



(12)发明专利

(10)授权公告号 CN 105797377 B

(45)授权公告日 2019.08.23

(21)申请号 201610134651.9

A63F 13/56(2014.01)

(22)申请日 2016.03.10

A63F 13/537(2014.01)

(65)同一申请的已公布的文献号

申请公布号 CN 105797377 A

(56)对比文件

CN 104392171 A, 2015.03.04,

CN 102567661 A, 2012.07.11,

CN 104331664 A, 2015.02.04,

CN 103096134 A, 2013.05.08,

(43)申请公布日 2016.07.27

审查员 陈丽静

(73)专利权人 武汉斗鱼网络科技有限公司

地址 430000 湖北省武汉市武汉东湖开发区软件园东路1号软件产业4.1期B1栋11楼

(72)发明人 周志刚

(74)专利代理机构 武汉东喻专利代理事务所

(普通合伙) 42224

代理人 胡星驰

(51)Int.Cl.

A63F 13/55(2014.01)

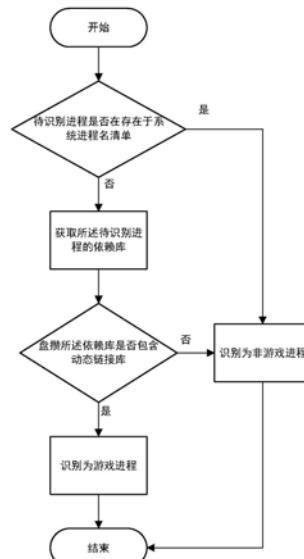
权利要求书2页 说明书6页 附图4页

(54)发明名称

一种游戏进程识别和显示方法及系统

(57)摘要

本发明公开了一种游戏进程识别及显示方法及系统。所述游戏进程识别方法，包括：(1)判断待识别进程是否为系统进程，如所述待识别进程为系统进程则判断所述进程为非游戏进程，否则进入步骤(2)；(2)获取所述待识别进程的依赖库；(3)判断所述依赖库是否包含图像渲染动态链接库；(4)当所述进程的依赖库中包含图像渲染动态链接库时，则识别所述待识别进程为游戏进程，否则识别所述待识别进程为非游戏进程。所述游戏进程显示方法，在识别游戏进程后，将游戏进程优先显示。所述游戏进程识别及显示系统，用于实现所述方法。所述游戏进程识别及显示方法及系统，能区分游戏进程并进行优先显示，方便用户浏览及查找，提高用户体验。



1. 一种游戏进程识别方法,其特征在于,包括以下步骤:

(1) 判断待识别进程是否为系统进程,如所述待识别进程为系统进程则判断所述进程为非游戏进程,否则进入步骤(2);

所述判断待识别进程是否为系统进程具体为:

在预先收集的系统进程名清单中查找所述待识别进程的进程名,如果所述待识别进程的进程名存在于所述系统进程名清单中,则判断所述进程为系统进程,否则判断所述进程为非系统进程;

(2) 获取所述待识别进程的依赖库;

所述获取所述待识别进程的依赖库具体为:

通过系统应用程序编程接口枚举所述待识别进程加载的动态链接库,所述待识别进程加载的所有动态链接库形成所述待识别进程的依赖库;

(3) 判断所述依赖库是否包含图像渲染动态链接库;

所述判断所述依赖库是否包含图像渲染动态链接库具体为:

判断所述依赖库是否包含directx图像渲染动态链接库和/或opengl图像渲染动态链接库;

(4) 当所述进程的依赖库中包含图像渲染动态链接库时,则识别所述待识别进程为游戏进程,否则识别所述待识别进程为非游戏进程。

2. 如权利要求1所述的游戏进程识别方法,其特征在于,步骤(3)所述判断所述依赖库是否包含directx图像渲染动态链接库,即判所述依赖库是否包含文件名为d3d\*.d11的动态链接库文件,其中\*为通配符,匹配一个或多个数字字符。

3. 如权利要求1所述的游戏进程识别方法,其特征在于,步骤(3)所述判断所述依赖库是否包含opengl图像渲染动态链接库,即判断所述依赖库是否包含文件名为opengl32.dll的动态链接库文件。

4. 一种游戏进程识别系统,其特征在于,包括:

系统进程判断模块:用于判断待识别进程是否为系统进程,如果所述待识别进程为系统进程,则将结果提交至游戏进程判断模块;否则将进入依赖库获取模块;

所述系统进程判断模块包括系统进程名收集子模块,用于预先收集的系统进程名形成系统进程名清单,如果所述待识别进程的进程名存在于所述系统进程名清单中,则判断所述进程为系统进程,否则判断所述进程为非系统进程;

所述依赖库获取模块,用于获取所述待识别进程的依赖库,并将所述依赖库提交至图像渲染动态链接库判断模块;所述获取所述待识别进程的依赖库具体为:

通过系统应用程序编程接口枚举所述待识别进程加载的动态链接库,所述待识别进程加载的所有动态链接库形成所述待识别进程的依赖库;

所述图像渲染动态链接库判断模块,用于判断所述待识别进程依赖库中是否包含有图像渲染动态链接库,并将判断结果提交至游戏进程判断模块;所述图像渲染动态链接库为directx图像渲染动态链接库和/或opengl图像渲染动态链接库;

所述游戏进程判断模块,用于判断所述待识别进程是否为游戏进程,当所述待识别进程为系统进程判断模块提交的被判断为系统进程的进程,则识别所述待识别进程为非游戏进程;当所述待识别进程为所述图像渲染动态链接库判断模块提交的被判断为其依赖库不

包含图像渲染动态链接库的待识别进程，则识别所述待识别进程为非游戏进程；当所述识别待识别进程为所述图像渲染动态链接库判断模块提交的被判断为其依赖库包含图像渲染动态链接库的待识别进程，则识别所述待识别进程为游戏进程。

5. 一种游戏进程优先显示方法，其特征在于，包括以下步骤：

S1、获取系统的进程列表；

S2、对于步骤S1中获取的进程列表中的每一个进程，按照权利要求1至3任意一项所述的方法识别所述进程是否为游戏进程；

S3、将步骤S2识别出的游戏进程的显示优先级提高；

S4、根据进程的显示优先级按照从高到低的顺序显示进程列表。

6. 如权利要求5所述的游戏进程优先显示方法，其特征在于，所述步骤S4，对于进程的显示优先级相同的进程按照进程名的字母顺序显示。

7. 一种游戏进程优先显示系统，其特征在于，包括：

进程列表获取模块，用于获取系统的进程列表；

游戏进程识别模块，用于按照权利要求1至3任意一项所述方法识别进程列表获取模块获取的进程列表中的每一个进程是否为游戏进程；

进程显示优先级重排模块，用于将游戏进程识别模块识别出的游戏进程的显示优先级提高，得到调整后的显示优先级；

进程显示模块，用于根据进程显示优先级重排模块得到的调整后的显示优先级按照从高到低的顺序显示进程列表。

## 一种游戏进程识别和显示方法及系统

### 技术领域

[0001] 本发明属于计算机技术领域,更具体地,涉及一种游戏进程识别和显示方法及系统。

### 背景技术

[0002] 目前,游戏直播软件都是通过注入一个动态链接库到游戏中来获取游戏的显示内容,然后通过网络发送显示内容,从而就实现了游戏直播。所以获取游戏进程是第一步,目前通用的做法是获取到系统的所有进程,通过一个列表进行显示,然而一般系统进程个数都有50个以上,用户从50个以上的进程中挑选到游戏进程是一个体验很不友好的操作。

[0003] 目前,并没有方法对进程列表进行优化,将所有的游戏进程排列到进程列表最前面,用户比较容易的就能勾选到游戏进程。

### 发明内容

[0004] 针对现有技术的以上缺陷或改进需求,本发明提供了一种游戏进程识别及显示方法,其目的在于通过游戏进程图像渲染特征的分析,识别游戏进程并进行优先显示,由此解决现有技术不能方便迅速的找到游戏进程的技术问题。

[0005] 为实现上述目的,按照本发明的一个方面,提供了一种游戏进程识别方法,包括以下步骤:

[0006] (1) 判断待识别进程是否为系统进程,如所述待识别进程为系统进程则判断所述进程为非游戏进程,否则进入步骤(2);

[0007] (2) 获取所述待识别进程的依赖库;

[0008] (3) 判断所述依赖库是否包含图像渲染动态链接库;

[0009] (4) 当所述进程的依赖库中包含图像渲染动态链接库时,则识别所述待识别进程为游戏进程,否则识别所述待识别进程为非游戏进程。

[0010] 优选地,所述游戏进程识别方法,其步骤(1)所述判断待识别进程是否为系统进程具体为:

[0011] 在预先收集的系统进程名清单中查找所述待识别进程的进程名,如果所述待识别进程的进程名存在于所述系统进程名清单中,则判断所述进程为系统进程,否则判断所述进程为非系统进程。

[0012] 优选地,所述游戏进程识别方法,其步骤(2)所述获取所述待识别进程的依赖库具体为:

[0013] 通过系统应用程序编程接口枚举所述待识别进程加载的动态链接库,所述待识别进程加载的所有动态链接库形成所述待识别进程的依赖库。

[0014] 优选地,所述游戏进程识别方法,其步骤(3)所述判断所述依赖库是否包含图像渲染动态链接库具体为:

[0015] 判断所述依赖库是否包含directx图像渲染动态链接库和/或opengl图像渲染动态链接库。

[0016] 优选地,所述游戏进程识别方法,其步骤(3)所述判断所述依赖库是否包含directx图像渲染动态链接库,即判断所述依赖库是否包含文件名为d3d\*.dll的动态链接库文件,其中\*为通配符,匹配一个或多个数字字符。

[0017] 优选地,所述游戏进程识别方法,其步骤(3)所述判断所述依赖库是否包含opengl图像渲染动态链接库,即判断所述依赖库是否包含文件名为opengl32.dll的动态链接库文件。

[0018] 按照本发明的另一个方面提供了一种游戏进程识别系统,包括:

[0019] 系统进程判断模块:用于判断待识别进程是否为系统进程,如果所述待识别进程为系统进程,则将结果提交至游戏进程判断模块;否则将进入依赖库获取模块;

[0020] 所述依赖库获取模块,用于获取所述待识别进程的依赖库,并将所述依赖库提交至图像渲染动态链接库判断模块;

[0021] 所述图像渲染动态链接库判断模块,用于判断所述待识别进程依赖库中是否包含有图像渲染动态链接库,并将判断结果提交至游戏进程判断模块;

[0022] 所述游戏进程判断模块,用于判断所述待识别进程是否为游戏进程,当所述待识别进程为系统进程判断模块提交的被判断为系统进程的进程,则识别所述待识别进程为非游戏进程;当所述待识别进程为所述图像渲染动态链接库判断模块提交的被判断为其依赖库不包含图像渲染动态链接库的待识别进程,则识别所述待识别进程为非游戏进程;当所述识别待识别进程为所述图像渲染动态链接库判断模块提交的被判断为其依赖库包含图像渲染动态链接库的待识别进程,则识别所述待识别进程为游戏进程。

[0023] 按照本发明的另一个方面,提供了一种游戏进程优先显示方法,包括以下步骤:

[0024] S1、获取系统的进程列表;

[0025] S2、对于步骤S1中获取的进程列表中的每一个进程,按照权利要求1至6任意一项所述的方法识别所述进程是否为游戏进程;

[0026] S3、将步骤S2识别出的游戏进程的显示优先级提高;

[0027] S4、根据进程的显示优先级按照从高到低的顺序显示进程列表。

[0028] 优选地,所述游戏进程优先显示方法,其步骤S4,对于进程的显示优先级相同的进程按照进程名的字母顺序显示。

[0029] 按照本发明的另一个方面,提供了一种游戏进程优先显示系统,其特征在于,包括:

[0030] 进程列表获取模块,用于获取系统的进程列表;

[0031] 游戏进程识别模块,用于按照权利要求1至6任意一项所述方法识别进程列表获取模块获取的进程列表中的每一个进程是否为游戏进程;

[0032] 进程显示优先级重排模块,用于将游戏进程识别模块识别出的游戏进程的显示优先级提高,得到调整后的显示优先级;

[0033] 进程显示模块,用于根据进程显示优先级重排模块得到的调整后的显示优先级按照从高到低的顺序显示进程列表。

[0034] 总体而言,通过本发明所构思的以上技术方案与现有技术相比,抓住游戏进程需

要进行图像渲染的特征,通过获取进程依赖库,判断进程依赖库中是否包含有图像渲染动态链接库文件来识别游戏进程,并进一步的优先显示游戏进程,从而方便用户浏览游戏进程,提高用户体验。

## 附图说明

- [0035] 图1是本发明实施例1提供的游戏进程识别方法流程图;
- [0036] 图2是本发明实施例2提供的游戏进程识别系统结构示意图;
- [0037] 图3是本发明实施例3提供的游戏进程显示方法流程图;
- [0038] 图4是本发明实施例4提供的游戏进程显示系统结构示意图。

## 具体实施方式

[0039] 为了使本发明的目的、技术方案及优点更加清楚明白,以下结合附图及实施例,对本发明进行进一步详细说明。应当理解,此处所描述的具体实施例仅仅用以解释本发明,并不用于限定本发明。此外,下面所描述的本发明各个实施方式中所涉及到的技术特征只要彼此之间未构成冲突就可以相互组合。

- [0040] 本发明提供的游戏进程的识别方法,包括以下步骤:
  - [0041] (1) 判断待识别进程是否为系统进程,如所述待识别进程为系统进程则判断所述进程为非游戏进程,否则进入步骤(2);
  - [0042] 所述系统进程为系统运行常用的非游戏进程,如操作系统创建的进程。
  - [0043] 所述判断待识别进程是否为系统进程,具体为:
  - [0044] 在预先收集的系统进程名清单中查找所述待识别进程的进程名,如果所述待识别进程的进程名存在于所述系统进程名清单中,则判断所述进程为系统进程,否则判断所述进程为非系统进程。
  - [0045] (2) 获取所述待识别进程的依赖库;
  - [0046] 具体步骤为:通过系统应用程序编程接口,如Windows API,枚举所述待识别进程加载的动态链接库,所述待识别进程加载的所有动态链接库形成所述待识别进程的依赖库。
  - [0047] (3) 判断所述依赖库是否包含图像渲染动态链接库;
  - [0048] 优选地,判断所述依赖库是否包含directx图像渲染动态链接库和/或opengl图像渲染动态链接库。
  - [0049] 具体而言,判断所述依赖库是否包含directx图像渲染动态链接库,即判断所述依赖库是否包含文件名为d3d\*.dll的动态链接库文件,其中\*为通配符,匹配一个或多个数字字符,如d3d8.dll或d3d9.dll或d3d10.dll或d3d11.dll,其中d3d后面的数字就是directx对应的版本号。
  - [0050] 判断所述依赖库是否包含opengl图像渲染动态链接库,即判断所述依赖库是否包含文件名为opengl32.dll的动态链接库文件。
  - [0051] (4) 当所述进程的依赖库中包含图像渲染动态链接库时,则识别所述待识别进程为游戏进程,否则识别所述待识别进程为非游戏进程。
  - [0052] 本发明提供的游戏进程识别系统,包括:

[0053] 系统进程判断模块:用于判断待识别进程是否为系统进程,如果所述待识别进程为系统进程,则将结果提交至游戏进程判断模块;否则将进入依赖库获取模块;

[0054] 所述系统进程为系统运行常用的非游戏进程,如操作系统创建的进程。

[0055] 所述进程判断模块包括系统进程名收集子模块,用于预先收集的系统进程名形成系统进程名清单,如果所述待识别进程的进程名存在于所述系统进程名清单中,则判断所述进程为系统进程,否则判断所述进程为非系统进程。

[0056] 所述依赖库获取模块,用于获取所述待识别进程的依赖库,并将所述依赖库提交至图像渲染动态链接库判断模块;

[0057] 所述图像渲染动态链接库判断模块,用于判断所述待识别进程依赖库中是否包含有图像渲染动态链接库,并将判断结果提交至游戏进程判断模块;

[0058] 所述图像渲染动态链接库,优选为directx图像渲染动态链接库和/或opengl图像渲染动态链接库。

[0059] directx图像渲染动态链接库文件其文件名为d3d\*.d11的动态链接库文件,其中\*为通配符,匹配一个或多个数字字符,如d3d8.d11或d3d9.d11或d3d10.d11或d3d11.d11,其中d3d后面的数字就是directx对应的版本号。

[0060] Opengl图像渲染动态链接库其文件名为opengl32.d11。

[0061] 所述游戏进程判断模块,用于判断所述待识别进程是否为游戏进程,当所述待识别进程为系统进程判断模块提交的被判断为系统进程的进程,则识别所述待识别进程为非游戏进程;当所述待识别进程为所述图像渲染动态链接库判断模块提交的被判断为其依赖库不包含图像渲染动态链接库的待识别进程,则识别所述待识别进程为非游戏进程;当所述待识别进程为所述图像渲染动态链接库判断模块提交的被判断为其依赖库包含图像渲染动态链接库的待识别进程,则识别所述待识别进程为游戏进程。

[0062] 本发明提供的游戏进程优先显示方法,包括以下步骤:

[0063] S1、获取系统的进程列表;

[0064] S2、对于步骤S1中获取的进程列表中的每一个进程,按照本发明提供的方法识别所述进程是否为游戏进程;

[0065] S3、将步骤S2识别出的游戏进程的显示优先级提高;

[0066] 具体如共设2个优先级,从高到低为优先级1、0;将步骤S2识别出的游戏进程的显示优先级设为优先级1,非游戏进程优先级设为优先级0。

[0067] S4、根据进程的显示优先级按照从高到低的顺序显示进程列表。

[0068] 对于进程的显示优先级相同的进程按照进程名的字母顺序显示。

[0069] 本发明提供的游戏进程优先显示系统,包括:

[0070] 进程列表获取模块,用于获取系统的进程列表;

[0071] 游戏进程识别模块,用于按照权利要求本发明提供的游戏进程识别方法识别进程列表获取模块获取的进程列表中的每一个进程是否为游戏进程;

[0072] 进程显示优先级重排模块,用于将游戏进程识别模块识别出的游戏进程的显示优先级提高,得到调整后的显示优先级;具体地:

[0073] 可采用2层优先级,游戏进程是优先级1,非游戏进程是优先级0,优先级1高于优先级0。

[0074] 进程显示模块,用于根据进程显示优先级重排模块得到的调整后的显示优先级按照从高到低的顺序显示进程列表。对于进程的显示优先级相同的进程按照进程名的字母顺序显示。

[0075] 以下为实施例:

[0076] 实施例1

[0077] 一种游戏进程识别方法,如图1所示,包括以下步骤:

[0078] (1)对于待识别进程在预先收集的系统进程名清单中查找所述待识别进程的进程名,如果所述待识别进程的进程名存在于所述系统进程名清单中,则判断所述进程为系统进程,否则判断所述进程为非系统进程。

[0079] 所述系统进程为系统运行常用的非游戏进程,如操作系统创建的进程。

[0080] 系统进程名清单如下表所示:

[0081] 如:system、explorer、winlogon、csrss、smss。

[0082] (2)获取所述待识别进程的依赖库;

[0083] 通过Windows API枚举所述待识别进程加载的动态链接库,所述待识别进程加载的所有动态链接库形成所述待识别进程的依赖库。

[0084] (3)判断所述依赖库是否包含图像渲染动态链接库;

[0085] 判断所述依赖库是否包含directx图像渲染动态链接库和/或opengl图像渲染动态链接库。

[0086] 具体而言,判断所述依赖库是否包含directx图像渲染动态链接库,即判断所述依赖库是否包含文件名为d3d\*.d11的动态链接库文件,其中\*为通配符,匹配一个或多个数字字符,如d3d8.d11或d3d9.d11或d3d10.d11或d3d11.d11,其中d3d后面的数字就是directx对应的版本号。

[0087] 判断所述依赖库是否包含opengl图像渲染动态链接库,即判断所述依赖库是否包含文件名为opengl32.d11的动态链接库文件。

[0088] (4)当所述进程的依赖库中包含图像渲染动态链接库时,则识别所述待识别进程为游戏进程,否则识别所述待识别进程为非游戏进程。

[0089] 实施例2

[0090] 一种游戏进程识别系统,如图2所示,包括:

[0091] 系统进程判断模块:用于判断待识别进程是否为系统进程,如果所述待识别进程为系统进程,则将结果提交至游戏进程判断模块;否则将进入依赖库获取模块;

[0092] 所述系统进程为系统运行常用的非游戏进程,如操作系统创建的进程。

[0093] 所述进程判断模块包括系统进程名收集子模块,用于预先收集的系统进程名形成系统进程名清单,如果所述待识别进程的进程名存在于所述系统进程名清单中,则判断所述进程为系统进程,否则判断所述进程为非系统进程。

[0094] 所述依赖库获取模块,用于采用windows API获取所述待识别进程的依赖库,并将所述依赖库提交至图像渲染动态链接库判断模块;

[0095] 所述图像渲染动态链接库判断模块,用于判断所述待识别进程依赖库中是否包含有图像渲染动态链接库,并将判断结果提交至游戏进程判断模块;

[0096] 所述图像渲染动态链接库,为directx图像渲染动态链接库或opengl图像渲染动

态链接库。directx图像渲染动态链接库文件其文件名为d3d8.dll或d3d9.dll或d3d10.dll或d3d11.dll。

[0097] Opengl图像渲染动态链接库其文件名为opengl32.dll。

[0098] 所述游戏进程判断模块,用于判断所述待识别进程是否为游戏进程,当所述待识别进程为系统进程判断模块提交的被判断为系统进程的进程,则识别所述待识别进程为非游戏进程;当所述待识别进程为所述图像渲染动态链接库判断模块提交的被判断为其依赖库不包含图像渲染动态链接库的待识别进程,则识别所述待识别进程为非游戏进程;当所述识别待识别进程为所述图像渲染动态链接库判断模块提交的被判断为其依赖库包含图像渲染动态链接库的待识别进程,则识别所述待识别进程为游戏进程。

[0099] 实施例3

[0100] 一种游戏进程优先显示方法,如图3所示,包括以下步骤:

[0101] S1、获取系统的进程列表;

[0102] S2、对于步骤S1中获取的进程列表中的每一个进程,按照实施例1提供的方法识别所述进程是否为游戏进程;

[0103] S3、将步骤S2识别出的游戏进程的显示优先级按照如下方法调整:

[0104] 共设2个优先级,从高到低为优先级1、0;将步骤S2识别出的游戏进程的显示优先级设为优先级1,非游戏进程优先级设为优先级0。

[0105] S4、根据进程的显示优先级按照从高到低的顺序显示进程列表,对于进程的显示优先级相同的进程按照进程名的字母顺序显示。

[0106] 实施例4

[0107] 一种游戏进程优先显示系统,如图4所示,包括:

[0108] 进程列表获取模块,用于获取系统的进程列表;

[0109] 游戏进程识别模块,用于按照实施例1所述方法识别进程列表获取模块获取的进程列表中的每一个进程是否为游戏进程;

[0110] 进程显示优先级重排模块,用于将游戏进程识别模块识别出的游戏进程的显示优先级提高,得到调整后的显示优先级;具体地:

[0111] 采用2层优先级,游戏进程是优先级1,非游戏进程是优先级0,优先级1高于优先级0。

[0112] 进程显示模块,用于根据进程显示优先级重排模块得到的调整后的显示优先级按照从高到低的顺序显示进程列表。对于进程的显示优先级相同的进程按照进程名的字母顺序显示。

[0113] 本领域的技术人员容易理解,以上所述仅为本发明的较佳实施例而已,并不用以限制本发明,凡在本发明的精神和原则之内所作的任何修改、等同替换和改进等,均应包含在本发明的保护范围之内。

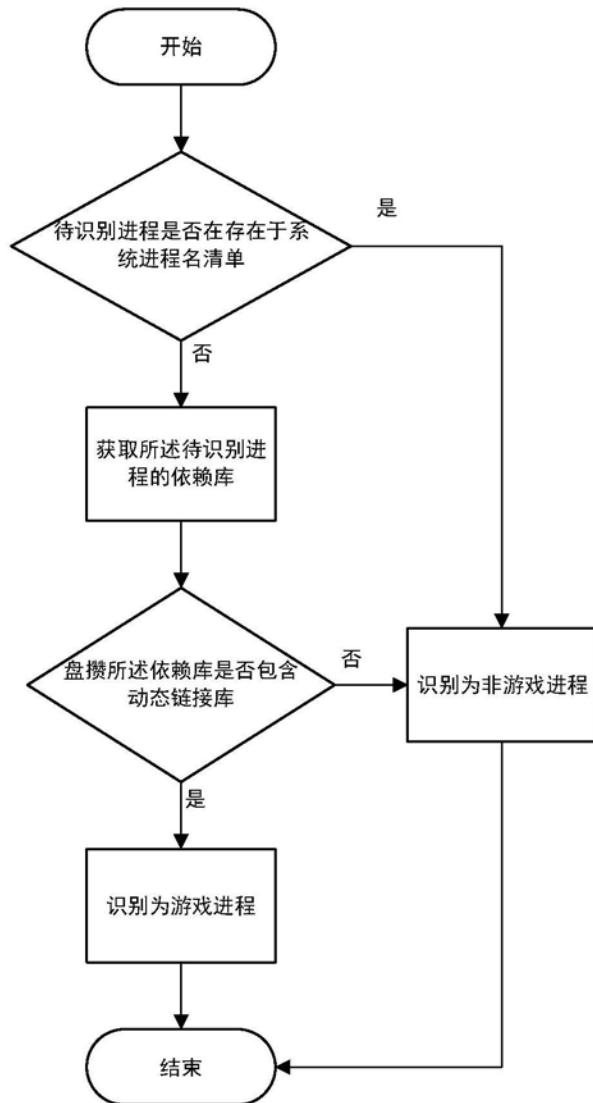


图1

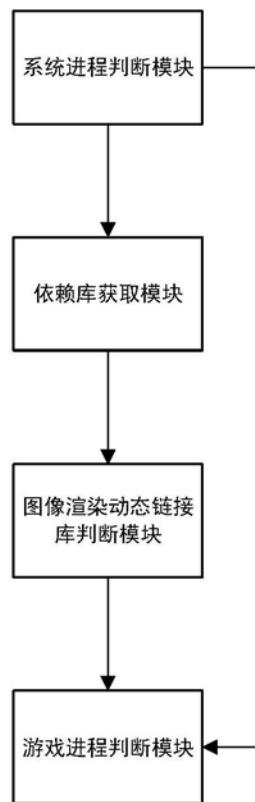


图2

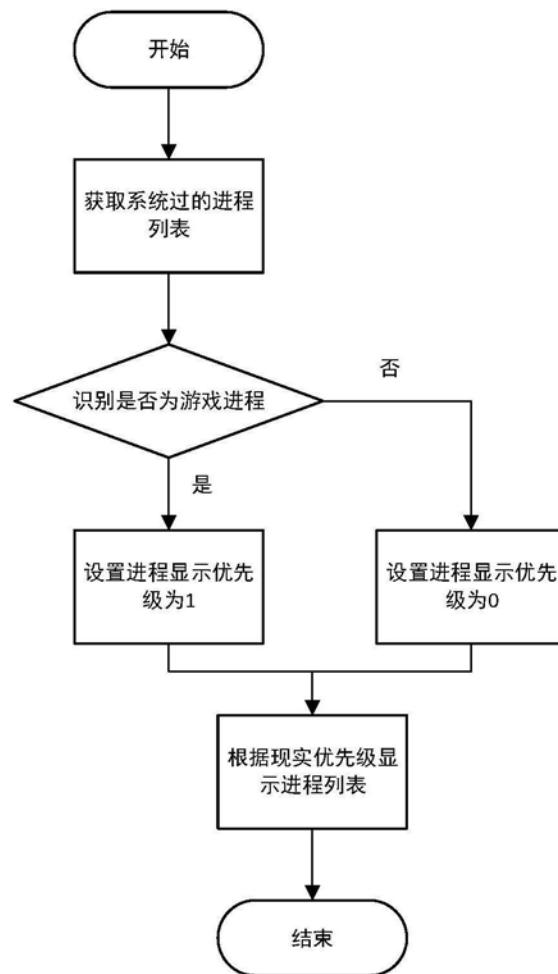


图3



图4