



(12)发明专利

(10)授权公告号 CN 105094584 B

(45)授权公告日 2019.04.05

(21)申请号 201410206909.2

G06F 9/451(2018.01)

(22)申请日 2014.05.16

(56)对比文件

(65)同一申请的已公布的文献号
申请公布号 CN 105094584 A

CN 101140516 A,2008.03.12,
CN 1845054 A,2006.10.11,
CN 103324734 A,2013.09.25,全文.
CN 102999274 A,2013.03.27,全文.
CN 101140516 A,2008.03.12,

(43)申请公布日 2015.11.25

(73)专利权人 北大方正集团有限公司
地址 100871 北京市海淀区成府路298号方
正大厦9层

审查员 黄剑飞

专利权人 北京北大方正电子有限公司

(72)发明人 林好 高辰柱

(74)专利代理机构 北京友联知识产权代理事务
所(普通合伙) 11343

代理人 尚志峰 汪海屏

(51)Int.Cl.

G06F 3/0484(2013.01)

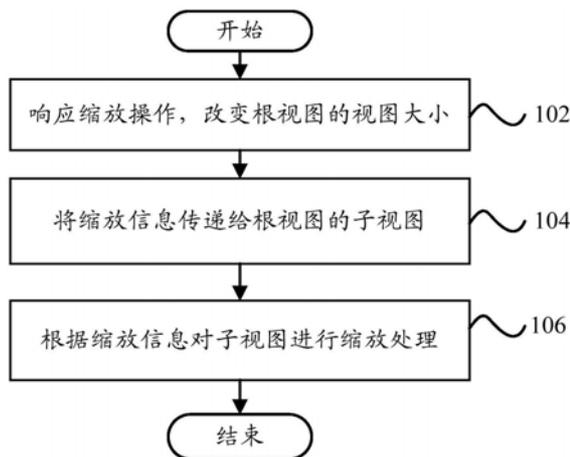
权利要求书1页 说明书6页 附图1页

(54)发明名称

视图缩放处理方法和视图缩放处理装置

(57)摘要

本发明提出了一种视图缩放处理方法和一种视图缩放处理装置,其中视图缩放处理方法包括:响应缩放操作,改变根视图的视图大小;将缩放信息传递给所述根视图的子视图;根据所述缩放信息对所述子视图进行缩放处理。通过本发明的技术方案能够实现基于安卓系统的应用程序内的页面级缩放,提高阅读操作体验。



1. 一种视图缩放处理方法,其特征在于,包括:
 - 响应缩放操作,改变根视图的视图大小;
 - 将缩放信息传递给所述根视图的子视图;
 - 根据所述缩放信息对所述子视图进行缩放处理;
 - 在所述子视图的类型为布局视图时,所述缩放信息为缩放参数,调用相应的缩放函数并基于所述缩放参数对所述子视图进行缩放处理;
 - 在所述子视图的类型为直接视图时,所述缩放信息为所述子视图缩放后的大小信息,调用测量方法对所述子视图进行缩放处理;
 - 还包括:记录所述根视图和所述子视图的原位置和大小以及原比例;
 - 在所述根视图接收到缩放操作和缩放参数后,根据原比例确定长和宽的新值。
2. 根据权利要求1所述的视图缩放处理方法,其特征在于,还包括:
 - 为子视图中的字体设置预定像素大小,以及随根视图缩放的变化量;
 - 在所述根视图缩放时,若子视图为继承自文本视图,则使用所述变化量改变所述子视图中的字体大小。
3. 根据权利要求1或2所述的视图缩放处理方法,其特征在于,还包括:在对所述子视图进行缩放处理的同时,将所述缩放信息下发至下一级子视图。
4. 一种视图缩放处理装置,其特征在于,包括:
 - 缩放单元,用于响应缩放操作,改变根视图的视图大小,以及根据所述缩放信息对所述根视图的子视图进行缩放处理;
 - 数据传递单元,用于将缩放信息传递给所述子视图;
 - 所述缩放单元包括:
 - 调用单元,在所述子视图的类型为布局视图时,所述缩放信息为缩放参数,调用相应的缩放函数并基于所述缩放参数对所述子视图进行缩放处理,以及在所述子视图的类型为直接视图时,所述缩放信息为所述子视图缩放后的大小信息,调用测量方法对所述子视图进行缩放处理;
 - 还包括:
 - 存储单元,记录所述根视图和所述子视图的原位置和大小以及原比例;
 - 所述缩放单元还用于在所述根视图接收到缩放操作和缩放参数后,根据原比例确定长和宽的新值。
5. 根据权利要求4所述的视图缩放处理装置,其特征在于,还包括:
 - 设置单元,用于为子视图中的字体设置预定像素大小,以及随根视图缩放的变化量;
 - 所述缩放单元还用于在所述根视图缩放时,若子视图为继承自文本视图,则使用所述变化量改变所述子视图中的字体大小。
6. 根据权利要求4或5所述的视图缩放处理装置,其特征在于,所述数据传递单元还用于在对所述子视图进行缩放处理的同时,将所述缩放信息下发至下一级子视图。

视图缩放处理方法和视图缩放处理装置

技术领域

[0001] 本发明涉及计算技术领域,具体而言,涉及一种视图缩放处理方法和一种视图缩放处理装置。

背景技术

[0002] 安卓系统(Android)的视图包括布局视图(直接继承ViewGroup)和直接视图(直接继承View)。目前Android中三个常用的布局视图为Framelayout,Linearlayout,Relativelayout。在终端阅读领域,多数是以图片等方式进行浏览,当随着用户需求的增加需要很多复杂的交互,而在交互组件的缩放上,至今还没有相关处理方法,虽然有使用一种伪缩放的方法,即用Android自带的webview控件加载纯静态的网页方式,但没有在本质上解决Android子View(子视图)随父View(根视图)缩放的问题。

发明内容

[0003] 考虑到相关技术中出现的技术问题,本发明提供了一种新的视图缩放处理技术,能够真正实现页面布局缩放的应用。

[0004] 根据本发明的一个方面,提供了一种视图缩放处理方法,包括:响应缩放操作,改变根视图的视图大小;将缩放信息传递给所述根视图的子视图;根据所述缩放信息对所述子视图进行缩放处理。

[0005] 在对根视图进行缩放时,同时向子视图下发缩放参数,实现了子视图跟随跟视图进行缩放的效果。

[0006] 在上述技术方案中,优选的,在所述子视图的类型为布局视图时,所述缩放信息为缩放参数,调用相应的缩放函数并基于所述缩放参数对所述子视图进行缩放处理;在所述子视图的类型为直接视图时,所述缩放信息为所述子视图缩放后的大小信息,调用测量方法对所述子视图进行缩放处理。

[0007] 在根视图使用一个Viewgroup(布局视图),然后实现一个手势缩放的行为,控制缩放的参数,使用这个参数来对当前页面的大小做重置,然后在onMeasure方法中将这个参数传递下去,如果子view是单独的view则直接传递计算之后的值,如果子view是ViewGroup,则传递这个参数的具体值,这样递归的对各个子view做大小和位置的重置,来实现缩放的效果。

[0008] 在上述技术方案中,优选的,还可以包括:为子视图中的字体设置预定像素大小,以及随根视图缩放的变化量;在所述根视图缩放时,若子视图为继承自文本视图,则使用所述变化量改变所述子视图中的字体大小。

[0009] 在子视图随根视图进行缩放时,其中的字体也能随之改变,并且通过预先设置变化量的方式提高了缩放效率。

[0010] 在上述技术方案中,优选的,还可以包括:记录所述根视图和所述子视图的原位置和大小以及原比例;在所述根视图接收到缩放操作和缩放参数后,根据原比例确定长和宽

的新值。缩放后的比例与缩放前的比例相同。

[0011] 在上述技术方案中,优选的,还可以包括:在对所述子视图进行缩放处理的同时,将所述缩放信息下发至下一级子视图。

[0012] 根据本发明的另一方面,还提供了一种视图缩放处理装置,包括:缩放单元,用于响应缩放操作,改变根视图的视图大小,以及根据所述缩放信息对所述根视图的子视图进行缩放处理;数据传递单元,用于将缩放信息传递给所述子视图。

[0013] 在对根视图进行缩放时,同时向子视图下发缩放参数,实现了子视图跟随根视图进行缩放的效果。

[0014] 在上述技术方案中,优选的,所述缩放单元包括:调用单元,在所述子视图的类型为布局视图时,所述缩放信息为缩放参数,调用相应的缩放函数并基于所述缩放参数对所述子视图进行缩放处理,以及在所述子视图的类型为直接视图时,所述缩放信息为所述子视图缩放后的大小信息,调用测量方法对所述子视图进行缩放处理。

[0015] 在根视图使用一个Viewgroup(布局视图),然后实现一个手势缩放的行为,控制缩放的参数,使用这个参数来对当前页面的大小做重置,然后在onMeasure方法中将这个参数传递下去,如果子view是单独的view则直接传递计算之后的值,如果子view是ViewGroup,则传递这个参数的具体值,这样递归的对各个子view做大小和位置的重置,来实现缩放的效果。

[0016] 在上述技术方案中,优选的,还可以包括:设置单元,用于为子视图中的字体设置预定像素大小,以及随根视图缩放的变化量;所述缩放单元还用于在所述根视图缩放时,若子视图为继承自文本视图,则使用所述变化量改变所述子视图中的字体大小。

[0017] 在子视图随根视图进行缩放时,其中的字体也能随之改变,并且通过预先设置变化量的方式提高了缩放效率。

[0018] 在上述技术方案中,优选的,还可以包括:存储单元,记录所述根视图和所述子视图的原位置和大小以及原比例;所述缩放单元还用于在所述根视图接收到缩放操作和缩放参数后,根据原比例确定长和宽的新值。缩放后的比例与缩放前的比例相同。

[0019] 在上述技术方案中,优选的,所述数据传递单元还用于在对所述子视图进行缩放处理的同时,将所述缩放信息下发至下一级子视图。

[0020] 本发明公开了一种在安卓系统应用程序中实现应用程序内的页面级缩放的解决方案。在现有的安卓系统应用程序中,这种需求很不常见,但对于在手机上的杂志阅读很有实际意义。首先在根布局使用一个Viewgroup,然后实现一个手势缩放的行为,控制缩放的参数,使用这个参数来对当前页面的大小做重置,然后在onMeasure方法中将这个参数传递下去,如果子view是单独的view,则直接传递计算之后的值,如果子view是ViewGroup,则传递这个参数的具体值,这样递归的对各个子view做大小和位置的重置,来实现缩放的效果。

附图说明

[0021] 图1示出了根据本发明的实施例的视图缩放处理方法的示意流程图;

[0022] 图2示出了根据本发明的实施例的视图缩放处理装置的示意框图。

具体实施方式

[0023] 为了能够更清楚地理解本发明的上述目的、特征和优点,下面结合附图和具体实施方式对本发明进行进一步的详细描述。需要说明的是,在不冲突的情况下,本申请的实施例及实施例中的特征可以相互组合。

[0024] 在下面的描述中阐述了很多具体细节以便于充分理解本发明,但是,本发明还可以采用其他不同于在此描述的方式来实施,因此,本发明并不限于下面公开的具体实施例的限制。

[0025] 图1示出了根据本发明的实施例的文字处理方法的示意图。

[0026] 如图1所示,根据本发明的实施例的视图缩放处理方法可以包括以下步骤:

[0027] 步骤102,响应缩放操作,改变根视图的视图大小;步骤104,将缩放信息传递给所述根视图的子视图;步骤106,根据所述缩放信息对所述子视图进行缩放处理。

[0028] 在对根视图进行缩放时,同时向子视图下发缩放参数,实现了子视图跟随跟视图进行缩放的效果。

[0029] 在上述技术方案中,优选的,在所述子视图的类型为布局视图时,所述缩放信息为缩放参数,调用相应的缩放函数并基于所述缩放参数对所述子视图进行缩放处理;在所述子视图的类型为直接视图时,所述缩放信息为所述子视图缩放后的大小信息,调用测量方法对所述子视图进行缩放处理。

[0030] 在根视图使用一个Viewgroup (布局视图),然后实现一个手势缩放的行为,控制缩放的参数,使用这个参数来对当前页面的大小做重置,然后在onMeasure方法中将这个参数传递下去,如果子view是单独的view则直接传递计算之后的值,如果子view是ViewGroup,则传递这个参数的具体值,这样递归的对各个子view做大小和位置的重置,来实现缩放的效果。

[0031] 在上述技术方案中,优选的,还可以包括:为子视图中的字体设置预定像素大小,以及随根视图缩放的变化量;在所述根视图缩放时,若子视图为继承自文本视图,则使用所述变化量改变所述子视图中的字体大小。

[0032] 在子视图随根视图进行缩放时,其中的字体也能随之改变,并且通过预先设置变化量的方式提高了缩放效率。

[0033] 在上述技术方案中,优选的,还可以包括:记录所述根视图和所述子视图的原位置和大小以及原比例;在所述根视图接收到缩放操作和缩放参数后,根据原比例确定长和宽的新值。缩放后的比例与缩放前的比例相同。

[0034] 在上述技术方案中,优选的,还可以包括:在对所述子视图进行缩放处理的同时,将所述缩放信息下发至下一级子视图。

[0035] 视图(View)在重绘时会以一种递归的方式来逐层的下发缩放信息,并且会经过裁剪大小,定位和绘制这三个基本过程,每个过程都要递归遍历所有的View。这样在父View缩放的时候,也向子View下发缩放参数,子View就可以自己使用后再次下发下去。

[0036] 综上,本发明提供的视图缩放处理方法包括以下步骤:

[0037] A步骤、选择一个基础的布局视图作为根视图。实现其包含手势的缩放。可以采用自己的代码实现,也可以使用android自带的手势类来实现。为了方便扩展,使用前者比较合适。内部实现了这个手势缩放之后,要时刻记录缩放的变化量,此时可以做一个防抖的处

理,不能多大的缩放都响应,这里可选取10px。在缩放之前,记录子视图和根视图的原始位置和大小,以便保证始终在原始比例,不在缩放时变形。

[0038] B步骤、父View(根视图)的onMeasure方法。传递给各个子View一个变换的缩放参数。如果子视图是ViewGroup,则递归传递给子View缩放参数;如果不是,则直接计算出子View缩放后应有的大小,直接调用子View的measure方法进行缩放处理。

[0039] 详细的,B步骤可以再分为以下步骤:

[0040] B-1、在FrameLayout记录其本身和所有的子View的位置大小信息,通过一个boolean的变量来控制是否记录;

[0041] B-2、每次处于根的FrameLayout在接收了缩放的手势和缩放参数之后,直接去计算本View的新位置和大小。计算时选取宽或者高更长的,计算其缩放后的新值,然后根据之前保存的比例来计算宽或者高比较短的那个,这样就保证了原始比例不被破坏。

[0042] B-3、主动调用requestLayout(),在重新绘制时,通过一个boolean的变量来判断是否是缩放需要重绘的请求,如果需要重绘,则如下:

[0043] 在onMeasure中遍历所有的子View,计算它们的新大小和新的位置,按照计算其本身的方法相同。

[0044] 然后判断子View是否是一个ViewGroup,是则调用递归的调用其相应缩放的函数,否则直接measure子View,其中MeasureSpec的类型必须使用EXACTLY。

[0045] 完成了以上的步骤,基本上就可以实现简单的页面缩放了。但还是会有很多细节问题,在实际的应用中。比如Textview(文本视图)的字体大小,子View是Viewgroup嵌套的问题,子Viewgroup不是FrameLayout的情况等等。下面介绍实际应用中的解决方案。

[0046] Textview直接继承自View,在父View应用缩放时,会直接改变Textview本身的视图大小,而里面的文字不会发生改变,甚至文字的margin(边页空白)都不会发生变化。而button是继承自Textview,里面的文字也不会改变,必须要单独设置。优选使用px为单位,设置字体大小。因为sp应用在系统改变大小的情况下,会对应用产生不必要的适配问题,而dp应用仅在xml布局中比较好,而在代码中需要转换,而且不合适在字体中使用。所以直接使用字体的像素值是最好的选择。具体的,可以在设置原始大小时,调整一个比较合适的字体像素大小,和一个随着Textview本身视图缩放,字体像素值变化的一个增量(减量)。这样在每次父View应用缩放时,如果发现子View是继承自Textview,则使用之前设置好的增量(减量)来直接改变字体大小,需要注意的还应传递一个单位值,这里传递像素即可。

[0047] 解决子View也是ViewGroup的问题。当子View也是一个布局时,其本身也会包含布局视图(继承自ViewGroup的)或者直接视图(直接继承自View的)。这样在传递缩放参数时就会产生问题,因为实际的应用中我们并不能总是保证所有的布局视图都是FrameLayout。因此在实现缩放时使用一个通用的Viewgroup类,让所有的布局视图都继承自这个布局。在这个ViewGroup中添加手势缩放,传递缩放参数的代码。我们保留常用布局的默认实现。只在缩放时调用相应的函,这样做易于维护。

[0048] 还有一种解决方法就是把常用的三种布局视图都实现这样的缩放控制,这样会有一些冗余的代码,但是实现简单,技术难度较小。也可以根据不同的布局视图做一些微调。在ViewGroup嵌套ViewGroup的问题上,以上两种方案各有利弊,针对实际的电子杂志页面缩放,可使用后者方法,因为我们的布局基本都是在FrameLayout上实现的。

[0049] 最后一个问题就是在实际应用中,子View可能比较复杂,可能是重写过的基础控件。里面的逻辑不能简单的使用以上的规则来定义。这时一个比较直观的方法就是针对有问题且不能使用通用方法来解决的问题,需要单独处理。根据实际情况来微调,且不能破坏其他的规则,需要保证以下几点:

[0050] a.单独微调的规则之间不能互相影响;b.缩放的流程不能破坏;c.如果调整后的View还有其他的子View,则对其子View的调整需要放在单独的实现的View中,不能再放在基础的布局视图中。

[0051] 作为扩展的实现,还可以使用View的layout方法来完成位置和大小变化,这样比较直观,就是在子View和自身缩放时,传递缩放后位置和大小。这样做直接跳过了measure的步骤,因为这样不会保存measure之后的大小,所以在其他有重绘的时机时,会采用未经修改过的裁剪大小,而不是我们主动缩放后的大小,所以会导致一些不可预知的问题,因为Android的View重绘机制有三种,一是主动请求,二是View的状态改变,三是View的大小位置改变。layout的方法是1和3的条件,而我们不能控制2的条件发生的重绘。但如果不必保存View的裁剪大小,我们也可以使用这种方式。例如在控制Imageview的变化大小时,不必考虑其状态的改变。

[0052] 图2示出了根据本发明的实施例的视图缩放处理装置的示意框图。

[0053] 如图2所示,根据本发明的实施例的视图缩放处理装置200可以包括:缩放单元202,用于响应缩放操作,改变根视图的视图大小,以及根据所述缩放信息对所述根视图的子视图进行缩放处理;数据传递单元204,用于将缩放信息传递给所述子视图。

[0054] 在对根视图进行缩放时,同时向子视图下发缩放参数,实现了子视图跟随跟视图进行缩放的效果。

[0055] 在上述技术方案中,优选的,所述缩放单元202包括:调用单元2022,在所述子视图的类型为布局视图时,所述缩放信息为缩放参数,调用相应的缩放函数并基于所述缩放参数对所述子视图进行缩放处理,以及在所述子视图的类型为直接视图时,所述缩放信息为所述子视图缩放后的大小信息,调用测量方法对所述子视图进行缩放处理。

[0056] 在根视图使用一个Viewgroup(布局视图),然后实现一个手势缩放的行为,控制缩放的参数,使用这个参数来对当前页面的大小做重置,然后在onMeasure方法中将这个参数传递下去,如果子view是单独的view则直接传递计算之后的值,如果子view是ViewGroup,则传递这个参数的具体值,这样递归的对各个子view做大小和位置的重置,来实现缩放的效果。

[0057] 在上述技术方案中,优选的,还包括:设置单元206,用于为子视图中的字体设置预定像素大小,以及随根视图缩放的变化量;所述缩放单元202还用于在所述根视图缩放时,若子视图为继承自文本视图,则使用所述变化量改变所述子视图中的字体大小。

[0058] 在子视图随根视图进行缩放时,其中的字体也能随之改变,并且通过预先设置变化量的方式提高了缩放效率。

[0059] 在上述技术方案中,优选的,还可以包括:存储单元208,记录所述根视图和所述子视图的原位置和大小以及原比例;所述缩放单元202还用于在所述根视图接收到缩放操作和缩放参数后,根据原比例确定长和宽的新值。缩放后的比例与缩放前的比例相同。

[0060] 在上述技术方案中,优选的,所述数据传递单元204还用于在对所述子视图进行缩

放处理的同时,将所述缩放信息下发至下一级子视图。

[0061] 以上结合附图详细说明了根据本发明的技术方案,本发明公开了一种在Android应用程序中实现应用程序内的页面级缩放的解决方案。在现有的Android应用程序中,这种需求很不常见,但对于在手机上的杂志阅读很有实际意义。首先在根布局使用一个Viewgroup,然后实现一个手势缩放的行为,控制缩放的参数,使用这个参数对当前页面的大小做重置,然后在onMeasure方法中将这个参数传递下去。如果子view是单独的view,则直接传递计算之后的值,如果子view是ViewGroup,则传递这个参数的具体值,这样递归的对各个子view做大小和位置的重置,来实现缩放的效果。

[0062] 以上所述仅为本发明的优选实施例而已,并不用于限制本发明,对于本领域的技术人员来说,本发明可以有各种更改和变化。凡在本发明的精神和原则之内,所作的任何修改、等同替换、改进等,均应包含在本发明的保护范围之内。

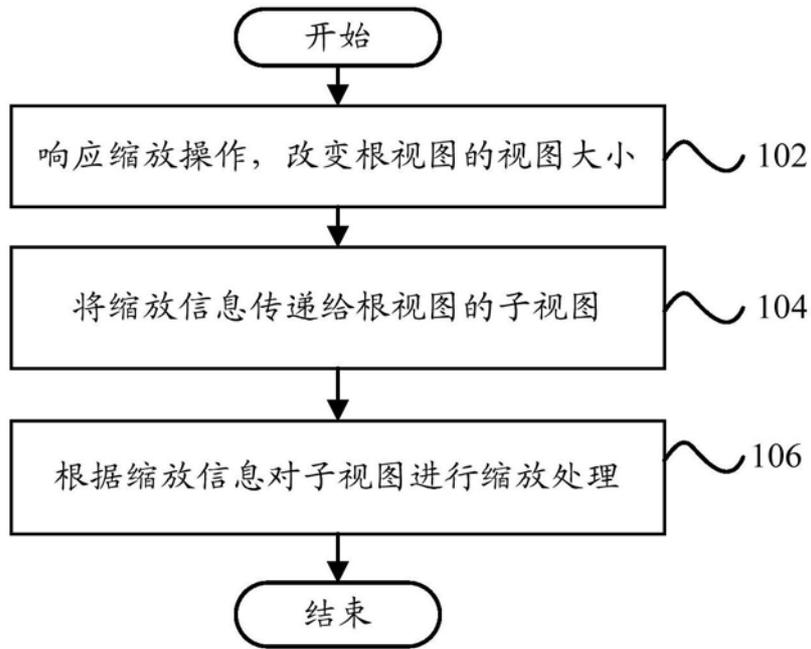


图1

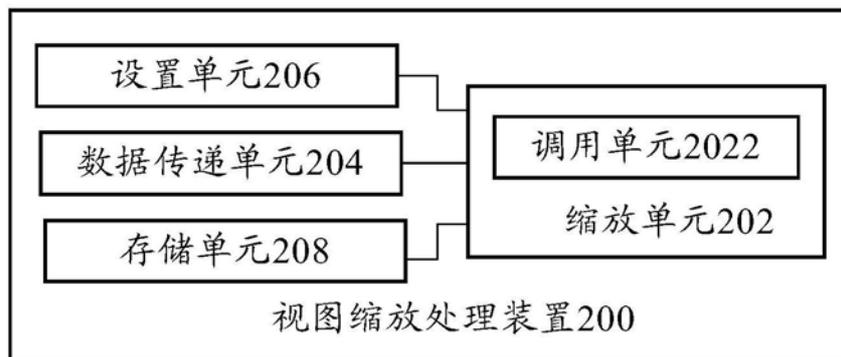


图2