大的孔，其适于精确定位所述孤立的“错误视差值”。

在第二非限制性实施例中，过滤被执行如下。就投影像素值的数目而言，通过检查投影图上的其他孤立像素来检测错误值也是可能的。所述孤立的投影像素就是那些与它们周围的像素特性不一致的像素。

举例说，如果一个包括3个值的像素被包括2个值的像素包围，那么其中可能包含有至少一个错误值。通过将这个像素值组与其周围的像素值组进行比较，列表上的错误值就能被检测到。

注意，投影视差图上一个像素的值的数量范围在0到 n (n 决定于原始视差图)之间。孤立像素检测所需要的精度将确定想要处理哪种孤立像素：被孔包围的孤立的像素，然后是具有两个或更多个值的像素，以及被只包括1个值的像素包围的那些像素，依此类推。处理的情况越多，视差图的精度就越高，然而算法也就越为计算机密集 (computer-intensive)。根据所想要的应用的种类，可以在精度和效率之间折衷。

改善这一过滤过程的一种方式是对系数 α 取不同的值进行测试，通过这种测试，可以检测更多的错误值。改变系数 α 有两种不同的方式：

- 通过改变它的符号，即改变投影方向：采用这种方式，在一幅视图中对象的两面都能得到增强。例如，如果一幅视图包括一座山，若做“右”投影，则过滤所述山的右侧的孔，若做“左”投影，则过滤所述山的左侧的孔。

- 通过改变它的大小，利用更大的孔，可以加强视差损害并且能检测其中一些当其大小较小时不够明显的损害。

注意，用不同的 α 值运行过滤器将提高检测效率但是也增大了复杂度。要测试的视差值数目的选择也依赖于应用。

在第二步骤2)，填充投影视差图上的孔。

在重建过程中，跟在投影视差图生成后的第二步骤是通过插值来填充该视差图上的孔。一般来说，插值类型的选择是如此考虑的，即在最后重建的视图中，所述孔与属于背景的东西相对应而不是与前景对象相对应（这是“现实生活 (real life)”的最普通的情形）。下面

给出一个例子。在投影视差域中，通过一个填充过程，在孔的整条线上重复孔边界的视差值。所述填充过程在于取离孔最近的边界视差值，并用其填充所述孔的整条线。

无论如何，插值非常依赖于孔边界的视差值。问题是，当采用压缩过的（也就是说受损的）视差图时，用于该填充过程的孔边界值可能由于压缩算法而明显改变了，因此可能是错误的，此外，从一条线到另一条线的改变方式也有变化：那么，填充过程就是在这些错误的值的基础上执行的，这些错误值将传播到孔的整条线。在视觉上，这将产生重建视图中所称的“磨损边缘（worn edge）”现象。

根据本发明的方法，为孔边界值采用了一种新的特定的后处理过滤器 F2，其可以避免上述问题。所述过滤器应用于孔的边界而不是孔本身。

首先，由于实际上有些孔并没有大到会引起明显的错误插值数值（例如，一个像素宽的孔），因此仅在大于给定（依赖于应用的）阈值 T2 的孔的边界处执行过滤，例如，T2 为 3 个像素。

然后，对于这些选择出来的孔 H，考虑周围的投影像素值（当然不包括孔），在边界像素值上执行中值过滤，如图 4 所示。

在图 4 的例子中，投影视差图 Dp 包括两个区域；一个归为像素 P，另一个归为孔 H。

选择投影视差图 Dp 上的孔 H1。围绕所认为的孔 H1 的边界像素值 Vk1，选取像素值（不是孔）的一个窗口 W。一旦窗口 W 中的所有像素值都被排序，就可以取得中值，并用其取代边界像素值 Vk1。

在通过中值过滤改变像素值 Vk1 后，在孔 H1 本身上用新的像素值 Vk1 执行常规的填充过程。

如果该中值过滤引起边界像素值变化，那么原始视差图边界像素值被相应地标记为错误值，并且向若被分派了新的值 Vk1 就本应被投影到孔 H1 的边界像素上的原始视差图上的像素分派新的值 Vk1。也可以使用 α 的不同值（如前述那样改变它的符号或大小）时做这些修改。

例如，投影视差图 Dp 上的所选择的孔的一个边界像素具有像素号 11，其值为 10。在原始视差图 Do 上的对应的像素是像素号 1，其值为 10。边界像素号 11 经过中值过滤后，它的新值为 5。因此，在原始视差图 Do 上的对应的像素号 1 取得“错误”的值。此外，向若被分派了

新值 5 就本应被映射到边界像素号 11 的原始视差图 D_0 上的像素分派新值 5。事实上，原始视差图 D_0 上的像素号 6 被分派了值 5。

在第三步骤 3)，优选地进行最后的规则化。这在于在各条线上通过中值过滤 F3 对投影视差图进行规则化（这能使投影视差更加规则，并能平滑一些不能被前面的过滤检测到的不规则性，比如由于这些不规则的地方不位于大的视差间断附近因而未被检测到）。如前所述，可能的变化都被转换到原始视差图上。

在第四步骤 4)，基于孔填充后的投影视差图，生成最终的 I_N 视图。

上述不同的处理步骤是在投影视差域执行的。因此，在第一非限制性实施例，如本领域技术人员所熟知的那样，最后的重建的视图 R_VIEW 可以由该过滤后的投影视差图生成，这参见文献“Disparity field and depth map coding for multi-view 3D image generation, by D.Tzovaras, N.ZGrammalidis, G.Strinzis, Signal Processing: Image Communication II (1998) 205-230 页”。

在第二非限制性实施例，重建视图由原始视差图的修改版产生，如下所述。

注意在步骤 1 和 2 的不同过滤过程中，只要在投影视差图上有一个像素被修改或者标记为“错误视差值”，原始视差图也如前述那样被标记和修改。从而，这些修改或标记的像素被转换到原始视差图上。

这样就得到了一个未定义值 WV 或“错误视差值”的列表。

为了得到最终的增强的视差图，在“孔填充”程序中应替换这个列表。这样做的更可靠的程序见下面内容。

在各条线上扫描具有未定义值的原始解码后的视差图，也就是说，修改后的原始视差图。为了填充这些未定义的值，拷贝未定义值的边界像素值的其中之一：通常有两个可能的边界像素值候选可供选择，一个在包含未定义值的像素的左边，一个在右边。一种选择是采用双线性或最近邻域插值，但是孔的不同尺寸可能导致插值在空间上的不一致。因此，优选地选择这样的边界像素，该边界像素的值最接近原始解码视差图上的原始像素的值。虽然压缩可能会对原始未压缩的值有较大改变，但最后的值一般都更接近于相同对象周围的像素，并且后处理过的视差图也接近于原始未压缩的视差图（采用这种插值

的实验结果较好)。

如图 5 所示, 可以看到原始解码视差图 Dco 和修改后的带有未定义值的原始解码视差图 Dcco。在后一个视差图 Dcco 中, 可以看到一个必须校正的未定义的值 WV1。该值与原始解码视差图 Dco 上的像素 Pco1 对应。对于未定义值 WV1, 候选的有两个边界像素 Ps1 和 Ps2。像素 Ps1 的值最接近于原始解码视差图 Dco 上的对应的原始像素 Pco1 的值。用所述像素 Ps1 替换未定义值 WV1。

一旦完成了插值, 校正后的视差图就生成了, 该图接近于原始未压缩的视差图。创建这样一个图的优势有两点:

- 其对于视图重建过程可以再次使用。由于已从该校正的视差图上消除了假象, 可以用其做最终的重建(不需要再次过滤, 除了可能在规则化步骤 3 中需要)。并且它给出了一个比利用投影视差图更好的重建视图。

- 所生成的校正后的图通常更为尖锐, 并且没有异常值, 这使得它更适于其它的基于深度/视差值的应用, 例如本领域人员熟知的合成和 Z 键控(Z-keying)。

需要了解的是, 本发明并不局限于前面提到的实施例, 在不脱离所附权利要求书所限定的本发明的精神和范围的情况下, 可以进行各种变动和修改。在这方面, 做出以下结语。

需要了解的是, 本发明并不局限于前面提到的视频应用。也可以用于考虑压缩视差图来处理信号的系统中的任何应用。特别地, 本发明应用于其他 MPEG 标准族(MPEG-1, MPEG-2)的视频压缩算法, 若应用于视差图编码, 则还适用于 ITU H26X 族(H261、H263 及扩展, 现今 H261 是最近的, 参考号 Q15-K-59)。

需要了解的是, 依据本发明的方法并不局限于前面提到的实现方式。

有许多种实现根据本发明的方法的功能的方式, 可以依靠硬件项或软件项或借助于二者, 其中单独的一个硬件或软件项可以执行多个功能。这不排除通过软件项或硬件项或二者的组合来完成一个功能, 因此, 不需要修改本发明的后处理视频信号的方法, 就能形成一个单一的功能。

所述软件或硬件项可以以多种方式实现, 例如通过连线电子电

路，或适当编程的集成电路。所述集成电路可以包含在计算机或解码器里。在后面这种情况中，如上所述，解码器包括执行前面所述的后处理视频信号方法中的步骤1、2、3和4的装置，所述装置如上面所述可以是软件或硬件项。

集成电路包括指令集。例如，所述指令集包含在计算机编程存储器或解码器存储器中，所述指令集使得计算机或解码器执行后处理方法中的不同步骤。

该指令集可以通过读取磁盘一类的数据载体加载进编程存储器中。服务提供商也可以通过通信网络（例如因特网）来使所述指令集可用。

权利要求书中的任何附图标记都不应解释为对权利要求的限制。很明显，动词“包括”的用法并没有把那些未在权利要求中限定的步骤或元件排除在外。步骤或元件前的冠词“一个”也并不排除含有多个此类步骤或元件。

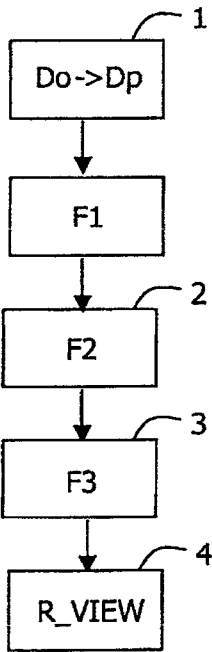


图 1

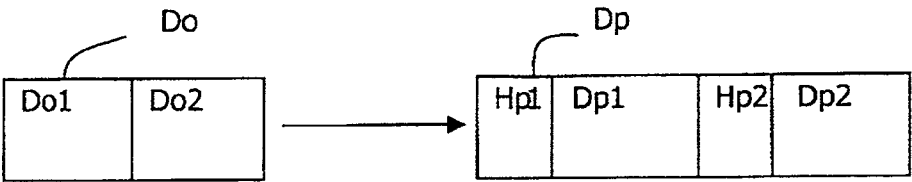


图 2

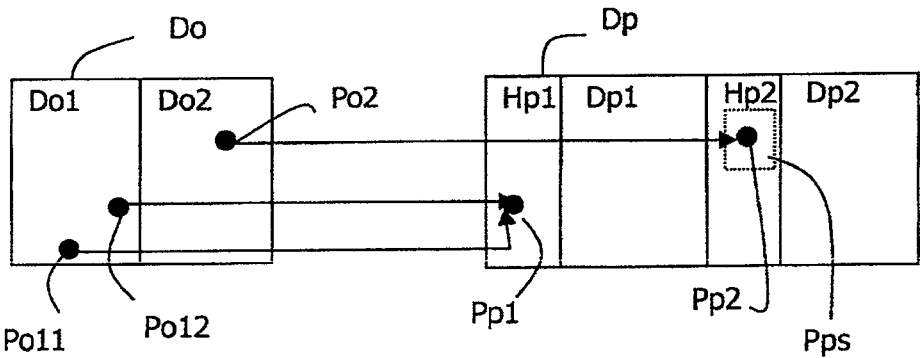


图 3

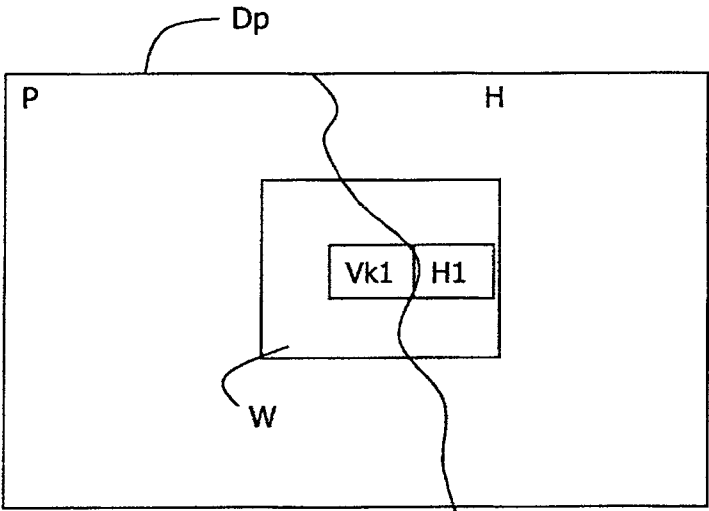


图 4

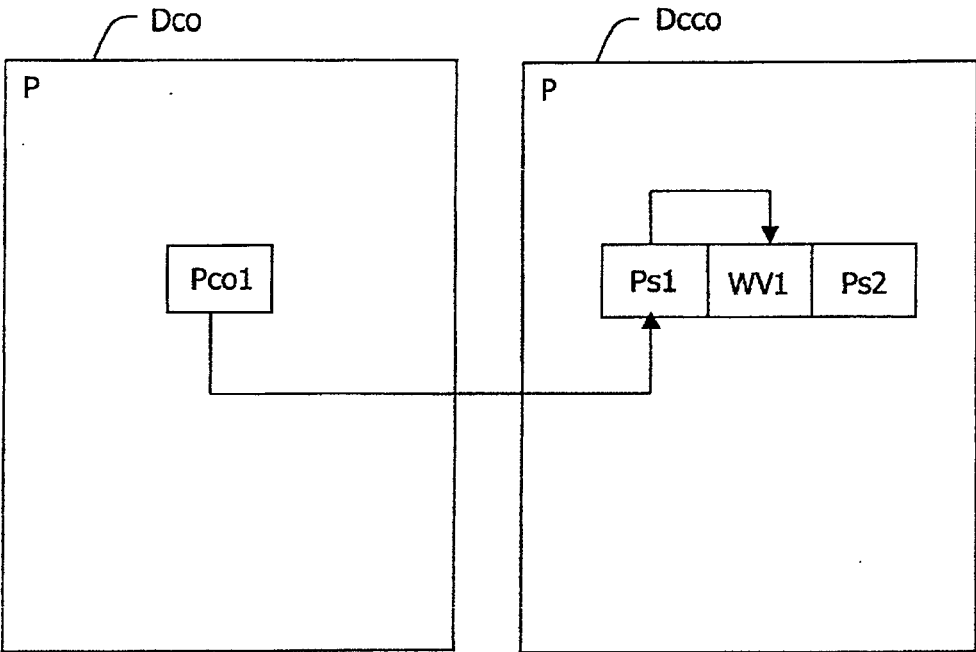


图 5