

# Tailoring Horror Games by Integrating Bodily Sensors

Student Name: Steven Lowes

Supervisor Name: Magnus Bordewich

Submitted as part of the degree of BSc Computer Science to the  
Board of Examiners in the School of Engineering and Computing Sciences, Durham University  
2019-01-25

## *Abstract —*

**Context/Background** Horror games are a large industry, with thousands of games and hundreds of millions of copies owned. Players are thrilled by experiencing fear in a safe and secure environment. However, each player reacts differently and the one-size-fits-all approach can prove constricting for players.

**Aims** The project aims to explore how bodily sensors can be integrated into horror games to improve the user's experience by adjusting gameplay based on data readings. A secondary aim of the project is that jump-scares can be triggered without pre-existing knowledge of the environment that the player is in, or the stage of the play-through that they are at. This will make gameplay more dynamic and reduce the difficulty of creating new environments.

**Method** A simple horror game with the ability to read data from sensors will be created. The rate and timing of scares will be adjusted based on the sensor readings. Different ways of adjusting the rate and timing of scares will be explored and measured in a user study. The dynamic methods will be compared to 'static' methods which are not based on the sensor readings.

**Proposed Solution** The game Minecraft will be modified using the community-made 'Forge' modding api. The modifications will add a jump-scare which can be triggered programmatically. After a jump-scare is activated, the mod will read data from a galvanic skin response sensor and allow more time between scares if it believes that the user is becoming desensitized. This adaptive method will be compared to two 'static' methods - one which scares at random intervals, and one with scares at regular intervals.

**Keywords —** Galvanic Skin Response, Electrodermal Activity, Human-Computer Interaction, Video Games

## I INTRODUCTION

- Horror industry
  - Examine horror industry, size, especially gaming
  - Comment on lack of tailoring to the individual
  - Compare with tailored scare experiences
  - Maybe there is some research on this

Horror games are a large industry, with over 2000 unique titles on steam and almost 500 million copies owned /citehorrorSteamSpy

- Bodily Sensors

- The game minecraft
- Project Purpose
  - Do users enjoy playing a horror game more when its scares are tailored based on feedback from bodily sensors? Is galvanic skin response sufficient to understand a user's response to a scare?
- Deliverables
  1. Minimum Deliverables
    - Create an immersive game environment in which you can control events such as the arrival of new enemies and their location.
    - Create a system for tracking the user EDA measurements and game events simultaneously.
    - Create a standardised game setting for users to play through and record EDA measurements as they progress. Record some user experiences.
    - Determine what game events trigger responses and select events to use in next deliverables.
  2. Intermediate Deliverables
    - Analyse data from users and try to determine: susceptibility to expected new shock events, e.g. underlying tension or delay since last event.
    - Create one or more adaptive systems for triggering events at moments of maximum impact.
    - Create a (null hypothesis) system for random generation of events.
  3. Advanced Deliverables
    - Conduct a user study to determine whether the adaptive systems give a better user experience than the random system.
    - Revise the adaptive system based upon empirical evidence obtained.

## II DESIGN

### ***A Requirements***

- Functional Requirements
- Nonfunctional Requirements

### ***B Galvanic Skin Response***

- Explain the mechanism of GSR
- Response time, average curve + recovery
- Explain cause, size of effect that is needed to cause a response
- Talk about other uses
- What kind of things is it best at measuring

## ***C In-Game Environment***

The environment will be a haunted house. Users will be given the goal of finding a number of items throughout the house, but they will not be able to reasonably complete the task within the time. This just ensures that the users have a purpose and continue to explore the environment

## ***D Jump Scares***

Describe the jump scare briefly, show a screenshot from the game and explain the sound  
Show some GSR data from a jump scare?

### **D.1 Choice of Game**

The game Minecraft was chosen due to its comprehensive modding api, cross-platform compatibility, customisability, and ease of use when it comes to creating environments. The modding api, called 'forge' is community-maintained, but incredibly complete and mature, having originally been created x years ago (CITE).

### **D.2 System Architecture**

What arduino, what sensor

What code is the arduino running?

How is communication happening? (serial + jrxtx library)

Talk about data buffer, how gsr is interpreted, that the system just sets a timer until the next scare and it's not based on the environment whatsoever.

The language Kotlin was chosen due to my familiarity with it, and its interoperability with java, which Minecraft and its modding api use. Kotlin also allows code to be written more succinctly with less boilerplate to improve productivity and expressiveness.

## ***E User Study***

Talk about number of participants, length of participation, whether they play more than once

Success will be measured based on a combination of data from the sensors, and players self reported enjoyment + scaredness on a 10-point scale.

Each user will play the game once. Repeat playthroughs are not reasonable as it is expected that the game will become less scary after one playthrough. Each adaptive-method playthrough will be matched with some naive method playthroughs to control for the number of scares and total playtime. (Adaptive algorithm will have a differing number of scares each time, and for each adaptive run-through we'll do one of each naive method pre-set to have the same number of scares)

Playthroughs are anticipated to last around 10 minutes (will be predetermined) and contain around 10 scares (will be determined based on the adaptive algorithm).

## ***F Future Ideas***

Add heartbeat sensor

Add multiple kinds of scares, tailor to what scares individuals rather than just rate + timing, i.e. spiders for people scared on them

## ***G*** *References*

- 

## **References**