



(12) 发明专利申请

(10) 申请公布号 CN 103546816 A

(43) 申请公布日 2014. 01. 29

(21) 申请号 201310509984. 1

(51) Int. Cl.

(22) 申请日 2013. 10. 25

H04N 21/472(2011. 01)

H04N 21/422(2011. 01)

(66) 本国优先权数据

201310158245. 2 2013. 05. 02 CN

(71) 申请人 乐视网信息技术(北京)股份有限公司

地址 100026 北京市海淀区学院南路 68 号
19 号楼六层 6184 号房间

(72) 发明人 贾跃亭 高健明 李亮 关佐龙
陈芾

(74) 专利代理机构 北京驰纳智财知识产权代理
事务所(普通合伙) 11367

代理人 谢亮 唐与芬

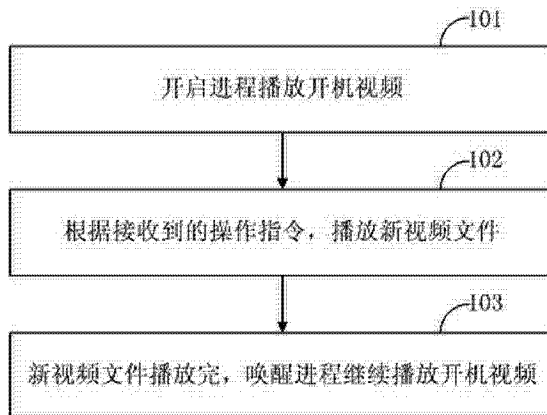
权利要求书1页 说明书7页 附图4页

(54) 发明名称

用户界面处理及视频播放方法及用户界面系统

(57) 摘要

本发明提供一种用户界面处理及视频播放方法及用户界面系统。该方法包括:将全视频用户界面设计为两个层,终端开启第一进程播放视频,将当前播放的视频层设置为背景层;当终端点播新的视频时,沉睡第一进程,开启第二进程呈现图形操作层,将图形操作层设置为前景层;第二进程将图形操作层展现在播放的视频层之上,并播放新视频文件;播放完新视频文件后,唤醒第一进程继续播放原有视频。采用本发明的方法及系统,利用多线程方案,将背景层视频和前景层用户界面作为两个进程,两个进程相对独立运行,互不影响,以确保用户在操作前景层用户界面时不会影响到背景层播放的视频,让软件的操作变得简单。



1. 一种用户界面处理及视频播放方法,其特征在于,包括以下步骤:

步骤 101,将全视频化用户界面分为两个层,终端开启第一进程播放视频,将当前播放的视频层设置为背景层;

步骤 102,当终端点播新的视频时,沉睡第一进程,开启第二进程呈现图形操作层,将图形操作层设置为前景层;所述第二进程将图形操作层展现在播放的视频层之上,并播放新视频文件;使用第一进程控制背景层,第二进程控制背景层,所述第一进程和第二进程单独运行;

步骤 103,播放完新视频文件后,唤醒第一进程继续播放原有视频。

2. 如权利要求 1 所述的用户界面处理及视频播放方法,其特征在于,所述终端开启第一进程播放视频的步骤包括:

启动 android 平台的 SurfaceView 和 VideoView 组件来播放视频。

3. 如权利要求 1 所述的用户界面处理及视频播放方法,其特征在于,

所述步骤 102 中还包括,用户选择系统菜单操作或选择节目菜单操作来选定新视频文件。

4. 如权利要求 1 所述的用户界面处理及视频播放方法,其特征在于,所述图形操作层为导航栏用户界面;

步骤 102 中,通过使用 Android 中多图层布局来实现利用 Android 中多图层布局实现文本控件和图形控件展示导航栏用户界面。

5. 如权利要求 1 所述的用户界面处理及视频播放方法,其特征在于:

所述全视频化用户界面包括视频界面全视频化、导航栏界面全视频化、首页界面全视频化、节目单全视频化。

6. 如权利要求 1 所述的用户界面处理及视频播放方法,其特征在于:

所述前景层为半透明的图形操作层,包括显示导航界面。

7. 如权利要求 6 所述的用户界面处理及视频播放方法,其特征在于:

所述步骤 102 还包括:开启第三进程将所述显示导航界面展现在直播视频层之上。

8. 如权利要求 7 所述的用户界面处理及视频播放方法,其特征在于:

所述第三进程是通过使用 Android 中多图层布局来实现文本控件和 / 或图形控件展现在直播视频层之上;或者所述第三进程通过设置 Fragment 组件为透明来实现将导航界面展现在直播视频层之上。

9. 如权利要求 5 所述的用户界面处理及视频播放方法,其特征在于:

所述将背景层作为第一进程,包括将背景层视频播放作为第一进程,所述将前景层作为第二进程,包括将前景层人机交互作为第二进程,所述第一进程和第二进程单独运行,由程序来维护这两个进程的资源平衡。

10. 一种用户界面系统,其特征在于,该系统包括:

分层模块,将视频化用户界面分为背景层和前景层;所述前景层置于所述背景层之上;

背景层模块,将当前播放视频的节目层设置为背景层;

前景层模块,将图形操作层设置为前景层;

多进程模块,根据指令,开启第一进程控制背景层模块,开启第二进程控制前景层模块。

用户界面处理及视频播放方法及用户界面系统

[0001] 本发明请求本申请人于 2013 年 5 月 2 日向中国国家知识产权局提交的申请号为 CN201310158245.2, 发明名称为“一种全视频化用户界面的实现方法”的中国发明专利申请的优先权, 上述申请的全部内容以引用方式并入本文。

技术领域

[0002] 本发明涉及智能电视的用户界面领域, 尤其涉及一种用户界面处理及视频播放方法及用户界面系统。

背景技术

[0003] 用户界面指软件的人机交互、操作逻辑、界面美观的整体。好的人机界面不仅是让软件变得有个性的味道, 还要让软件的操作变得舒适、简单、自由、充分体现软件的定位和特点。

[0004] 一个友好美观的界面会给人带来舒适的视觉享受, 拉近人与电视的距离, 为商家创造卖点。用户界面的三大原则是: 置界面于用户的控制之下; 减少用户的记忆负担; 保持界面的一致性。

[0005] 然而现有技术中的用户界面存在诸多问题, 主要缺陷在于用户界面过于复杂, 影响用户正常视频观看。例如用户使用智能电视的核心目的是观看喜爱的视频节目, 而现在的智能电视经常让用户陷入繁复的查找、检索和搜索之中, 浪费了很多精力却无法看到自己想要的內容, 或者在操作过程中忘记了自己想要看的视频內容。例如在申请号为 CN201110460522.6 的中国专利申请中, 提及一种智能电视界面的显示方法, 在该方法中就存在用户界面布局复杂, 占用视频播放页面大, 影响用户持续正常观看电视的问题。

发明内容

[0006] 针对现有技术的缺陷, 如何解决用户界面过于复杂, 影响用户正常视频观看成为本方法现在需要解决的主要问题。

[0007] 为了解决现有技术存在的以上问题, 本发明先提供一种用户界面处理及视频播放方法, 包括以下步骤: 步骤 101, 将全视频化用户界面分为两个层, 终端开启第一进程播放视频, 将当前播放的视频层设置为背景层; 步骤 102, 当终端点播新的视频时, 沉睡第一进程, 开启第二进程呈现图形操作层, 将图形操作层设置为前景层; 所述第二进程将图形操作层展现在播放的视频层之上, 并播放新视频文件; 使用第一进程控制背景层, 第二进程控制背景层, 所述第一进程和第二进程单独运行; 步骤 103, 播放完新视频文件后, 唤醒第一进程继续播放原有视频。

[0008] 上述方法中, 所述终端开启第一进程播放视频的步骤包括:

启动 android 平台的 SurfaceView 和 VideoView 组件来播放视频。

[0009] 较佳地, 所述步骤 102 中还包括, 用户选择系统菜单操作或选择节目菜单操作来选定新视频文件。

[0010] 上述方法中,所述图形操作层为导航栏用户界面;

步骤 102 中,通过使用 Android 中多图层布局来实现利用 Android 中多图层布局实现文本控件和图形控件展示导航栏用户界面。

[0011] 上述方法中,所述全视频化用户界面包括视频界面全视频化、导航栏界面全视频化、首页界面全视频化、节目单全视频化。

[0012] 上述方法中,所述前景层为半透明的图形操作层,包括显示导航界面。

[0013] 较佳地,所述步骤 102 还包括:开启第三进程将所述显示导航界面展现在直播视频层之上。

[0014] 上述方法中,所述第三进程是通过使用 Android 中多图层布局来实现文本控件和/或图形控件展现在直播视频层之上;或者所述第三进程通过设置 Fragment 组件为透明来实现将导航界面展现在直播视频层之上。

[0015] 上述方法中,所述将背景层作为第一进程,包括将背景层视频播放作为第一进程,所述将前景层作为第二进程,包括将前景层人机交互作为第二进程,所述第一进程和第二进程单独运行,由程序来维护这两个进程的资源平衡。

[0016] 本发明提供了一种用户界面系统,该系统包括:

分层模块,将视频化用户界面分为背景层和前景层;所述前景层置于所述背景层之上;

背景层模块,将当前播放视频的节目层设置为背景层;

前景层模块,将图形操作层设置为前景层;

多进程模块,根据指令,开启第一进程控制背景层模块,开启第二进程控制前景层模块。

[0017] 上述系统中,所述前景层包括显示导航界面,所述前景层模块通过使用 Android 文本控件或图形控件将所述导航界面作为前景层,或者通过使用 Android 中多图层布局将文本控件和/或图形控件展现在所述背景层之上。

[0018] 本发明书中涉及一些术语,特此解释:

SurfaceView 和 VideoView :android 开发中使用到的的两个类,用于图形处理。

[0019] UI :用户界面。

[0020] m3u8 :一种视频格式。

[0021] Fragment 组件 :一种代表应用局部局部区域并能单独与用户进行交互的组件。

[0022] ActivityManager 类 :android 开发中使用到的一个类。

[0023] Relative Layout :一种布局方式,允许子元素指定它们相对于其父元素或兄弟元素的位置。

[0024] 本发明用户界面处理及视频播放方法及用户界面系统中,全视频化用户界面的实现,主要是通过智能电视使用过程中将视频播放作为背景层,将用户界面作为前景层,两个层分别由两个进程来分开运行,互不干扰来实现的,本发明目的是让软件的操作变得舒适、简单、自由、充分体现软件的定位和特点,实现为一个友好美观的界面,给人带来人性化的用户体验,拉近人与电视的距离。

附图说明

[0025] 图 1 为作为按照本发明的全视频化用户界面的一个优选实施例,视频界面全视频化的示意图。

[0026] 图 2 为作为按照本发明的全视频化用户界面的一个优选实施例,导航栏界面全视频化的示意图。

[0027] 图 3 为作为按照本发明的全视频化用户界面的一个优选实施例,系统流程示意图。

[0028] 图 4 为作为按照本发明的全视频化用户界面的一个优选实施例,首页界面全视频化的示意图。

[0029] 图 5 为作为按照本发明的全视频化用户界面的一个优选实施例,节目单全视频化的示意图。

[0030] 图 6 为本发明用户界面处理及视频播放方法的流程图。

具体实施方式

[0031] 下面将结合本发明实施例中的附图,对本发明实施例中的技术方案进行清楚、完整地描述,显然,所描述的实施例仅仅是本发明一部分实施例,而不是全部的实施例。基于本发明中的实施例,本领域普通技术人员在没有做出创造性劳动前提下所获得的所有其他实施例,都属于本发明保护的范围。

[0032] 本发明第一实施例提供一种智能电视全视频化用户界面的实现方法,包括视频界面的全视频化、导航栏界面的全视频化、首页界面的全视频化以及节目单的全视频化,让用户更方便的使用用户界面,无干扰的观看视频。

[0033] 本发明提供的一种用户界面处理及视频播放方法包括以下步骤:将全视频化用户界面分为两个层,终端开启第一进程播放视频,将当前播放的视频层设置为背景层;当终端点播新的视频时,沉睡第一进程,开启第二进程呈现图形操作层,将图形操作层设置为前景层;第二进程将图形操作层展现在播放的视频层之上,并播放新视频文件;使用第一进程控制背景层,第二进程控制背景层,第一进程和第二进程单独运行;播放完新视频文件后,唤醒第一进程继续播放原有视频。

[0034] 本发明用户界面处理及视频播放方法包括以下步骤:

步骤 101,开启进程播放开机视频。

[0035] 该步骤包括:根据外部对客户端的第一操作指令,开启用以进行播放控制的进程;用以进行播放控制的进程根据第一操作指令播放视频文件,并将该视频作为开机视频。其中,第一操作指令中还可携带需播放的视频文件的标识信息,比如:名称、类型、创建时间等。

[0036] 以 Android 平台为例进行描述,当然本领域的技术人员应当清楚,在 IOS、Windows 等平台中实现本发明中的步骤也是可以的。

[0037] 用以进行播放控制的进程启动 android 平台的 SurfaceView 和 VideoView 组件来播放视频。其中, surfaceview 的核心在于提供了两个线程:UI 线程和渲染线程;VideoView 用于播放视频文件,可以从不同的来源通过视频请求协议读取图像,计算和维护视频的画面尺寸以使其适用于任何布局管理器,并提供一些诸如缩放、着色之类的显示选项。VideoView 组件所采用的视频请求协议为 http 协议,目标文件为 m3u8。图 1 描述了客

户端根据外部的操作指令进行当前视频播放的示意图,该当前视频可以为开机视频。

[0038] 步骤 102,根据接收到的操作指令,播放新视频文件。

[0039] 该步骤包括:根据接收到的第二操作指令,沉睡(sleeping)步骤 101 中的用以进行播放控制的进程,开启用以进行操作控制的进程;用以进行操作控制的进程将步骤 101 中播放视频文件的层作为背景层,将根据第二操作指令调用的图形操作层置于背景层之前,作为前景层;根据接收到的第三操作指令,用以进行操作控制的进程播放新视频文件。

[0040] 其中,第二操作指令可为用户在观看步骤 101 中播放的视频文件时进行的需要进行新视频文件播放的指令;图形操作层可以是导航栏界面、首页界面或节目单界面;第三操作指令为用户对图形操作层进行操作后根据具体的操作内容生成的指令。

[0041] 例如:当用户点播其他视频时,播放用户选择的新视频文件。在此之前还包括该进程,开启另一个进程,展示导航栏用户界面,该进程将导航用户界面展现在直播视频层之上,保证用户在操作导航栏界面的时候所观看的视频不会受到影响。图 2 展示了导航栏界面全视频化的场景。

[0042] 参见附图 3,上述方法可以描述为,客户端开启进程播放直播流并等待用户操作。用户选择系统菜单操作或选择节目菜单操作,开启另一个进程,展示导航栏用户界面,该进程将导航用户界面展现在直播视频层之上,保证用户在操作导航栏界面的时候所观看的视频不会受到影响。将系统菜单全凭化处理;则播放用户选择的新视频文件。

[0043] 例如,当用户点击遥控器【上】按键时,程序开启另一个进程,展示导航栏用户界面。在该步骤中,进程接受用户输入的信息,然后通知后台程序展示导航栏用户界面。在该步骤中,进程数据通过 http 协议取自后台服务器。

[0044] 优选地,步骤 102 中进程将导航用户界面展现在直播视频层之上是通过使用 Android 中多图层布局来实现的,利用 Android 中多图层布局实现文本控件和图形控件展示导航栏用户界面。在该步骤中,进程通过设置 Fragment 组件为透明来实现导航栏界面的透明程度使之展现在直播视频层之上,如果用户需要,还可以调节导航栏界面的透明程度以适应自身的需要,保证不影响到背景视频的正常播放。

[0045] 图 4 展示了按照本发明的一个实施例首页界面的全视频化的场景。

[0046] 用户在打开首页界面时,屏幕上显示出首页界面。同时正在播放的视频作为背景视频存在。用户在操作首页界面时,不影响背景视频的正常播放。首页界面是第一进程,背景层视频播放作为第二进程,由一个程序来维护两个进程的资源平衡。更进一步,第一进程为一个人机交互进程,第二进程为一个视频播放进程。

[0047] 在一个非限制性的实现上,维护两个进程的资源程序通过图片按需加载,使用 ActivityManager 类相关 API 来监测内存资源和显存资源,然后通过对两个进程的资源管理分配来保证用户操作的无卡顿和背景视频的流畅播放。

[0048] 图 5 展示了按照本发明的一个实施例节目单的全视频化的场景。

[0049] 用户在全屏观看视频时,系统后台静默缓冲视频相关数据。保证用户观看视频的流畅。当用户想查看节目单时,按遥控器【下】键,在屏幕下方显示节目单以供用户操作,同时后台背景视频正常播放,确保用户查看节目单时,不影响背景视频正常播放。非限制性地,该步骤中使用 android 中按钮事件处理弹出 Frament 浮层,设置背景透明来实现节目单浮现于视频层之上,android 中 Relative Layout 相对布局来让节目单只呈现于屏幕底部

固定位置,人机交互进程唤醒底部节目单,方便用户找到需要的目标,同时不影响用户视频的正常观看。

[0050] 步骤 103,新视频文件播放完,唤醒 (running) 进程继续播放开机视频。

[0051] 该步骤包括:在新视频文件播放完后,沉睡用以进行操作控制的进程,唤醒用以进行播放控制的进程,用以进行播放控制的进程将开机视频置于前景层,播放开机视频。

[0052] 本发明的第二实施方式中,提供一种用户界面全视频化方法,以方便用户使用及保证用户持续正常的观看视频。该方法可通过如下步骤来实现:

步骤 A、将全视频化用户界面分为两个层,其中,将当前播放的影视节目层设置为背景层,将半透明的图形操作层设置为前景层;

步骤 B、将背景层作为第一进程,将前景层作为第二进程,第一进程和第二进程单独运行,互不影响,从而实现用户在操作前景层用户界面时不会影响到背景层播放的视频。

[0053] 其中,全视频化,是指终端在播放视频时,尤其在整個显示界面都播放视频时,客户端或视频网站仍旧可以根据终端发送的操作指令进行相应的操作,不影响视频的播放,并可以将正在播放的视频作为背景,将供用户操作的界面显示在前端。

[0054] 图形操作层可以为半透明的,并可以显示更多的用户可操作的信息,甚至是全屏在半透明状态下,将提供给用户的操作信息显示在视频前端。

[0055] 优选的是,步骤 A 中全视频化用户界面包括视频界面全视频化、导航栏界面全视频化、首页界面全视频化、节目单全视频化。的视频界面包括智能电视机一开机就显示的界面。

[0056] 步骤 A 中,将当前播放的影视节目层设置为背景层,包括开机播放视频作为一个层独立运行,只有当用户点播了其他视频的时候,才会切换,其他视频播放完毕的时候,依然播放原视频。

[0057] 更优选的是,播放视频是通过第三进程来播放的。例如第三进程是通过启动 android 平台的 SurfaceView 和 VideoView 组件来播放视频。Surfaceview 提供了两个线程:UI 线程和渲染线程。VideoView 用于播放视频文件,可以从不同的来源通过视频请求协议读取图像,计算和维护视频的画面尺寸,以使其适用于任何布局管理器,并提供一些诸如缩放、着色之类的显示选项。来源为资源文件、内容提供者,视频请求协议为 http 协议,目标文件为 m3u8。

[0058] 在步骤 A 中,将半透明的图形操作层设置为前景层,包括显示导航界面,将导航界面作为一个层展现在直播视频层之上,供用户操作控制。

[0059] 此外,显示导航界面并将之作为一个层展现在直播视频层之上是通过程序开启第四进程来实现的。第四进程是通过使用 Android 中多图层布局来实现文本控件和图形控件或其中任一项展现在直播视频层之上;或者第四进程通过设置 Fragment 组件为透明来实现将导航界面展现在直播视频层之上。更优选的是,第四进程的调用后台服务器是通过 http 协议取自后台服务器以实现的。

[0060] 在步骤 B 中,将背景层作为第一进程,包括将背景层视频播放作为第一进程,将前景层作为第二进程,包括将前景层人机交互作为第二进程,第一进程和第二进程单独运行,互不影响,包括由程序来维护这两个进程的资源平衡。维护两个进程的资源平衡的程序,是通过按需加载图片,ActivityManager 类相关 API 来监测内存资源和显存资源,以安排可

以在第一进程和第二进程中使用的资源,从而保证用户操作的无卡顿和背景视频的流畅播放。

[0061] 节目单界面全视频化包括当用户需要的时候,人机交互进程唤醒底部节目单,让用户获取焦点实现交互,来保证提供用户的需要。调用节目单是当用户需要的时候,android 中按钮事件处理弹出 Frament 浮层,设置背景透明来实现节目单浮现于视频层之上。可以通过 android 中 Relative Layout 相对布局来让节目单只呈现于屏幕底部固定位置,人机交互进程唤醒底部节目单,让用户获取焦点实现交互。

[0062] 在本发明的另一实施方式中,提供一种全视频化用户界面系统,包括:

一个分层模块,用于将视频化用户界面设计为两个层;该两个层分别为背景层和前景层,前景层置于背景层之上。

[0063] 一个背景层模块,用于将当前播放视频的节目层设置为背景层;

一个前景层模块,用于将图形操作层设置为前景层;其中,图形操作层为半透明状。

[0064] 一个多进程模块,根据指令,开启第一进程控制背景层模块,开启第二进程控制前景层模块。具体的,将背景层作为第一进程,将前景层作为第二进程,利用多线程技术,使第一进程和第二进程单独运行,互不影响,从而实现用户在操作前景层用户界面时不会影响到背景层播放的视频。

[0065] 优选的是,分层模块中的视频化用户界面包括全视频化视频界面、全视频化导航界面、全视频化首页界面、全视频化节目单界面。

[0066] 上述任一方案中优选的是,背景层模块中,将当前播放的影视节目层设置为背景层,是通过开机启动第三进程来实现的,该进程启动 android 平台的 SurfaceView 和 VideoView 组件来播放视频,并将视频播放作为一个单独的层来运行。

[0067] 更优选的是, Surfaceview 提供了两个线程:UI 线程和渲染线程。

[0068] 更优选的是, VideoView 用于播放视频文件,可以从不同的来源通过视频请求协议读取图像,计算和维护视频的画面尺寸,以使其适用于任何布局管理器,并提供一些诸如缩放、着色之类的显示选项。

[0069] 更优选的是,来源为资源文件。

[0070] 更优选的是,来源为内容提供者。

[0071] 更优选的是,视频请求协议为 http 协议。

[0072] 更优选的是,目标文件为 m3u8。

[0073] 更优选的是,将视频播放作为一个单独的层来运行,包括只有当用户点播了其他视频的时候,才会切换到其他视频,待其他视频播放完毕后,依然播放原视频来实现的。

[0074] 前景层模块中,将半透明的图形操作层设置为前景层,包括当用户需要显示导航界面时,开启第四进程,该进程通过使用 Android 文本控件和图形控件或其中任一种将导航界面作为前景层展现在直播视频层之上。第四进程通过使用 Android 中多图层布局来实现文本控件和图形控件或其中任一种展现在直播视频层之上。第四进程通过设置 Fragment 组件为透明,来实现弹出导航栏设置背景透明,即将导航栏作为前景层展现在直播视频层之上,供用户操作控制。第四进程中的数据通过 http 协议取自后台服务器。

[0075] 多进程模块中,将背景层作为第一进程,包括将背景层视频播放作为第一进程,将前景层作为第二进程,包括将前景层人机交互作为第二进程,第一进程和第二进程单独运

行,互不影响,包括由程序来维护这两个进程的资源平衡。

[0076] 更优选的是,该系统还可通过图片按需加载,通过 ActivityManager 类相关 API 来监测内存资源和显存资源,以安排可以在第一进程和第二进程中使用的资源,从而保证用户操作的无卡顿和背景视频的流畅播放。

[0077] 全视频化节目单界面包括当用户需要的时候,人机交互进程唤醒底部节目单,让用户获取焦点实现交互,来保证提供用户的需要。当用户需要的时候,唤醒 android 中按钮事件,弹出 Frament 浮层,设置背景透明来实现节目单浮现于视频层之上。进一步,使用 android 中 Relative Layout 相对布局来让节目单只呈现于屏幕底部固定位置。

[0078] 以下以智能电视点播影片《钢铁侠 3》为例,描述上述方法的实际应用:智能电视开机,播放开机视频。用户使用遥控器点播影片《钢铁侠 3》,智能电视停止播放背景视频,开始播放影片《钢铁侠 3》。用户点击遥控器【上】按键,屏幕播放影片《钢铁侠 3》,同时在屏幕上方显示导航栏界面供用户操作。用户操作遥控器,在导航栏界面选择首页界面,屏幕播放影片《钢铁侠 3》,同时在屏幕上显示首页界面。用户操作遥控器,选择全屏播放视频。将影片《钢铁侠 3》放大至全屏幕播放。用户操作遥控器,点击遥控器【下】键,全屏播放影片《钢铁侠 3》,同时在屏幕下方显示节目单界面。用户操作遥控器,选择结束观看,屏幕结束播放影片《钢铁侠 3》,开始播放开机视频。

[0079] 本发明中全视频化用户界面的实现,主要是通过智能电视使用过程中将视频播放作为背景层,将用户界面作为前景层,两个层分别由两个进程来分开运行,互不干扰来实现的,本发明目的是让软件的操作变得舒适、简单、自由、充分体现软件的定位和特点,实现为一个友好美观的界面,给人带来人性化的用户体验,拉近人与电视的距离。

[0080] 技术人员不难看出,上述各优选方案的任意组合所构成的方案都是本发明的一部分。

[0081] 应该理解,本发明的方法中所涉及的硬件或软件,可以采用任何现有技术中适用的硬件或软件,或在适当时结合两者的组合来实现。因此,本发明的方法,可以采用包含在诸如软盘、CD-ROM、硬盘驱动器或任何其他机器可读存储介质等有形介质中的程序代码(即,指令)的形式,其中,当程序代码在可编程计算机上执行的情况下,计算设备通常包括处理器、该处理器可读的存储介质(包括易失性存储器和/或存储元件)、至少一个输入设备、以及至少一个输出设备。一个或多个程序可以例如,通过使用 API,可重用控件等来实现或利用结合本发明描述的过程。这样的程序优选地用高级过程语言或面向对象编程语言来实现,以与计算机系统通信。然而,如果需要,该程序可以用汇编语言或机器语言来实现。在任何情形中,语言可以是编译语言或解释语言,且与硬件实现相结合。

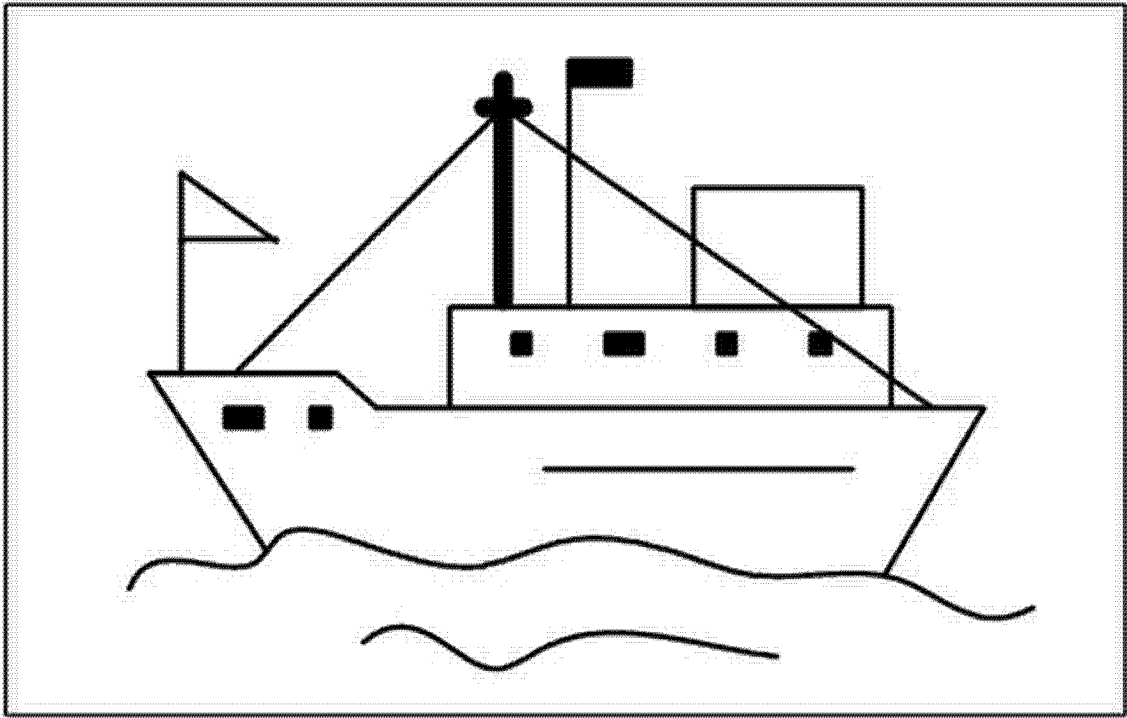


图 1

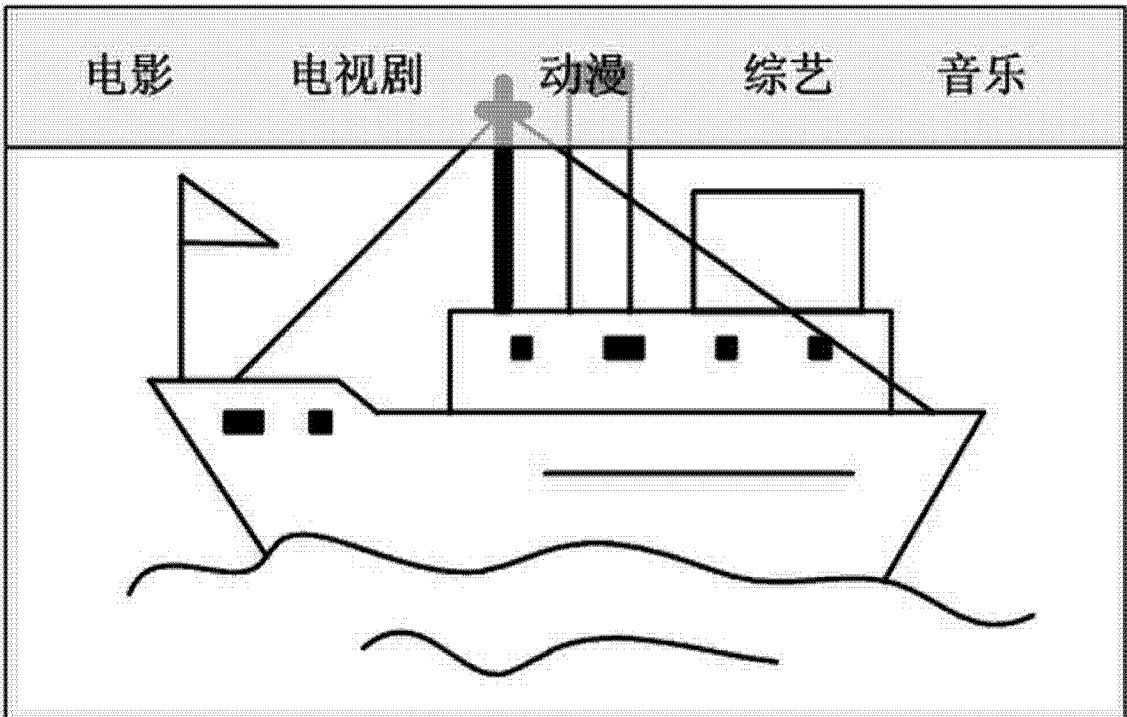


图 2

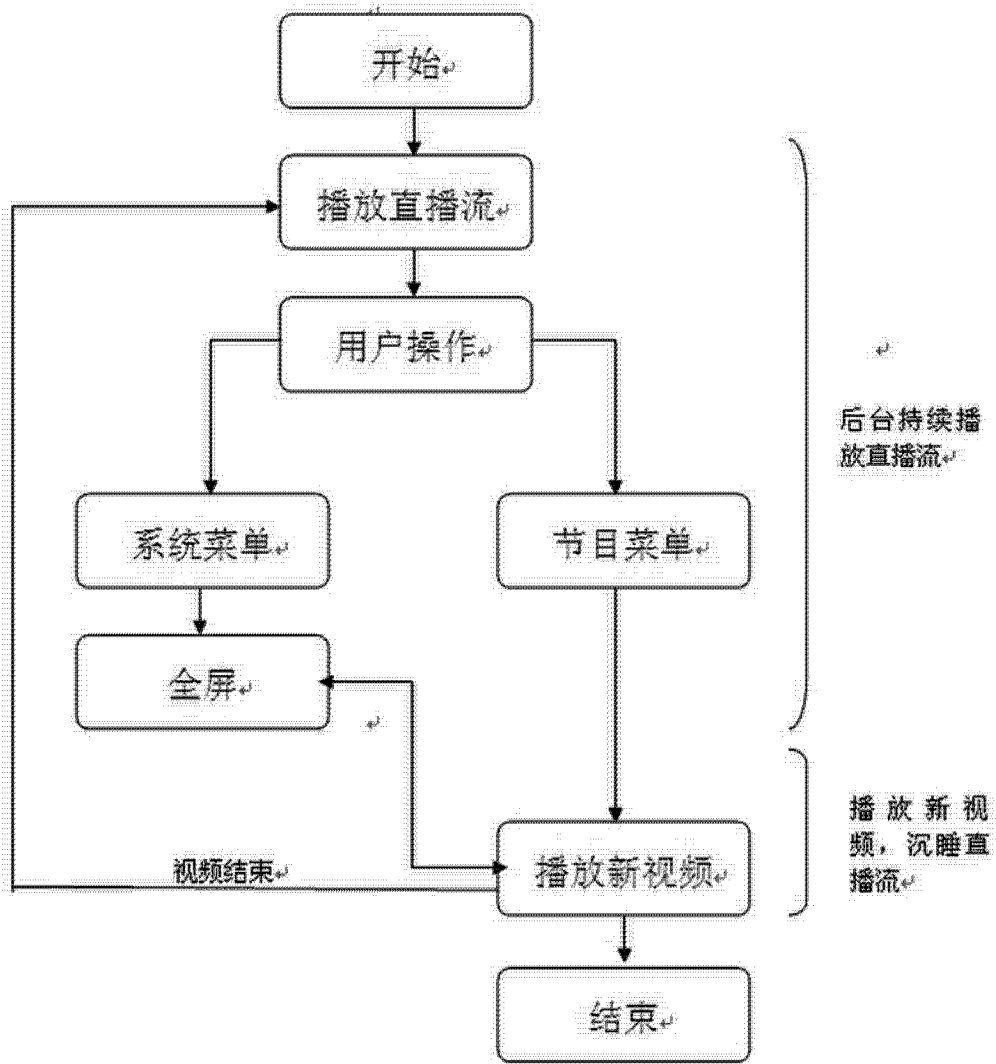


图 3

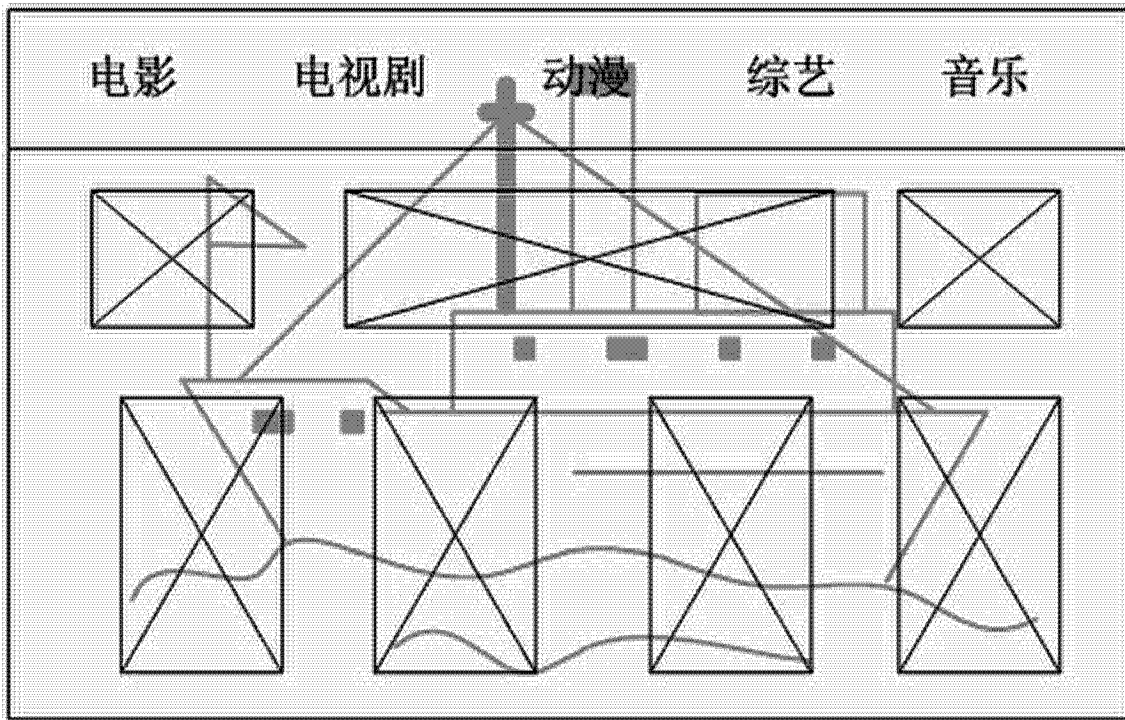


图 4

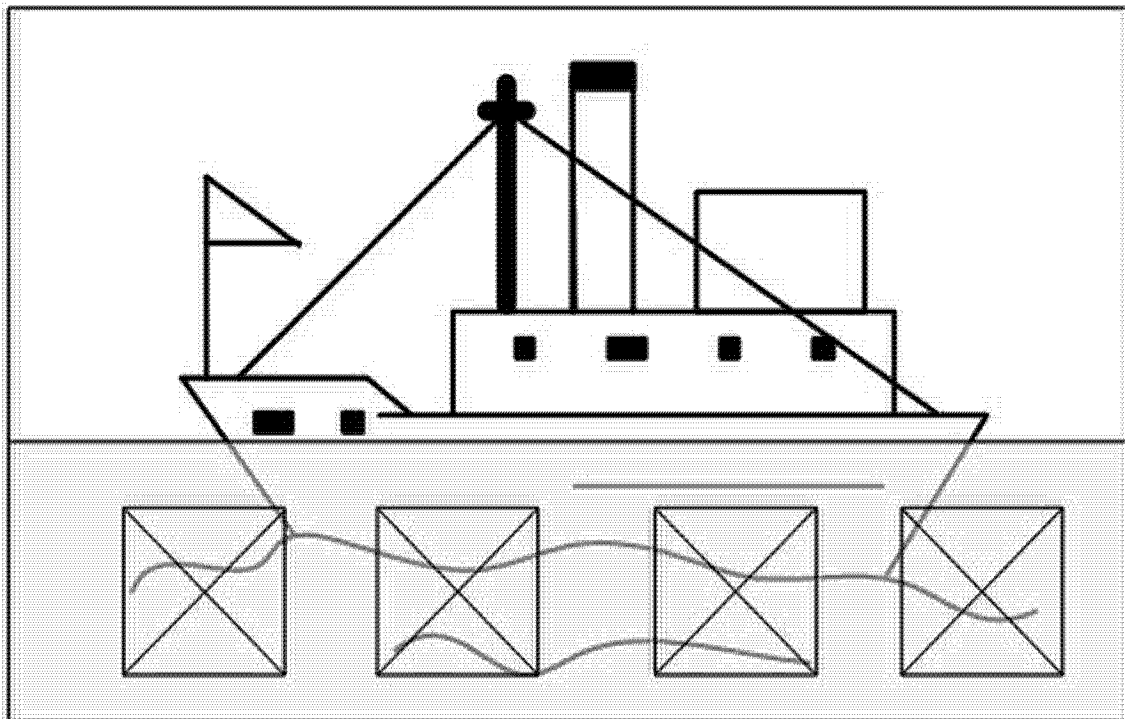


图 5

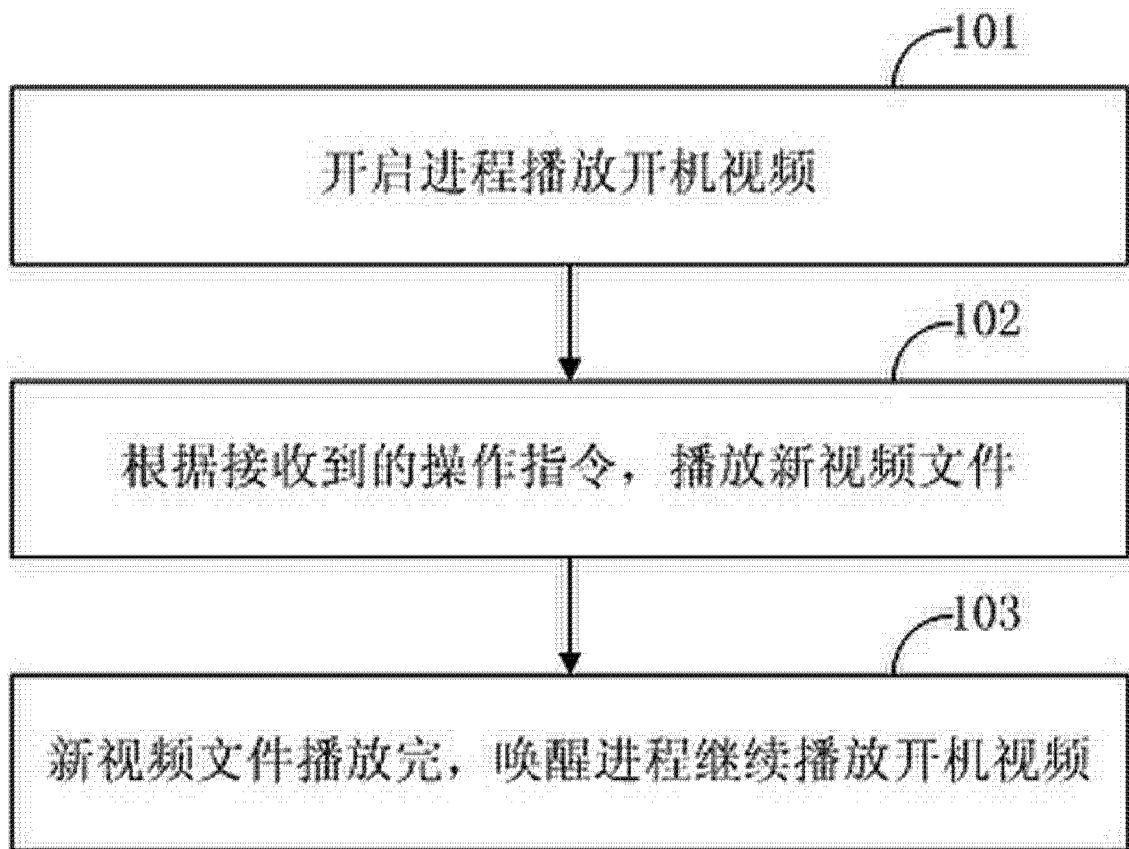


图 6