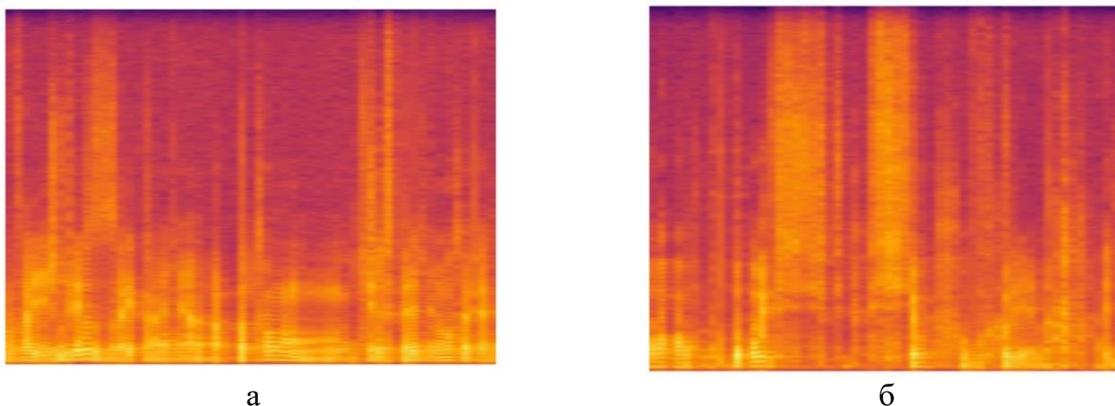


в данный момент времени аудиосигнала. Для отображения спектрограммы используется, *specshow*. Поскольку все действие происходит в нижней части спектра, можно преобразовать ось частот в логарифмическую.

Сравнивая полученные изображения с заиканием и без заикания, можно найти существенные отличия.

Однако, использования исключительно спектрограммы для последующего обучения нейронной сети недостаточно, требуется больше данных. Именно по этой причине необходимо использовать метод вычисления спектрального цетроидцетроида и центрального спада. В этом случае получится заполнить более точную картину.



а

б

а- без заикания, б- с заиканием

Рисунок 1. Спектрограмма аудиозаписи

Тем не менее, на основании уже имеющихся данных можно однозначно сказать, что в процессе разговора человек с речью без дефектов обладает демонстрирует совершенно иные темпо-ритмические показатели, нежели говорящий, обладающий клоническим или тоническим типом заикания, что наглядно отображается на спектрограммах.

Заключение

Подробный спектральный анализ аудиозаписи способен стать крайне эффективным методом распознавания тонического и клонического типов заикания. Полученные данные в последующем могут быть использованы для быстрого обнаружения фрагментов с представленным форматом речевой патологии.

Литература

1. Климов, Г. В. Теория приближений и спектральный анализ [Текст] / Г. В. Климов, Б. С. Михайлов, В. В. Чесноков. – СПб.: Лань, 2017. – 376 с.
2. Смирнов, И. С. Полная математика [Текст] / И. С. Смирнов. – М.: Колос, 2015. – 592 с.
3. Чернышев, С. В. Машинное обучение и анализ данных [Текст] / С. В. Чернышев. – М.: ДМК Пресс, 2018. – 360 с.

РАЗРАБОТКА ИГРОВОГО ПРИЛОЖЕНИЯ «ЕСНО-10» В ЖАНРЕ «КОНЕЧНЫЙ РАННЕР» С ИСПОЛЬЗОВАНИЕМ СОБСТВЕННОЙ ГРАФИКИ В СРЕДЕ UNITY

Овчинина О. В. (студент гр. ИТИ-42)

Гомельский государственный технический университет имени П.О.Сухого, Гомель, Республика Беларусь

Научный руководитель – **Захаренко Владимир Сергеевич**

(к.т.н., доцент кафедры «Информационные технологии» ГГТУ им. П.О. Сухого)

Аннотация: В данной работе рассматриваются основные проблемы при разработке игрового приложения «Echo-10» и их решения, такие как выбор средств разработки, архитектуры, процесс создания собственной графики, механик игры и сюжета.

Ключевые слова: игровое приложение, архитектура, графика, сюжет, разработка.

Введение

Игры в жанре «Раннер» давно укрепились в мире видеоигр. Они характеризуются процедурной генерацией уровней, созданием препятствий и системой подсчета очков. Поджанр «Конечный раннер» включает в себя наличие нескольких уровней и завершающий этап игры, что обеспечивает более детальное развитие сюжета игры.

Результаты и обсуждение

Средства разработки. Для разработки игры был выбран язык программирования C#, игровой движок Unity, интегрированная среда разработки Rider, графический редактор Adobe Photoshop для создания оригинальной графики и сервис по управлению проектами Notion. Благодаря кроссплатформенной среде разработки Unity приложение адаптируется как для персонального компьютера, так и для мобильного устройства.

Собственная графика. С помощью инструмента Adobe Photoshop была создана собственная графика для игрового приложения, с целью придания уникального и привлекательного визуального стиля, а также для создания неповторимой атмосферы в игре. Кат-сцена игры в качестве примера представлена на рис. 1.



Рис. 1. Кат-сцена игры

Архитектура. Архитектурные проблемы в играх, разработанных с использованием Unity – среды разработки компьютерных игр, включают в себя сложности с контролем состояний игровых объектов, а также зависимость от компонентов Unity, что может затруднить переносимость и интеграцию проекта. Неэффективное использование ресурсов и недостаточное разделение обязанностей также могут привести к проблемам производительности и сложностям в модификации проекта.

Чтобы решить эти проблемы была разработана собственная архитектура для приложения «Echo-10». Схема архитектуры представлена на рис. 2.

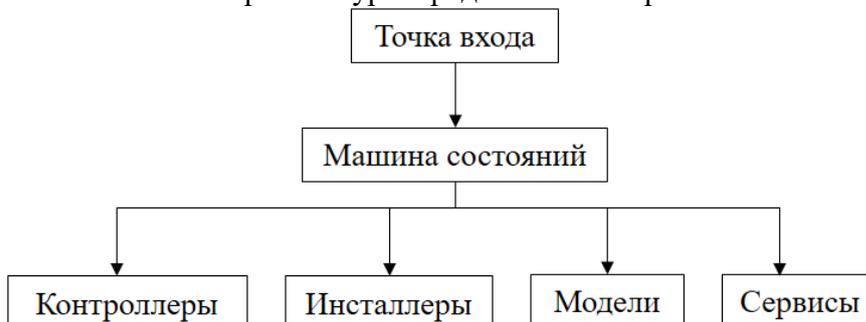


Рис. 2. Архитектура игрового приложения

Точка входа инициирует запуск игры и управляет её жизненным циклом. Она содержит методы, которые вызываются при старте игры, перезагрузке сцены и других событиях, обеспечивая основное управление приложением. Машина состояний определяет все возможные состояния игры, их изменения в ответ на события или условия, а также действия, выполняемые в каждом из состояний. Модели представляют собой ядро приложения, содержащее структуры данных, методы для доступа и обновления данных, а также

реализацию игровых механик. Инсталлеры – это конфигурационные точки, где происходит подготовка различных компонентов игры к работе. Контроллеры ответственны за управление машиной состояний. Они координируют переходы между различными состояниями игровых объектов или систем, а также контролируют выполнение действий, связанных с каждым состоянием. Сервисы обеспечивают функциональность, которую можно использовать повторно.

Сюжет. В приложении «Echo-10» игрок управляет Мигелем – летучей мышью, лишенной эхолокации и вынужденной полагаться на новые технологии для ориентации. Сюжет раскрывается по ходу игры в виде небольших кат-сцен перед прохождением уровня.

Механики. Игра состоит из нескольких уровней, где игрок должен избегать препятствий и добраться до конца. Препятствия не видны на экране до тех пор, пока игрок не использует эхолокационный заряд. При этом они на короткое время подсвечиваются, и игрок должен запомнить их расположение. Количество зарядов на уровень ограничено до 10. Игровой уровень представлен на рис. 3.



Рис. 3. Скриншот для игровой уровень

В процессе игры игрок собирает монеты в виде мошек, количество которых влияет на конечный исход игры.

Заключение

В результате была разработана архитектура и прототип игрового приложения «Echo-10» в жанре «Конечный раннер» с использованием собственной графики в среде Unity.

ИННОВАЦИОННЫЙ ИСКУССТВЕННЫЙ ИНТЕЛЛЕКТ В UNITY

Пешевич А.С. (студент гр. ИТИ-41)

Гомельский государственный университет имени П.О. Сухого, Гомель, Республика Беларусь

Научный руководитель – **Комракова Е.В.**

(ст. преп. кафедры «Информационные технологии» ГГТУ им. П.О. Сухого)

Аннотация: в данном докладе рассматриваются две инновационных платформы искусственного интеллекта – Unity Muse и Unity Sentis. Unity Muse предлагает обширные возможности для разработки игр и интерактивного контента, включая трехмерную графику, анимацию, физику и монетизацию проектов. С другой стороны, Unity Sentis предназначена для создания и интеграции интерактивного контента, включая виртуальную и дополненную реальность, игры и промышленное моделирование.

Ключевые слова: искусственный интеллект, AI, игры, разработка игр, Unity, Unity Muse, Unity Sentis.

Введение

В настоящее время многие компании активно используют AI для улучшения пользовательского опыта, оптимизации производства, обработки данных и других задач.

Цель работы и актуальность данной темы обусловлены стремительным развитием технологий искусственного интеллекта.