

资源配置方法、装置、设备和存储介质

本申请要求于 2020 年 01 月 09 日提交的申请号为 202010022010.0、发明名称为“资源配置方法、装置、设备和存储介质”的中国专利申请的优先权，其全部内容通过引用结合在本申请中。

技术领域

本申请实施例涉及计算机技术领域，特别涉及一种资源配置方法、装置、设备和存储介质。

背景技术

随着客户对游戏体验的要求逐渐提高，游戏客户端也需要经常更新。

在相关技术中，通过客户端本地表格配置一系列游戏包体内的图片路径，然后在配置表格中根据随机数据以及关键词索引，找到对应的图片路径，通过该图片路径加载该图片并在用户界面中进行显示。

在上述相关技术中，由于需要在客户端本地进行配置，因此在需要更换图片时，只能通过客户端的版本更新才能实现，导致更换图片的成本较高，耗时较长。

发明内容

本申请实施例提供了一种资源配置方法、装置、设备和存储介质，可用于节省更换配置资源的成本，缩短更换配置资源的耗时。所述技术方案如下：

根据本申请的一个方面，提供了一种资源配置方法，所述方法应用于终端，所述方法包括：

获取配置数据，所述配置数据包括用于对目标应用程序中的目标项目进行配置的 m 个候选配置资源，所述 m 为正整数；

根据所述 m 个候选配置资源分别对应的场景标识信息，从所述 m 个候选配置资源中选取与所述目标应用程序的当前运行场景相匹配的 n 个候选配置资源，其中，所述场景标识信息用于指示所述候选配置资源所适用的运行场景，所述 n

为小于或等于所述 m 的正整数；

根据所述 n 个候选配置资源分别对应的权重，从所述 n 个候选配置资源中选取目标配置资源，其中，所述权重用于指示所述候选配置资源的被选择概率；
采用所述目标配置资源对所述目标项目进行配置。

根据本申请的另一方面，提供了一种资源配置方法，所述方法应用于终端，所述方法包括：

获取配置数据，所述配置数据包括用于对游戏应用程序中的用户界面进行配置的 m 个候选图片资源，所述 m 为正整数；

从所述 m 个候选图片资源中选取与所述游戏应用程序的当前运行场景相适配的 n 个候选图片资源，所述 n 为小于或等于所述 m 的正整数；

从所述 n 个候选图片资源中选取目标图片资源；

在所述用户界面中显示所述目标图片资源。

根据本申请的另一方面，提供了一种资源配置装置，所述装置包括：

配置数据获取模块，用于获取配置数据，所述配置数据包括用于对目标应用程序中的目标项目进行配置的 m 个候选配置资源，所述 m 为正整数；

候选资源选取模块，用于根据所述 m 个候选配置资源分别对应的场景标识信息，从所述 m 个候选配置资源中选取与所述目标应用程序的当前运行场景相匹配的 n 个候选配置资源，其中，所述场景标识信息用于指示所述候选配置资源所适用的运行场景，所述 n 为小于或等于所述 m 的正整数；

目标资源选取模块，用于根据所述 n 个候选配置资源分别对应的权重，从所述 n 个候选配置资源中选取目标配置资源，其中，所述权重用于指示所述候选配置资源的被选择概率；

目标资源配置模块，用于采用所述目标配置资源对所述目标项目进行配置。

根据本申请的另一方面，提供了一种资源配置装置，所述装置包括：

配置数据获取模块，用于获取配置数据，所述配置数据包括用于对游戏应用程序中的用户界面进行配置的 m 个候选图片资源，所述 m 为正整数；

候选图片选取模块，用于从所述 m 个候选图片资源中选取与所述游戏应用程序的当前运行场景相适配的 n 个候选图片资源，所述 n 为小于或等于所述 m 的正整数；

目标图片选取模块，用于从所述 n 个候选图片资源中选取目标图片资源；

目标图片显示模块，用于在所述用户界面中显示所述目标图片资源。

根据本申请的另一方面，提供了一种计算机设备，所述计算机设备包括处理器和存储器，所述存储器中存储有至少一条指令、至少一段程序、代码集或指令集，所述至少一条指令、所述至少一段程序、所述代码集或指令集由所述处理器加载并执行以实现如上述方面所述的资源配置方法。

根据本申请的另一方面，提供了一种计算机可读存储介质，所述计算机可读存储介质中存储有至少一条指令、至少一段程序、代码集或指令集，所述至少一条指令、所述至少一段程序、所述代码集或指令集由处理器加载并执行以实现如上述方面所述的资源配置方法。

根据本申请的另一方面，提供了一种计算机可读存储介质，该计算机可读存储介质中存储有计算机程序，该计算机程序在被处理器执行时实现上述资源配置方法。

根据本申请的另一方面，提供了一种计算机程序产品，所述计算机程序产品包括计算机指令，所述计算机指令存储在计算机可读存储介质中。计算机设备的处理器从所述计算机可读存储介质读取所述计算机指令，所述处理器执行所述计算机指令，使得所述计算机设备执行如上述各方面所述的资源配置方法。

附图说明

- 图1是本申请一个示例性实施例提供的实施环境的示意图；
- 图2是本申请一个示例性实施例提供的资源配置方法的流程图；
- 图3是本申请另一个示例性实施例提供的资源配置方法的流程图；
- 图4是本申请一个示例性实施例提供的客户端获取配置数据的流程图；
- 图5是本申请一个示例性实施例提供的随机数区间的示意图；
- 图6是本申请另一个示例性实施例提供的资源配置方法的流程图；
- 图7是本申请一个示例性实施例提供的用户界面的示意图；
- 图8是本申请另一个示例性实施例提供的用户界面的示意图；
- 图9是本申请一个示例性实施例提供的资源配置装置的框图；
- 图10是本申请另一个示例性实施例提供的资源配置装置的框图；
- 图11是本申请另一个示例性实施例提供的资源配置装置的框图；
- 图12是本申请一个示例性实施例提供的终端的结构框图；

图 13 是本申请一个示例性实施例提供的服务器的结构示意图。

具体实施方式

为使本申请的目的、技术方案和优点更加清楚，下面将结合附图对本申请实施方式作进一步地详细描述。

图 1 示出了本申请一个示例性实施例提供的实施环境的示意图。该实施环境可以包括：终端 10 和服务器 20。

上述终端 10 可以是台式计算机、膝上型便携计算机、手机、平板电脑、电子书阅读器、MP3（Moving Picture Experts Group Audio Layer III，动态影像专家压缩标准音频层面 3）播放器、MP4（Moving Picture Experts Group Audio Layer IV，动态影像专家压缩标准音频层面 4）播放器等等。本申请实施例对此不作限定。

可选地，该终端 10 中安装运行有目标应用程序的客户端，该目标应用程序可以为游戏类应用程序，如其可以是第三人称射击（Third Person Shooting Game，TPS）游戏、第一人称射击（First Person Shooting Game，FPS）游戏、多人在线战术竞技（Multiplayer Online Battle Arena，MOBA）游戏、多人枪战类生存游戏等等。此外，该目标应用程序还可以是即时通信类应用程序、视频类应用程序、社交类应用程序、多媒体类应用程序、购物类应用程序、支付类应用程序、生活服务类应用程序，等等。本申请实施例对此不作限定。

可选地，上述终端 10 包括不止一个。

上述服务器 20 可以通过有线或者无线网络与终端 10 进行通信。

可选地，上述服务器 20 可以是独立的物理服务器，也可以是多个物理服务器构成的服务器集群或者分布式系统，还可以是提供云服务、云数据库、云计算、云函数、云存储、网络服务、云通信、中间件服务、域名服务、安全服务、CDN（Content Delivery Network，内容分发网络）、以及大数据和人工智能平台等基础云计算服务的云服务器，本申请实施例对此不作限定。

下面，通过几个实施例对本申请技术方案进行介绍说明。

图 2 示出了本申请一个示例性实施例提供的资源配置方法的流程图。在本

实施例 10 中，主要以该方法应用于图 1 所示的终端 10 中来举例说明，在终端 10 中安装有运行的目标应用程序的客户端中。该方法包括如下几个步骤：

步骤 201，获取配置数据，该配置数据包括用于对目标应用程序中的目标项目进行配置的 m 个候选配置资源，该 m 为正整数。

终端中安装运行有目标应用程序的客户端，关于该目标应用程序已在上文进行了介绍，此处不再赘述。

客户端可以通过网络与服务器进行通信，从而，客户端可以基于网络从服务器获取配置数据。

配置数据是指在实现目标应用程序的相关功能时的一些参数。配置数据中包括用于对目标应用程序中的至少一个项目进行配置的候选配置资源。对于目标应用程序，其可以包括多个需要配置的项目。以目标应用程序为游戏类应用程序为例，其需要配置的项目可以包括界面显示项目、界面文字项目、音效项目等，其中，界面显示项目是指该游戏类应用程序在运行时显示的用户界面；界面文字项目是指该游戏类应用程序在运行时显示的用户界面中包括的一些文字说明；音效项目是指该游戏类应用程序中的音效。

目标项目可以是多个项目中的任意一个项目。对于目标项目，其可以包括多个候选配置资源，该多个候选配置资源各不相同，且可以在不同的条件下配置给目标项目。以目标应用程序为游戏类应用程序，以目标项目为游戏类应用程序对应的界面显示项目为例，配置数据中的对该目标项目进行配置的 m 个候选配置资源可以是 m 张不同的图片，在不同的条件下配置给目标项目，实现该游戏类应用程序在不同的用户界面上显示不同的图片。示意性的，配置数据中的对该目标项目进行配置的 m 个候选资源可以是 m 段动画，在不同的条件下配置给目标项目，实现该游戏类应用程序在不同的用户界面上显示不同的动画。示意性的，配置数据中的对该目标项目进行配置的 m 个候选配置资源可以是 m 段不同的视频，在不同条件下配置给目标项目，实现游戏类应用程序在不同的用户界面上显示不同的视频。

步骤 202，根据 m 个候选配置资源分别对应的场景标识信息，从 m 个候选配置资源中选取与目标应用程序的当前运行场景相匹配的 n 个候选配置资源，该 n 为小于或等于 m 的正整数。

场景标识信息用于指示候选配置资源所适用的运行场景。该运行场景是指

客户端所处的运行情况。

在获取到上述 m 个候选配置资源之后，根据 m 个候选配置资源分别对应的场景标识信息，确定出与目标应用程序的当前运行场景相匹配的 n 个候选配置资源。其中，当前运行场景是指目标应用程序当前所处的运行场景；与当前场景相匹配的候选配置资源是指适用于当前运行场景的候选配置资源。

步骤 203，根据 n 个候选配置资源分别对应的权重，从 n 个候选配置资源中选取目标配置资源。

其中，权重用于指示候选配置资源的被选择概率。在确定出 n 个候选配置资源之后，可以根据该 n 个候选配置资源分别对应的权重，从该 n 个候选配置资源中选择目标配置资源。

步骤 204，采用目标配置资源对目标项目进行配置。

在确定出目标配置资源之后，可以采用该目标配置资源对目标项目进行配置。

综上所述，本申请实施例提供的技术方案，通过目标应用程序在不同场景下的场景标识信息与配置资源之间的对应关系，使得目标应用程序的客户端在获取到 m 个候选配置资源后，根据场景标识信息选取与当前运行场景相匹配的 n 个候选配置资源，利用 n 个候选配置资源分别对应的权重进一步确定目标配置资源。

图 3 示出了本申请另一个示例性实施例提供的资源配置方法的流程图。在本实施例中，主要以该方法应用于图 1 所示的终端 10 中来举例说明，在终端 10 中安装运行的目标应用程序的客户端。该方法包括如下几个步骤：

步骤 301，接收第一时间戳。

用户在运行目标应用程序的客户端时，可以登录用户在该目标应用程序中所使用的用户帐号，之后可以接收服务器发送的第一时间戳。

在一些实施例中，还可以在目标应用程序的客户端启动运行的第一时间接收服务器发送的第一时间戳。

第一时间戳用于指示配置数据库中存储的配置数据最后更新的时刻。配置数据库用于存储配置数据。目标应用程序的开发人员在对目标应用程序的项目开发的过程中，可以根据实际需求对配置数据进行更新。

在一些实施例中，在更新配置数据之后，配置数据库可以将该配置数据的最后更新的时刻，也即第一时间戳，发送给服务器。对应地，服务器可以接收该第一时间戳，以便之后将该第一时间戳发送给客户端。

步骤 302，比较第一时间戳和第二时间戳。

在获取到上述第一时间戳之后，可以将该第一时间戳与第二时间戳进行比对；其中，第二时间戳用于指示本地存储的配置数据最后更新的时刻，本地存储是指目标应用程序的客户端进行的存储。

在一些实施例中，客户端在每次获取到上述配置数据之后，可以将该配置数据存储在本地上，并对应记录该配置数据在配置数据库中最后更新的时刻。

在一些实施例中，客户端可以以 json 的格式存储在本地上，json（JavaScript Object Notation，JS 对象简谱）是一种轻量级的数据交换格式。在另一些实施例中，还可以采用其它格式进行存储，本申请实施例对此不作限定。

步骤 303，响应于第一时间戳晚于第二时间戳，发送配置获取请求。

当第一时间戳晚于第二时间戳时，也就是说，第一时间戳下的配置数据是最新的配置数据。

在这种情况下，客户端可以向服务器发送配置获取请求。配置获取请求用于请求获取配置数据库中存储的配置数据。

在一些实施例中，当第一时间戳早于或等于第二时间戳时，表示当前本地存储的配置数据与当前配置数据库中存储的配置数据相同，则可以不向服务器发送配置获取请求。

步骤 304，接收配置数据库中存储的配置数据。

服务器在获取到上述配置获取请求之后，可以从配置数据库中获取上述配置数据，并将获取到的配置数据发送给客户端，即客户端可以接收服务器发送的配置数据库中存储的配置数据。

在一些实施例中，客户端在接收到服务器发送的配置数据库中存储的配置数据之后，还可以更新上述第二时间戳。

通过在第一时间戳晚于第二时间戳的情况下获取配置数据，避免了每次登录客户端均拉取配置数据所造成的网络资源浪费，以及服务器资源的浪费。

示例性地，图 4 示例性示出了客户端获取配置数据的流程图。客户端获取配置数据的方法包括以下几个子步骤：步骤 41，配置数据库向服务器发送配置

数据，对应地，服务器接收上述配置数据；步骤 42，服务器向客户端发送第一时间戳，对应地，客户端接收服务器发送的第一时间戳；步骤 43，客户端比较第一时间戳和第二时间戳；步骤 44，响应于第一时间戳晚于第二时间戳，客户端向服务器发送配置获取请求，对应地，服务器接收客户端发送的配置获取请求；步骤 45，服务器向客户端发送配置数据，对应地，客户端接收配置数据。

步骤 305，获取目标应用程序的当前运行场景的场景标识信息。

在获取到上述配置数据之后，可以根据 m 个候选配置资源分别对应的场景标识信息，从 m 个候选配置资源中选取与目标应用程序的当前运行场景相匹配的 n 个候选配置资源。配置数据中包括用于对目标应用程序中的目标项目进行配置的 m 个候选配置资源。 m 为正整数， n 为小于或等于 m 的正整数。

客户端需要获取目标应用程序的当前运行场景的场景标识信息，该场景标识信息用于指示候选配置资源所适用的运行场景。

在一些实施例中，获取目标应用程序的当前运行场景的场景标识信息，可以包括以下几个步骤：

(1) 获取目标应用程序的当前运行场景对应的场景因子。

场景因子用于指示运行场景的属性。该场景因子包括以下至少一项：地区因子、模式因子、时间因子、功能因子、情感因子。上述地区因子用于指示目标应用程序的发行地区，如韩国、越南、台湾等；上述模式因子用于指示目标应用程序当前所处的模式，以目标应用程序为游戏类应用程序为例，该模式因子可以包括对抗模式、逃杀模式、格斗模式等；上述时间因子用于指示当前目标应用程序的运行时间；上述功能因子用于指示目标应用程序中不同的功能模块；上述情感因子用于指示目标应用程序中不同情感类型的内容。

(2) 根据场景因子的标识，确定当前运行场景的场景标识信息。

场景因子的标识用于标识场景因子。通过场景因子的标识确定当前运行场景的场景标识信息。

在一些实施例中，上述根据场景因子的标识，确定当前运行场景的场景标识信息，包括：将场景因子的标识进行拼接，得到当前运行场景的场景标识信息。例如，当前运行场景对应的场景因子包括地区因子和模式因子时，可以将该地区因子的标识和模式因子的标识进行拼接，将拼接得到的结果作为当前运行场景的场景标识信息。

在一些实施例中，上述模式因子还可以是模式组合因子。比如，将对抗模式对应的第一模式因子与逃杀模式对应的第二模式因子进行拼接，得到模式组合因子。

需要说明的是，若当前运行场景对应的场景因子包括一项，如地区因子，则直接将该地区因子的标识，确定为当前运行场景的场景标识信息。

考虑地区因子和模式因子进行资源配置，可以针对程序发行地域和模型进行差异性配置。

上述配置数据可以包括第一配置表和第二配置表，其中，第一配置表中包括候选配置资源、候选配置资源的资源标识、候选配置资源的权重、候选资源的生效时段（包括开始生效时间和结束生效时间）等；上述第二配置表包括场景标识信息、候选配置资源的资源标识等。

示例性地，以目标应用程序为游戏类应用程序，资源为图片资源，且场景因子包括地区因子和模式因子为例，表-1 示例性示出了一种第一配置表。

资源标识	图片资源	权重	开始时间	结束时间
1	https://www.pubgmobile.com/event/pics/loading/loading15.jpg	1	2019-10-1 00:00:00	2030-07-15 00:00:00
2	https://www.pubgmobile.com/event/pics/loading/loading32.jpg	100	2019-04-01 00:00:00	2030-04-03 23:59:59
3	https://www.pubgmobile.com/event/pics/loading/loading34.jpg	100	2019-04-04 00:00:00	2030-04-05 23:59:59
4	https://www.pubgmobile.com/event/pictures/events/blackandwhitewhoareyou.jpg	150	2019-14-18 00:00:00	2030-04-20 23:59:59

表-1

其中，第一列 ID 表示候选图片资源的资源标识，第二列为图片资源，第三列为候选图片资源的权重，第四列为候选资源的开始生效时间，第五列为候选资源的结束生效时间。

表-2 示例性示出了一种第二配置表。

场景标识信息	地区因子的标识	模式因子的标识	模式名称	资源标识
GLOBAL1	GLOBAL	1	全球绝地	1; 2

			海島	
DEFAULT		9999	默认模式配置	4

表-2

其中，第一列为场景标识信息，该场景标识信息是由地区因子的标识和模式因子的标识进行拼接得到的；第二列为地区因子的标识；第三列为模式因子的标识，该模式因子的标识可以采用模式组合因子的标识表示，以表-3 表示模式因子的标识与模式组合因子的标识之间的对应关系。

模式因子的标识	备注	模式组合因子的标识
101	逃杀模式单人配置	1
102	逃杀模式双人配置	1
103	逃杀模式四人配置	1

表-3

表-4 示例性示出了另一种第二配置表。

地区因子的标识	地区因子的代码	地区名称	资源标识
GLOBAL	840	全球	1; 2; 3; 4
KOREA	410	韩国	1
JAPAN	392	日本	2; 3
VNG	422	越南	1; 4
TW	156	台湾	3; 4
CE	344	CE	2; 4
DEFAULT	9999	默认区域配置	4

表-4

步骤 306，获取与当前运行场景的场景标识信息相对应的资源标识。

在确定当前运行场景的场景标识信息之后，获取与该场景标识相对应的资源标识。上述资源标识用于唯一标识配置资源。

示例性地，如表-2 所示，假设场景标识信息“GLOBAL1”是地区因子的标识“GLOBAL”和模式因子的标识“1”进行拼接得到的，该场景标识信息“GLOBAL1”对应的资源标识（索引 ID 序列）包括 1 和 2。

步骤 307, 从 m 个候选配置资源中选取资源标识所指示的候选配置资源, 得到 n 个候选配置资源。

在获取到与当前运行场景的场景标识信息相对应的资源标识之后, 从 m 个候选配置资源中选取资源标识所指示的候选配置资源, 从而得到 n 个候选配置资源。

示例性地, 如上述表-2 所示, 假设资源标识包括 1 和 2, 则表示该 m 个候选配置资源中, 将资源标识为 1 和 2 的两个候选配置资源作为上述 n 个候选配置资源。

在一些实施中, 若采用场景标识信息无法获取到资源标识, 则采用默认标识信息进行获取。

如上述表-2 和表-4 所示, 采用 DEFAULT 获取到资源标识为 4, 则将资源标识为 4 的候选配置资源作为上述 n 个候选配置资源。

在一些实施中, 在从 m 个候选配置资源中得到 n 个候选配置资源之后, 还可以执行下述步骤: 获取 n 个候选配置资源分别对应的生效时间; 根据 n 个候选配置资源分别对应的生效时间, 从 n 个候选配置资源中选取当前时刻有效的 k 个候选配置资源, 该 k 为小于或等于 n 的正整数。

由于候选配置资源存在生效时间, 因此, 根据上述 n 个候选配置资源分别对应的生效时间, 从该 n 个候选配置资源中选取当前时刻有效的 k 个候选配置资源, 以便从该 k 个候选配置资源中选取目标配置资源。

在一些实施例中, 上述获取 n 个候选配置资源分别对应的生效时间; 根据 n 个候选配置资源分别对应的生效时间, 从 n 个候选配置资源中选取当前时刻有效的 k 个候选配置资源的步骤还可以在上述步骤 305 之前执行, 本申请实施例对此不作限定。

通过采用生效时间进一步对候选配置资源进行筛选, 可以实现时效内的资源配置, 使得不处于生效时间 (如处于生效时间之前或生效时间之后) 的资源不进行配置。

步骤 308, 根据 n 个候选配置资源分别对应的权重, 生成 n 个候选配置资源分别对应的随机数区间。

在确定上述 n 个候选配置资源之后, 可以根据第一配置表获取该 n 个候选配置资源分别对应的权重。该权重用于指示候选配置资源的被选择概率。

之后，根据该 n 个候选配置资源分别对应的权重，可以生成 n 个候选配置资源分别对应的随机数区间。在一些实施例中，上述随机数区间的长度和权重呈正相关关系。

在一些实施例中，上述根据 n 个候选配置资源分别对应的权重，生成 n 个候选配置资源分别对应的随机数区间，包括以下几个步骤：

(1) 将 n 个候选配置资源分别对应的权重相加，得到权重和。

(2) 根据权重和确定随机数总区间。

(3) 按照 n 个候选配置资源分别对应的权重，对随机数总区间进行划分，得到 n 个候选配置资源分别对应的随机数区间。

示例性地，如图 5 所示，假设 n 等于 3，该 3 个候选配置资源的权重分别为 30、49 和 11，将该 3 个候选配置资源分别对应的权重相加，得到权重和，即 $30+49+11=90$ ；之后，确定随机数总区间为 $[1, 90]$ ；结合 3 个候选配置资源的权重，将随机数总区间 $[1, 90]$ 划分为 $[1, 30]$ 、 $[31, 79]$ 和 $[80, 90]$ 共 3 个随机数区间。

步骤 309，生成随机数。

上述随机数可以是上述随机数总区间的任意一个数。

步骤 310，响应于随机数处于 n 个随机数区间中的第 i 个随机数区间，将第 i 个随机数区间对应的第 i 个候选配置资源，确定为目标配置资源，该 i 为小于或等于 n 的正整数。

在生成一个随机数之后，确定该随机数处于 n 个随机数区间的第 i 个随机数区间，由于每个随机数区间对应一个候选配置资源，因此，在确定上述随机数处于第 i 个随机数区间之后，可以将该第 i 个随机数区间对应的第 i 个候选配置资源确定为目标配置资源。

示例性地，继续参考图 5，假设随机数为 50，该 50 落在第二个随机数区间 $[31, 79]$ 之间，则确定该第二个随机数区间 51 对应的候选配置资源为目标配置资源。

步骤 311，采用目标配置资源对目标项目进行配置。

此步骤与图 2 实施例中步骤 204 的内容相同或类似，此处不再赘述。

在一些实施例中，在确定上述目标配置资源之后，判断该目标配置资源是本地资源还是网络资源，若该目标配置资源是本地资源，则可以直接采用该目

标配置资源对目标项目进行配置；若该目标配置资源是网络资源且未下载至本地，则先从服务器下载该目标配置资源，将该目标配置资源存储在本地，然后采用该目标配置资源对目标项目进行配置。

示例性地，以上述目标配置资源是图片资源为例，若该图片资源是存储或缓存在本地的图片，则直接显示该图片；若该图片资源是以文本传输协议（http）格式的图片且未下载至本地，则从服务器（如图片服务器）下载该图片资源，然后进行加载显示。

另外，通过在第一时间戳晚于第二时间戳的情况下获取配置数据，避免了每次登录客户端均拉取配置数据所造成的网络资源浪费，以及服务器资源的浪费。

另外，通过采用生效时间进一步对候选配置资源进行筛选，可以实现时效内的资源配置，使得不处于生效时间（如处于生效时间之前或生效时间之后）的资源不进行配置。

另外，考虑地区因子和模式因子进行资源配置，可以针对程序发行地域和模型进行差异性配置。

图 6 示出了本申请另一个示例性实施例提供的资源配置方法的流程图。在本实施例中，主要以该方法应用于图 1 所示实施环境的终端 10 中来举例说明，在终端 10 中安装运行有目标应用程序（如游戏应用程序）的客户端。该方法包括如下几个步骤：

步骤 601，获取配置数据，配置数据包括用于对游戏应用程序中的用户界面进行配置的 m 个候选图片资源，该 m 为正整数。

终端中可以安装运行有目标应用程序的客户端，该目标应用程序可以是游戏应用程序。对于游戏应用程序，其用户界面的显示依赖于图片资源。

该游戏应用程序的客户端基于网络从服务器获取配置数据，该配置数据包括用于对游戏应用程序中的用户界面进行配置的多个候选图片资源。

在一些实施例中，上述配置数据中还包括用于对游戏应用程序中的其它项目进行配置的多个配置资源，本申请实施例对此不作限定。

示例性地，配置数据的代码格式如下：

cfg = (

```
{
  "SheetInfo":
    ["loading 图索引表.xls", "图片库", "LoadingPicsCfg", "ID"],
  "NameMap":
    [
      ["ID", "ID", "number"],
      ["图片资源", "path", "string"],
      ["权重", "weight", "number"],
      ["开始生效时间", "start_time", "string"],
      ["结束生效时间", "end_time", "string"],
    ]
},
{
  "SheetInfo":
    ["loading 图索引表.xls", "战斗 loading 索引", "LoadingBattleCfg", "拼接
key"],
  "NameMap":
    [
      ["拼接 key", "ID", "string"],
      ["索引 id 序列", "pictures", "string"],
    ]
},
{
  "SheetInfo":
    ["loading 图索引表.xls", "登录 loading 索引", "LoadingLoginCfg", "发布
地区"],
  "NameMap":
    [
      ["发布地区", "ID", "string"],
      ["索引 id 序列", "pictures", "string"],
    ]
}
```

```

        ]
    },
    {
        "SheetInfo":
            ["loading 图索引表.xls", "模式索引", "LoadingMode2GroupCfg", "主模式 id"],
        "NameMap":
            [
                ["主模式 id", "modeID", "number"],
                ["模式组合 id", "groupID", "number"],
            ]
    },
)

```

步骤 602, 从 m 个候选图片资源中选取与游戏应用程序的当前运行场景相适配的 n 个候选图片资源, 该 n 为小于或等于 m 的正整数。

在获取到上述 m 个候选图片资源之后, 从该 m 个候选图片资源中选取与游戏应用程序的当前运行场景相适配的 n 个候选图片资源。

关于从该 m 个候选图片资源中选取与游戏应用程序的当前运行场景相适配的 n 个候选图片资源的详细过程, 与参考图 3 实施例步骤 305-307 的内容相同或类似, 此处不再赘述。

步骤 603, 从 n 个候选图片资源中选取目标图片资源。

之后, 从上述 n 个候选图片资源中进一步确定出目标图片资源。

关于从 n 个候选图片资源中选取目标图片资源的详细过程, 与参考图 3 实施例中步骤 308-311 的内容相同或类似, 此处不再赘述。

步骤 604, 在用户界面中显示目标图片资源。

在确定上述目标图片资源之后, 在用户界面中显示该目标图片资源。

在一些实施例中, 若该图片资源是存储或缓存在本地的图片, 则直接显示该图片; 若该图片资源是 http 格式图片且未下载至本地, 则从服务器 (如图片服务器) 下载该图片资源, 然后进行加载显示。

示例性地, 如图 7 和图 8 所示, 在用户界面 71 中显示的图片资源均是基于

网络服务器得到的，在配置数据发生更新时，可以动态变更该用户界面中的显示。

综上所述，本申请实施例提供的技术方案，通过游戏应用程序在不同场景下的场景标识信息与图片资源之间的对应关系，使得游戏应用程序的客户端在获取到 m 个候选图片资源后，根据场景标识信息选取与当前运行场景相匹配的 n 个候选图片资源，利用 n 个候选图片资源分别对应的权重进一步确定目标图片资源。

下述为本申请装置实施例，可以用于执行本申请方法实施例。对于本申请装置实施例中未披露的细节，请参照本申请方法实施例。

图 9 示出了本申请一个示例性实施例提供的资源配置装置的框图。该装置具有实现上述资源配置方法示例的功能，所述功能可以由硬件实现，也可以由硬件执行相应的软件实现。该装置可以是上文介绍的终端，也可以设置在终端上。该装置 900 可以包括：配置数据获取模块 910、候选资源选取模块 920、目标资源选取模块 930 和目标资源配置模块 940。

配置数据获取模块 910，用于获取配置数据，配置数据包括用于对目标应用程序中的目标项目进行配置的 m 个候选配置资源， m 为正整数。

候选资源选取模块 920，用于根据 m 个候选配置资源分别对应的场景标识信息，从 m 个候选配置资源中选取与目标应用程序的当前运行场景相匹配的 n 个候选配置资源，其中，场景标识信息用于指示候选配置资源所适用的运行场景， n 为小于或等于 m 的正整数。

目标资源选取模块 930，用于根据 n 个候选配置资源分别对应的权重，从 n 个候选配置资源中选取目标配置资源，其中，权重用于指示候选配置资源的被选择概率。

目标资源配置模块 940，用于采用目标配置资源对目标项目进行配置。

综上所述，本申请实施例提供的技术方案，通过目标应用程序在不同场景下的场景标识信息与配置资源之间的对应关系，使得目标应用程序的客户端在获取到 m 个候选配置资源后，根据场景标识信息选取与当前运行场景相匹配的 n 个候选配置资源，利用 n 个候选配置资源分别对应的权重进一步确定目标配置资源。

在一些可能的设计中，如图 10 所示，所述候选资源选取模块 920，包括：标识信息获取单元 921、资源标识获取单元 922 和候选资源选取单元 923。

所述标识信息获取单元 921，用于获取目标应用程序的当前运行场景的场景标识信息。

所述资源标识获取单元 922，用于获取与当前运行场景的场景标识信息相对应的资源标识。

所述候选资源选取单元 923，用于从 m 个候选配置资源中选取资源标识所指示的候选配置资源，得到 n 个候选配置资源。

在一些可能的设计中，所述标识信息获取单元 921，用于获取目标应用程序的当前运行场景对应的场景因子，场景因子包括以下至少一项：地区因子、模式因子、时间因子、功能因子、情感因子；根据场景因子的标识，确定当前运行场景的场景标识信息。

在一些可能的设计中，如图 10 所示，目标资源选取模块 930，包括：随机区间生成单元 931、随机数生成单元 932 和目标资源确定单元 933。

所述随机区间生成单元 931，用于根据 n 个候选配置资源分别对应的权重，生成 n 个候选配置资源分别对应的随机数区间。

所述随机数生成单元 932，用于生成随机数。

所述目标资源确定单元 933，用于响应于随机数处于 n 个随机数区间中的第 i 个随机数区间，将第 i 个随机数区间对应的第 i 个候选配置资源，确定为目标配置资源， i 为小于或等于 n 的正整数。

在一些可能的设计中，所述随机区间生成单元 931，用于将 n 个候选配置资源分别对应的权重相加，得到权重和；根据权重和确定随机数总区间；按照 n 个候选配置资源分别对应的权重，对随机数总区间进行划分，得到 n 个候选配置资源分别对应的随机数区间。

在一些可能的设计中，所述配置数据获取模块 910，用于接收第一时间戳，第一时间戳用于指示配置数据库中存储的配置数据最后更新的时刻；比较第一时间戳和第二时间戳，第二时间戳用于指示本地存储的配置数据最后更新的时刻；响应于第一时间戳晚于第二时间戳，发送配置获取请求，配置获取请求用于请求获取配置数据库中存储的配置数据；接收配置数据库中存储的配置数据。

在一些可能的设计中，如图 10 所示，所述装置 900 还包括：

生效时间获取模块 950，用于获取 n 个候选配置资源分别对应的生效时间。

候选配置选取模块 960，用于根据 n 个候选配置资源分别对应的生效时间，从 n 个候选配置资源中选取当前时刻有效的 k 个候选配置资源， k 为小于或等于 n 的正整数；其中，目标配置资源从 k 个候选配置资源中选取。

图 11 示出了本申请另一个示例性实施例提供的资源配置装置的框图。该装置具有实现上述资源配置方法示例的功能，所述功能可以由硬件实现，也可以由硬件执行相应的软件实现。该装置可以是上文介绍的终端，也可以设置在终端上。该装置 1100 可以包括：配置数据获取模块 1110、候选图片选取模块 1120、目标图片选取模块 1130 和目标图片显示模块 1140。

所述配置数据获取模块 1110，用于获取配置数据，配置数据包括用于对游戏应用程序中的用户界面进行配置的 m 个候选图片资源， m 为正整数。

候选图片选取模块 1120，用于从 m 个候选图片资源中选取与游戏应用程序的当前运行场景相适配的 n 个候选图片资源， n 为小于或等于 m 的正整数。

目标图片选取模块 1130，用于从 n 个候选图片资源中选取目标图片资源。

目标图片显示模块 1140，用于在用户界面中显示所述目标图片资源。

综上所述，本申请实施例提供的技术方案，通过游戏应用程序在不同场景下的场景标识信息与配置资源之间的对应关系，使得游戏应用程序的客户端在获取到 m 个候选图片资源后，根据场景标识信息选取与当前运行场景相匹配的 n 个候选图片资源，利用 n 个候选图片资源分别对应的权重进一步确定目标图片资源。

需要说明的是，上述实施例提供的装置，在实现其功能时，仅以上述各功能模块的划分进行举例说明，实际应用中，可以根据需要而将上述功能分配由不同的功能模块完成，即将设备的内部结构划分成不同的功能模块，以完成以上描述的全部或者部分功能。另外，上述实施例提供的装置与方法实施例属于同一构思，其具体实现过程详见方法实施例，这里不再赘述。

图 12 示出了本申请一个示例性实施例提供的终端的结构框图。通常，终端 1200 包括有：处理器 1201 和存储器 1202。

处理器 1201 可以包括一个或多个处理核心，比如 4 核心处理器、8 核心处

理器等。处理器 1201 可以采用 DSP (Digital Signal Processing, 数字信号处理)、FPGA (Field Programmable Gate Array, 现场可编程门阵列)、PLA (Programmable Logic Array, 可编程逻辑阵列) 中的至少一种硬件形式来实现。处理器 1201 也可以包括主处理器和协处理器, 主处理器是用于对在唤醒状态下的数据进行处理的处理单元, 也称 CPU (Central Processing Unit, 中央处理器); 协处理器是用于对在待机状态下的数据进行处理的低功耗处理器。在一些实施例中, 处理器 1201 可以在集成有 GPU (Graphics Processing Unit, 图像处理单元), GPU 用于负责显示屏所需要显示的内容的渲染和绘制。一些实施例中, 处理器 1201 还可以包括 AI (Artificial Intelligence, 人工智能) 处理器, 该 AI 处理器用于处理有关机器学习的计算操作。

存储器 1202 可以包括一个或多个计算机可读存储介质, 该计算机可读存储介质可以是非暂态的。存储器 1202 还可包括高速随机存取存储器, 以及非易失性存储器, 比如一个或多个磁盘存储设备、闪存存储设备。在一些实施例中, 存储器 1202 中的非暂态的计算机可读存储介质用于存储至少一条指令、至少一段程序、代码集或指令集, 该至少一条指令、至少一段程序、代码集或指令集用于被处理器 1201 所执行以实现本申请中方法实施例提供的资源配置方法。

在一些实施例中, 终端 1200 还可选包括有: 外围设备接口 1203 和至少一个外围设备。处理器 1201、存储器 1202 和外围设备接口 1203 之间可以通过总线或信号线相连。各个外围设备可以通过总线、信号线或电路板与外围设备接口 1203 相连。具体地, 外围设备可以包括: 通信接口 1204、显示屏 1205、音频电路 1206、摄像头组件 1207、定位组件 1208 和电源 1209 中的至少一种。

本领域技术人员可以理解, 图 12 中示出的结构并不构成对终端 1200 的限制, 可以包括比图示更多或更少的组件, 或者组合某些组件, 或者采用不同的组件布置。

图 13 示出了本申请一个示例性实施例提供的服务器的结构示意图。具体来讲:

所述服务器 1300 包括 CPU (Central Processing Unit, 中央处理器) 1301、包括 RAM (Random Access Memory, 随机存取存储器) 1302 和 ROM (Read Only Memory, 只读存储器) 1303 的系统存储器 1304, 以及连接系统存储器 1304 和

中央处理单元 1301 的系统总线 1305。所述服务器 1300 还包括帮助计算机内的各个器件之间传输信息的基本 I/O (Input/Output 输入/输出) 系统 1306, 和用于存储操作系统 1313、应用程序 1314 和其他程序模块 1315 的大容量存储设备 1307。

所述基本输入/输出系统 1306 包括有用于显示信息的显示器 1308 和用于用户输入信息的诸如鼠标、键盘之类的输入设备 1309。其中所述显示器 1308 和输入设备 1309 都通过连接到系统总线 1305 的输入输出控制器 1310 连接到中央处理单元 1301。所述基本输入/输出系统 1306 还可以包括输入输出控制器 1310 以用于接收和处理来自键盘、鼠标、或电子触控笔等多个其他设备的输入。类似地, 输入输出控制器 1310 还提供输出到显示屏、打印机或其他类型的输出设备。

所述大容量存储设备 1307 通过连接到系统总线 1305 的大容量存储控制器 (未示出) 连接到中央处理单元 1301。所述大容量存储设备 1307 及其相关联的计算机可读介质为服务器 1300 提供非易失性存储。也就是说, 所述大容量存储设备 1307 可以包括诸如硬盘或者 CD-ROM (Compact Disc Read-Only Memory, 只读光盘) 驱动器之类的计算机可读介质 (未示出)。

不失一般性, 所述计算机可读介质可以包括计算机存储介质和通信介质。计算机存储介质包括以用于存储诸如计算机可读指令、数据结构、程序模块或其他数据等信息的任何方法或技术实现的易失性和非易失性、可移动和不可移动介质。计算机存储介质包括 RAM、ROM、EPROM (Erasable Programmable Read Only Memory, 可擦除可编程只读存储器)、闪存或其他固态存储其技术, CD-ROM、DVD (Digital Video Disc, 高密度数字视频光盘) 或其他光学存储、磁带盒、磁带、磁盘存储或其他磁性存储设备。当然, 本领域技术人员可知所述计算机存储介质不局限于上述几种。上述的系统存储器 1304 和大容量存储设备 1307 可以统称为存储器。

根据本申请的各种实施例, 所述服务器 1300 还可以通过诸如因特网等网络连接到网络上的远程计算机运行。也即服务器 1300 可以通过连接在所述系统总线 1305 上的网络接口单元 1311 连接到网络 1312, 或者说, 也可以使用网络接口单元 1311 来连接到其他类型的网络或远程计算机系统 (未示出)。

所述存储器还包括至少一条指令、至少一段程序、代码集或指令集, 所述至少一条指令、至少一段程序、代码集或指令集存储于存储器中, 且经配置以

由一个或者一个以上处理器执行，以实现上述资源配置方法。

在示例性实施例中，还提供了一种计算机设备。该计算机设备可以是终端或服务器。该计算机设备包括处理器和存储器，该存储器中存储有至少一条指令、至少一段程序、代码集或指令集，该至少一条指令、该至少一段程序、该代码集或指令集由处理器加载并执行以实现上述资源配置方法。

在示例性实施例中，还提供了一种计算机可读存储介质，该计算机可读存储介质中存储有至少一条指令、至少一段程序、代码集或指令集，该至少一条指令、该至少一段程序、该代码集或该指令集在被处理器执行时实现上述资源配置方法。

在示例性实施例中，还提供了一种计算机可读存储介质，该计算机可读存储介质中存储有计算机程序，该计算机程序在被处理器执行时实现上述资源配置方法。

在示例性实施例中，还提供了一种计算机程序产品，该计算机程序产品包括计算机指令，该计算机指令存储在计算机可读存储介质中。计算机设备的处理器从该计算机可读存储介质读取该计算机指令，该处理器执行该计算机指令，使得该计算机设备执行如上述各方面所述的资源配置方法。

应当理解的是，在本文中提及的“多个”是指两个或两个以上。“和/或”，描述关联对象的关联关系，表示可以存在三种关系，例如，A 和/或 B，可以表示：单独存在 A，同时存在 A 和 B，单独存在 B 这三种情况。字符“/”一般表示前后关联对象是一种“或”的关系。

以上所述仅为本申请的示例性实施例，并不用以限制本申请，凡在本申请的精神和原则之内，所作的任何修改、等同替换、改进等，均应包含在本申请的保护范围之内。

权 利 要 求 书

1、一种资源配置方法，其特征在于，所述方法应用于终端，所述方法包括：
获取配置数据，所述配置数据包括用于对目标应用程序中的目标项目进行配置的 m 个候选配置资源，所述 m 为正整数；

根据所述 m 个候选配置资源分别对应的场景标识信息，从所述 m 个候选配置资源中选取与所述目标应用程序的当前运行场景相匹配的 n 个候选配置资源，其中，所述场景标识信息用于指示所述候选配置资源所适用的运行场景，所述 n 为小于或等于所述 m 的正整数；

根据所述 n 个候选配置资源分别对应的权重，从所述 n 个候选配置资源中选取目标配置资源，其中，所述权重用于指示所述候选配置资源的被选择概率；
采用所述目标配置资源对所述目标项目进行配置。

2、根据权利要求 1 所述的方法，其特征在于，所述根据所述 m 个候选配置资源分别对应的场景标识信息，从所述 m 个候选配置资源中选取与所述目标应用程序的当前运行场景相匹配的 n 个候选配置资源，包括：

获取所述目标应用程序的当前运行场景的场景标识信息；

获取与所述当前运行场景的场景标识信息相对应的资源标识；

从所述 m 个候选配置资源中选取所述资源标识所指示的候选配置资源，得到所述 n 个候选配置资源。

3、根据权利要求 2 所述的方法，其特征在于，所述获取所述目标应用程序的当前运行场景的场景标识信息，包括：

获取所述目标应用程序的当前运行场景对应的场景因子，所述场景因子包括以下至少一项：地区因子、模式因子、时间因子、功能因子、情感因子；

根据所述场景因子的标识，确定所述当前运行场景的场景标识信息。

4、根据权利要求 1 所述的方法，其特征在于，所述根据所述 n 个候选配置资源分别对应的权重，从所述 n 个候选配置资源中选取目标配置资源，包括：

根据所述 n 个候选配置资源分别对应的权重，生成所述 n 个候选配置资源

分别对应的随机数区间；

生成随机数；

响应于所述随机数处于所述 n 个随机数区间中的第 i 个随机数区间，将所述第 i 个随机数区间对应的第 i 个候选配置资源，确定为所述目标配置资源，所述 i 为小于或等于所述 n 的正整数。

5、根据权利要求 4 所述的方法，其特征在于，所述根据所述 n 个候选配置资源分别对应的权重，生成所述 n 个候选配置资源分别对应的随机数区间，包括：

将所述 n 个候选配置资源分别对应的权重相加，得到权重和；

根据所述权重和确定随机数总区间；

按照所述 n 个候选配置资源分别对应的权重，对所述随机数总区间进行划分，得到所述 n 个候选配置资源分别对应的随机数区间。

6、根据权利要求 1 所述的方法，其特征在于，所述获取配置数据，包括：

接收第一时间戳，所述第一时间戳用于指示配置数据库中存储的配置数据最后更新的时刻；

比较所述第一时间戳和第二时间戳，所述第二时间戳用于指示本地存储的配置数据最后更新的时刻；

响应于所述第一时间戳晚于所述第二时间戳，发送配置获取请求，所述配置获取请求用于请求获取所述配置数据库中存储的配置数据；

接收所述配置数据库中存储的配置数据。

7、根据权利要求 1 至 6 任一项所述的方法，其特征在于，所述根据所述 m 个候选配置资源分别对应的场景标识信息，从所述 m 个候选配置资源中选取与所述目标应用程序的当前运行场景相匹配的 n 个候选配置资源之后，还包括：

获取所述 n 个候选配置资源分别对应的生效时间；

根据所述 n 个候选配置资源分别对应的生效时间，从所述 n 个候选配置资源中选取当前时刻有效的 k 个候选配置资源，所述 k 为小于或等于所述 n 的正整数；

其中，所述目标配置资源从所述 k 个候选配置资源中选取。

8、一种资源配置方法，其特征在于，所述方法应用于终端，所述方法包括：
获取配置数据，所述配置数据包括用于对游戏应用程序中的用户界面进行配置的 m 个候选图片资源，所述 m 为正整数；

从所述 m 个候选图片资源中选取与所述游戏应用程序的当前运行场景相匹配的 n 个候选图片资源，所述 n 为小于或等于所述 m 的正整数；

从所述 n 个候选图片资源中选取目标图片资源；

在所述用户界面中显示所述目标图片资源。

9、一种资源配置装置，其特征在于，所述装置包括：

配置数据获取模块，用于获取配置数据，所述配置数据包括用于对目标应用程序中的目标项目进行配置的 m 个候选配置资源，所述 m 为正整数；

候选资源选取模块，用于根据所述 m 个候选配置资源分别对应的场景标识信息，从所述 m 个候选配置资源中选取与所述目标应用程序的当前运行场景相匹配的 n 个候选配置资源，其中，所述场景标识信息用于指示所述候选配置资源所适用的运行场景，所述 n 为小于或等于所述 m 的正整数；

目标资源选取模块，用于根据所述 n 个候选配置资源分别对应的权重，从所述 n 个候选配置资源中选取目标配置资源，其中，所述权重用于指示所述候选配置资源的被选择概率；

目标资源配置模块，用于采用所述目标配置资源对所述目标项目进行配置。

10、根据权利要求 9 所述的装置，其特征在于，所述候选资源选取模块包括：标识信息获取单元、资源标识获取单元、候选资源选取单元；

所述标识信息获取单元，用于获取所述目标应用程序的当前运行场景的场景标识信息；

所述资源标识获取单元，用于获取与所述当前运行场景的场景标识信息相对应的资源标识；

所述候选资源选取单元，用于从所述 m 个候选配置资源中选取所述资源标识所指示的候选配置资源，得到所述 n 个候选配置资源。

11、根据权利要求 10 所述的装置，其特征在于，所述标识信息获取单元，用于获取所述目标应用程序的当前运行场景对应的场景因子，所述场景因子包括以下至少一项：地区因子、模式因子、时间因子、功能因子、情感因子；根据所述场景因子的标识，确定所述当前运行场景的场景标识信息。

12、根据权利要求 9 所述的装置，其特征在于，所述目标资源选取模块包括：随机区间生成单元、随机数生成单元和目标资源确定单元；

所述随机区间生成单元，用于根据所述 n 个候选配置资源分别对应的权重，生成所述 n 个候选配置资源分别对应的随机数区间；

所述随机数生成单元，用于生成随机数；

所述目标资源确定单元，用于响应于所述随机数处于所述 n 个随机数区间中的第 i 个随机数区间，将所述第 i 个随机数区间对应的第 i 个候选配置资源，确定为所述目标配置资源，所述 i 为小于或等于所述 n 的正整数。

13、根据权利要求 12 所述的装置，其特征在于，所述随机区间生成单元，用于将所述 n 个候选配置资源分别对应的权重相加，得到权重和；根据所述权重和确定随机数总区间；按照所述 n 个候选配置资源分别对应的权重，对所述随机数总区间进行划分，得到所述 n 个候选配置资源分别对应的随机数区间。

14、根据权利要求 9 所述的装置，其特征在于，所述配置数据获取模块，用于接收第一时间戳，所述第一时间戳用于指示配置数据库中存储的配置数据最后更新的时刻；比较所述第一时间戳和第二时间戳，所述第二时间戳用于指示本地存储的配置数据最后更新的时刻；响应于所述第一时间戳晚于所述第二时间戳，发送配置获取请求，所述配置获取请求用于请求获取所述配置数据库中存储的配置数据；接收所述配置数据库中存储的配置数据。

15、根据权利要求 9 至 14 任一所述的装置，其特征在于，所述装置还包括：生效时间获取模块，用于获取所述 n 个候选配置资源分别对应的生效时间。候选配置选取模块，用于根据所述 n 个候选配置资源分别对应的生效时间，

从所述 n 个候选配置资源中选取当前时刻有效的 k 个候选配置资源，所述 k 为小于或等于所述 n 的正整数；其中，所述目标配置资源从所述 k 个候选配置资源中选取。

16、一种资源配置装置，其特征在于，所述装置包括：

配置数据获取模块，用于获取配置数据，所述配置数据包括用于对游戏应用程序中的用户界面进行配置的 m 个候选图片资源，所述 m 为正整数；

候选图片选取模块，用于从所述 m 个候选图片资源中选取与所述游戏应用程序的当前运行场景相适配的 n 个候选图片资源，所述 n 为小于或等于所述 m 的正整数；

目标图片选取模块，用于从所述 n 个候选图片资源中选取目标图片资源；

目标图片显示模块，用于在所述用户界面中显示所述目标图片资源。

17、一种计算机设备，其特征在于，所述计算机设备包括处理器和存储器，所述存储器中存储有至少一条指令、至少一段程序、代码集或指令集，所述至少一条指令、所述至少一段程序、所述代码集或指令集由所述处理器加载并执行以实现如权利要求 1 至 7 任一项所述的资源配置方法，或者实现如权利要求 8 所述的资源配置方法。

18、一种计算机可读存储介质，其特征在于，所述计算机可读存储介质中存储有计算机程序，所述计算机程序由处理器加载并执行以实现如权利要求 1 至 7 任一项所述的资源配置方法，或者实现如权利要求 8 所述的资源配置方法。