



(12)发明专利

(10)授权公告号 CN 106294640 B

(45)授权公告日 2020.02.21

(21)申请号 201610627154.2

(22)申请日 2016.08.02

(65)同一申请的已公布的文献号
申请公布号 CN 106294640 A

(43)申请公布日 2017.01.04

(73)专利权人 三星电子(中国)研发中心
地址 210012 江苏省南京市雨花台区安德
门大街57号6幢5-12楼
专利权人 三星电子株式会社

(72)发明人 周志顺 高强 王岩 丁训然

(74)专利代理机构 北京德琦知识产权代理有限
公司 11018
代理人 衣淑凤 宋志强

(51)Int.Cl.
G06F 16/957(2019.01)

(56)对比文件

US 6374273 B1,2002.04.16,
US 9400776 B1,2016.07.26,
CN 1278340 A,2000.12.27,
CN 102096690 A,2011.06.15,
WO 2014151693 A1,2014.09.25,

审查员 赵识谦

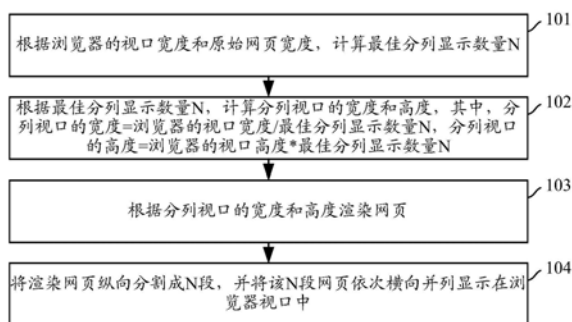
权利要求书2页 说明书7页 附图5页

(54)发明名称

网页显示方法及装置

(57)摘要

本申请提出网页显示方法及装置。方法包括:根据浏览器的视口宽度和原始网页宽度,计算最佳分列显示数量N;根据最佳分列显示数量N,计算分列视口的宽度和高度,其中,分列视口的宽度=浏览器的视口宽度/最佳分列显示数量N,分列视口的高度=浏览器的视口高度*最佳分列显示数量N;根据分列视口的宽度和高度渲染网页;将渲染网页纵向分割成N段,并将该N段网页依次横向并列显示在浏览器视口中。本申请优化了网页显示效果。



1. 一种网页显示方法,其特征在于,该方法包括:

根据浏览器的视口宽度和原始网页宽度,计算最佳分列显示数量N;

根据最佳分列显示数量N,计算分列视口的宽度和高度,其中,分列视口的宽度=浏览器的视口宽度/最佳分列显示数量N,分列视口的高度=浏览器的视口高度*最佳分列显示数量N;

根据分列视口的宽度和高度渲染网页;

将渲染网页纵向分割成N段,并将该N段网页依次横向并列显示在浏览器视口中;

所述将渲染网页纵向分割成N段之后、将该N段网页依次横向并列显示在浏览器视口中之前进一步包括:

对于任意两个相邻的分段网页,检测该两个分段网页的分割线是否正位于一个网页元素上,若是,将该分割线向上移动d距离到该网页元素上方的空白处,同时,将后续相邻分段网页的分割线都向上移动d距离。

2. 根据权利要求1所述的方法,其特征在于,所述计算最佳分列显示数量N为:

最佳分列显示数量N=浏览器的视口宽度/原始网页宽度。

3. 根据权利要求1所述的方法,其特征在于,所述将该N段网页依次横向并列显示在浏览器视口中之后进一步包括:

当检测到滚动条被拖动时,根据拖动轨迹,对各分段网页进行连续替换更新或者逐列替换更新。

4. 根据权利要求3所述的方法,其特征在于,所述对各分段网页进行逐列替换更新包括:

当用户拖动滚动条或者触发与拖动滚动条等价的操作时,根据拖动方向和拖动距离,将分列视口相对网页内容的位置向上或向下移动,每次移动的步长是浏览器视口的高度,触发浏览器渲染引擎将新的网页内容渲染显示到浏览器视口中。

5. 根据权利要求1所述的方法,其特征在于,所述将该N段网页依次横向并列显示在浏览器视口中之后进一步包括:

将操作在浏览器窗口上的位置转换为在分列视口中的位置,根据在分列视口中的位置确定出在对应网页元素上的位置,根据所述在对应网页元素上的位置渲染操作。

6. 根据权利要求1所述的方法,其特征在于,所述计算最佳分列显示数量N之后进一步包括:

当浏览器的视口宽度发生变化时,根据新的浏览器视口宽度,重新计算最佳分列显示数量;

或者,当屏幕分辨率发生变化时,计算新分辨率与原分辨率的比值,将当前最佳分列显示数量N与该比值相乘,得到新的最佳分列显示数量N;

或者,当网页发生缩放时,将当前最佳分列显示数量N除以缩放系数,得到新的最佳分列显示数量N。

7. 一种网页显示装置,其特征在于,包括:

分列显示数量计算单元:根据浏览器的视口宽度和原始网页宽度,计算最佳分列显示数量N;

布局渲染适配单元:根据最佳分列显示数量N,计算分列视口的宽度和高度,其中,分列

视口的宽度=浏览器的视口宽度/最佳分列显示数量N,分列视口的高度=浏览器的视口高度*最佳分列显示数量N;

浏览器渲染引擎:根据布局渲染适配单元计算出的分列视口的宽度和高度渲染网页;将渲染网页纵向分割成N段,并将该N段网页依次横向并列显示在浏览器视口中;

所述布局渲染适配单元进一步用于,

对于任意两个相邻的分段网页,检测该两个分段网页的分割线是否正位于一个网页元素上,若是,将该分割线向上移动d距离到该网页元素上方的空白处,同时,将后续相邻分段网页的分割线都向上移动d距离。

8.根据权利要求7所述的装置,其特征在于,所述分列显示数量计算单元计算最佳分列显示数量N为:

最佳分列显示数量N=浏览器的视口宽度/原始网页宽度。

9.根据权利要求7所述的装置,其特征在于,所述布局渲染适配单元进一步用于,

当检测到滚动条被拖动时,根据拖动轨迹,调用所述浏览器渲染引擎对各分段网页进行连续替换更新或者逐列替换更新。

10.根据权利要求9所述的装置,其特征在于,所述调用所述浏览器渲染引擎对各分段网页进行逐列替换更新包括:

当用户拖动滚动条或者触发与拖动滚动条等价的操作时,根据拖动方向和拖动距离,将分列视口相对网页内容的位置向上或向下移动,每次移动的步长是浏览器视口的高度,触发浏览器渲染引擎将新的网页内容渲染显示到浏览器视口中。

11.根据权利要求7所述的装置,其特征在于,所述布局渲染适配单元进一步用于,

将操作在浏览器窗口上的位置转换为在分列视口中的位置,根据在分列视口中的位置确定出在对应网页元素上的位置;

所述浏览器渲染引擎进一步用于,根据布局渲染适配单元计算出的所述在对应网页元素上的位置渲染操作。

12.根据权利要求7所述的装置,其特征在于,所述分列显示数量计算单元计算最佳分列显示数量N之后进一步包括:

当浏览器的视口宽度发生变化时,根据新的浏览器视口宽度重新计算最佳分列显示数量;

或者,当屏幕分辨率发生变化时,计算新分辨率与原分辨率的比值,将当前最佳分列显示数量N与该比值相乘,得到新的最佳分列显示数量N;

或者,当网页发生缩放时,将当前最佳分列显示数量N除以缩放系数,得到新的最佳分列显示数量N。

网页显示方法及装置

技术领域

[0001] 本申请涉及浏览器技术领域,尤其涉及网页显示方法及装置。

背景技术

[0002] 浏览器是电脑、手机和智能电视上实现网页浏览等操作的主要工具。随着硬件设备的更新换代,浏览器越来越多地工作在高清晰大面积的显示器上,屏幕的宽高比趋向于多样化,同时多屏幕拼接的场景也越来越多。

[0003] 当屏幕宽度远大于网页宽度时,对于目前互联网上的绝大部分网页,如果不放大显示,其内容只覆盖在在屏幕的局部区域,留出大面积空白,带来很多空间浪费。虽然较新式网页的设计会根据浏览器宽度和分辨率进行适配,但是这种网页自动适配往往会使得网页显示无法达到最优的观看效果。

[0004] 目前利用这种由于大面积高分辨率显示器带来的空间留白的技术主要是分屏显示,但是目前的分屏显示技术中各个窗口中显示的内容是各自独立渲染的不同网页。另一方面,相比于电脑和手机,电视浏览器操控交互过程中对于滚动条的拖动相对不便利,通常希望尽量减少用户对于滚动条的操作,在屏幕上把一张网页上的更多内容一次性展示出来。

发明内容

[0005] 本申请实施例提供网页显示方法及装置,以优化网页显示效果。

[0006] 本申请的技术方案是这样实现的:

[0007] 一种网页显示方法,该方法包括:

[0008] 根据浏览器的视口宽度和原始网页宽度,计算最佳分列显示数量 N ;

[0009] 根据最佳分列显示数量 N ,计算分列视口的宽度和高度,其中,分列视口的宽度=浏览器的视口宽度/最佳分列显示数量 N ,分列视口的高度=浏览器的视口高度*最佳分列显示数量 N ;

[0010] 根据分列视口的宽度和高度渲染网页;

[0011] 将渲染网页纵向分割成 N 段,并将该 N 段网页依次横向并列显示在浏览器视口中。

[0012] 所述计算最佳分列显示数量 N 为:

[0013] 最佳分列显示数量 N =浏览器的视口宽度/原始网页宽度。

[0014] 所述将渲染网页纵向分割成 N 段之后、将该 N 段网页依次横向并列显示在浏览器视口中之前进一步包括:

[0015] 对于任意两个相邻的分段网页,检测该两个分段网页的分割线是否正位于一个网页元素上,若是,将该分割线向上移动 d 距离到该网页元素上方的空白处,同时,将后续相邻分段网页的分割线都向上移动 d 距离。

[0016] 所述将该 N 段网页依次横向并列显示在浏览器视口中之后进一步包括:

[0017] 当检测到滚动条被拖动时,根据拖动轨迹,对各分段网页进行连续替换更新或者

逐列替换更新。

[0018] 所述对各分段网页进行逐列替换更新包括：

[0019] 当用户拖动滚动条或者触发与拖动滚动条等价的操作时，根据拖动方向和拖动距离，将分列视口相对网页内容的位置向上或向下移动，每次移动的步长是浏览器视口的高度，触发浏览器渲染引擎将新的网页内容渲染显示到浏览器视口中。

[0020] 所述将该N段网页依次横向并列显示在浏览器视口中之后进一步包括：

[0021] 将操作在浏览器窗口上的位置转换为在分列视口中的位置，根据在分列视口中的位置确定出在对应网页元素上的位置，根据所述在对应网页元素上的位置渲染操作。

[0022] 所述计算最佳分列显示数量N之后进一步包括：

[0023] 当浏览器的视口宽度发生变化时，根据新的浏览器视口宽度，重新计算最佳分列显示数量；

[0024] 或者，当屏幕分辨率发生变化时，计算新分辨率与原分辨率的比值，将当前最佳分列显示数量N与该比值相乘，得到新的最佳分列显示数量N；

[0025] 或者，当网页发生缩放时，将当前最佳分列显示数量N除以缩放系数，得到新的最佳分列显示数量N。

[0026] 一种网页显示装置，包括：

[0027] 分列显示数量计算单元：根据浏览器的视口宽度和原始网页宽度，计算最佳分列显示数量N；

[0028] 布局渲染适配单元：根据最佳分列显示数量N，计算分列视口的宽度和高度，其中，分列视口的宽度=浏览器的视口宽度/最佳分列显示数量N，分列视口的高度=浏览器的视口高度*最佳分列显示数量N；

[0029] 浏览器渲染引擎：根据布局渲染适配单元计算出的分列视口的宽度和高度渲染网页；将渲染网页纵向分割成N段，并将该N段网页依次横向并列显示在浏览器视口中。

[0030] 所述分列显示数量计算单元计算最佳分列显示数量N为：

[0031] 最佳分列显示数量N=浏览器的视口宽度/原始网页宽度。

[0032] 所述布局渲染适配单元进一步用于，

[0033] 对于任意两个相邻的分段网页，检测该两个分段网页的分割线是否正位于一个网页元素上，若是，将该分割线向上移动d距离到该网页元素上方的空白处，同时，将后续相邻分段网页的分割线都向上移动d距离。

[0034] 所述布局渲染适配单元进一步用于，

[0035] 当检测到滚动条被拖动时，根据拖动轨迹，调用所述浏览器渲染引擎对各分段网页进行连续替换更新或者逐列替换更新。

[0036] 所述调用所述浏览器渲染引擎对各分段网页进行逐列替换更新包括：

[0037] 当用户拖动滚动条或者触发与拖动滚动条等价的操作时，根据拖动方向和拖动距离，将分列视口相对网页内容的位置向上或向下移动，每次移动的步长是浏览器视口的高度，触发浏览器渲染引擎将新的网页内容渲染显示到浏览器视口中。

[0038] 所述布局渲染适配单元进一步用于，

[0039] 将操作在浏览器窗口上的位置转换为在分列视口中的位置，根据在分列视口中的位置确定出在对应网页元素上的位置；

[0040] 所述浏览器渲染引擎进一步用于,根据布局渲染适配单元计算出的所述在对应网页元素上的位置渲染操作。

[0041] 所述分列显示数量计算单元计算最佳分列显示数量N之后进一步包括:

[0042] 当浏览器的视口宽度发生变化时,根据新的浏览器视口宽度重新计算最佳分列显示数量;

[0043] 或者,当屏幕分辨率发生变化时,计算新分辨率与原分辨率的比值,将当前最佳分列显示数量N与该比值相乘,得到新的最佳分列显示数量N;

[0044] 或者,当网页发生缩放时,将当前最佳分列显示数量N除以缩放系数,得到新的最佳分列显示数量N。

[0045] 可见,本申请能够根据浏览器的视口宽度,调整分列显示数量,从而充分利用浏览器的可用显示面积,针对同一网页,扩大显示区域同时又保持原有网页的最佳显示比例,增加一次性显示内容,减少了滚动条操作。

附图说明

[0046] 图1为本申请一实施例提供的网页显示方法流程图;

[0047] 图2为本申请另一实施例提供的网页显示方法流程图;

[0048] 图3为本申请实施例提供的当 $N=2$ 时,分列视口的宽度和高度的示意图;

[0049] 图4为本申请实施例提供的当 $N=2$ 时,浏览器分列显示网页的示意图;

[0050] 图5为本申请实施例提供的当分割线正位于一个网页元素上时,对分割线进行调整的示意图;

[0051] 图6为当分割线正位于一个网页元素上时,若不调整分割线时的分列网页显示示意图;

[0052] 图7为本申请实施例提供的当分割线正位于一个网页元素上时,调整分割线后的分列网页显示示意图;

[0053] 图8-1为用户向下拖动滚动条前浏览器视口的显示示意图;

[0054] 图8-2为用户向下拖动滚动条后,采用本申请实施例提供的连续替换更新方式时浏览器视口的显示示意图;

[0055] 图9-1为用户向下拖动滚动条前浏览器视口的显示示意图;

[0056] 图9-2为用户向下拖动滚动条后,采用本申请实施例提供的逐列替换更新方式时浏览器视口的显示示意图;

[0057] 图10为本申请实施例提供的网页显示装置的组成示意图。

具体实施方式

[0058] 下面结合附图及具体实施例对本发明再作进一步详细的说明。

[0059] 图1为本申请一实施例提供的网页显示方法流程图,其具体步骤如下:

[0060] 步骤101:根据浏览器的视口宽度和原始网页宽度,计算最佳分列显示数量N。

[0061] 具体地,最佳分列显示数量 $N = \text{浏览器的视口宽度} / \text{原始网页宽度}$ 。

[0062] 步骤102:根据最佳分列显示数量N,计算分列视口的宽度和高度,其中,分列视口的宽度 $= \text{浏览器的视口宽度} / \text{最佳分列显示数量}N$,分列视口的高度 $= \text{浏览器的视口高度} * N$ 。

最佳分列显示数量N。

[0063] 步骤103:根据分列视口的宽度和高度渲染网页。

[0064] 步骤104:将渲染网页纵向分割成N段,并将该N段网页依次横向并列显示在浏览器视口中。

[0065] 图2为本申请另一实施例提供的网页显示方法流程图,其具体步骤如下:

[0066] 步骤201:设备的分列显示数量计算单元检测到用户启用了浏览器多列显示功能,计算最佳分列显示数量,将计算得到的最佳分列显示数量发送给布局渲染适配单元。

[0067] 设备包括:电脑、电视、移动终端(如:手机等)。

[0068] 具体地,计算最佳分列显示数量可通过如下几种方式:

[0069] 方式一、若用户配置了分列显示数量,则直接将其作为最佳分列显示数量。

[0070] 方式二、若用户未配置分列显示数量,则根据预设的最佳分列显示数量计算公式计算得到,例如:

[0071] 最佳分列显示数量 $N = \text{浏览器的视口宽度} / \text{原始网页宽度}$ 。

[0072] 浏览器的视口宽度指的是,浏览器上可用于显示网页的那部分界面的总宽度;

[0073] 原始网页宽度指的是,原始网页设计时设定的网页宽度,可从网页描述文档中读取到。

[0074] 若上述公式的计算结果不为整数,则进一步进行向上取整运算,得到整数N。

[0075] 步骤202:设备的布局渲染适配单元根据最佳分列显示数量,计算分列视口的宽度和高度。

[0076] 分列视口的宽度 $= \text{浏览器的视口宽度} / \text{最佳分列显示数量}N$,

[0077] 分列视口的高度 $= \text{浏览器的视口高度} * \text{最佳分列显示数量}N$ 。

[0078] 图3给出了当 $N=2$ 时,分列视口的宽度和高度的示意图。

[0079] 步骤203:设备的布局渲染适配单元将分列视口的宽度和高度发送给浏览器渲染引擎,浏览器渲染引擎根据分列视口的宽度和高度渲染网页。

[0080] 步骤204:浏览器渲染引擎在将网页渲染结果与浏览器UI (User Interface,用户接口) 界面进行合成时,将网页渲染结果纵向分割成N段,并将该N段网页依次横向并列显示在浏览器视口中。

[0081] 图4为 $N=2$ 时,浏览器分列显示网页的示意图。

[0082] 其中,对于任意两个相邻的分段网页,布局渲染适配单元可进一步检测该两个分段网页的分割线是否正位于一个网页元素上(例如:正位于一行文字或者一行图片上),若是,则确定将该分割线向上移动到该网页元素上方的空白处所对应的移动距离 d ,将该移动距离 d 发送给浏览器渲染引擎,浏览器渲染引擎根据该移动距离 d 渲染网页,即,将当前分割线及后续相邻分段网页的分割线都向上移动距离 d 。

[0083] 图5给出了当分割线正位于一个网页元素上时,对分割线进行调整的示意图。如图5所示,分割线 $f1$ 正好位于网页元素X上,则将分割线 $f1$ 向上调整 $d1$ 到达空白处,后续的分割线 $f2$ 、 $f3$ 同时向上调整 $d1$,但是分割线向上调整 $d1$ 后,又正好位于网页元素Y上,则将分割线 $f2$ 再向上调整 $d2$ 到达空白处,同时分割线 $f3$ 再向上调整 $d2$,此时,分割线 $f1$ 、 $f2$ 、 $f3$ 都不位于网页元素上,则调整结束。

[0084] 图6给出了若不调整分割线时的分列网页显示示意图,图7给出了调整分割线后的

分列网页显示示意图。

[0085] 特殊地,若两个相邻的分段网页的分割线正处于一个网页元素上,但该网页元素的高度超出单列分段网页的高度,则无需调整该分割线,因为在这种情况下该网页元素无法在单列分段网页上完整显示。

[0086] 当设备的布局渲染适配单元检测到用户拖动滚动条时,根据用户的拖动轨迹,触发浏览器渲染单元对网页进行连续替换更新或者逐列替换更新。

[0087] 具体地,对网页进行连续替换更新指的是,当用户拖动滚动条或者触发与拖动滚动条等价的操作时,根据拖动方向和拖动距离,分列视口相对网页内容的位置向上或向下移动,触发浏览器渲染引擎将新的网页内容渲染显示到浏览器视口中。从浏览器使用者的角度来看,当向下拖动滚动条或进行等价操作时,则浏览器视口中最左边列网页的部分内容向上移出浏览器视口,其右边列网页中的部分内容更新到最左边列网页,右边后续各列网页依此更新;当向上拖动滚动条或进行等价操作时,最右边列网页中的部分内容移出浏览器视口,其左边列网页的部分内容逐渐更新到最右边列网页,左边后续各列网页依此更新。

[0088] 图8-1为用户向下拖动滚动条前浏览器视口的显示示意图,此时,共有三列网页,最左边列网页中包含了网页元素a、b,中间列网页中包含了网页元素c、d,最右边列网页包含了网页元素e、f。

[0089] 图8-2为用户向下拖动滚动条后采用连续替换更新方式时浏览器视口的显示示意图,此时,最左边列网页中的网页元素a移出浏览器视口,中间列网页中的网页元素c移入最左边列,最右边列网页中的网页元素e移入中间列,后续新的网页元素g更新到最右边列。

[0090] 对网页进行逐列替换更新指的是,当用户拖动滚动条或者触发与拖动滚动条等价的操作时,根据拖动方向和拖动距离,分列视口相对网页内容的位置向上或向下移动,每次移动的步长是浏览器视口的高度,触发浏览器渲染引擎将新的网页内容渲染显示到浏览器视口中,即每次至少有一列的网页内容更新出和更新入浏览器视口区域。从浏览器使用者的角度来看,如果是向下拖动滚动条或进行等价操作时,最左边整列网页内容移出浏览器视口,其右边列网页内容整体更新到最左边列中,右边后续各列网页内容依此更新;如果是向上拖动滚动条或进行等价操作时,最右边列整列网页内容移出浏览器视口,其左边列网页内容整体更新到最右边列,左边后续各列网页内容依此更新。

[0091] 图9-1为用户向下拖动滚动条前浏览器视口的显示示意图,此时,共有三列网页,最左边列网页中包含了网页元素a、b,中间列网页中包含了网页元素c、d,最右边列网页包含了网页元素e、f。

[0092] 图9-2为用户向下拖动滚动条后采用逐列替换更新方式时浏览器视口的显示示意图,此时,最左边列网页中的所有网页元素a、b移出浏览器视口,中间列网页中的所有网页元素c、d移入最左边列,最右边列网页中的所有网页元素e、f移入中间列,后续新的网页元素g、h更新到最右边列。

[0093] 在步骤204之后,对于用户针对浏览器窗口上的鼠标、触控等操作,设备的布局渲染适配单元根据分列视口的宽度和高度,将操作在浏览器窗口上的位置转换为在分列视口中的位置,并进而确定出在对应网页元素上的位置,将在对应网页元素上的位置信息发送给浏览器渲染引擎,浏览器渲染引擎在对应位置渲染鼠标鼠标、触控等操作。

[0094] 进一步地,当浏览器的视口宽度发生变化时,返回步骤201重新计算最佳分列显示数量。

[0095] 在具体应用中,在计算最佳分列显示数量时,可进一步考虑到分辨率,具体地,可预先设置一个默认分辨率,步骤201中,最佳分列显示数量的计算公式可调整为:

[0096] 最佳分列显示数量 $N = (\text{浏览器的视口宽度} / \text{原始网页宽度}) * (\text{当前分辨率} / \text{默认分辨率})$,当分辨率发生变化时,返回步骤201根据该计算公式重新计算最佳分列显示数量。

[0097] 在具体应用中,当网页放大或缩小时,可返回步骤201重新计算最佳分列显示数量,更新后的最佳分列显示数量的计算公式为:

[0098] 最佳分列显示数量 $N = (\text{浏览器的视口宽度} / (\text{原始网页宽度} * \text{网页缩放系数}))$ 。网页缩放系数即,当网页放大了 m 倍时,则缩放系数为 m ,当网页缩小为原来的 $1/m$ 时,则缩放系数为 $1/m$ 。

[0099] 需要说明的是,网页的缩放可以是由用户点击放大/缩小按钮触发的,对于一些设备来说,其自身就支持根据用户到显示器的距离自动进行网页的缩、放。在这两种情况下,都会根据上述公式重新计算最佳分列显示数量。

[0100] 需要说明的是,当计算出了新的最佳分列显示数量 N 时,设备还可提醒用户是否根据新的最佳分列显示数量 N 进行网页的渲染合成,若用户返回是,则根据新的最佳分列显示数量 N 执行步骤202~204;否则,不对网页进行调整。

[0101] 本申请的有益技术效果如下:

[0102] 1)能够根据浏览器的视口宽度,调整分列显示数量,从而充分利用浏览器的可用显示面积,尤其是在浏览器的视口较宽时;且,针对同一网页,扩大显示区域同时又保持原有网页的最佳显示比例,增加一次性显示内容,减少了滚动条操作。

[0103] 特别针对智能电视,其具有屏幕宽分辨率高的特点,即使不进行屏幕拼接也具有更宽大的显示空间,同时使用电视遥控器进行浏览器滚动条操作比电脑流量和移动浏览器更不便利,本申请充分使用浏览器显示空间增加浏览器一次渲染显示的内容,减少了用户的滚动条操作,带来了最佳的浏览体验。

[0104] 图10为本申请实施例提供的网页显示装置的组成示意图,其主要包括:

[0105] 分列显示数量计算单元101:根据浏览器的视口宽度和原始网页宽度,计算最佳分列显示数量 N ,将最佳分列显示数量 N 发送给布局渲染适配单元102。

[0106] 布局渲染适配单元102:根据分列显示数量计算单元101发来的最佳分列显示数量 N ,计算分列视口的宽度和高度,其中,分列视口的宽度=浏览器的视口宽度/最佳分列显示数量 N ,分列视口的高度=浏览器的视口高度*最佳分列显示数量 N ,将分列视口的宽度和高度发送给浏览器渲染引擎103。

[0107] 浏览器渲染引擎:根据布局渲染适配单元102发来的分列视口的宽度和高度渲染网页;将渲染网页纵向分割成 N 段,并将该 N 段网页依次横向并列显示在浏览器视口中。

[0108] 一实施例中,分列显示数量计算单元101计算最佳分列显示数量 N 为:

[0109] 最佳分列显示数量 $N = \text{浏览器的视口宽度} / \text{原始网页宽度}$ 。

[0110] 一实施例中,布局渲染适配单元102进一步用于,

[0111] 对于任意两个相邻的分段网页,检测该两个分段网页的分割线是否正位于一个网页元素上,若是,将该分割线向上移动 d 距离到该网页元素上方的空白处,同时,将后续相邻

分段网页的分割线都向上移动d距离。

[0112] 一实施例中,布局渲染适配单元102进一步用于,

[0113] 当检测到滚动条被拖动时,根据拖动轨迹,调用浏览器渲染引擎103对各分段网页进行连续替换更新或者逐列替换更新。

[0114] 一实施例中,布局渲染适配单元102调用浏览器渲染引擎103对各分段网页进行逐列替换更新包括:

[0115] 当用户拖动滚动条或者触发与拖动滚动条等价的操作时,根据拖动方向和拖动距离,将分列视口相对网页内容的位置向上或向下移动,每次移动的步长是浏览器视口的高度,触发浏览器渲染引擎将新的网页内容渲染显示到浏览器视口中。

[0116] 一实施例中,布局渲染适配单元102进一步用于,

[0117] 将操作在浏览器窗口上的位置转换为在分列视口中的位置,根据在分列视口中的位置确定出在对应网页元素上的位置;

[0118] 浏览器渲染引擎103进一步用于,根据布局渲染适配单元102计算出的所述在对应网页元素上的位置渲染操作。

[0119] 一实施例中,分列显示数量计算单元101计算最佳分列显示数量N之后进一步包括:

[0120] 当浏览器的视口宽度发生变化时,根据新的浏览器视口宽度重新计算最佳分列显示数量;

[0121] 或者,当屏幕分辨率发生变化时,计算新分辨率与原分辨率的比值,将当前最佳分列显示数量N与该比值相乘,得到新的最佳分列显示数量N;

[0122] 或者,当网页发生缩放时,将当前最佳分列显示数量N除以缩放系数,得到新的最佳分列显示数量N。

[0123] 以上所述仅为本申请的较佳实施例而已,并不用以限制本申请,凡在本申请的精神和原则之内,所做的任何修改、等同替换、改进等,均应包含在本申请保护的范围之内。

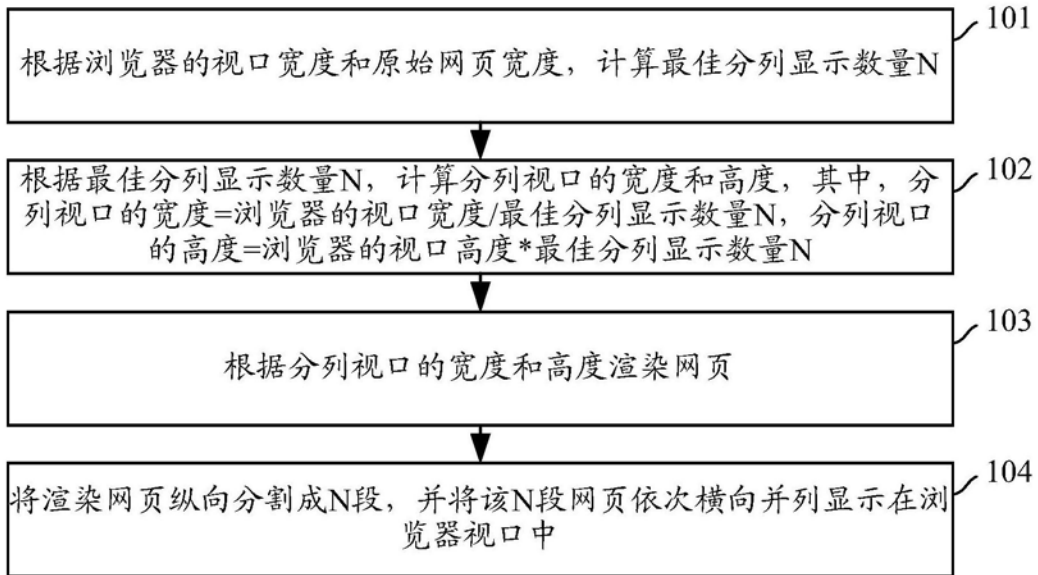


图1

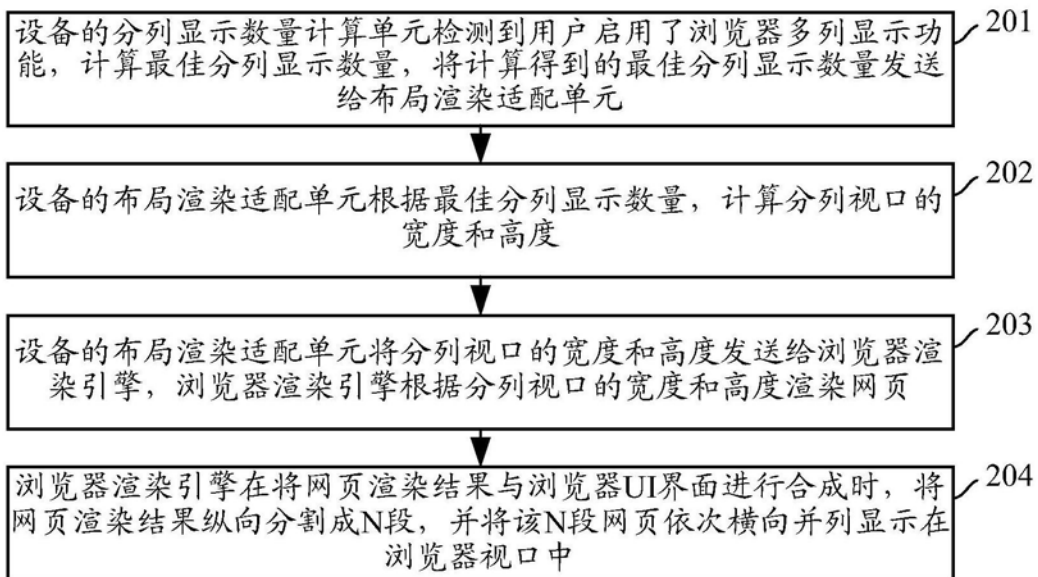


图2

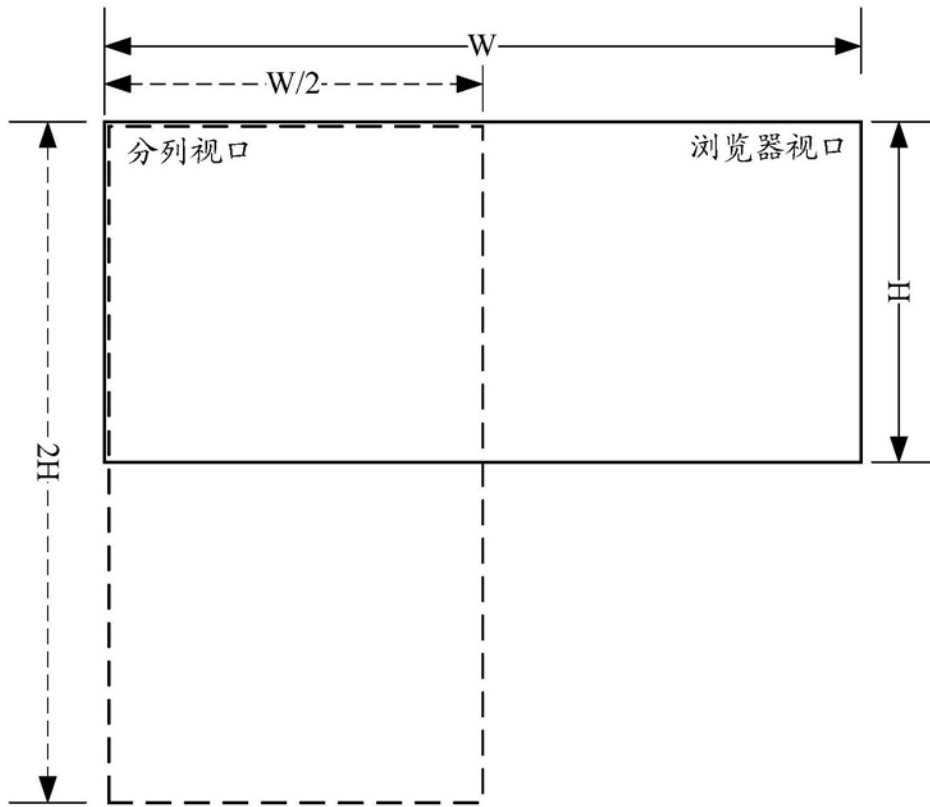


图3

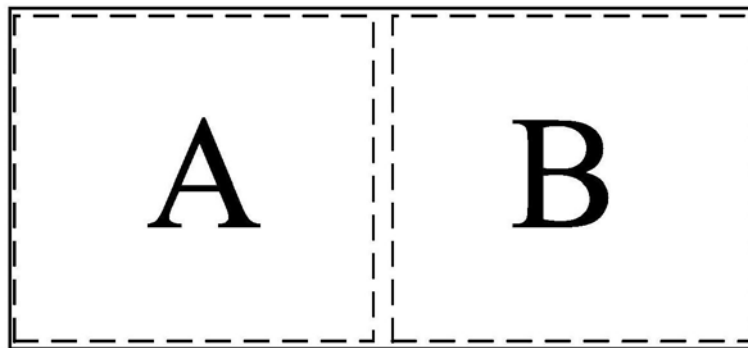


图4

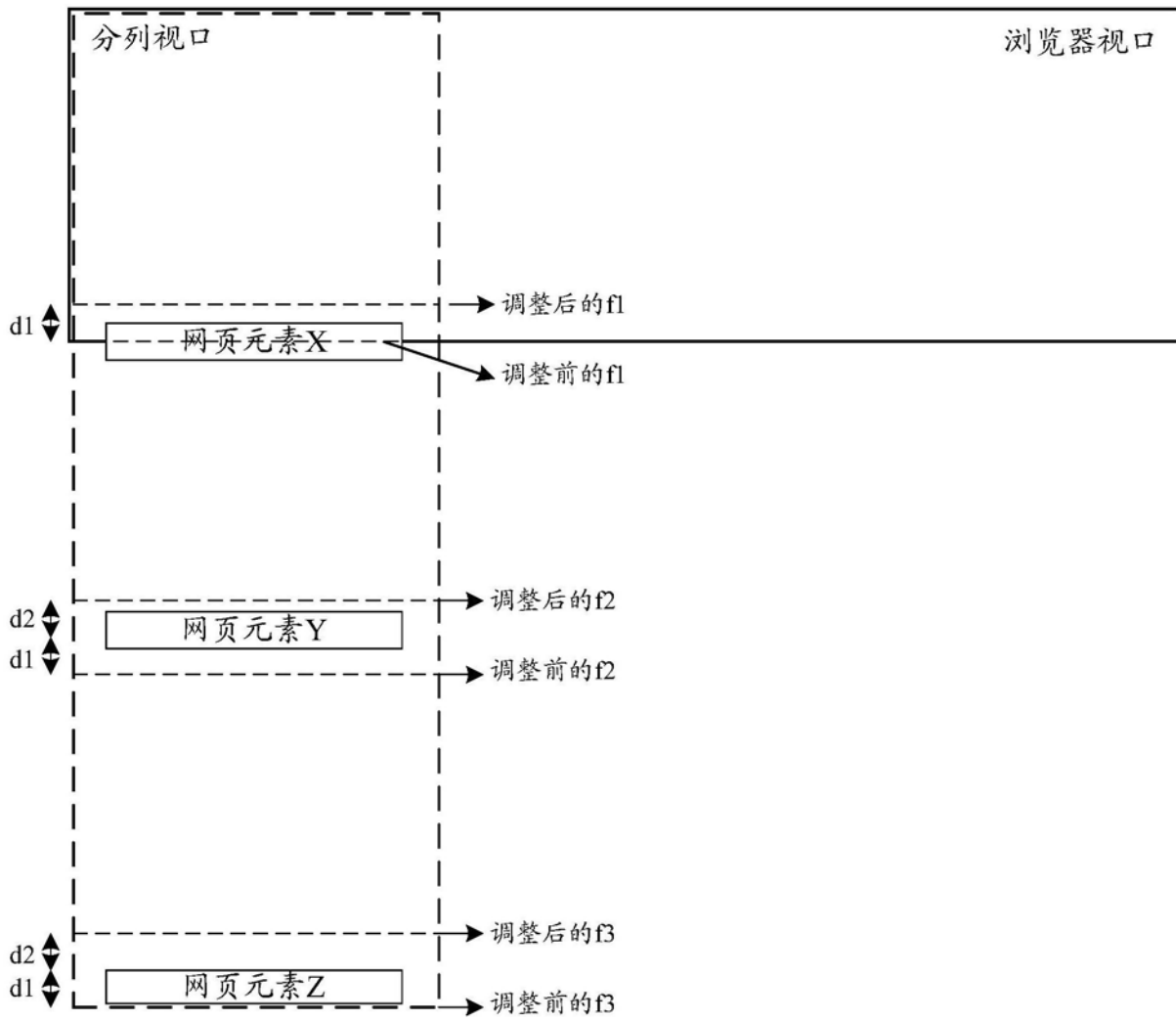


图5



图6



图7

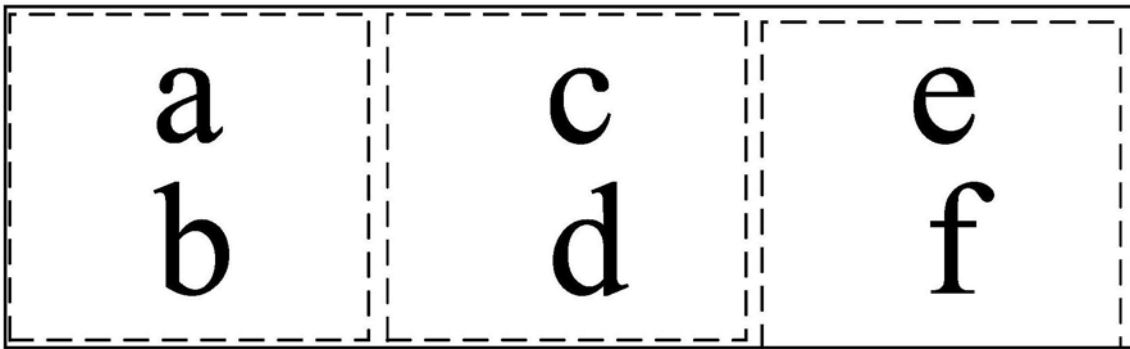


图8-1

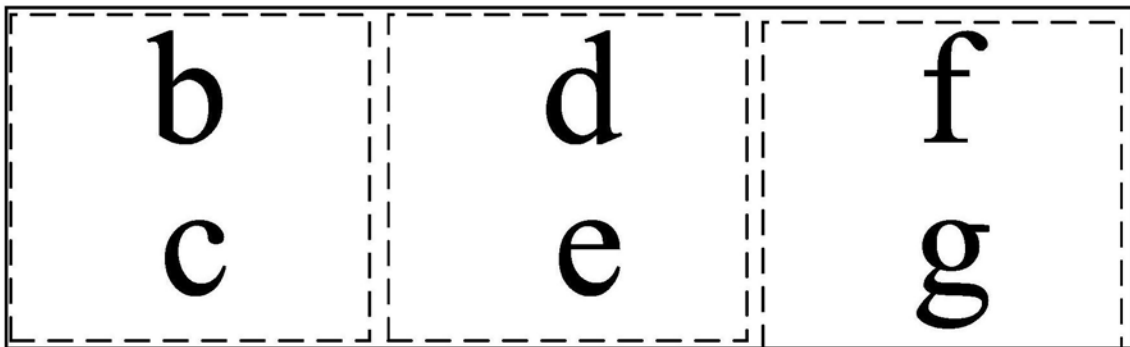


图8-2

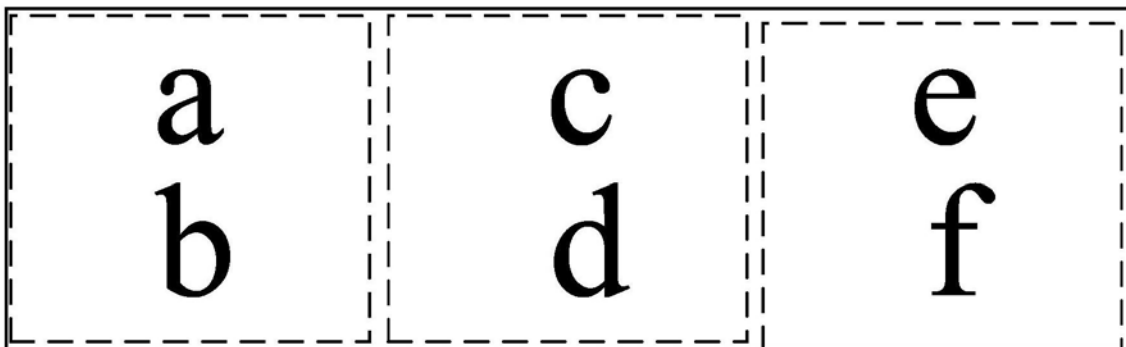


图9-1

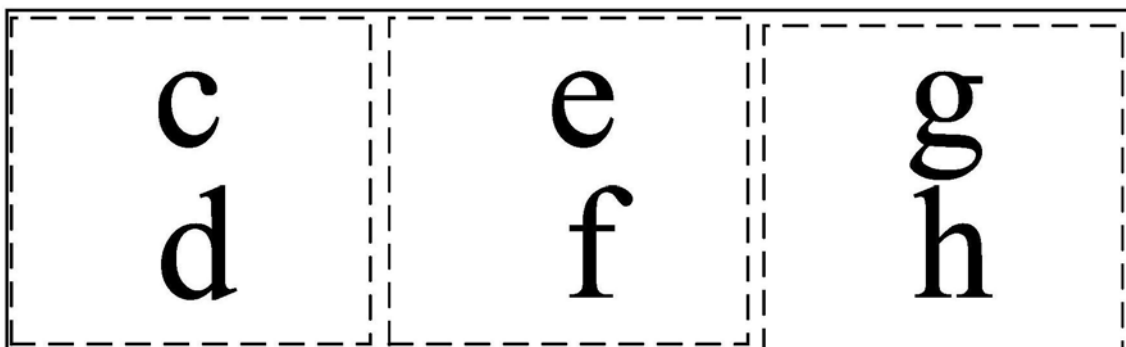


图9-2

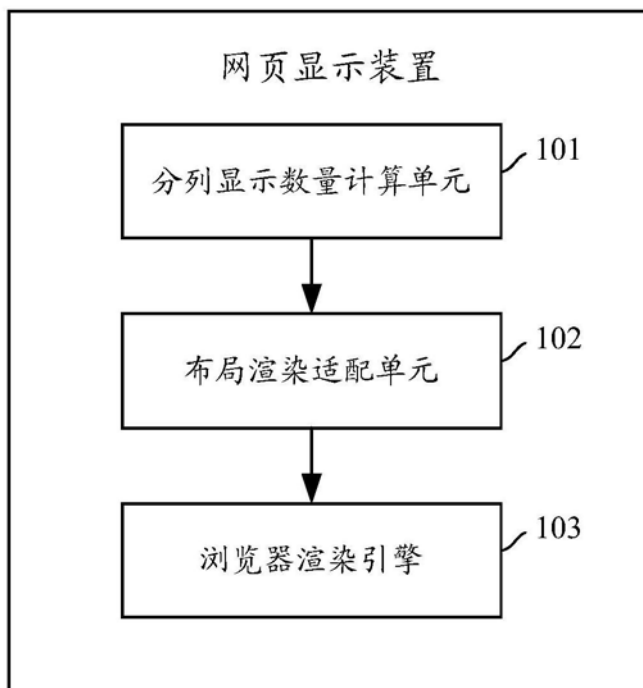


图10