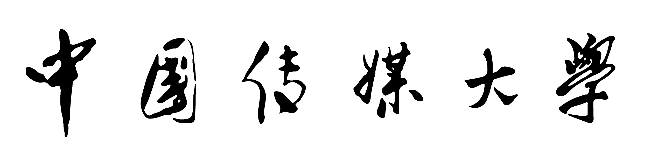
分类号： TP311.5 单位代码： 10033

密 级： 学号：



硕士学位论文



**中文论文题目： 游戏测试可视化辅助系统的**

**设计与实现**

**英文论文题目：**

申请人姓名： 温伟航

指导教师： 韩红雷

专业名称： 计算机技术

研究方向： 游戏制作

所在学院： 动画与数字艺术学院

**论文提交日期**

中国传媒大学研究生学位论文独创性声明

本人声明所呈交的学位论文是本人在导师指导下进行的研究工作及取得的研究成果。除了文中特别加以标注和致谢的地方外，论文中不包含其他人已经发表或撰写过的研究成果，也不包含为获得 **中国传媒大学** 或其他教育机构的学位或证书而使用过的材料。与我一同工作的同志对本研究所做的任何贡献均已在论文中作了明确的说明并表示谢意。

学位论文作者签名： 签字日期： 2022年 00月00日

学位论文版权使用授权书

本学位论文作者完全了解 **中国传媒大学** 有权保留并向国家有关部门或机构送交本论文的复印件和磁盘，允许论文被查阅和借阅。本人授权**中国传媒大学**可以将学位论文的全部或部分内容编入有关数据库进行检索和传播，可以采用影印、缩印或扫描等复制手段保存、汇编学位论文。

（保密的学位论文在解密后适用本授权书）

学位论文作者签名： 导师签名：

签字日期： 2022年 00月 00日 签字日期： 2022年 00月00日

# 致 谢

游戏测试可视化辅助系统的设计与实现

# 摘 要

随着游戏产业的飞速发展，高质量的游戏会为玩家带来优质的游戏体验。而游戏测试是游戏开发过程中的重要组成部分，旨在通过收集游戏数据和发掘游戏缺陷，帮助开发者让游戏更贴近自己的设计意图，从而提高游戏质量。

本文提出一种基于数据可视化的游戏测试辅助系统，可以有效的辅助开发者测试游戏中的设计内容。系统首先对游戏体验时玩家的生理和游戏数据进行采集。然后，系统对收集到的数据进行特征提取，同时以特定的规则对不同类型的数据进行集成和绑定，以便能够充分利用每种数据集的优点。最后，系统利用数据聚类和可视化技术将游戏测试的数据结果呈现。其中，在系统的数据处理模块中，本文提出了一种对心电数据中R波的提取方法；在数据挖掘和可视化模块中，提出一种MRGTree的聚类方法，可以根据用户所需呈现的细节等级，对场景中的数据进行数据聚类，以便将数据内所表现的玩家运动趋势和行为传达给开发者，从而使开发者对游戏的设计内容进行评估。

本文对基于数据可视化的游戏测试辅助系统的需求分析、总体设计、详细设计与实现细节进行了详细的介绍，并使用 UML对系统的架构和设计进行阐述，核心算法使用伪代码进行说明。最后，该系统以C#进行开发实现，并利用Unity引擎开发的2D平台动作游戏作为实验材料，对系统进行了功能性和有效性进行了测试，确定了系统功能完整性，且有效性能达到要求，满足开发团队和设计团队的需求。

关键字：游戏测试；生理数据采集；游戏数据分析；聚类算法；数据可视化

**XXXXXXXXXXX**

# ABSTRACT

# 目 录

[致 谢 I](#_Toc98158943)

[摘 要 II](#_Toc98158944)

[ABSTRACT III](#_Toc98158945)

[目 录 IV](#_Toc98158946)

[1. 绪论 1](#_Toc98158947)

[1.1 论文背景及研究意义 1](#_Toc98158948)

[1.2 国内外研究现状 3](#_Toc98158949)

[1.2.1 游戏测试相关工作研究 3](#_Toc98158950)

[1.2.2 游戏用户体验与生理信号数据研究 4](#_Toc98158951)

[1.2.3 游戏数据挖掘及可视化工作研究 6](#_Toc98158952)

[1.3 本文的研究内容分析 6](#_Toc98158953)

[1.4 论文结构安排 8](#_Toc98158954)

[2. 游戏测评辅助系统的需求分析与总体设计 9](#_Toc98158955)

[2.1 游戏测评辅助系统业务分析 9](#_Toc98158956)

[2.2 游戏测评辅助系统需求分析 10](#_Toc98158957)

[2.2.1 游戏内数据采集需求分析 10](#_Toc98158958)

[2.2.2 玩家生理数据采集需求分析 11](#_Toc98158959)

[2.2.3 数据处理需求分析 11](#_Toc98158960)

[2.2.4 数据可视化需求分析 11](#_Toc98158961)

[2.3 系统总体设计 12](#_Toc98158962)

[2.3.1 游戏数据采集模块 14](#_Toc98158963)

[2.3.2 生理数据采集模块 14](#_Toc98158964)

[2.3.3 玩家信息及问卷采集模块 14](#_Toc98158965)

[2.3.4 数据处理模块 15](#_Toc98158966)

[2.3.5 数据挖掘和可视化模块 15](#_Toc98158967)

[2.4 开发运行环境 15](#_Toc98158968)

[2.5 本章小结 16](#_Toc98158969)

[3. 数据采集模块的设计与实现 17](#_Toc98158970)

[3.1 生理数据采集模块设计架构 17](#_Toc98158971)

[3.1.1 心电数据信号采集原理 17](#_Toc98158972)

[3.1.2 面部数据信号采集原理 18](#_Toc98158973)

[3.1.3 心电数据采集详细设计与实现 18](#_Toc98158974)

[3.1.4 面部数据采集详细设计与实现 19](#_Toc98158975)

[3.1.5 生理采集工具整合及实现 20](#_Toc98158976)

[3.2 游戏数据采集模块设计架构 22](#_Toc98158977)

[3.3 玩家信息及问卷采集模块设计架构 24](#_Toc98158978)

[3.4 本章小结 25](#_Toc98158979)

[4. 数据处理模块的详细设计与实现 27](#_Toc98158980)

[4.1 心电信号中R波的识别方法 27](#_Toc98158981)

[4.1.1 小波变换法 27](#_Toc98158982)

[4.1.2 差分阈值法 28](#_Toc98158983)

[4.2 本文采用的特征R波识别方法 29](#_Toc98158984)

[4.2.1 可行性验证 30](#_Toc98158985)

[4.3 玩家面部摘要提取 31](#_Toc98158986)

[4.4 本章小结 32](#_Toc98158987)

[5. 数据挖掘及可视化 33](#_Toc98158988)

[5.1 数据挖掘 33](#_Toc98158989)

[5.1.1 划分法 34](#_Toc98158990)

[5.1.2 层次法 34](#_Toc98158991)

[5.1.3 密度法 35](#_Toc98158992)

[5.1.4 网格法 35](#_Toc98158993)

[5.2 本文所用的方法 35](#_Toc98158994)

[5.2.1 MRGTree算法流程 36](#_Toc98158995)

[5.2.2 构建层次结构 36](#_Toc98158996)

[5.2.3 玩家数据填充 37](#_Toc98158997)

[5.2.4 聚类结点数据 39](#_Toc98158998)

[5.2.5 调用层次结构方法 40](#_Toc98158999)

[5.3 可视化 40](#_Toc98159000)

[5.3.1 基于场景区域的特征数据静态可视化功能 40](#_Toc98159001)

[5.3.2 多路径动态可视化功能设计及实现 43](#_Toc98159002)

[5.4 本章小结 44](#_Toc98159003)

[6. 系统的实验和测试 46](#_Toc98159004)

[6.1 实验材料 46](#_Toc98159005)

[6.2 实验设计与安排 48](#_Toc98159006)

[6.3 实验结果分析 48](#_Toc98159007)

[6.4 本章小结 49](#_Toc98159008)

[7. 结论 50](#_Toc98159009)

[参 考 文 献 52](#_Toc98159010)

#### 

## 绪论

### 论文背景及研究意义

游戏产业作为第九艺术，仅在国内2021年前半年，游戏市场销售达到1504.93亿[1]。以多样化的电子设备为载体，游戏逐渐突破了在时间和空间上束缚，使得人们无论在何时何地均可以享受到游戏带来的乐趣。在游戏行业和游戏市场的规模日渐庞大的当下，游戏也渗透到诸多领域进行融合，利用游戏中内在娱乐性和趣味性的特点，与教育、军事和医疗保健等不同行业结合而生的“严肃游戏”[2]。游戏无论是娱乐方向还是严肃方向，游戏目的不同并不会改变游戏可玩性的本质。在市场竞争日益激烈的环境下，玩家对游戏产品的品质要求越来越高，同时对游戏体验也越来越重视。

为了在游戏发行之前要满足相应的品质要求，游戏测试在整个游戏开发的周期中至关重要。通过游戏测试可以发现游戏中存在或潜在的缺陷，通过发现这些缺陷到修复缺陷来提升游戏品质。虽然游戏本质上也是软件，软件测试和游戏测试在测试工程的角度上看是相同的。但是，游戏的开发过程中的独特的之处会使得游戏测试应该更具有针对性。首先游戏开发活动中典型的团队架构包括：程序开发团队、美术团队、音视频团队、测试团队和游戏设计(策划)团队。其中各个团队的规模也不尽相同，美术团队规模较大，而音视频团队的规模较小。此架构对于典型的软件开发团队而言不仅规模庞大，而且游戏团队所涉及的领域也更全面。这意味着缺陷会从不同团队引入到游戏中。其次，游戏开发中任务也逐渐呈现团队协作和跨领域融合的态势，例如：技术美术，关卡策划等。那么功能性上的缺陷，产生原因变的多元化且难以被发现。最后则是游戏相比软件而言，对其品质的界定上包括一个重要的方面，即游戏的可玩性，换句话说，也是玩家的游戏体验情况。通过测试对游戏品质进行衡量，并能发现品质上潜在的缺陷，定位并修复缺陷则在游戏测试工作中尤为重要了。

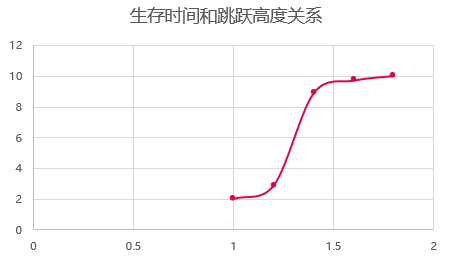
游戏的可玩性会直接影响到玩家的游戏体验，但是，游戏可玩性不是简单的对游戏中优质系统进行拼接，也不是对其好评关卡进行复刻，而是非常复杂的一个综合体[3][4]。因为游戏本身是由策划、程序、美术等不同领域的开发人员协作而成。在游戏的设计之初，开发人员会以自身经验来对游戏进行策划和设计，所以在游戏开发进行中对游戏可玩性的测 试和衡量就很有必要。在游戏测试的生命周期中，始终贯彻的任务目的是要帮助游戏开发者让游戏更贴近自己的设计意图，另一方面是要帮助游戏开发者发现游戏设计中的不足。

当游戏开发流程进行到beta测评阶段，其中的一项阶段性目标就是要锁定游戏设计[5]。在通常情况下，游戏测试人员组织外部人员进行游戏测试。针对游戏设计的相关内容，在测试结束后会收集测试参与者的两部分数据：一部分是偏向于玩家的主观数据，对游戏可玩性以及体验感受等相应的内容进行事后问卷调查。另一部分是玩家的客观数据，例如玩家在游戏中触发事件的操作，虚拟货币交易数据等。然后将大量的体验数据进行统计分析，根据分析后的结果，评估游戏的可玩性，进一步指导游戏开发人员进行相应的修改和迭代。

我们发现在当下的游戏测评流程中，在对游戏设计内容进行的测试活动中，存在着如下几个可待提升之处。

**问题1**. 用户调研问卷作为量化体验程度的调研方法，可行性毋庸置疑[6]。虽然精心设计的问卷可以反映一定程度的客观数据，但是仍然存在主观成分。且仅利用单一的问卷结果来佐证游戏可玩性及其玩家情感体验程度，可信度是有待提高的。而且，在问卷中的有限离散题目是很难反应出玩家在游戏体验中某些情感的连续变化。

**问题2**. 为了让不同职能的开发人员都能够直观地理解数据的统计情况，可视化是一个不错的呈现手段[7]。但在游戏测试中，例如图 1.1所示，将收集到的数据通过曲线图进行展现，从而能够从曲线图中读出，随着跳跃高度的提升玩家角色的生存时常显著得到提升，但真实情况是，这是在场景设计中某些关卡(机关)难度较大，影响了玩家在挑战关卡时候的流畅性。我们考虑，如果数据可视化能够基于场景关卡信息进行展示，则可以有效的呈现出传统图表无法反映出的信息。



##### 图 .1.1（左）游戏中生存时间和跳跃高度的曲线图（右）基于场景的运动路径可视化，则可以反映出场景的设计问题

**问题3**.玩家在游戏内的行为，能够反映关卡的设计是否达到预期，或是反映出玩家的情绪是否得到调动，但目前是无法直接通过传统的可视化技术来体现，而且自动化的关卡测试技术也难以对语义层面的设计进行测试。游戏优秀关卡设计者会在场景中添加可以隐式表达一些语义的场景元素，玩家则会在体验中获取这些语义信息，设计者从而可以利用这些元素来达到指引玩家的目的。但失败的设计则会让游戏体验变的很差。如图 1.2所示，塞尔达中有裂痕的墙壁意味着可以用炸弹打开隐藏道路中所示：在塞尔达中，当玩家遇到具有裂痕的墙体，会通过使用炸弹来开辟隐藏的道路入口。但是设计者仅仅将有裂痕墙体的场景元素突兀的表达对墙体的装饰效果，则会让玩家错误地理解为这些场景元素是可以进行交互的游戏对象，从而浪费了许多时间在尝试与之交互上，此处失败的设计会影响玩家的游戏体验。但是玩家在此处的驻足行为，在自动化的对游戏场景的测试中，甚至于将玩家路径直接进行可视化的呈现上，均很难发现问题所在。



##### 图 1.2塞尔达中有裂痕的墙壁意味着可以用炸弹打开隐藏道路

本文的测试辅助系统将会整合在游戏beta测试阶段对游戏设计内容的测试用例，集成和提升此测试工作流中的功能模块。系统通过额外收集玩家连续产生的客观的数据，对相关数据进行绑定，再对绑定后数据进行聚类挖掘，最终分析结果会结合场景在多分辨率下进行可视化展示。这样系统可以在一定程度上游戏关卡设计提供指向性的修改和优化方向，同时为衡量游戏表现和玩家体验方面提供较为客观的数据参考。测试团队和程序开发团队利用此系统相互协作，以完成游戏beta测试阶段的任务目标。

### 国内外研究现状

本节调研了目前与游戏测试相关的国内外研究现状，同时又对系统所涉及到的用户体验相关的心理生理学和游戏内数据可视化进行了概述，并归纳总结了当前相关工作中存在的问题与不足。

#### 游戏测试相关工作研究

在游戏测试相关工作的调研中，浙江大学的程文华利用图像特征识别和机器学习的方法，在对采集连续游戏画面数据进行机器学习后，为游戏自动化测试[8]。同样华中科技大学的康怡琳等人，利用自动化测试技术，为游戏软件测试其性能好坏和完整性[9]。随着研究方向的细分，重庆大学的王颐聚焦于游戏中用户界面(GUI)的测试，其规划游戏中GUI测试流程，对所包含的测试用例、预期输入、执行测试等多个环节进行了规范和论述[10]。上海交通大学的王志森，对测试的方法进行了新的探索，并在网络游戏中予以测试实践[11]。另一方面在近年来，西安电子科技大学的赵洋艺，从游戏发行公司的移动数据业务流程进行需求的调研和分析，开发数据统计分析系统。这样可以面向游戏发行公司的后台数据，有针对性的对玩家留存度和活跃度等进行统计分析[12]。开发这样的系统所提供的设计思路是正确的。但是，此系统仅面向游戏运维后台的数据统计，并非游戏产品本身的设计方向，而且所面向的用户仅为游戏运维人员。

总的来说，国内在游戏测试方面的研究是比较少的。上述研究工作仅限于用自动化方法来测试游戏的性能及完整性，或是对游戏测试环节进行规范和对测试方法进行探索。游戏本体的设计意图的测试并未涉及，也没有通过玩家的客观数据进行反馈。

在国外，由于游戏行业起步较早，所对应在游戏完整的开发流程中，游戏测试技术的研究已经相对全面，在Charles[5]所著的Game Testing All in One一文中，从软件测试推演到游戏测试，对游戏测试方法及行业规范进行了系统而全面的介绍。在游戏测试技术的分支中，游戏进行自动化测试及其辅助系统的相关研究已有很多，如Seth Cooper[13]利用代理寻路不但可以自动测试关卡，还可以修复无法使用的关卡。Shivam Agarwal[14]在自动测试2D游戏场景中，利用深度学习的遗传算法，来使AI代理探索场景并记录每帧数据。Paranthaman[15] 通过名为ARAPID的系统对游戏进行高效的迭代测试，为了克服参与线下测试的玩家等不便因素，此系统可以在线收集大量玩家的游戏数据，最终汇集到终端进行分析，为游戏开发者提供游戏测试和可视化结果。在对测试用户体验的方向上，学者Pejman Mirza[16]首次将生理测量数据，问卷调查或访谈结果以及游戏内玩家移动数据结合，然后通过可视化技术辅助理解数据集之间的关系。Aaron Baue[17]r通过基于关卡场景平台，对玩家的路径进行聚类，然后通过可视化的方式结合关卡场景对路径的聚类结果进行展示。

国外在游戏测试方面的相关研究比国内的研究较为全面，这与游戏产业在国内发展缓慢有关，游戏测试在国内还未受到相应的重视。虽然游戏测试领域在自动化测试方向的研究相对较多，但是自动化仅限于对地图空间路径的正确性进行批量测试，而在关卡设计上，玩家对设计元素的理解和体验是自动化所无法替代的，所以游戏设计方面的测试研究逐渐向可视化辅助系统的方向延伸。本文系统游戏的测试不拘泥于测试流程的创新，同时融合更多的技术来辅助完成游戏测试的任务需求，支持向游戏开发者和设计者提供游戏的体验反馈。

#### 游戏用户体验与生理信号数据研究

现代心理生理学研究表明，许多内在情感过程并不一定伴随着视觉、听觉等可感知的物理表现，生理信号的变化往往等更接近人的真实内在的情感感受[18]。而游戏作为一种可交互的诱发素材，玩家在体验时，会激发玩家的内在的多种情感变化，它们对积极的游戏体验有着重要影响[19]。

国内在利用生理数据对情绪进行分类的研究中，近些年开始有所发展。江苏大学流体中心生物医学工程室在对HR、HRV进等信号进行分析后，得出有效的情绪识别方法是以生理信号分析为主，以表情行为观察和情绪主观感受报告为辅相结合[20]。浙江大学的张乐凯[21]利用生理信号进行产品设计和体验评估，利用机器学习的方法，对用户的情感进行分类和回归。但是其实验案例中的诱发素材主要是声音、视频以及实体交互产品，但未对游戏作为诱发素材进行研究。在心流理论[22]的基础上，游戏领域也就诞生了游戏心流[23]，由此，北京科技大学的叶晓帧基于多种生理信号，利用机器学习对游戏用户进行情绪的识别和分析,证明了用户的心流体验与SC和HR信号具有显著相关性。与此同时，利用单一生理信号，诸如ECG信号，GSR信号，呼吸率等进行多种情绪的识别，可以有不错的识别率。

相比国内心理生理学的研究情况，其研究在国外已有很长的时间。一些学者在生理心理的研究中，可以用生理数据反应用户在人机交互中的情绪状态。Muhammad[24] 列出了情绪状态和不同生理信号状态的关系(见表 1.1)：

##### 表 1.1情绪状态和不同生理信号状态的关系

| **情绪** | **生理反应** |
| --- | --- |
| 压力 | 低SC和EMG，高HR |
| 生气 | 高SC和EMG，高呼吸率 |
| 愉悦 | 高SC、EMG和HR，低呼吸率 |

游戏作为人机交互的子类，用户在游戏中的体验感受也逐渐从传统的调研报告到现在对生理心理信号进行测量。Drachen[28]等学者经过实验发现游戏体验问卷与心率的直接关系。另外，Yannakakis[27]指出，心率作为一种心理生理指标，与问卷报告数据中的积极和消极体验有很强的相关性。之后更深入的研究中，Harmat[26] 指出心流状态与较低HRV的低频功率有相关性。近年来，更有学者通过机器学习的方法，将不同类型的生理数据收集后，结合游戏体检测评结果进行训练，得出仅利用生理数据可以对游戏进行体验评估的系统[25]。

综上所述，游戏用户体验与心理生理学的研究方向上，国内的研究逐步成熟，相关的诱发素材开始从单向接收逐渐转变到双向交互相关的产品中。同时，通过生理信号对应特定情绪的高识别率，则可以使得利用生理信号来反映玩家体验游戏时的情感变化成为可能，通过对不同特定情感的调动，可以提升玩家在游戏内的体验。而且，对于测试和设计团队，可以通过将与游戏体验时采集的生理数据进行可视化，则可以反映玩家游戏特定时间和空间内的情绪连续的变化情况。这也是本文系统所做的工作之一。

#### 游戏数据挖掘及可视化工作研究

可视化技术由来已久，在从海量数据中发现规律、增强数据表现、提升交互效率等方面，数据的可视化有着非常重要的作用[刘滨，数据可视化研究综述]。伴随着游戏行业的迅速发展，游戏开发人员可以利用玩家在游戏中的数据进而分析玩家行为和游戏表现[29]。针对游戏运营人员和市场人员，可通过后台数据，挖掘用户特征[30]，进而能够对相关人员的决策做出科学的依据和指引[31]。在另一方面，结合游戏开发实践中，游戏数据可视化的相关研究处于初级阶段。有浙江大学的学者指出，游戏内数据可视化是为了解决游戏开发者关注的两大问题：评估游戏的表现和分析玩家的行为[32]。游戏的表现所指游戏的性能好坏以及游戏体验的好坏。玩家的行为却可以从侧面反映出游戏设计人员的意图，因为玩家在游戏中的行为是能够受到游戏内元素所引导的。

可以看到，在国内相关工作的研究中，利用数据可视化技术来分析玩家行为，促进开发人员对游戏进行修改和迭代开发的相关工作非常有限。我们的系统能够辅助分析玩家的体验和行为，从而有助于设计人员对游戏进行修改来提升游戏用户体验是非常有必要的。

在国外，利用在游戏测试阶段获得的数据，然后对其进行聚合及可视化的研究相比国内更丰富。Medler[34] 利用视觉游戏分析系统对游戏中的数据进行可视化展示，以便让不同技术种类开发人员增强他们对所关注数据的认知。在对玩家的游戏数据进行空间分析和可视化的研究中，提出了四个关键领域：空间离群点检测、空间聚类、空间预测模型、空间模式和规则挖掘。这些领域的发展为游戏的分析提供了新的途径[33]。经过发展，时空可视化也在游戏数据分析中变得越来越流行，并通过可视化系统对玩家的交互数据中提取意义。学者Günter Wallner[34]在Pejman[4]的基础上，使用聚合可视化技术，即利用聚类、区域镶嵌和轨迹聚合来同时显示混合的游戏测试数据。以便于游戏团队中不同的开发人员都可以直观看到数据所反映的玩家行为。基于此研究思路的，用不同算法来对游戏内的数据轨迹进行聚类及可视化，可以开发者看到玩家对场景关卡设计的反馈，又可以让玩家看到自己体验游戏时的过程，以便提升玩家在游戏中的体验。

### 本文的研究内容分析

我们针对本章第一小节提出三个可提升项，根据国内外相关工作的调研情况，以游戏测试阶段性目标为导向，从测试团队及游戏设计人员的角度进行如下分析：

**问题1分析**：在当下，事后问卷作为的玩家体验反馈的数据收集手段有些单薄，短板就在它的非实时性和一定程度上的主观性。而这些短板在明显无法满足测试团队在beta阶段的任务目标，同时在游戏设计层面的反馈上，关卡设计师无法得到玩家情绪调动和关卡设计的反馈信息。1.2.2小节中指出，生理数据和情感体验是具有相关的，玩家的生理数据 ECG、GSR的变化可以反应某些情绪状态（诸如：高兴、愤怒、恐惧、悲伤、平静等）的变化。而且，自主神经系统(ANS)，它控制着心跳、皮电等无意识的生理反应。因此，这些生理数据是不能被有意识地影响或控制，这样就减少了玩家主观因素对游戏可玩性评判的影响[24]。只有玩家通过完整的游戏体验才可以真实反映出游戏的可玩性。对于玩家来说，良好的游戏体验则能够让玩家的特定情感产生变化直至心流。

当玩家在体验时将完整生理数据提供给测试团队和游戏关卡设计者，则也就相当于玩家完整的情感波动变化反馈给了开发人员。数据经过统计处理后，设计人员就可以验证他们的设计是否符合预期，或者对设计进行针对性的调整。由此提升了玩家的游戏体验，也就意味着确保了游戏品质。

另一方面，作为外在生理数据的面部表情，它可以直白的、形象的来表达我们的情绪,因此情绪是识别始于面部表情[肖雅萌,基于面部表情和生理信号的情绪识别方法研究],在非主动控制的状态下，是可以用表情对玩家的情绪状态加以验证。

综上所述，在游戏进行到bate测试阶段，生理数据与玩家的问卷，对于反映玩家情绪变化，可以起到互补作用，将二者在此阶段有效配合，可以帮助开发者得到更真实的玩家体验分析结果。

**问题2分析**：游戏数据可视化的目的是以直观的，形象化的形式将数据展示给开发者。结合游戏场景关卡，将玩家的游戏数据进行呈现，更有利于关卡设计者对玩家在关卡中的体验情况进行整体感知。我们可以用一定的采样频率直接记录玩家角色的空间位置，视野方向，以及触发事件的位置等数据。这些数据具有无噪声干扰而且读取方便等优质数据的特征，这些数据经过统计后，通过可视化技术，不但可以得到数据间的统计图表，在结合场景关卡后，还可以得到场景中区域的访问量，场景的探索程度，场景中玩家的漫游路径等，

为了让可视化更灵活且更符合用户的习惯和特点，数据可视化与交互结合，可以让游戏开发人员指向性地关注场景中感兴趣区域，自定义图元颜色，多分辨率展示和挖掘数据。并让场景关卡也作为可视化内容的元素，结合热力图、轨迹动画、标注提示等方式来表现，对开发人员而言会更加直观显著的了解玩家在游戏内的活动，以及场景设计内容的对比情况。

**问题3分析**： 玩家在游戏中的某些行为，可以通过挖掘游戏数据获得，并结合场景以可视化的技术呈现给开发者。跟随这个思路，当我们采集到玩家在游戏中多种数据后，将不同类型的数据进行组合绑定，并将绑定后的数据进行聚类，则能够反应玩家的某些行为特征。如将玩家的时间和空间数据绑定后进行聚类分析，不但可以得到基本的场景区块访问量，场景的探索程度及其变化量，而且可以反映出区域内玩家的驻足行为[35][36]。利用玩家的这些行为就可以反馈出关卡设计是否达到预期的程度，从而设计者根据反馈来判定是否需要在某个元素上进行修改。

另一方面，此方法还能够辅助关卡设计人员发现玩家群体行为的变化，而此变化产生在特定的时空中却是非有意设计而为。例如在恐怖游戏中，玩家走到一个长廊前，开始加速穿过长廊，玩家此时心跳加速，那么意味着场景调动了玩家恐怖情绪，导致其幅值增强，然而此情景的发生却并非是由设计所为。此系统捕捉到数据反应的玩家行为，并以可视化呈现，是可以让游戏设计者受益的。

综上分析所述，本文系统面向游戏开发人员，着重针对游戏关卡设计人员，设计并实现游戏的可视化测评辅助系统。利用此系统，测试团队可以在游戏beta测试阶段对玩家的游戏数据以及生理数据进行采集和处理，并自动批量的对玩家数据进行分析挖掘和可视化呈现。整合游戏测试工作流可以提升测试团队工作效率，针对性的处理阶段任务。同时系统为设计人员提供友好的交互方式，让用户可以有选择的聚焦数据，在不同的分辨率下理解数据，在静态图表与动画模式下呈现数据。这样不但可以有助于让游戏设计人员验证自己的设计意图是否达到，而且能够让他们快速定位到游戏中需要调整的位置，同时在游戏的调整方向上给予指引。

### 论文结构安排

## 游戏测评辅助系统的需求分析与总体设计

本文在第一章列出了游戏测试阶段存在的不足和问题，详细讨论并提出了相应优化和改进方法。在本章我们要对游戏测评辅助系统进行需求分析，从而明确系统的总体结构，研究并设计系统内各模块间的解决方案，最终完成对此系统的总体设计。

### 游戏测评辅助系统业务分析

本系统面向的用户主要是游戏开发人员，其中，游戏测试团队和游戏设计团队的相关人员是本系统的主要使用者。对游戏测试团队而言，整合测试流程，规范的测试步骤会大大提升测试效率。因为在游戏beta版测试时，团队需要请到项目之外的玩家对游戏进行测试。此阶段测试除了与alpha版阶段对发现游戏关于功能性、稳定性的缺陷外，还需要对游戏的设计和性能做相关测试，并在游戏beta版之后锁定设计。

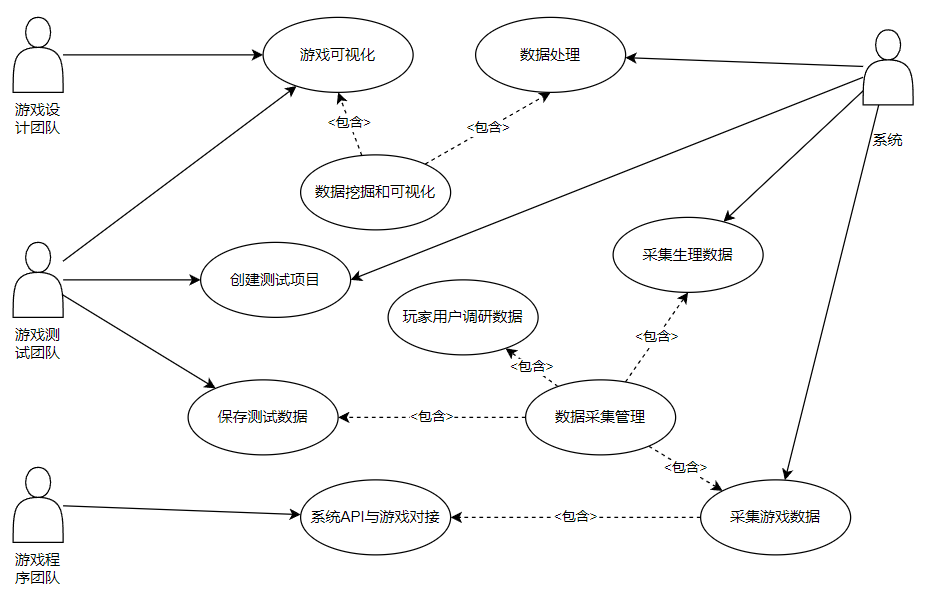
正是开发中的游戏第一次展示给项目组之外的玩家进行体验，这些测试数据就相对更有价值。收集这些玩家的游戏数据，为发现游戏设计中的缺陷并对其调整修改提供了数据支撑。游戏公司在通常的测试活动中，测试团队会录制每一个体验者的屏幕，再游戏体验完成后填写问卷或做用户访谈。然而，录制的视频虽可以连续记录玩家真实的操作实况，但是视频的弊端也显现出来。很难有精确的数据进行分析，同时也很难批量对玩家数据进行处理。

我们的系统则是统一完善了测试的工作流，不但能够从采集到的数据中重现玩家在场景中的体验情况，而且可以进一步对数据进行批量处理和挖掘，来发现玩家不同的行为活动。与此同时，系统还可以根据测试的具体需求，动态的调整场景区域信息和玩家行为的判定参数。系统的业务流程包括以下步骤：

1. 游戏开发人员将系统的API接入所需测试的游戏中。
2. 启动系统的生理数据采集工具，待玩家准备就绪后，将外置生理数据采集传感器正确部署。
3. 玩家进行游戏体验，同时系统的数据采集功能激活，并将数据保存。
4. 游戏开发人员将所采集的数据导入，并为系统配置参数。
5. 系统自动解析分析数据。
6. 开发者在系统中查看数据可视化结果。

在这套业务流程完毕后，关卡设计人员就可以通过可视化技术了解关卡中存在的设计问题，同时还可以与自己的设计意图相对比，以验证自己的设计意图是否达到了预期。而测试人员也可以方便有效的完成此阶段的测试任务。

### 游戏测评辅助系统需求分析

根据对系统的业务流程分析，系统会根据功能的需求，分为三个部分：数据的采集部分、数据的处理部分和数据的可视化部分。我们总结了系统总体用例，系统总体用例示意图如图 2.1所示。其中，按照用于分析的数据来源不同，我们将数据采集模块分为：游戏内数据采集和玩家的生理数据采集两种。其中游戏内数据指的是玩家的空间位置和，玩家的生理数据指的是心电数据、面部视频以及用户对游戏的问卷调研评分。

##### 图 2.1系统总体用例示意图

#### 游戏内数据采集需求分析

在系统中，游戏数据采集功能的需求是可以在游戏实际运行中，将任何所需收集的数据同时间戳绑定后进行格式化，保存成文本格式的记录文件。系统会根据游戏内容的不同，经计算给出数据采样频率的默认值，同时也可以支持用户进行自定义设置。作为通用的游戏测评辅助系统，需要与游戏的具体实现细节解耦，同时在不影响游戏体验的前提下，高效的完成数据采集的任务。其功能的启用与否是能够在游戏的开发版和发布版之间方便地切换。

另一方面，游戏数据的采集与生理数据采集是同时作业的，但是两个采集模块又所属不同的进程。故两个模块之间需要一个通讯机制来确保可以正确的采集数据，我们会在2.3小节进行介绍。

#### 玩家生理数据采集需求分析

生理数据的采集的基本流程是，玩家的生理信号通过传感器，转为数字信号，并将数字信号发送至终端电脑。我们的需求将会聚焦在将不同功能的传感器整合在同一个系统模块中，同时此模块也要保持扩展性。这样可以在不同的测试任务中，按照任务需求方便为此系统扩展添加新的生理信号传感器。在对诸如ECG、GSR或EMG等生理数据进行格式化并保存至文件的流程，与2.2.1小节相似，。但玩家的面部视频数据却是例外。我们考虑能否进行封装后，抽象出统一的接口以便后续系统使用。

从2.2.1小节得知，此模块是需要同游戏数据采集模块分为独立的进程来同时作业。鉴于外置传感器设备的不稳定性，且游戏自身的性能和体验效果是最优先的，如果传感器发生错误，则不能影响游戏的体验进程，同时模块后台需要第一时间重新扫描端口，重新与传感器建立链接，保障数据收集任务的继续进行。

#### 数据处理需求分析

系统用户对数据处理的主要需求在对生理数据的处理上，因为生理数据信号采集获得的原数据是无法使用的。这是由于传感器的采集频率高，而且数据有被不同程度的噪声所污染。那么将数据进行噪声过滤后提取特征值就十分有必要。在数据处理的过程中，需要在性能和效率上达到一个折中。虽然本系统并无对数据进行实时处理的需求，但鉴于对生理数据特征值的需求，以及所需处理数据的工作量，对系统而言也是不小的压力，所以在处理速度和精准度上需要一个平衡，以提升系统用户的操作体验。

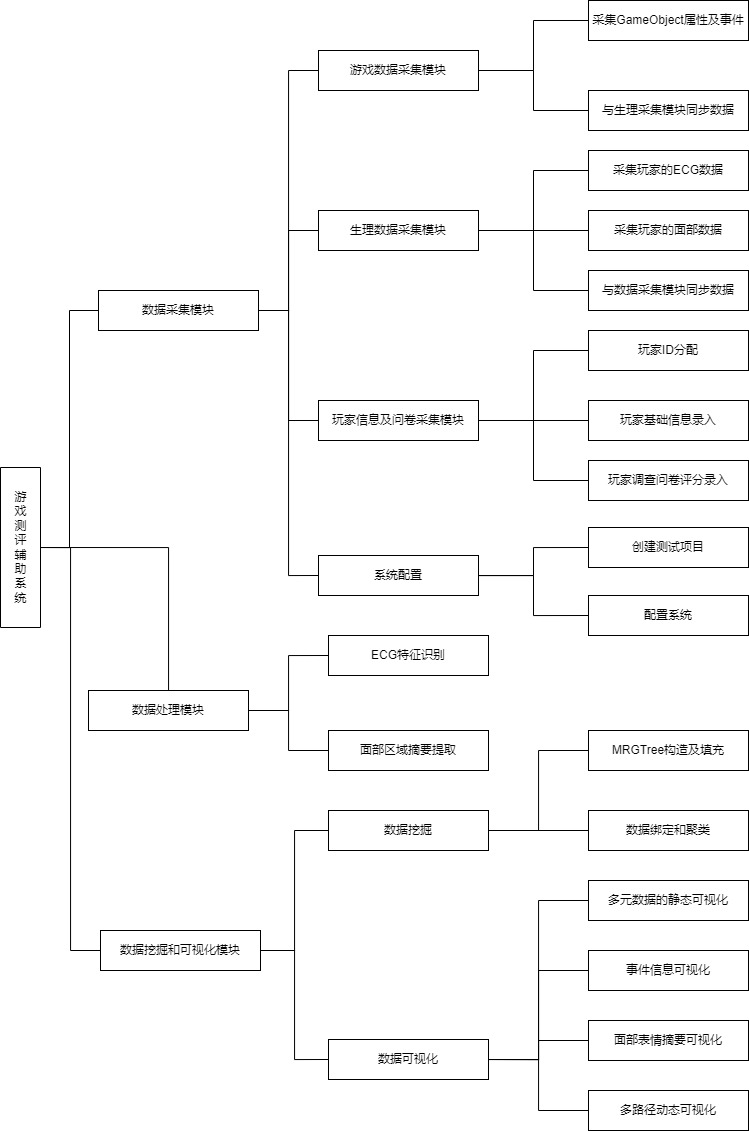
#### 数据可视化需求分析

不同游戏在场景规模上不尽相同，系统用户可以根据需要可以动态的调整可视化的分辨率，根据关卡设计人员所需关注的内容出发，对玩家在场景中的探索程度上，不但需要数值上的统计，也需要直观看到场景区域的访问热度。另一方面来说，场景区域的探索时间前后关系也非常重要，因为关卡中的某些设计需要隐式引导玩家的探索方向。所以可以让用户直观的看到这些基于时间下的区域探索情况也非常重要。设计人员有意营造的气氛和调动的玩家情感的效果，也需要通过可视化技术让他们了解。

系统在可视化模块需要给关卡设计人员需要提供交互功能，例如：分辨率设置，自定义标记的颜色和形状，满足用户自己的敏感度。以一种可以更友好，更合适用的方式让关卡设计人员聚焦在游戏设计的测试结果上。在场景中的特定位置、特定数据集用特定方式向设计人员展示测试结果。

### 系统总体设计

根据上述对系统需求分析，我们将系统划分为五个功能性模块：游戏数据采集模块、生理数据采集模块、玩家信息及问卷采集模块、数据处理模块、数据挖掘和可视化模块。系统的功能结构设计如图 2.2所示。



##### 图 2.2系统功能结构设计图

#### 游戏数据采集模块

鉴于此模块的功能是记录游戏状态和游戏对象（GameObject）的数值属性。游戏的角色就是由游戏对象所派生出来的，以及游戏内的事件数据也理解为是没有渲染组件的游戏对象。我们会将此模块功能封装后以API的形式接入到游戏项目当中，以组件的开发模式作为游戏对象的成员组件对象，并利用游戏数据录制管理器统一进行管理。游戏开发人员可以根据场景的规模对数据的采样频率进行设置，此模块会自动将数据采集并写入到硬盘内文本类型的文件中，发送至的数据处理模块使用。同时也保证在游戏项目的测试和发布等不同阶段能够灵活地控制。

另一方面，为了数据采集的健壮性，将此模块与生理数据采集模块划分为两个独立的进程模块，那么就需要模块间进行状态同步。将此模块抽象为Server将生理数据采集模块抽象为Client。在游戏启动时，此模块在Client请求建立链接后同步状态，并在准备工作就绪后回调给游戏逻辑执行接下来的游戏启动过程。

#### 生理数据采集模块

此模块可以利用外部的非侵入性传感器设备，对玩家的生理数据进行采集，最终保存为数据文件的功能。同时支持多个不同类型的传感器同时采集玩家的生理数据。而且在采集过程中能够实时观测可视化后的生理数据，能够实时控制生理数据的采集状态。

此模块为了确保此游戏数据采集模块进行状态同步，我们还进行一些利用I/O通信（可选Socket、Process Pipeline）的状态同步机制。另一方面，作为生理数据采集端，我们在设计时会利用适配器模式，令其内部的逻辑解耦。从而可以为日后增加所支持传感器设备的种类留下接口，以便于更快速清晰地进行对模块的扩展。

在本系统中，此模块当前支持ECG数据和面部数据的采集。ECG数据通过NeuroSky双电极BMD101芯片传感器，面部数据利用通用摄像头进行30FPS录制采样。

#### 玩家信息及问卷采集模块

在传统的游戏测试流程中，调研问卷至关重要，而在我们的游戏测试流程中，用户调研问卷也予以保留。利用此模块可以将游戏的基本信息录入在系统中。录入的数据会推送到数据挖掘和可视化，支持将录入后的基本信息做简单的统计。

#### 数据处理模块

此模块用于自动化批量的分析处理所有的测试玩家的数据。针对每个玩家，要对采集到的生理数据进行必要的预处理：数据去噪、特征值提取，数据格式化。生理数据作为传感器收集的自然数据，原始数据是被噪声所污染的，所以必须要进行噪声过滤。过滤后的生理数据才可以进行下一步的特征识别，特征识别后的生理数据会同游戏数据一并解析至内存中，以提供给系统进行后续的数据挖掘和可视化。通过这三步的数据处理，原始数据才可以成为能够使用的数据。

#### 数据挖掘和可视化模块

我们将数据挖掘和可视化放在同一个模块中，因为在底层的数据结构的设计中，两者应该同时被兼顾和优化。

最基本的，此模块会提供对玩家信息属性和问卷评分的简单统计，诸如：最大值、最小值、平均值、中位数等。最后会以图表方式进行静态的可视化展示。

模块内会对不同类型的数据进行绑定，然后对其进行聚类操作，从而结合场景来分析出玩家在场景特定区域中的行为表现和情绪变化。同时为设计人员提供对可视化功能的科学友好的交互手段，以便于开发者能够聚焦数据源个体，场景区域，场景探索路径等，还可以选择适合他们理解的可视化方式对数据进行了解。可视化会将玩家行为表现和情绪变化区域以一定的静态方式呈现，同时玩家探索场景的轨迹时序会以动画的方式呈现。

作为情绪识别的辅助功能，玩家的面部表情被记录后，会以摘要的形式在特定的时间点从视频中提取出来静帧，并在特定的空间路径中展示，以便于游戏设计人员对其进行辅助识别。

### 开发运行环境

操作系统：Windows 10

处理器：AMD Ryzen 5 3600 6-Core

显卡：NVIDIA GeForce RTX 2060

开发环境：.NET Framework 4.7 、C#7.1

开发工具：Visual Studio 2017、Unity3D 2020.3

本文所述的系统是一个通用的可视化测评辅助系统，但在本次项目实践中，我们测试的游戏是由Unity3D开发，所以我们的系统会在两个部分与Unity3D引擎相关联的，其一是为游戏数据采集模块编写相应Controller组件的API，其二就是在渲染引擎和GUI模块的底层功能上我们利用了Unity的接口，因为开发一套完整的渲染引擎（模块）绝非易事，且渲染引擎的实现也并不是本文要讨论的内容。

### 本章小结

本章在游戏beta测试阶段从业务和功能需求出发，分别从数据采集、数据处理、数据挖掘和可视化展示层面对系统进行了重点分析，并对此阶段游戏测试流程进行了整合，更有利于测试团队操作。而且根据设计团队的不同要求，选择合适的游戏数据和玩家生理数据进行采集和挖掘，并自定义可视化方案进行展示，从而为设计团队评估设计预期和评价游戏可玩性提供了参考点。本章为系统的实现指明了方向，并且通过对系统模块的架构分析，确保系统能够稳定且高效的运行。



## 数据采集模块的设计与实现

上一章对测试辅助系统的组织架构设计和功能流程设计进行了论述，从本章开始，会对系统中各个组成模块的具体实现展开论述。数据采集是测试辅助系统中相对重要的部分，没有玩家的数据反馈，整个测试阶段也就没有意思。那么从数据采集的来源分为生理数据采集、游戏数据采集以及玩家问卷分数采集三个部分，本章会依次对这三部分进行设计和实现。

### 生理数据采集模块设计架构

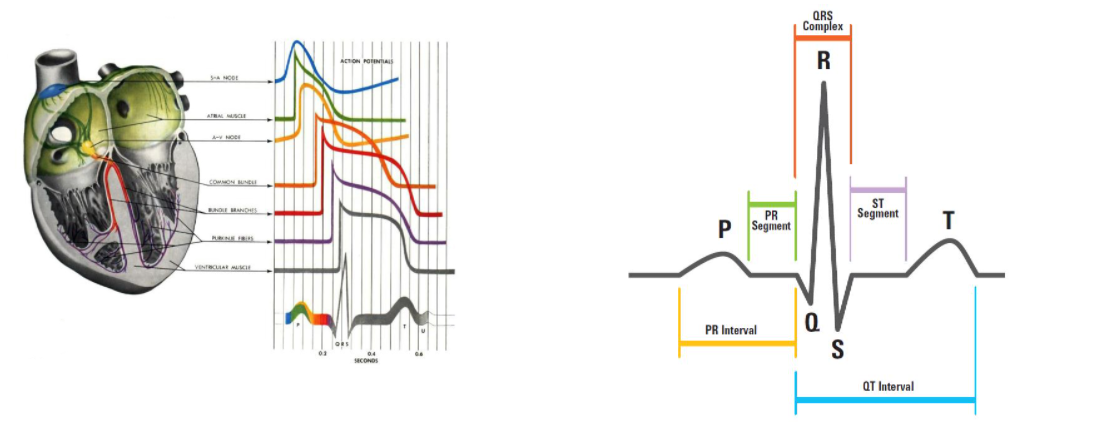
我们首先要对特定生理信号的产生进行相关的了解，随后根据第三方传感器提供的API接口，对其根据需要进行封装整合，然后通过生理数据采集管理器进行统一管理。最终在此应用模块的架构会按照MVC框架进行设计，此内容就在此不过多赘述了。

本小节会对两种不同类型的生理信号的采集原理进行描述：心电信号(ECG)的产生原理，从玩家面部视频中提取关键帧的原理。

#### 心电数据信号采集原理

心脏是人体的核心器官，而心电信号又是心脏完成一次搏动时，在心脏中不同位置的心肌细胞电激动一次动作的过程。在心电信号中，特征波有P波，QRS波群，T波，U波。其中，QRS 波是由一个向下的 Q 波，一个向上跳跃的 R 波和一个向下的S波组成的复合波[37][38]，而我们想要采集的HR信息仅需要关注每相邻两个QRS波中的R峰值的间隔时间。

因为心肌细胞电激动一次产生了电位差，如图 3.1所示，所以从人体体表通过金属传感器获得电位差的变化，就可以得到真实世界中的心电信号数据。ECG传感器设备分为单导联，三导联，十二导联等[39]，这里我们可以理解为导联越多，从人体不同部分采集的电信号叠加后越准确。但是，导联所用的电级贴敷在人体不同部位的采集首先会影响玩家的体验状态，而且有些位置具有侵入性。我们在实验中选择利用双电极肢体导联设备在玩家左右手指末端进行采集。因为，我们不需要过多的电信号细节，只需要获得R-Peak间隔数据，虽然心电信号是一种低频，微弱容易受到外界干扰的信号，但是它的准周期性会确保HR的准确性。故此方法可行。



##### 图 3.1 ECG电信号形成的原理

#### 面部数据信号采集原理

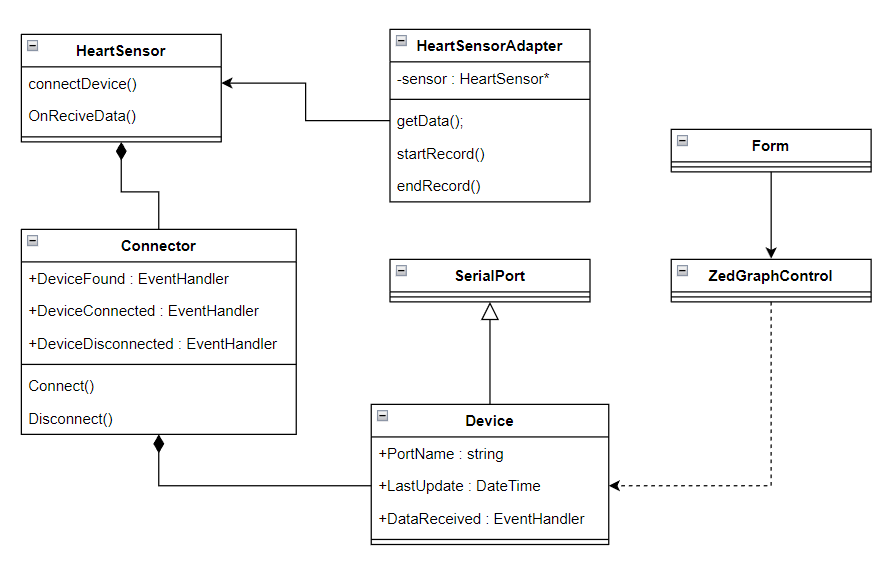
人类的感知、学习、交流、行为、判断和决策等各种功能会受情绪所影响，并会通过影响面部表情来反映个体的内心想法[40]。玩家的面部数据可通过从视频中抽取指定时间点的关键帧，然后对抽取出的静帧进行面部范围的识别，从而将面部区域提取出来。

通常视频文件中提取指定帧内的面部区域的流程如下：

1. 加载视频文件，获得视频文件头，从文件头中解析出视频媒体的时长、宽高、编码格式、码率、帧列表等关键信息。
2. 利用解析出的关键信息还原媒体信息。
3. 定位到视频文件中数据区域内的地址，将帧数据读取解析并暂存到内存中。
4. 将训练完成的人脸检测分类器数据导入到图像识别类库的相关分类器中。
5. 利用上述分类器对内存中暂存的帧数据进行人脸检测。
6. 将面部区域从暂存的帧数据中复制出。

#### 心电数据采集详细设计与实现

从心电信号生成地方原理以及得知，它是由测量体表电位差的传感器连续不断的获得信号，然后通过端口连入电脑后将捕捉的电信号打包传输。我们根据采集原理和流程，采用面向对象的方法，实现此功能的设计类图如图 3.2所示：



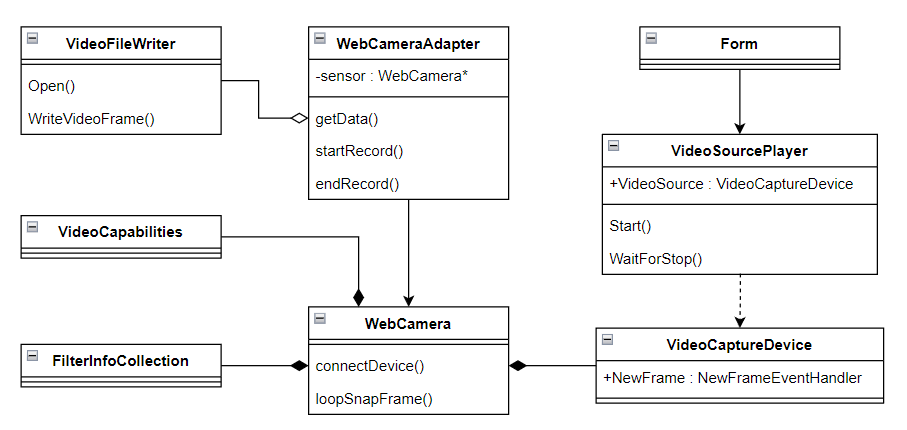
##### 图 3.2心电数据采集模块类示意图

其中，Connector类作为HeartSensor中的成员用来控制设备的链接状况。Connector对象在实例化后会扫描可用的串行端口，当发现设备后，就会执行事件回调DeviceFound。当用户以调用Connect方法和Disconnect方法成功后，会以设备的事件状态来执行回调函数。DeviceConnected设备成功链接时执行回调，DeviceDisconnected设备端口链接断开时执行回调。具体设备的实现细节在Device类中，Device是从System.IO.Ports域下的SerialPort派生而来，用来具体操作串行端口资源。我们将其封装，并添加新的属性诸如PortName，LastUpdate等，这些属性用来更新数据读取状态和在UI中展示串口概况等。而其中重要的是DataReceived函数，当有数据通过串口读入至系统时，则会执行此回调。在这里，ECG数据是在时间周期内打包传输的。我们在此函数内，需要对应ECG的芯片提供商（这里是BMD101）的Package协议文档对发来是数据包进行解析。此架构会在3.1.4小节中与适配器模式的接口对接，按照我们自定义的格式进行数据的序列化，将处理后的数据保存在ISensorAdapter中的容器内，并在用户调用SensorServer类的SaveDataToFile方法后，通过GetData获得ISensorAdapter内暂存的序列化后的数据，将这些数据写入文件中保存。

我们提到，需要数据采集时，提供一个可以实时监控ECG的数据的图表可视化功能。折线图的绘制，我们采用第三方插件ZedGraphControl进行绘制。

#### 面部数据采集详细设计与实现

面部数据采集，我们选择借助第三方AForge多媒体底层类库对视频进行硬件设备的链接和录制。其类架构如图 3.3所示：



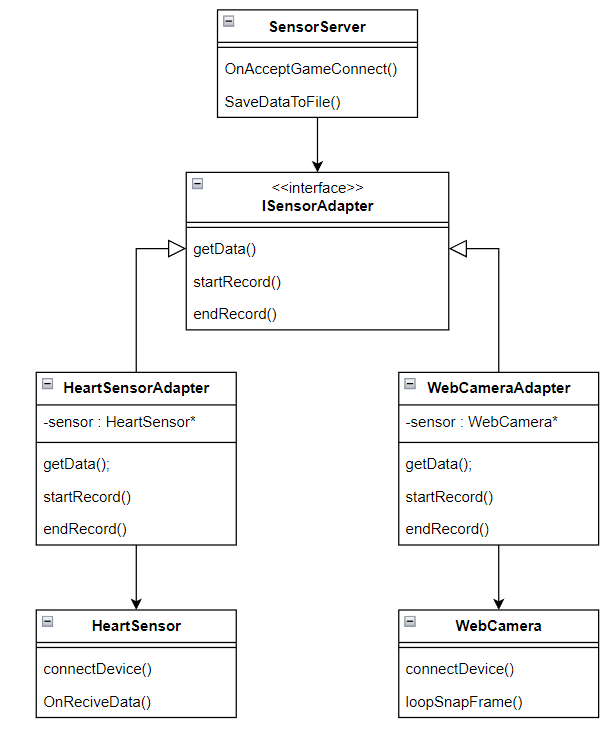
##### 图 .3面部数据采集模块类示意图

其中FilterInfoCollection用来收集连入本机的视频设备，并将收集来的视频设备信息展示到UI交互界面，以供用户选择本次录制视频的设备。在选择好设备并建立好链接后，则通过VideoCapabilities选择视频设备支持的录制分辨率。VideoCaptureDevice用来描述视频捕捉设备，并对设备进行操作。其中关键成员是委托变量 NewFrame，即当有捕捉到新的一帧画面后，回执行此回调。捕捉到的单帧画面会以Bitmap的形式传入。供WebCameraAdapter类中的VideoFileWriter对象将单帧数据写入至视频文件中。至此，在VideoSourcePlayer中引用VideoCaptureDevice对象，则VideoSourcePlayer会直接在UI界面上绘制出捕捉到的单帧(Bitmap)数据，以此会呈现出连贯的视频影像。

#### 生理采集工具整合及实现

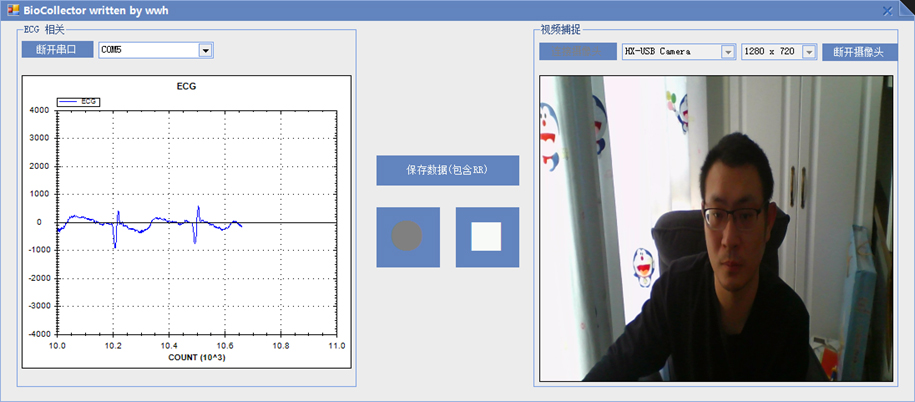
针对玩家的生理数据采集模块，为了便于扩展，我们对业务接口进行了抽象，在设计上我们选择了适配器模式(Adapter Pattern)，因为传感器芯片的厂商不同和业务逻辑不同，导致获得数据流和数据格式的方法均有差异，我们利用 ISensorAdapter 作为中间层，派生出适合多种传感器的Adapter，实现接口函数。然后SensorServer依赖对抽象接口的操作，从而获得数据后保存到文件中。

生理采集工具与游戏数据采集的通讯模块比较简单，鉴于我们是在本机进行的数据采集，我们选择了用进程管道数据流（NamedPipeServerStream）作为与游戏数据采集模块的通信。这样设计的意图在于可以将测试游戏和生理数据采集空间解耦合，并且今后可以扩展至对网络游戏进行评测辅助。如图 3.4所示。



##### 图 3.4生理采集模块利用适配器模式架构的类示意图

上述两个生理信号的采集功能，按照适配器模式架构，整合成为一个生理数据收集的工具，此应用程序的界面实现如图 3.5所示：



##### 图 3.5生理数据采集应用程序界面

界面中的ECG的下拉菜单可以选择有设备接入的串口，然后链接串口后则建立与传感器的链接。当成功建立链接后，按钮提示会变为断开链接。当有玩家佩戴好传感器后，下方的图表中可以实时绘制ECG的波形图，其中X轴方向表示接收到的采样数，Y轴表示心电的振幅。

在视频捕捉中，通过选择已经接入电脑的摄像设备后，设置选择设备所支持的采集分辨率。然后建立相应的链接。同时下发可以实时绘制摄像机采集到的体验游戏时玩家的画面情况

主界面中央位置，两个按钮分别是圆形对应开始录制和方块对应停止录制，同时还有保存数据按钮。用户录制完成点击保存后，心电数据和面部视频数据都将保存至文件中，文件名为格式为[ID\_时间戳\_数据类型.扩展名]。其中心电数据类型为ECG，扩展名为txt，玩家面部视频数据类型为FACE，扩展名为avi。

### 游戏数据采集模块设计架构

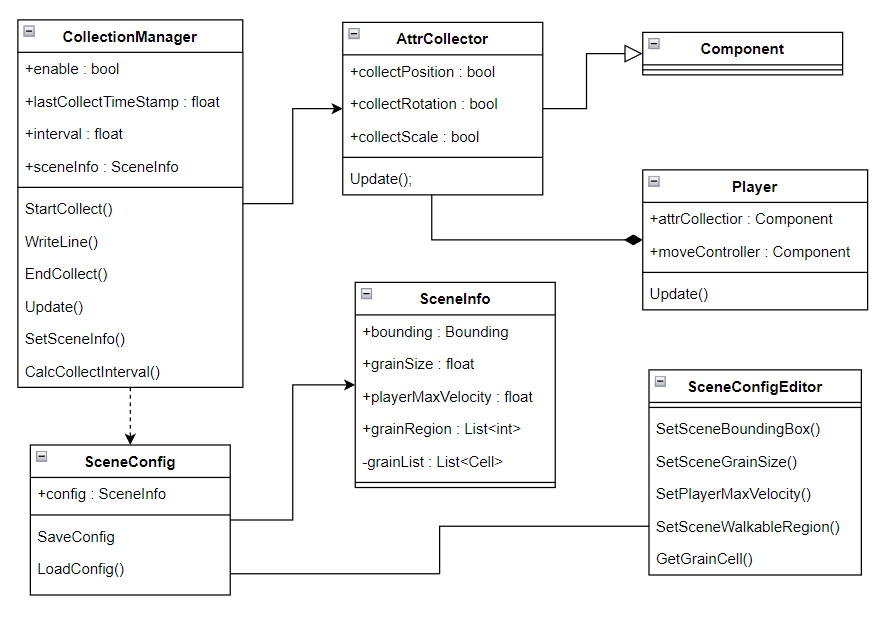
**游戏玩家角色属性的采集**：游戏在游戏中，测试团队会根据设计者所关注的内容来对玩家进行相应的游戏数据收集。在游戏中最常见的就是玩家的时空数据。虽然在游戏中，收集时空数据最简单的办法是可以在游戏的主循环中，读取角色transform属性并绑定时间戳进行保存，但是，数据量会非常大，而且数据冗余严重。故我们需要设置一个采样频率，可以确保玩家时空信息，同时还可以最大程度减少数据冗余。根据奈奎斯特采样定理，我们将场景网格化后，以给定最小可视化网格单元边长为L，在游戏中虚拟角色的速度V（米每秒），在采样的间隔时间内应有：

(1)

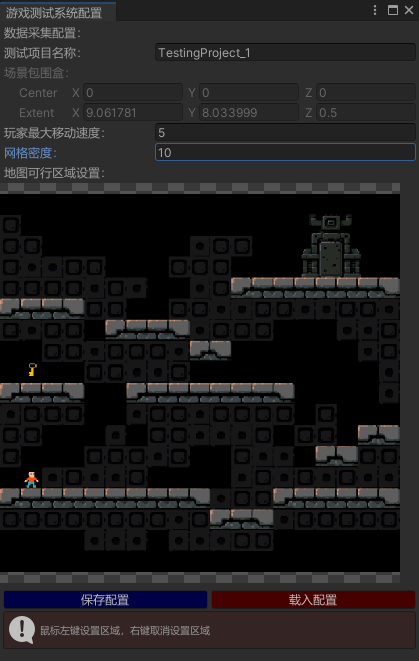
则有：

(2)

这样，我们就可以确定一个采样频率的最小值。然后我们将这个数值提供给用户作为采样频率的默认值，当然也支持玩家进行自定义设置。此功能的类架构图为图 3.6所示：



##### 图 3.6玩家角色属性采集功能的类示意图

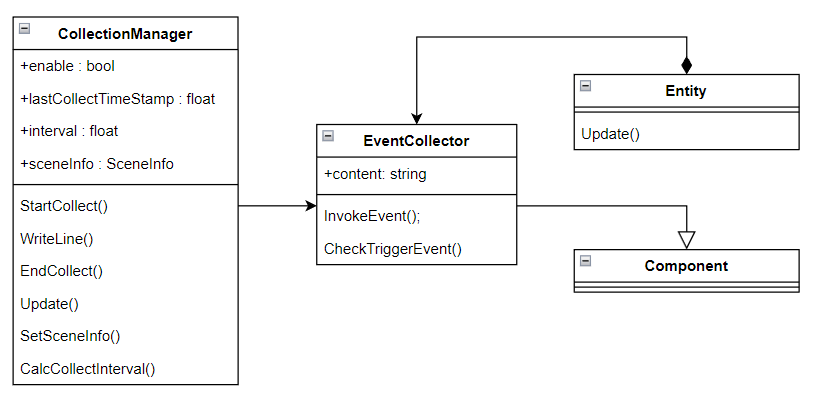
在玩家数据在采样前，需要为系统配置游戏测试场景的参数信息，用户与SceneConfigEditor实例化的窗体交互，可设置场景的基本属性：最小包围盒，最小的网格密度，场景中可行走的区域，玩家的最大移动速度等。设置最小的网格密度和玩家的最大移动速度会影响系统对玩家游戏数据的采样频率。最小的网格密度设置后，会生成覆盖场景的网格蒙版，用户可在网格蒙版中批量选取网格并设置为可行走区域，场景内的可行走区域与非行走区域的网格数目比值会被计算出，这些参数会保存在SceneInfo的对象中，并通过SceneConfig进行读写和管理。它们在数据挖掘和可视化模块被使用，详情见5.2.2章节。接下来CollectionManager通过配置的场景信息对采样频率进行计算，同时这个类还实现对文件流的格式化和读写等功能。玩家在其配置界面设置如图 3.7所示：

##### 图 3.7 测试系统配置界面示意图

其中，AttrCollector和Player的作为组件模式，可以在Player实例中灵活拆卸。在Player调用Update时，AttrCollector中的Update也被同时调用，Update中会不断向单件模式的CollectionManager请求保存收集来的数据。如果满足采样时间的话，那么数据被写入至文件中。

**游戏特定事件信息的数据采集**：通常情况下，游戏中触发的特定事件指的是游戏内实体(Entity)满足某种事件的触发条件，而对其属性变量产生变化。设计团队根据测试需要，会有选择的对触发事件时的信息进行记录，将事件产生的时间，位置，事件名称，变量值等保存至文件中。

此功能是较为特殊的，需要游戏程序团队和设计团队进行配合，因为事件的触发条件的情况是有时是非常复杂的，这就是需要程序团队帮助设计团队定制测试用的API，系统会将事件记录的接口与定制测试用的接口进行对接。但是有些记录的事件变量信息，可以通过游戏中同时触发此事件已知的回调函数获取，最常见的就是角色走进某个区域内，对角色自身的某些属性进行采集并保存。那么此功能的架构如图 3.8所示：

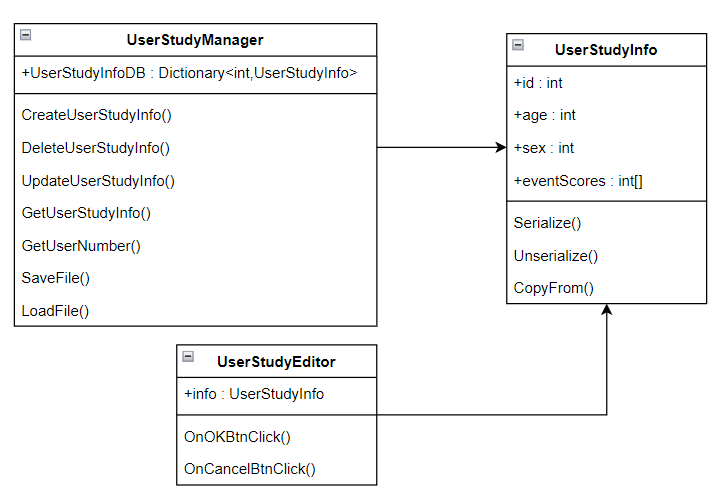


##### 图 3.8游戏中事件数据采集功能的类示意图

其中，EventCollector会在事件触发条件满足后，将事件信息的参数传入到其静态方法内，进行格式化，然后内部调用CollectionManager中文件流的操作方法。

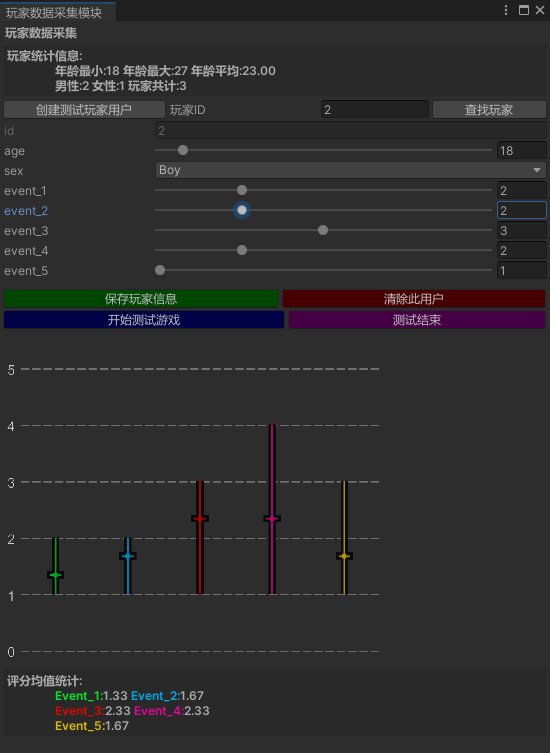
### 玩家信息及问卷采集模块设计架构

关于玩家信息的录入，我们会在游戏开始前，创建一位新的测试玩家，模块会自动分配玩家的UserID，当通过UserID查询到对应的玩家后，即可开始游戏测试。待体验结束后，将玩家的信息调研和问卷评分通过此模块汇总，并由本模块将采集的玩家信息（即个人信息及问卷评分数据）与玩家的UserID、游戏测试启动时刻的时间戳共同构成了三元组，并保存至文件内。游戏测试团队人员可在后期通过UserID进行索引查找符合UserID的玩家信息。同时在此模块的界面内，还对所有参与测试的玩家信息进行了数据的初步统计，并以静态图表的方式展示。此功能的设计类图如图 3.9所示：



##### 图 3.9玩家信息及问卷采集功能的类示意图

模块中架构非常简洁，用户与UserStudyEditor实例进行界面上的交互，界面设计如图 3.10所示。用户在填写完成用户信息后，会将UserStudyInfo数据更新保存至文件中，UserStudyEditor会调用UserStudyManager中的方法函数，对数据进行相关的新建、删除、修改、查询操作。还要提到的是UserStudyManager会给新建的玩家自动分配全局唯一的UserID。



##### 图 3.10 玩家数据采集交互界面示意图

由界面效果图中可以看到，在游戏测试开始后，当前测试玩家的UserID会同步至游戏数据采集模块，并与生理数据采集模块进行同步。待玩家体验结束后，需点选测试结束以便停止当前玩家的所有数据采集状态。接下来继续完成关于游戏体验的用户调研评分流程，并更新玩家的调研数据至UserStudyInfo结构中。

### 本章小结

本章节从系统的架构层面，介绍了此模块是基于适配器模式架构。此模式非常适合于模块在整体结构上抽象出统一的方法和接口，但是实际的业务逻辑层方法又各自独立的。适配器模式使用后会使得系统解耦、灵活又便于扩展开发。同时架构上利用Pipeline(或Socket)对两者不同模块的进程进行链接和通信，确保生理数据采集与游戏数据采集模块进程的相互独立，又可以对数据采集事件时间和采集状态的同步。

从功能的设计与实现上来看，我们又从硬件传感器获取信号的设备及原理方面进行了分析，介绍了用电极式ECG传感器来采集电信号的可行性，同时利用摄像头捕捉玩家面部表情的前后过程。最后分析了测试玩家在游戏中，设置关于虚拟角色的空间数据的采样频率。并在体验完成前后，将玩家的信息和对游戏体验的问卷分数录入至系统内。此系统整合了数据采集过程中流程结点，关于玩家的全部所需数据收集可以利用此系统模块完成。

## 数据处理模块的详细设计与实现

在收集到玩家的数据后，需要对原始数据进行处理。其中对于采集到的玩家游戏内的数据，符合高质量数据的多个特征，满足在数据挖掘及可视化阶段直接使用。另一方面，对于采集到的生理数据，需要进行数据处理来获得直接可用的特征数据。

### 心电信号中R波的识别方法

获得到的原始心电信号是带有时间戳的振幅数据，高采样率导致数据有过多的冗余信息，而传感器设备在采集信号的活动中，又使得数据被噪声污染，所以原始心电信号经过处理后才能获得我们需要的特征数据。我们需要的心率信息的关键在于可以识别QRS波群中R波时域中的位置，并利用相邻两个R波的时间差计算心率(HR)以及心率变异性(HRV)。

在心电信号特征数据识别中，通过小波变换对信号进行优质的去噪处理，去噪后的数据再经差分阈值法获得R波位置。

#### 小波变换法

在工程上，对信号进行噪声过滤的方法有许多，例如：中值滤波法，利用傅里叶变换去除噪声，无限冲击响应滤波器(IIR)[44]等。但是，上述方法在心电信号去噪方面表现不佳。而小波变换在去噪上却是强有力的工具，它是一种对信号进行时频分析的数学方法，在时频两域都具有表征信号局部特征的能力[62][65]。即在低频部分具有较高的频率分辨率和较低的时间分辨率，在高频部分具较低的频率分辨率和有较高的时间分辨率，很适合对心电信号进行分析，从而提取心电信号特征波中的R波。小波变换的原理是将信号分解为一系列的小波函数的叠加，而这些小波函数都是一个母小波经过平移和尺度伸缩得来的。连续小波变换的核心公式如下：

(3)

在公式(\*.\*)中，是母小波，是母小波的复共轭。a对母小波进行伸缩操作，被称为尺度因子；b对母小波进行平移操作，被称作平移因子；其中的是起到调和作用，使得具有不同a值的小波的能量保持相等。

通常使用小波变换处理心电信号的流程是：

1. 利用小波变换分解原始信号
2. 选取合适的母小波和分解层数，使噪声的能量分布于整个小波域内。
3. 使用合适的阈值将有用的信号系数保留，将噪声的信号系数置为0。
4. 最后再对经阈值过滤后的系数进行重构，得到去噪后的信号。

在实际应用当中，尤其利用计算机对采集来的信号进行小波变换时，通常数据呈离散化的，上述尺度因子a取 a = 2j，平移因子b取 b=k2j。此时的母小波变为：

(4)

#### 差分阈值法

差分阈值法是本质是通过在连续的信号中，通过在相邻时间点上振幅的差值来求斜率及其斜率的变化率。因为回顾QRS波群的图像，QRS波区域中的高频含量为[5至15]Hz，且不难发现R波的振幅是最显著的，故R波的上升和下降的斜率变化也最强烈。所以我们可以在给定的窗口内对信号进行分析，找出与设定阈值相匹配的斜率，就可以找到R波在信号时序中的位置。

差分阈值法的核心算法就是对离散的采样点进行差分计算，从而得到斜率，其中的一阶差分和二阶差分的形式为：

(5)

其中， 为预处理后的ECG信号序列，下标t为时间刻度。接下来，在阈值设定和查找的阶段中，不同的算法间会有差别。

利用差分法查找R波是基于公式6[45]：

(6)

利用二阶差分做平方，以增强QRS区域差分信号的幅度，同时减少了基线漂移对一阶差分的影响。阈值可以利用窗口内排序后的差分序列，进行动态的变化。算法会在寻找R波的过程上进行逻辑和经验值上的判断和处理。

经过对差分法的改良[43]，有公式7：

(7)

将一阶差分和二阶差分求其各自平方的线性和，即对信号中的R波信号进行了放大，但是同时也会受到高频噪声的干扰。最后我们设置阈值，有 其中 可以是随着窗口区间可以动态调节的，连续三个采样点满足不等式，则 t 时刻即为R峰值。此法可以抑制P波和T波对检测的干扰。

另一种变形后的差分法[41]基于公式8：

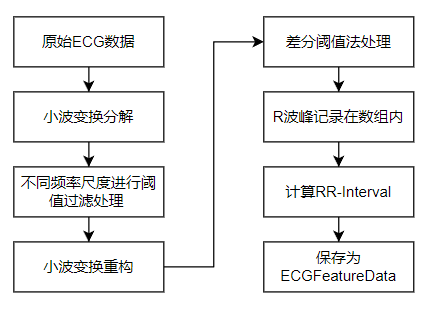
(8)

其中的 和 分别是窗口区间内一阶差分的最大值和二阶差分的最大值，此方法与方法二异曲同工，相当于在对窗口区间内的ECG信号进行了放大和整流。同时在我们可以得到区间内的,然后通过设置阈值 ,使得存在连续 ，但是其中的参数C为经验数值，可通过多次实验统计后得出。

通过常见的三种差分阈值法的算法分析，我们可以看到基于此方法的扩展算法已经相对比较成熟，且都大多利用二阶差分对信号进行相应的放大，算法简单故便于实时进行R波检测，但是，此方法都较易受高频噪声干扰。在查找R波的过程中如果利用经验数据引入修正规则和逻辑判断，虽可以提升准确度，但是又会相应的加重了算法复杂度。

### 本文采用的特征R波识别方法

因为差分阈值法容易受到高频噪声的影响，为了提升信噪比，保持信号的特征，同时又在非实时状态下去对数据进行处理，我们则可以将小波变换和差分阈值法结合，利用小波变换去过滤噪声，去噪后的数据再利用差分阈值法进行R波的检测。整体的流程如图 4.1所示：



##### 图 4.1本文心电信号处理方法流程图

其中，心电信号的能量几乎聚集在QRS波群，其频谱带宽为3至40Hz，R波波峰能量在6至18Hz附近集中。在对心电信号进行小波变换的实际应用时，我们选择Coiflets4作为母小波，对信号进行8尺度分解，其中的第1层和2层高频系数主要为噪声，而第8层低频系数主要为基线漂移。将这三层系数进行阈值判断，将满足阈值的系数置为零。最后利用小波逆变换重构信号，则对差分阈值法影响较大的噪声将被去除，重构后的信号信噪比会提高。

使用差分阈值法寻找R波的核心算法伪代码如算法1所示，其中widthOfSearchWindow表示再连续两次差分值符合后，搜索第三次差分值的窗口宽度。AdjustDiffThreshold()可以根据需要，利用上下文已获得数据，自定义动态调节D1、D2、D3阈值的函数。经过整个流程，系统将玩家的心电数据整理位有RR间隔时间的ECGFeatureData。

|  |
| --- |
| **算法1**差分阈值法找寻R波伪代码 |
| 设置 D1、D2、D3 为连续三次差分的阈值  loop ecgList //遍历ECGList中的数据  if(ecgList[i] - ecgList[i - 1] > D1) i++  if(ecgList[i] - ecgList[i - 1] > D2) i++  //最后依次差分阈值判断时，搜索后续指定长度的窗口内数据  for(; i < widthOfSearchWindow ; i++)  if(abs(ecgList[i] - ecgList[i-1]) > D3 &&  ecgList[i] - ecgList[i-1] < 0)  RList.Add(i-1)  RR = RList.Last() – LastR  //根据需要此处可以动态调节阈值  AdjustDiffThreshold()  End if  End for  End if  End if |

#### 可行性验证

在前期的文献调研中，虽然我们已经得知心电信号和特定情绪的相关性，但为确保真实性和可行性，我们在系统设计之初进行了前期的可行性实验。通过利用自定义的一款恐怖游戏作为情绪的诱发素材，对参与者21人进行了数据收集。

在执行具体的游戏体验前，我们给玩家戴好实验所用ECG传感器设备。让玩家静置5分钟后，在观看轻松的风景视频，收集在平稳情绪下玩家的生理数据。然后，让玩家体验游戏内容，游戏内会触发2次恐怖事件和1次无恐怖事件的对照。全程体验过程中无场外干扰。待玩家结束游戏后，数据采集也同时止。最后，玩家对游戏的内触发事件时的恐怖程度，进行五分制的评分。接下来，我们对问卷数据进行统计，以游戏内特定事件触发后的8次心跳间隔与恐怖事件的评分进行相关性分析，结果如表 4.1所示：

##### 表 4.1玩家平均RR与恐惧值相关分析

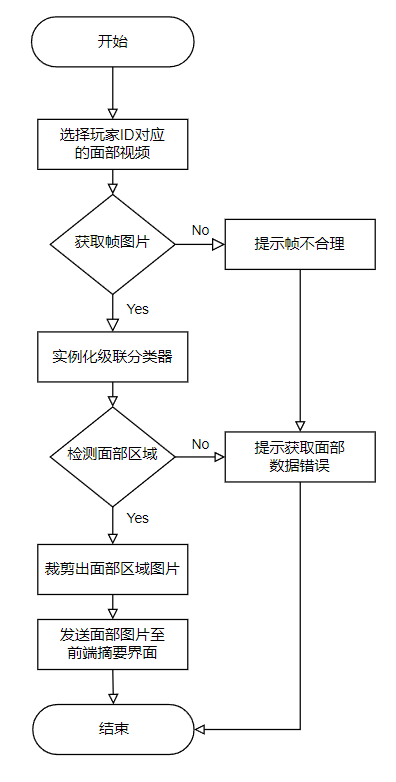
|  |  |  |  |
| --- | --- | --- | --- |
|  | | 平均RR | 恐惧值 |
| 平均RR | Pearson Correlation | 1 | -0.273 |
|  | Sig.(2-tailed) |  | 0.05 |
| 恐惧值 | Pearson Correlation | -0.273 | 1 |
|  | Sig.(2-tailed) | 0.05 |  |

通过恐惧值和平均RR的相关性分析可知，Sig <= 0.05 且皮尔森相关性指数为 -0.273 ，呈负相关, 即恐惧值越高，平均RR值越低。由此可知玩家的ECG数据与玩家的恐惧程度有相关性。所以，我们的系统要利用ECG信号来反映玩家的情绪变化是可行的。通过玩家的情绪变化与时空数据相结合，即可以给予游戏开发者了解游戏在不同阶段和不同场景下带给玩家的实时情感变化。

### 玩家面部摘要提取

本文系统中，会对玩家在体验游戏时面部表情提供摘要展示。为了让游戏设计团队人员更聚焦于玩家的面部及其表情上，我们要做的就是从记录的玩家面部的视频中抽取关键帧，然后对关键帧做分析，将玩家的面部图像从关键帧中裁剪出来。此方法的提取算法流程如图 4.2所示。

将面部从关键帧中截取出来，可利用OpenCV库的相关函数（如C#语言可以利用Emgu），我们目前会利用OpenCV中基础和便捷的Haar特征的级联分类器。我们目前会直接使用OpenCV库中已经训练完成的人脸捕捉的Haar特征数据。Haar特征分类器是一个XML格式的文件，该文件中的数据会描述人体各个部位的特征值。利用CascadeClassifier对象加载训练完成的Haar特征数据文件，然后则可以通过detectMultiScale方法并找出面部的区域，对其裁剪后方可进行展示。



##### 图 4.2提取面部区域算法流程图

### 本章小结

本章我们对心电信号的产生原理进行了介绍，同时分别对心电信号的数据处理和面部视频的面部摘要提取进行了论述。在心电信号的数据处理的方法上，我们将小波变换和差分阈值法进行结合，对信号进行噪声过滤和特征R波识别。同时，我们将识别后的R波结合玩家的恐怖感评价做了相关性分析，结果恰好满足相关指征。

在对玩家面部摘要进行提取的方法上，我们利用了OpenCV对视频进行了处理，抽取视频中特定时刻的关键帧，并通过训练完成的面部区域特征分类器对其进行检测，找出玩家的面部区域并将其返回。以此除去玩家背景信息对玩家验证玩家情绪的干扰。而且，提取面部的摘要数据，为后续对玩家的面部情感计算提供了便利的接口。

## 数据挖掘及可视化

对游戏的设计开发人员来说，将游戏中获得的数据进行挖掘和可视化是非常重要的事情。因为从游戏开发者的角度而言，他们关注的内容是可以由数据挖掘和可视化的方法来方便地提供。我们依照游戏开发人员的不同职能将游戏数据挖掘和可视化分为三个需求方向：策划设计人员对游戏设计意图的反馈，一般来说，设计意图会以玩家的行为特点和玩家的体验状态相关联。美术技术人员对游戏中风格化气氛的烘托，以及美术资源对玩家焦点的吸引力的程度上。最后就是程序开发人员对游戏自身的性能表现方面，诸如帧率流畅度、内存的占用变化、网络的延迟时常等。还有额外的方面，面向市场人员和运维人员对游戏投入市场后，所产生的广告曝光率，点击下载率或首充金额等方面的数据统计，这些并不属于游戏开发中数据，也就不在本文所开发系统的考虑范围之中了。

对于上述游戏自身性能表现方面，在程序开发中相关的工具已经趋于完整，因为此相关功能与玩家交互的耦合度低，故可做大量自动化的压力测试并分析结果。而另一方面，需要玩家参与交互的方向，则其相关的数据挖掘和可视化的研究就屈指可数了。所以我们也正是要以此为出发点，要结合玩家游戏数据，对其进行挖掘和可视化。

从第三章数据的采集中可知，我们通过此系统采集到了玩家的心电数据、面部数据、玩家虚拟角色的时空数据和事件触发数据、甚至玩家体验后的情绪自评数据。将这些不同维度的数据进行相应的组合，通过一定程度的数据挖掘和可视化能够满足游戏开发人员的需求。

总的来说，数据挖掘的目的是要发现玩家游戏数据背后所反映的玩家行为的共性和特点，以便给策划和美术人员提供游戏的修改方向；而数据可视化是为了将抽象冰冷的数据具象化，以特定的颜色，形状，甚至动画表现形式，结合游戏中具体的可交互元素，让美术和策划人员能明显的发现玩家在游戏中的兴趣区域、访问路径等。

### 数据挖掘

在本文中，由于游戏本身作为交互类型的限制，同时游戏又处于开发环节末期的测试阶段，能得到的玩家游戏数据量不足以作为分类学习的样本，我们在本文中会以聚类方法对游戏数据进行挖掘。而聚类的主要特点就是要把一个数据集分成不同的类，在同一个类中，数据的相似度尽可能高，而在不同的类之间，数据的相似度尽可能低，换句话说就是差异程度要尽可能高。

#### 划分法

在聚类算法中，经典的k-means，k-means++，bi-kmeans都属于划分法。以k-means为例，他所代表的划分法的主要思想是，通过预先指定k个数目的分类情况，并随机相应的聚类中心，通过反复迭代，以至于将一堆散点，聚类成为k类，迭代直到“类内的点都足够近，类间的点都足够远”[54]，我们的目标函数如下：

(9)

其中是数据点，是数据质心，上述的“远”和“近”可以用所需特殊的距离算法来判定,一般情况下，我们会用欧式距离。但是，此方法有一些特性，如需要指定k值，即预先确定类的数量；又对初始化的随机质心和数据中的异常数据都非常敏感。再观察游戏内的数据（如：时空数据，生理数据等），我们很难在初始化时决定k值大小，更无法给予合适的质心位置，故划分法在处理我们收集到的游戏数据时并不合适。

#### 层次法

层次法的原理是通过将整个数据集进行层(hierarchical)划分，每一层都包含有多个相应的聚类，本层的所有聚类是由上一层聚类通过聚合或者分裂生成的。聚合的生成方式又可以理解为是自下而上的方式，最开始数据点各自为独立的聚类，通过相似程度将最相似的数据点进行合并，生成新的聚类，反复迭代，直至所有数据点无法再进行合并，或聚类个数达到了上限。相反，分裂的生成方式是自上而下的，最开始所有的数据点归属于同一个聚类，根据相似度将聚类进行分裂，产生为多个新的聚类，反复这个过程，直到聚类无法进行新的分裂，或者聚类个数达到了上限。整个聚类的层次化流程更像是一颗树从根节点到子叶的生成，或是逆向的生成。

层次算法改进算法也比较多[48]，例如：BIRCH 算法、ROCK算法等，但是层次法也有不少缺点，其算法的效率是比较差的，且每层聚类中包含数据点的个数也是预先设定好的，同时也是对凸型的分布数据比较敏感。但是，层次法的思想给我们带来一些方向，树形结构的生成和游戏场景的切分有异曲同工之妙，而且在可视化的阶段，可以通过观察不同的层级的聚类情况，开发者就可以实时在不同分辨率下观察数据可视化的想发成为可能。更适合的聚类方法，我们在介绍网格法时会进一步说明。

#### 密度法

此方法不同于划分法，划分法在凸型数据聚类的分布上具有很好的效果，而作为场景中玩家的空间位置数据，划分法就不适用了，但是密度法正好是可以对不规则形状的聚类予以划分的[50][51]。的常见的密度法有 DBSCAN 和 OPTICS。以DBSCAN为例，其算法的描述为：

1. 通过设置两个参数，扫描半径r和最小包含点数M，从未访问过的数据集中的数据点P开始，进行邻域r范围内的搜索。
2. 如邻域内数据点个数小于M，则P被标记为噪声点。
3. 反之如邻域内数据点个数大于M，则将邻域内所有数据点及P点归为一个聚类C，此时P被标记为已访问。同时进行递归，以相同流程处理该聚类C中所有未被访问的点，从而对聚类C进行扩展，直至聚类C中所有的点均被访问。
4. 最后重复上述1至3步骤，直至数据集中的数据点全部被访问。

由此，我们可以看到在DBSCAN所反应的密度法中，是对噪声不敏感的，而且可以识别出任意形状的聚类。此方法对识别玩家路径和筛选噪声点是有行而效的。但是此方法也存在一些缺陷，首先算法带来的时间复杂度 O(n2),其次就是在数据密度发生太大变化时会发生一些问题。虽有上述缺点，我们会利用与网格法相结合来优化算法，减弱上述缺点。

#### 网格法

网格法可以理解为是分而治之的结构算法。保持层次法中层次化的思想，将层次法中对聚类集合的抽象层级，转而为对空间中固定划分的具体的容器网格。同时在数据集更新时，不需要重新计算单元格结构中的所有信息，只需要进行增量更新。

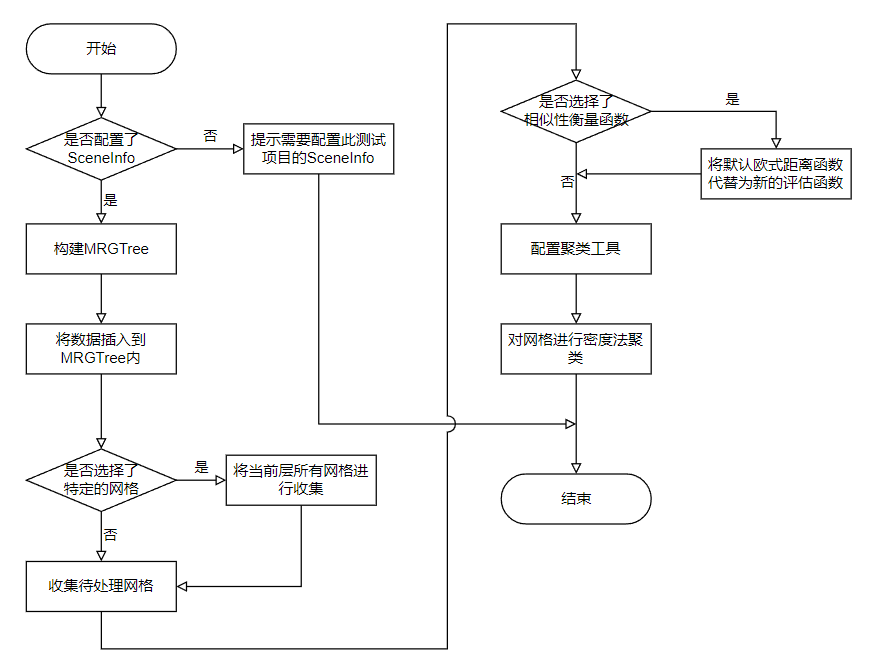
以其中的 STING[53]算法为例，STING全称为统计信息网格，算法对空间区域划分为矩形单元格，采用层次化的树形结构，2D场景以四叉树进行划分，对长宽进行对半细分至4个子节点，当扩展到3D场景后，以增加网格子节点的划分个数即可。算法按照分辨率的需要将场景进行特定深度的层次结构划分，此结构划分完成后，即可将数据集中的数据逐个从根节点开始逐层映射到网格中，数据在链式遍历层级网格时，即可对网格中的统计数据和参数进行更新。

### 本文所用的方法

由上一小节网格法可知，网格法的优势明显，层次结构的构建速度快，数据的增加和查询执行效率高，而且此结构与游戏场景管理中的四叉树或八叉树的层次化结构不谋而合。但网格法也并非完美：层次结构叶子节点粒度决定了聚类的效果，对高维数据的支持会以网格粒度会以指数增长。但乐观的是，在可控的游戏场景的空间下，层次结构的叶子节点粒度是可以由开发人员预定义的，基于场景的规模和数据的规模找到合适的粒度；同时，在玩家测试数据维度有限的背景下，维度灾难的弊端并不会显现。

另一方面，对节点内数据操作中融入密度聚类的思想，通过分而治之的方式，可以补充结点内统计数据的短板，在不同的辨率下进行数据聚类操作。在后续小节，我们将对改进后的算法进行详细的论述。

#### MRGTree算法流程

改进后的网格算法在本文中称作多分辨率网格树，这里简称MRGTree， MRGTree作为系统内数据挖掘的核心算法，整体流程如图 5.1所示：

##### 图 5.1 MRGTree聚类算法流程图

#### 构建层次结构

在3.2小节中，设计人员可为系统粗略勾绘出场景的可行走区域，从而粗略得出可行走区域所占整体场景区域的百分比。将所有体验玩家的数据进行统计，得出总体玩家收集到的游戏内空间数据点总量。然后通过计算，令有效叶子节点所包含的数据点个数可以达到阈值，从而得到网格层次结构的深度，作为给用户推荐的层次数。利用层次数和场景的最小包围盒矩形参数来构建层次结构。即此层次数应满足如下方程组，其中L为叶子节点的边长，W为场景的包围盒边长，R为可行走区域的百分比，N为空间位置的数据个数，为叶子结点包含数据的最大阈值，为叶子结点包含数据的最小阈值，d为层级数：

(10)

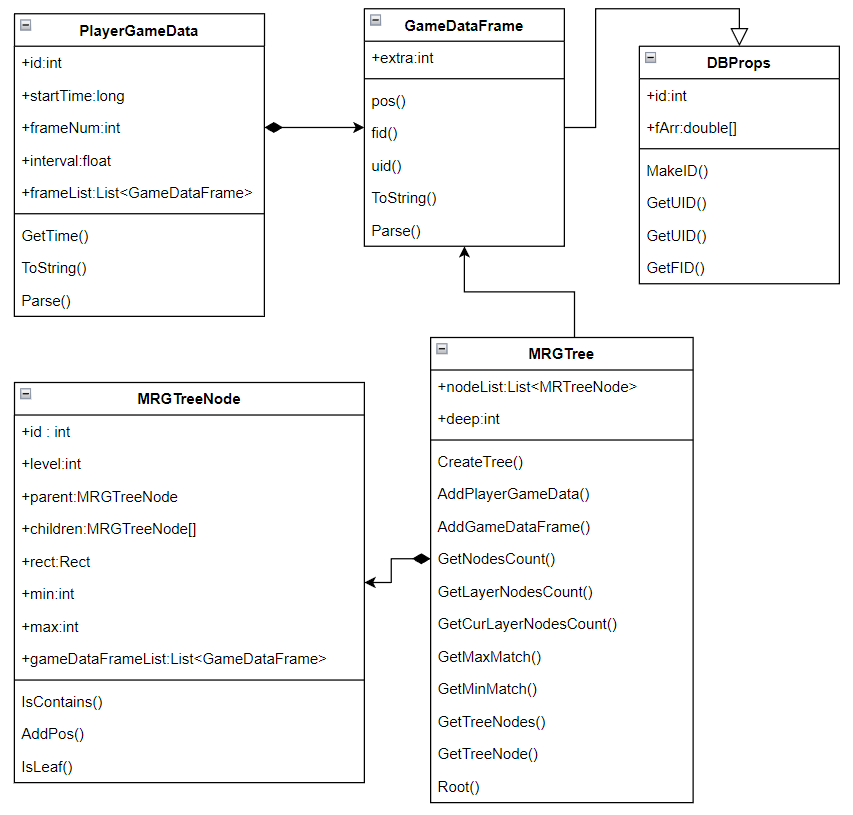
在构建层次结构时，此层次结构中的结点内不但需要保存其父节点和子节点的引用，根据需要，还可以保存其在空间中相邻“兄弟”结点的引用。以便在之后的查找运算中提升速度。创建层次结构的伪代码如算法2所示：

|  |
| --- |
| **算法2** MRGTree创建层次结构算法伪代码 |
| //创建nodeList作为结点容器  nodeList = new List<MRGTreeNode>(GetNodesCount(level))  Counter = 0;  //实例化结点，插入至容器中  For i = 0 To deep  For k = 0 To Math.Pow(4,i)  nodeList.Add(new MRGTreeNode(counter++,i));  End For  End For  //遍历nodeList中的所有结点，为每个结点创建链接关系和区域Rect  Foreach node in nodeList  start\_idx = GetChildStartIdx(node);  node.children = nodeList[start\_idx++]  node.children.parent = node;  …… //添加其他的关系链接  node.children.rect = new Rect()  End Foreach |

#### 玩家数据填充

将数据填充到层次结构中。即对此结构做数据插入操作。对于不同的功能，我们将数据分为两个类型，用于玩家运动路径和运动行为的时空数据的插入操作，其二是玩家在时空内的生理数据的插入操作。

1. 将采集到的玩家在游戏中的时空数据，重新解析为数据帧对象GameDataFrame，将此数据帧插入至与玩家运动相关的MRGTree中。数据帧对象GameDataFrame的成员和方法如下图 5.2所示：



##### 图 5.2 MRGTree结构类示意图

因为游戏测试规模的局限性，所以可以基于玩家的UserID和FrameID通过移位操作成为唯一的整形ID。数据帧中，玩家的空间位置、心电数据的RR间隔等填充到结构中的属性数组中，可在数据挖掘中作为判定相似性（匹配度）的衡量值。

1. 需要特殊处理的是将生理数据插入至数据帧对象，与时空数据不同，生理 数据中的FrameID即为ECG\_R\_Feature 对象在ECGFeatureData中的索引值，RR填入属性数据的固定放置额外数据的位置。其空间数据，是需要用ECG\_R\_Feature中保存的时间戳，基于玩家的时空数据进行插值后得到空间(位置)数据。最终将保存着生理信息的数据帧插入至与玩家生理相关的MRGTree中。

数据帧会调用MRGTree根节点提供的方法进行插入操作。在插入的过程中，通过对层次结构的先序遍历，将数据帧按照空间位置，逐级插入到满足网格区域矩形包含测试的结点中（也可以增加其它满足条件）。在遍历完成从子节点回溯时，可对结点匹配数据帧的个数进行统计，从而可以得到结点的匹配数目、子级的最大匹配数和最小匹配数。其中，匹配程度即为区域的访问热度值。将数据插入至MRGTree的伪代码如算法3所示：

|  |
| --- |
| **算法3** MRGTree插入数据算法伪代码 |
| //创建nodeList作为结点容器  //判定是否满足插入条件  if (IsContains(GameDataFrame.pos)  && PlayerFilter.IsMatch(GameDataFrame.uid)  && ……)  //将数据保存至节点容器内  GameDataFrameList.Add(GameDataFrame);  addCallBack(idx);  //将数据插入至其叶子节点内  children[].AddPos(GameDataFrame, addCallBack);  if(!IsLeaf())  //插入操作结束，回溯时计算网格的统计信息  min = CalcMin();  max = CalcMax();  ……  End if  End if |

#### 聚类结点数据

只通过对MRGTree是进行结点匹配计数的统计，能在宏观上认识整个场景的访问热度，对于游戏设计者而言，还缺少发现数据中隐含的关系。根据游戏数据的特定进行聚类操作。则能够挖掘出数据中包含的行为和关系。

针对用户指定的网格节点，对它们中的数据可以进行密度聚类操作，这样不仅让用户有选择的聚焦场景中的特定区域内容，而且减少了密度聚类法处理数据量。会大大提高聚类速度。系统为用户预设了两种的相似性衡量函数：欧式距离和加权的曼哈顿距离。

在相同的游戏测试项目中，玩家游戏数据的采样间隔是固定的，如果以时间差作为衡量相似度的其中一个参数变量，我们可以利用帧序列来代替时间序列，数据帧的差值即为时间差值。

此密度聚类法基于时空数据，且利用加权的曼哈顿距离来量化相似性，则可以反映出玩家的驻足行为；如想反应RR数据的变化率作为数据特性，可利用欧拉距离来量化相似性，此聚类可以反映出玩家在此节点所覆盖空间范围内的紧张情绪的程度。

#### 调用层次结构方法

在系统使用此层次结构时，会调用层次结构的公共方法，同时这些接口也便于为之后可视化模块进行扩展。

1. 支持在插入GameDataFrame时，网格对数据帧的包含测试可以设置玩家过滤器和帧过滤器等。
2. 支持对层次结构MRGTree当前层所有网格进行遍历。以层级深度传参，返回当前层所有网格的列表。
3. 支持对层次结构MRGTree当前层进行最值聚类查询，返回最值大小并最值聚类所处当前层的网格节点
4. 支持对层次结构MRGTree的特定网格进行聚类查询，同时返回该网格内的聚类计数，以及每个聚类包含的数据帧个数。
5. 支持空间位置到网格ID映射操作，以层级数做限制条件。

对层次结构MRGTree进行遍历和查询是非常重要的，但是，更新操作是不被公开允许的。因考虑到玩家数据对此系统模块而言，仅有只读的权限这一界定更符合模块的特有功能和架构意图。

### 可视化

可视化作为数据信息和用户之间的纽带，使得原本是多元且复杂数据背后的特征及变化，可以更直观的通过视觉语言传递给用户。上一小节我们对系统使用的数据挖掘算法进行了论述，即完成了从无序的原始数据到有序的特征数据的转化。本小节则会结合系统内用户交互与数据可视化进行了更详细的设计，同时，对相关可视化技术的实现进行了论述。

#### 基于场景区域的特征数据静态可视化功能

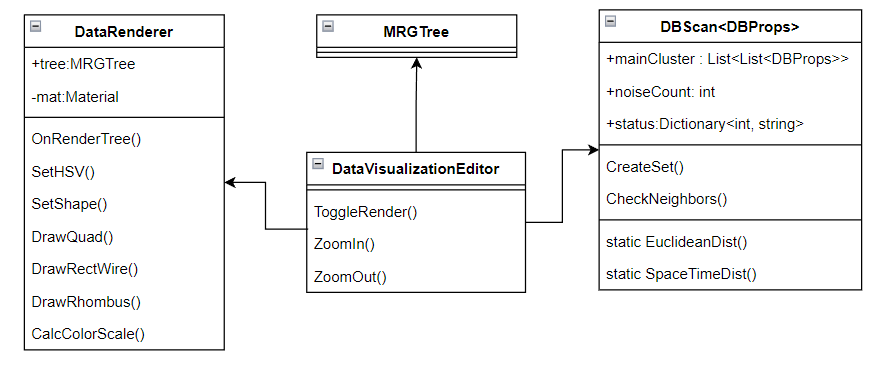
基于数据属性的不同，将数据的可视化表现分为静态和动态两种呈现方式。本小节对静态数据的信息内容分为了三个部分：对多元数进行结合绑定后进行可视化展示，对游戏事件触发信息在场景内的可视化展示，以及玩家面部表情摘要的可视化展示。

* + - 1. 多元数据的可视化功能的设计和实现

对多元数据进行结合绑定后，将玩家的路径空间数据、空间-心率数据组，可结合场景空间将数据以热力图的形式进行表现。另一方面，利用5.2小节所示的数据挖掘方法，可以将具有玩家特定行为，或者特定情绪的区域在网格上进行标记展示。

系统在可视化的表现方面，将可量化的游戏数据映射在可定量的视觉编码上[49][52]，例如颜色明度，空间位置，图形形状大小。映射方法由系统内部编码决定，而定性的视觉编码交给用户进行自定义操作，对图元的形状，颜色的色相，让用户选择合适自己理解和抽象的几何形状，或是适合自己的视觉效果。界面交互效果如图：

本章所涉模块在设计之初已经将数据的聚类和可视化进行了解耦，处理后的结构化数据更易被可视化渲染模块使用。我们此小节的功能类架构实现如下图 5.3所示：

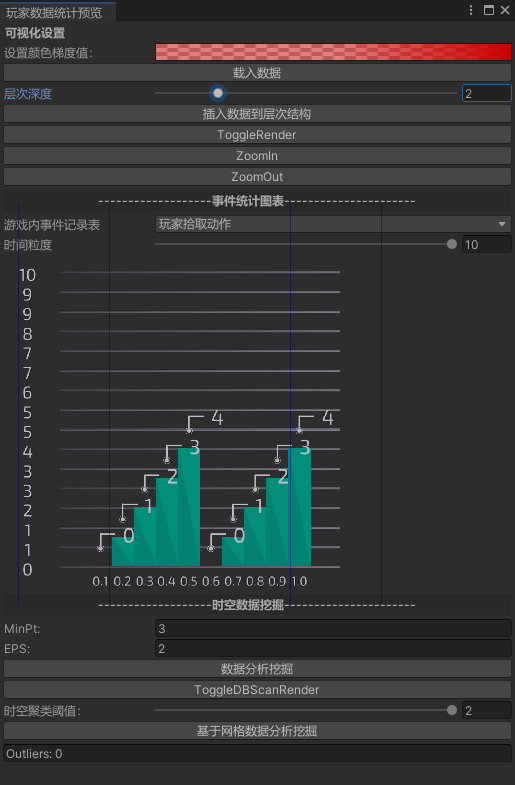


##### 图 5.3多元数据的可视化功能类示意图

上文已提到，功能模块的开发是符合MVC设计模式进行的。此类图中，DataRenderer负责对数据进行渲染， DataVisualizationEditor是作为用户交互的UI界面，来负责对渲染进行控制，数据的来源均来自于5.2小节的经MRGTree处理后的数据。其中DataRenderer借助Unity自带的渲染API，将数据可视化叠加于场景的表现上。在视觉表现上利用无光照材质，通过渲染管线向材质着色器的参数位传递游戏数据映射后的量值。

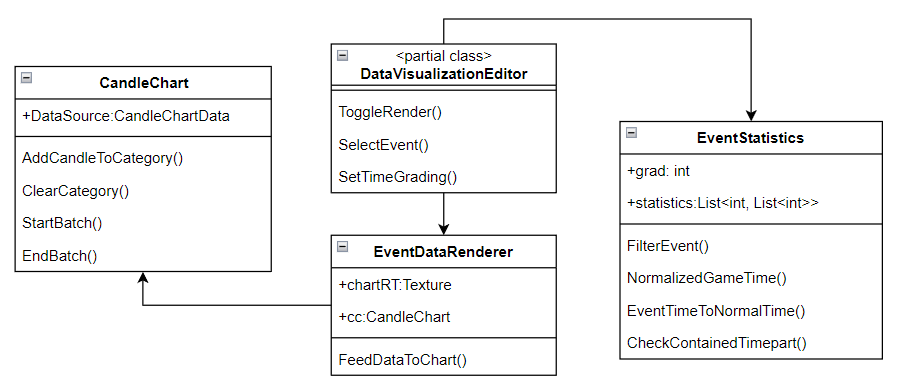
* + - 1. 事件信息可视化功能的设计和实现

事件信息的可视化展示需要载入参与测试的所有玩家的游戏数据，系统会将批量载入后的玩家数据进行过滤，将结果以静态柱状图呈现。用户的交互界面如图 5.4所示：



##### 图 5.4 数据挖掘及可视化相关的用户交互界面

在界面中用户通过选择加载事件数据，等待数据加载完成后，可以通过事件过滤器将特定事件提取，事件提取后进行快速的统计并绘制在界面上。柱状图的X轴为单位化后的游戏时间，Y轴为该事件触发的次数。的此功能实现的类架构如图 5.4所示：



##### 图 5.5 事件信息可视化功能类示意图

其中，类DataVisualizationEditor用于负责用户交互的相关控制操作，EventStatistics类会以玩家ID索引遍历出所有玩家的事件数据，此时将玩家的游戏开始和结束时间记录，同时将用户指定的事件过滤出，即可对此事件时间进行单位化。接下来，对单位化后的游戏时间进行粒度的细分，将游戏时常的0至1的区间内分为N阶粒度，将命中粒度区间内的单位化事件时间进行统计。并最终以柱状图渲染呈现。

* + - 1. 玩家面部表情摘要可视化功能的设计和实现

面部摘要的可视化功能主要是为了辅助用户来识别玩家的情绪状态。系统会在玩家指定的MRGTree的网格内，遍历网格内的聚类信息，以聚类为组，将聚类内的玩家UserID插入到集合中，并在其中每位玩家在聚类中的时常范围内取得中间的时间值，从对应捕捉的玩家面部视频中截取所有玩家的面部图像，并渲染在用户交互窗口界面中。游戏的伪代码如下：

|  |
| --- |
| 在用户指定网格内提取玩家的表情摘要伪代码： |
| //创建  nodeId = ScreenToNodeIdx(sx,sy);  node = MRGTree.GetTreeNode(nodeId);  clusters = dbScan.CreateSet(node.gameDataFrameList);  foreach cluster in clusters  foreach dataFrame in cluster.gameDataFrameList  //为玩家集合中的fid(FrameID)列表中添加fid  userset[dataFrame.uid].add(dataFrame.fid)  end foreach  foreach userData in userset  //取得该聚类中，当前玩家的平均fid  userData.fid = userData.AverageFID();  video = GetUserFaceVideo();  time = GetUserTimeByFID(userData.uid, userData.fid);  //找到此玩家fid对应的视频帧  frame = GetVideoFrame(video,time)  faceBitmap = GetFaceRegion(CascadeClassifier,frame)  //将人脸区域插入基于聚类的组别中  group[cluster.id].add(faceBitmap)  end foreach  end foreach |

#### 多路径动态可视化功能设计及实现

对于时序性数据来说，数据信息随时间不断变化，信息量也逐渐递增，就需要合理的手段以有效地展示数据中的信息。对单独的玩家而言，将多元数据中的时间部分、空间部分、心跳数据等进行拆分后，在时间轴上进行对空间位置、心跳值进行重组，利用折线图表展示，也是可行的。但是，如能用动画的可视化效果反应时序的先后性是相对更直接明了，同时将多玩家的运动路径统一进行动态的展示，有趣且富有挑战。

遍历场景中当前层级的所有网格，在对每一个网格内部的GameDataFrame数据列表进行遍历。并按照数据帧的帧序，在符合帧位置的网格ID列表中插入该网格ID，该容器的容量最大值应为参与测试的玩家个数。此时，构建出稀疏矩阵的数据结构，示例如图 5.5：

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 帧ID | 网格ID列表 | | | | | |
| 1 | 3 | 4 | 4 | 3 | 4 | 4 |
| 2 | 4 | 4 | 5 | 5 | 5 | 4 |
| 3 | 5 | 6 | 4 | 6 | 7 |  |
| 4 | 64 | 5 | 7 | 64 |  |  |
| 5 | 92 | 7 |  |  |  |  |
| … | … |  |  |  |  |  |
| 996 | 128 | 36 |  |  |  |  |

##### 图 5.6 因为每位玩家的游戏时常不同，所以呈稀疏矩阵

获得 “帧-网格”矩阵后，就可以根据帧序列的先后顺序播放网格路径了。当帧播放到该网格时，该网格呈现激活态，且网格的颜色透明度，随着触发网格的次数进行累加。一般的，系统会为网格颜色的alpha的累计值，按照公式对进行计算：

(11)

其中，为透明度每次累加值；为网格最终的透明度值，默认为255；为网格颜色初始的透明度值，默认为0；N为渐变次数，默认为参与测试的玩家数量，n为此网格在当前帧被激活的次数，为当前网格最值的透明度值。其多路径动画算法伪代码如算法4所示：

|  |
| --- |
| **算法4** 多路径动画算法伪代码： |
| //创建帧的稀疏数组  foreach node in layer  foreach dataframe in node  sparseMatrix[dataframe.fid].add(node)  end foreach  end foreach  //路径动画播放，在引擎API的Render中调用  timer -= deltaTime  if(timer < 0)  timer = frameIntervalTime;  curFrame++;  end if  //将初始alpha值填入alphaList  alphaList.Fill(alphaStart)  foreach node in sparseMatrix[curFrame]  alphaList[node.id] += deltaAlpha  renderer.draw(node,rgb,Min(alphaDest,alphaList[node.id]);  end foreach  alphaList.Clear(); |

再对动画播放控制，利用最基本的播放和停止功能进行控制，即可满足当前多路径动画展示的交互功能。在用户的UI上，提供了对网格激活态颜色透明度的颜色值、起始值、结束值以及渐变次数等。

### 本章小结

本章节介绍了四种常用的聚类方法，并对其实现方法以及优缺点进行了论述，从而引出本文可视化系统中的聚类算法，对此算法展开了详细的讨论，分别从顶层设计、创建层次结构MRGTree、插入玩家数据、使用MRGTree以及对MRGTree节点内数据的聚类方法。在可视化表现技术上，通过利用经MRGTree组织整理后的数据，对不同属性的数据类型，进行相应的可视化呈现。设计并实现了相应的可视化的表达与交互方法。

## 系统的实验和测试

为了验证游戏可视化辅助系统的有效性和功能性是否满足实际辅助游戏测试的需求，记录测试玩家的信息和数据，分析玩家的行为，并运用可视化技术结合场景展示玩家的路径轨迹，运动状态，情绪变化等。真正可以达到辅助游戏设计人员测试且评估对游戏设计内容。接下来将会以自定义设计的2D平台动作游戏作为实验游戏，来介绍系统的实验和测试。

### 实验材料

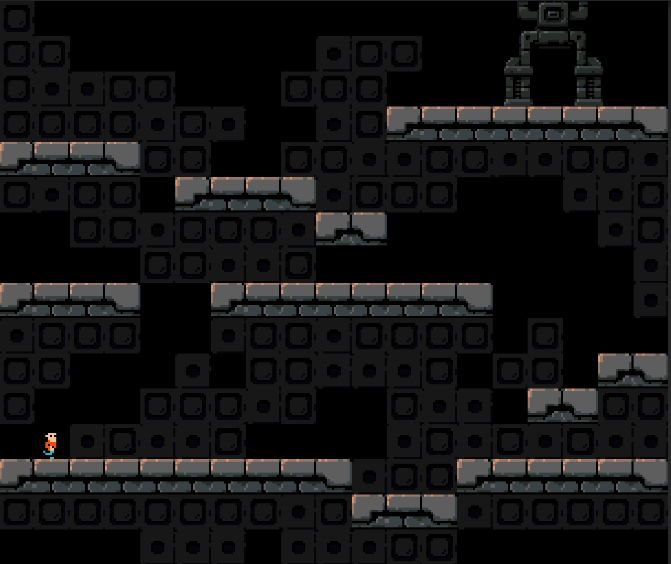
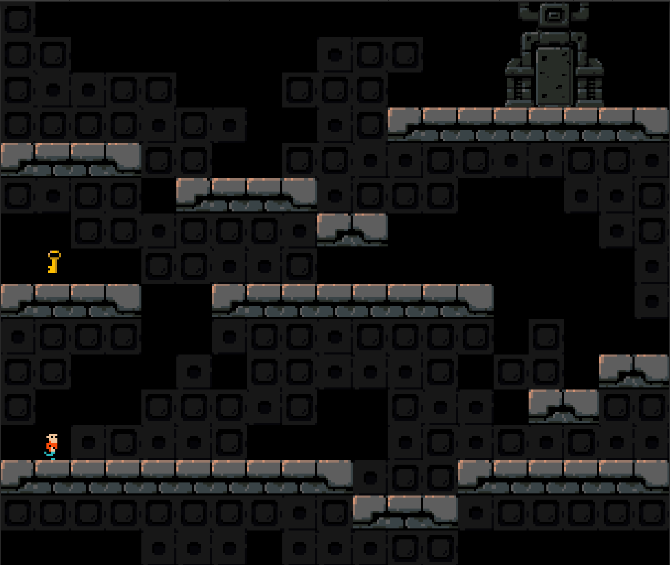
我们自定义一款2D平台动作游戏作为实验材料，在同样的场景空间中，对游戏中添加额外元素来改变场景，以作为四个对照实验的场景，利用系统中路径轨迹可视化，路径轨迹动态可视化，玩家情绪变化的可视化，玩家行为分析这四个功能，辅助设计者对游戏设计进行评估。

游戏场景的玩法很简单，玩家需控制角色消耗在最短的时间内进入到目的地即可成功，且角色在游戏中不会死亡。游戏内场景资源的语义是可以被玩家识别的，资源效果如表 6.1所示：

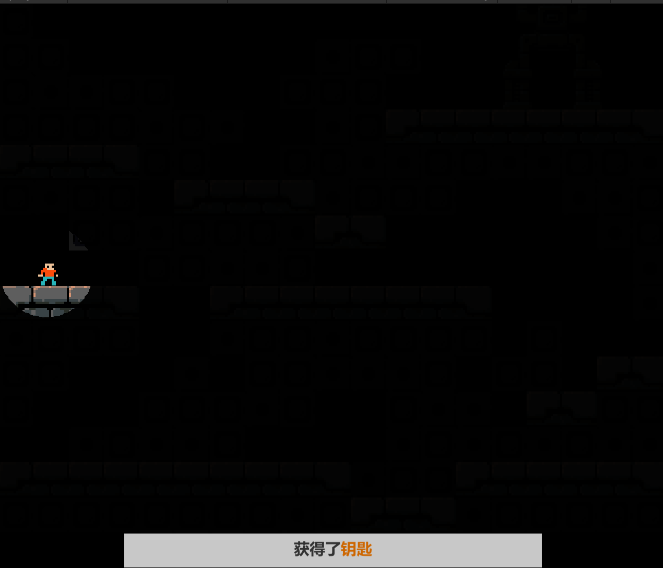
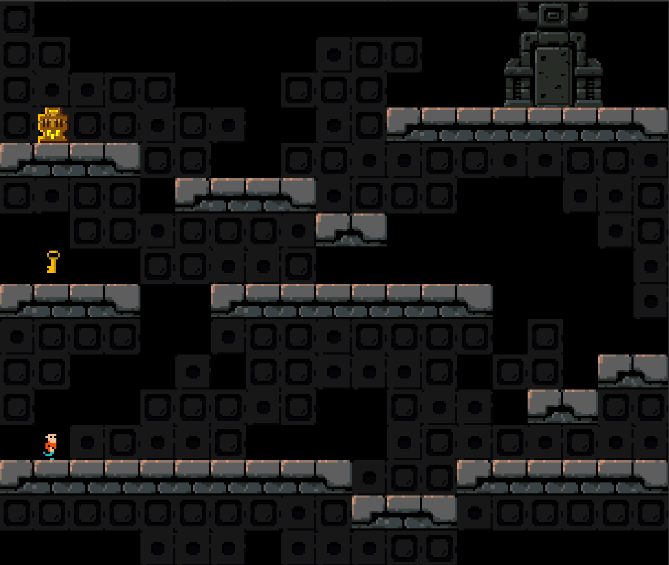
##### 表 6.1 游戏中美术资源汇总

|  |  |  |  |
| --- | --- | --- | --- |
| 前景平台资源 | 背景资源 | 目的地大门(敞开) | 目的地大门(关闭) |
|  |  |  |  |
| 钥匙 | 装饰物 | 角色资源  （动态） |  |
|  |  |  |  |

四个场景关卡的设计如图 6.1所示：

(a) (b)

(c) (d)

##### 图 6.1 (a)为测试场景a的关卡示意图，目的地的大门呈打开状态。(b)为测试场景b的关卡示意图，目的地的大门呈关闭状态，玩家在拾取钥匙后，大门会变为打开状态。(c)为测试场景c在钥匙被拾取后的示意图，关卡设计和任务流程与测试场景b相同。(d)为测试场景d的关卡示意图，在钥匙上方的平台上，放置了场景装饰物。

场景a：作为基础的游戏场景，玩家仅可以通过移动和跳跃来操纵玩家，寻找到目的地，即敞开着的大门, 进入大门后则玩家成功过关。利用此场景测试系统对玩家路径轨迹可视化功能。

场景b：在场景a的基础上，添加了一把钥匙，并将之前敞开状态的大门替换为关闭状态的大门。玩家的游戏目的不变，但需要通过设计者的暗示，找到钥匙并拾取它，然后到达大门前使用钥匙，则可打开大门，游戏成功结束。利用此场景测试路径轨迹动态的可视化功能。

场景c：相比于其游戏中的愉悦感和幸福感，恐怖感会更容易的调动玩家的情绪，使玩家的体验时的情感变化更加强烈[57]。在创作理论和技巧上，有学者指出恐怖游戏中音频感知对玩家体验的影响较为强烈[58]。同时，利用压迫感觉来给营造恐怖氛围[56]。我们在此场景的设计时，基于场景b的完整游戏流程，仅在玩家拾取钥匙后突然播放恐怖音效，同时玩家的视野变小，以此来创造恐怖气氛。利用此场景测试系统对玩家情绪变化的可视化功能。

场景d：在场景b的基础上，将钥匙的上方平台增加场景装饰物，装饰物的颜色突兀，传达着可以与角色交互的语义，但实际上，装饰物没有任何的交互功能。以此来模拟设计人员设计缺陷所导致给玩家传递错误的语义。

### 实验设计与安排

我们对10个参与者进行了实验，其中男生7人，女生3人，年龄分布在25至35岁之间。通过前期的游戏经验调查问卷得知，他们都对电子游戏感兴趣而且经常接触，同时我们自定义的游戏也属于2D平台动作游戏，我们在问卷中也调研了参与者对于2D平台动作游戏的经验情况，其中？？？%的人从不接触平台动作游戏，？？？%的参与者偶尔玩平台动作游戏，剩余？？？%的参与者经常平台动作游戏。

游戏测试前，测试工作人员，需要基于此测试游戏的场景对系统进行配置，之后给玩家安装好心电传感器，开启摄像头，使其面部保持在视频内主要位置，并告知玩家游戏规则和操作方式。硬件测试环境同2.4章节中的开发环境保持相同，同时让玩家身处在安静且不被打扰的外界环境中进行游戏体验。玩家在听从测试工作人员的指令下，依次对游戏中的场景a，场景b，场景c，场景d依次进行试玩体验。

当全部玩家完成四个场景的体验后，将可视化呈现效果与设计意图进行对比，以此让设计者对游戏设计是否达到预期做评估。

### 实验结果分析

待所有参与测试的玩家完成游戏体验后，我们将收集的游戏数据和生理数据载入系统，进行数据分析和可视化展示。

1. 场景a的实验结果：系统使用多元数据的可视化功能，静态的呈现玩家的探索场景的路径轨迹和访问热度。与设计者的设计意图的对比结果如图：

图？：此处有图

讨论：

1. 场景b的实验结果：系统使用路径轨迹动态可视化功能，动态的呈现玩家的探索场景的路径轨迹，与设计者的设计意图的对比结果如图：

图？：此处有图

讨论：

1. 场景c的实验结果：系统使用多元数据的可视化功能，静态的呈现玩家的运动轨迹和心跳变化，设计者期望在玩家获取钥匙后营造的恐怖气氛会对玩家达到预期的恐怖效果，结果如图所示：

图？：此处有图

讨论：

1. 场景d的实验结果：系统使用多元数据的可视化功能，以及事件信息可视化功能，静态的呈现玩家的驻足行为（不可交互宝箱，会使得玩家在此处逗留进行多次尝试）。同时，每当玩家按下拾取按键，事件进行一次激活计数增加1。设计者的设计意图是不希望玩家在装饰物位置处驻足，因为装饰物的真实情况是“设计上的缺陷”。换句话说，设计者希望通过此测试辅助系统捕捉到游戏设计上的缺陷。结果如图所示：

图？：此处有图

讨论：

### 本章小结

本章对游戏可视化测评辅助系统的测试工作进行了阐述。基于玩家体验2D平台动作游戏的测试活动，利用该系统对此测试活动进行可视化辅助。并通过四个具有相同玩法且不同设计的游戏场景，对系统的功能性和有效性进行了实验测试。经过测试，系统在辅助游戏测试的活动中，可达到测试团队和游戏设计者的需求。

## 结论

本文设计并实现了一种利用数据挖掘和可视化技术对游戏测试进行辅助的系统。从游戏测试的实际需求出发，结合游戏测试以及所涉及生理心理学、游戏数据可视化的研究和发展，论述了游戏测试辅助系统的意义及重要性。首先通过对系统业务的了解，对系统的需求进行全面的分析，设计并架构了系统的总体框架和功能模块。然后，对系统内各个功能模的类设计，结构设计和应用流程上，进行了更详细的设计和实现细节。最后，基于玩家在自定义的一款2D平台动作游戏的测试活动，对该系统的功能性和有效性上进行了实验测试，保证了系统在功能性完整，有效性上符合测试团队和设计团队的要求。

本论文主要完成了如下几点核心工作内容：

1. 系统整合了游戏beta测试阶段的工作流程，提高测试团队进行测试活动的效率，高效稳定的完成对游戏设计开展测试的工作内容。
2. 针对传统折现图表现游戏数据能力的局限性，系统将结合游戏场景对玩家的数据进行可视化表现，同时根据数据类型，是否有时序关系，在表现上会以静态和动态两种模式呈现。此功能可以让设计者对场景的空间规划和任务引导等设计内容进行评估。
3. 玩家在测试时的情绪变化可以利用该系统，通过采集到的生理数据，反应在场景空间内。可以让设计者对玩家的情绪调动进行评估。
4. 玩家在数据可视化模块中，可以对数据进行重组和聚类，从而反应出某些特定的玩家行为。多分辨率的统计网格树，自定义的可视化视觉编码，使用户以适合的方式聚焦于所关注的场景区域和数据。
5. 系统的模块架构上本着高内聚低耦合易扩展的思想，使得在生理采集设备上，数据处理上可以更方便的进行二次开发和扩展。同时，可视化模块的渲染接口可以根据游戏开发时项目采用的不同渲染引擎进行对接。

综上所述，本文设计和实现的辅助游戏测试的可视化系统，在功能性和有效性方面达到了预期目标，满足了测试团队和游戏设计团队的需求。但系统仍然存在着有待进一步提升和改进的地方：

1. 美化系统界面。在系统的实践过程中，良好的视觉效果能够有效增强用户体验。因此，可提升系统的交互界面美观程度。同时在可视化的视觉效果方面，也可进行额外的点缀。
2. 玩家的情感计算。针对玩家在游戏体验中的面部视频，对其进行自动化的情感计算，则可以为玩家情绪的识别和变化上提供更全面的数据支撑。

# 参 考 文 献

1. 柯文. 今年上半年我国移动游戏行业收入达1147.72亿元[N]. 人民邮电,2021-08-03(006).
2. Serious Games : An Overview[J]. Institutionen Fr Kommunikation Och Information, 2007.
3. Selinker M , Howell D , Tidball J . Kobold Guide to Board Game Design[J]. 2012.
4. Mirza-Babaei P, Wallner G, McAllister G, et al. Unified visualization of quantitative and qualitative playtesting data[M]//CHI'14 Extended Abstracts on Human Factors in Computing Systems. 2014: 1363-1368.Schultz C P, Bryant R D. Game testing: All in one[M]. Stylus Publishing, LLC, 2016.
5. Game user experience evaluation[M]. Springer, 2015.
6. 斯蒂尔. 数据可视化之美[M]. 机械工业出版社, 2011
7. 程文华. 基于机器学习的手机游戏软件自动化测试研究[D]. 浙江大学, 2014.
8. 康怡琳. 远程游戏自动化测试系统的设计与实现[D]. 华中科技大学, 2006.
9. 王颐. 基于图形用户界面的软件测试技术及其在儿童益智游戏产品中的应用研究[D]. 重庆大学.
10. 王志森. 探索式测试方法在网络游戏软件测试中的应用[D]. 上海交通大学, 2011.
11. 赵洋艺. 某游戏发行公司移动数据统计分析系统的设计与实现[D]. 陕西:西安电子科技大学,2020.
12. Cooper S, Sarkar A. Pathfinding Agents for Platformer Level Repair[C]//AIIDE Workshops. 2020.
13. Agarwal S. Data mining: Data mining concepts and techniques[C]//2013 international conference on machine intelligence and research advancement. IEEE, 2013: 203-207.
14. Paranthaman P K, Cooper S. ARAPID: towards integrating crowdsourced playtesting into the game development environment[C]//Proceedings of the Annual Symposium on Computer-Human Interaction in Play. 2019: 121-133.
15. Mirza-Babaei P, Nacke L E, Gregory J, et al. How does it play better? exploring user testing and biometric storyboards in games user research[C]//Proceedings of the SIGCHI conference on human factors in computing systems. 2013: 1499-1508.
16. Bauer A W, Cooper S, Popovic Z. Automated redesign of local playspace properties[J]. FDG, 2013, 13: 190-197.
17. 陈月芬, 崔跃利, 王三秀. 基于生理信号的情感识别技术综述[J]. 系统仿真技术, 2017, 13(1):5.
18. 王朝光, 于积理. 电子游戏体验的生理测量[C]// 心理学与社会和谐学术会议. 2013.
19. 陈继华, 李岚, 钱坤喜. 基于多生理信号的情绪初步识别[J]. 生物医学工程研究, 2006(3):6.
20. 张乐凯. 基于生理信号数据的产品设计与用户体验研究[D]. 浙江大学, 2018.
21. Csikszentmihalyi M . Flow: The Psychology of Optimal Experience[J]. Design Issues, 1991, 8(1).
22. 姜峰. 玩家心流数据分析及建模[D].电子科技学,2021.
23. Ismail, Muhammad, Qaraqe, et al. Epileptic seizure onset detection based on EEG and ECG data fusion[J]. Epilepsy & behavior: E&B, 2016.
24. Čertický, Martin, Čertický, Michal, Sinčák, Peter, et al. Psychophysiological Indicators for Modeling User Experience in Interactive Digital Entertainment[J]. Sensors, 2019, 19(5).
25. Physiological correlates of the flow experience during computer game playing.[J]. International Journal of Psychophysiology, 2015, 97(1):1-7.
26. Georgios, N, Yannakakis, et al. Entertainment capture through heart rate activity in physical interactive playgrounds[J]. User Modeling&User-Adapted Interaction, 2008.
27. Drachen A, Thurau C, Togelius J, et al. Game data mining[M]//Game analytics. Springer, London, 2013: 205-253..
28. 刘用飞. 基于标注与可视化的游戏AI效果反馈系统的设计与实现[D]. 哈尔滨工业大学.
29. 张宏鑫, 盛风帆, 徐沛原,等. 基于移动终端日志数据的人群特征可视化[J]. 软件学报, 2016(5):14.
30. 张彦俊. 游戏运营中的数据挖掘[D]. 复旦大学.
31. 兰吉, 巫英才. 游戏数据的可视分析[J]. 计算机辅助设计与图形学学报, 2017, 29(4):8.
32. Drachen A, Schubert M. Spatial game analytics and visualization[C]//2013 IEEE Conference on Computational Inteligence in Games (CIG). IEEE, 2013: 1-8.
33. Medler B, Magerko B. Analytics of play: Using information visualization and gameplay practices for visualizing video game data[J]. Parsons Journal for Information Mapping, 2011, 3(1): 1-12.
34. Wallner G, Halabi N, Mirza-Babaei P. Aggregated visualization of playtesting data[C]//Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems. 2019: 1-12.
35. 马小东，任芃锟，赵凡．起止点数据可视分析研究[J/OL]．图学学报. https://kns.cnki.net/kcms/detail/10.1034.T.20210928.1321.002.html
36. 陆妍玲, 韦晶闪, 赵雨萌,等. 提取热点区域的时空轨迹数据聚类分析[J]. 数学的实践与认识, 2021, 51(13):10.
37. 曾浩然. 基于ECG信号的心律失常识别研究[D]. 电子科技大学, 2019.
38. 刘乐乐. 心电信号去噪及特征波检测算法研究[D]. 重庆邮电大学, 2018.
39. 吴佳全. 基于神经网络的ECG分类算法及高能效架构研究. 浙江大学.
40. 宋佳, 蔡峰权, 顾天晴,等. 人脸情绪识别研究综述[J]. 现代计算机, 2021(22):8.
41. 赵羿欧, 刘扬. 一种改进的差分阈值心电检测算法[J]. 计算机工程, 2011(S1):3.
42. 顾秀秀, 朱明亮, 王璐,等. 基于小波变换的心电信号特征提取[J]. 软件导刊, 2021, 20(5):5.
43. 董红生. 心电波形检测与心率变异性分析方法研究[D]. 兰州理工大学.
44. Nancy Betancourt and Marco Flores-Calero and Carlos Almeida. ECG Denoising by using FIR and IIR Filtering Techniques[C]., 2019
45. Sadhukhan D, Mitra M. R-peak detection algorithm for ECG using double difference and RR interval processing[J]. Procedia Technology, 2012, 4: 873-877.
46. 彭自然,王国军.用小波变换对ECG信号进行去噪研究[J].信号处理,2017,33(08):1122-1131.
47. 纪汉霖, 李兆信. 多种聚类算法性能的比较分析[J]. 计算机技术与发展, 2020, 30(8):8.
48. 尚家泽, 安葳鹏, 郭耀丹. 基于阈值的BIRCH算法改进与分析[J]. 重庆邮电大学学报：自然科学版, 2020, 32(3):8.
49. 刘滨, 刘增杰, 刘宇, et al. 数据可视化研究综述[J]. 河北科技大学学报, 2021, 42(6):12.
50. 宋柱, 付锐, 张名芳,等. 分区基于密度的聚类算法在激光雷达行人检测系统中的应用[J]. 科学技术与工程, 2017, 17(18):6.
51. 高涛, 王钊, 雷正潮,等. 基于改进DBSCAN算法的汽车行驶热区提取方法[J]. 测控技术, 2017, 36(3):4.
52. 杨明欣. 数据可视化通用视觉语言分析研究[D].江南大学,2021.
53. Wang W , Yang J , Muntz R . STING: A statistical information grid approach to spatial data mining[C]// VLDB'97, Proceedings of 23rd International Conference on Very Large Data Bases, August 25-29, 1997, Athens, Greece. DBLP, 1997.
54. 李航. 统计学习方法[M]. 清华大学出版社, 2012.
55. 裴丹丹. 基于ECharts的数据可视化实现[D].北京邮电大学,2018..
56. Garner, Tom & Grimshaw-Aagaard, Mark & Nabi, Debbie. (2010). A preliminary ex-periment to assess the fear value of preselected sound parameters in a survival horror game. Games Computing and Creative Technologies: Conference Papers (Peer-Reviewed). 10.1145/1859799.1859809.
57. Pedro A. Nogueira, Vasco Torres, Rui Rodrigues, et al. Vanishing scares: biofeedback modulation of affective player experiences in a procedural horror game. 2016, 10(1):31-62.
58. Katja Rogers, Giovanni Ribeiro, Rina R. Wehbe, Michael Weber, and Lennart E. Nacke. 2018. Vanishing Importance: Studying Immersive Effects of Game Audio Per-ception on Player Experiences in Virtual Reality. In Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems, CHI 2018, Montreal, QC, Canada, April 21-26, 2018. 328.