

Абакумов А.В.

к.т.н., доцент С.В. Еремеев

*Муромский институт (филиал) федерального государственного образовательного учреждения высшего образования «Владимирский государственный университет имени Александра Григорьевича и Николая Григорьевича Столетовых»
602264, г. Муром, Владимирская обл., ул. Орловская, 23*

Персистентный анализ растровых спутниковых снимков

В основе предлагаемого алгоритма лежит классическая персистентная обработка [1] растровых изображений, но с дополнительными условиями и обработкой.

Прежде всего стоит отметить, что анализ происходит на основании яркостей соседних пикселей, что делает обработку крайне чувствительной к мелким искажениям. Для решения этой проблемы предлагается сжать исходное изображение в два раза, и, таким образом, избавиться от лишних поперечных значений, которые обычно возникают на цифровых изображениях. Также подобный подход поможет ускорить обработку.

Далее происходит персистентный анализ. Стоит уточнить, что изображение переводится в полутоны.

Во время анализа алгоритм проходит по каждому пикселю на изображении в порядке уменьшения их яркости (от значения 255 до 0). Выбран именно такой порядок из-за того, что как правило крыши ярче фона, а в качестве результата ожидается, что они станут отдельными компонентами.

При анализе каждого пикселя смотрятся его соседи: если один из них является компонентой, то пиксель присоединяется к ней, иначе появляется новая компонента, что содержит анализируемый пиксель.

При присоединении пикселя происходит медианная проверка: для начала находится разница между яркостями первого добавленного пикселя в компоненту и последнего (благодаря линейному проходу они являются наибольшим и наименьшими яркостями). Эта разница делится на два, таким образом находится медианная яркость, и вычитается из яркости первого добавленного пикселя. Таким образом, получается максимально возможное расстояние.

Затем медиана вычитается уже из яркости добавляемого в компоненту пикселя (претендента), и получившиеся расстояние сравнивается с максимально допустимым. Если допустимое оказалось меньше, то данный пиксель не включается в компоненту.

Также стоит отметить, что проверка происходит только если количество пикселей в компоненте превышает 1% от площади изображения.

Если при анализе соседей обнаруживается две или более компоненты, то они объединяются. Причём та, что имеет меньшее количество пикселей, присоединяется к той, у которой пикселей больше. Таким образом образуется древовидная структура.

Когда все пиксели обработаны, все компоненты без «родителя» (те, которые не были присоединены к каким-либо другим компонентам) присоединяются к корневой компоненте. Таким образом, завершается формирование дерева.

После получения компонент происходит поиск крыш. Для начала создаётся маска путём нанесения на пустое изображения только инклюзивных (принадлежащих только дочерним компонентам) точек от каждой компоненты.

Теперь происходит фильтрация. Каждая компонента проверяется на следующие условия:

1. Имеет минимум 50 пикселей;
2. Глубина не более 4-го уровня (в дереве компонент корневая компонента имеет уровень 0)
3. Точки этой компоненты есть на маске

Если компонента проходит все эти условия, то её контур аппроксимируется, чтобы сгладить края.

В финальном этапе происходит ещё одна фильтрация. На основании средней площади всех отобранных компонент мы вычисляем допустимые диапазоны. Если площадь контура не

попадает в пределы, то компонента исключается. Таким образом, в итоге остаются контуры, максимально приближенные к реальным.

Литература

1. Карлссон, Г. Persistent Homology and Applied Homotopy Theory. Г. Карлссон — arXiv, 2020, 33 с.

Веселов И.П., Пажуков К.А.
Научный руководитель: к.т.н., доцент Зяблицева О.В.
ФГБОУ ВО «Ковровская государственная технологическая Академия
им. В.А. Дегтярева»
Владимирская обл., г. Ковров, ул. Маяковского 19
pmsapr@dksta.ru

Симуляция гидрологической эрозии ландшафта

С каждым годом граница между реалистичностью и компьютерной графикой становится все менее заметной. Многие современные фильмы и видео игры нельзя представить без бескрайних ландшафтов, для их создания постоянно ищутся новые методы генерации поверхностей, чтобы сократить время и ресурсы для производства. Одним из таких методов и является гидрологическая эрозия.[1]

Процесс гидравлической эрозии состоит из нескольких отдельных этапов. Во-первых, в каком-то месте или местах появляется вода. Это может быть связано с наличием дождя, источников воды или из-за потока воды. Вода поглощает материал либо из-за потока, разрушающего поверхность, либо из-за растворения. Затем вода, а также захваченный осадок транспортируются в соответствии с внутренними и внешними силами. Наиболее важным фактором, влияющим на поток воды, является гравитация, хотя внутренние силы также играют важную роль. Вода сбрасывает переносимый и подвешенный материал в определенных местах. На этот процесс осаждения влияют два основных фактора. Во-первых, вода замедляется, так что тяжелые частицы материала больше не могут переноситься. Вторым фактором, способствующим процессу осаждения, является превышение емкости водного осадка. Это вызвано испарением воды. Таким образом, процесс гидравлической эрозии может быть описан следующими четырьмя независимыми этапами:

- 1) Создаем частицу в случайном месте на ландшафте
- 2) Перемещаем нашу частицу по ландшафту, используя законы динамики
- 3) Выполним осаждение между поверхностью и частицей
- 4) Если частица слишком мала, то просто удалим её

Наша частица имеет позицию и скорость, определяющую то, как она перемещается. Кроме того, она имеет объём и долю, определяющую, какая часть объёма является осадочными породами. При создании новой частицы нам будет достаточно прописать ее позиции в пространстве, все остальные характеристики будут общие для всех частиц. В начале каждой итерации мы создаем новую частицу, которая расположена в случайных координатах, принадлежащих нашей карте высот. Частица испытывает направленное вниз ускорение, вызванное гравитацией, но она находится на поверхности, что делает ускорение вниз невозможным. Поэтому вместо этого частица подвергается силе, направленной вдоль поверхности и пропорциональной нормали к поверхности. Процесс образования осадочных пород физически происходит как перенос осадочных пород с земли на частицу и обратно в точке расположения частицы. Массообмен пропорционален разности между концентрацией поверхности и равновесной концентрацией.[2] Если наша равновесная концентрация выше текущей, то частица поглощает осадочные породы. Если ниже, то теряет их. Если они равны, то никаких изменений происходить не будет. Если наша итоговая концентрация будет меньше нуля, то просто приравняем ее к нулю. Изменение концентрации внутри нашей частицы полностью описывается уравнением массообмена. Изменение в карте высот дополнительно умножается на объём частицы, потому что мы меняем его пропорционально не концентрации, а массе. В конечном итоге мы получим коэффициент, который мы можем вычесть из высоты в данной точке. В конце каждого шага времени частица теряет немного массы в соответствии со своей скоростью испарения. Данный процесс повторяется для десятков тысяч частиц, которые создаются в случайных позициях и начинают симулироваться по отдельности.

Для того, чтобы продемонстрировать нашу симуляцию, нам потребуются какие-то первоначальные данные. Для работы с ландшафтом используется карта высот. Карта высот представляет из себя двумерную карту, которую можно использовать для того, чтобы создать

иллюзию того, что одни части ландшафта выступают сильнее, чем другие, то есть имеют большую высоту. Мы будем хранить наши данные в массиве, где 1 будет самой высокой точкой ландшафта, а 0 самой низшей.

Для генерации ландшафта был выбран алгоритм Diamond Square, т.к. он позволял сгенерировать максимально приближенный к реальности ландшафт. Алгоритм начинает работу с двумерного массива размера $2^n + 1$. В четырёх угловых точках массива устанавливаются начальные значения высот. Шаги diamond и square выполняются поочередно до тех пор, пока все значения массива не будут установлены. Шаг diamond - для каждого квадрата в массиве, устанавливается срединная точка, которой присваивается среднее арифметическое из четырёх угловых точек плюс случайное значение. Шаг square - берутся средние точки граней тех же квадратов, в которые устанавливается среднее значение от четырёх соседних с ними по осям точек плюс случайное значение. Случайное число обычно выбирается в промежутке $[-R^i, R^i]$, где R это фактор неровности в промежутке от 0 до 1, а i это номер итерации (шаг diamond и шаг square это одна итерация). Соответственно, при каждой итерации случайное значение, прибавляющееся к срединным точкам, уменьшается.

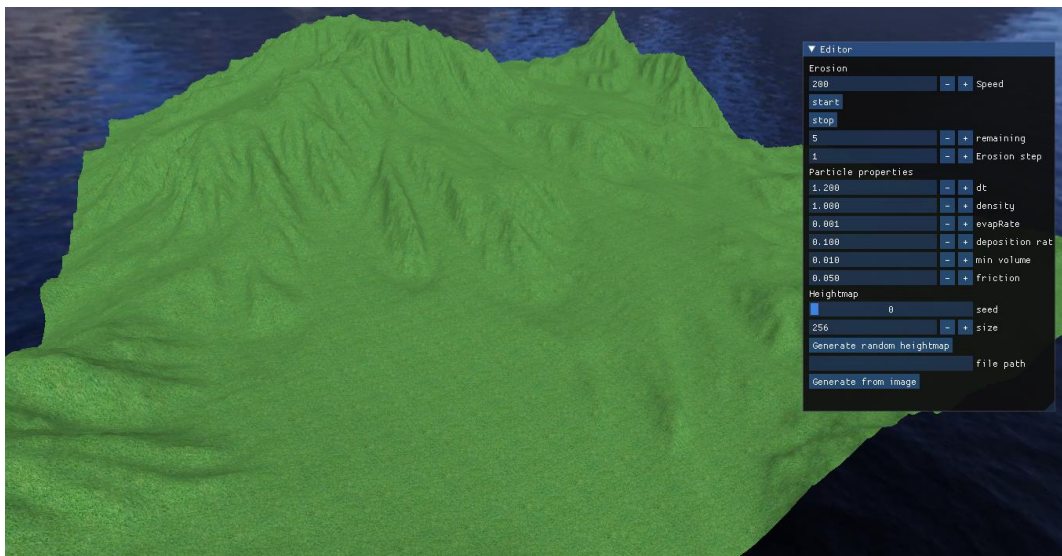


Рис.1 результат симуляции

В результате у нас есть полноценная система, которая позволяет процедурно генерировать случайные ландшафты и симулировать гидрологическую эрозию, которую мы сможем отслеживать в реальном времени (рис. 1).

Литература

1. Jean-David Genevau. Terrain Generation Using Procedural Models Based on Hydrology / Jean-David Genevau, Eric Galin, Eric Guérin, Adrien Peytavie, Bedrich Benes // HAL Open Science. – 2020
2. Vinogradov, Yu. B. HYDROLOGY OF SLOPING TERRAIN / Yu. B. Vinogradov // State Hydrological Institute, St. Petersburg, Russia

Висенте Д.А.

Научный руководитель: к.т.н., доц. Е. Е. Канунова

*Муромский институт (филиал) федерального государственного образовательного учреждения высшего образования «Владимирский государственный университет имени Александра Григорьевича и Николая Григорьевича Столетовых»
602264, г. Муром, Владимирская обл., ул. Орловская, 23
oid@mivlgu.ru*

Разработка модуля «Equipment menu» для компьютерных игр

Самый распространенный метод разработки программ – это объектно-ориентированный. Суть объектно-ориентированного программирования состоит в том, что задачу можно разложить на составные части, каждая из которых становится самостоятельным объектом. Каждая составляющая содержит свои собственные данные и методы их обработки, также, возможно, методы отношений с другими объектами. В итоге получается некая совокупность относительно независимых объектов в их взаимосвязи.

В докладе рассматриваются вопросы разработки модуля «Equipment menu». Этот модуль является одним из составляющих геймплея, компонента игры, отвечающий за интерактивное взаимодействие игры и игрока. Модуль не предназначен для проведения досуга или решения каких-либо задач, ему необходимы и другие составляющие для образования игрового процесса. Но даже так, он является всего лишь одним из компонентов компьютерной игры. Данная тема демонстрирует преимущество объектно-ориентированного программирования, а именно декомпозицию, разложение задачи на подзадачи, что позволяет облегчить достижение поставленных целей.

Приложение реализовано в среде разработки компьютерных игр Unity на версии редактора 2020.3.27f1.

Модуль имеет интуитивно понятный пользовательский интерфейс и интерактивные модули управления. Пользователю предоставлена возможность настроить и экипировать своего персонажа. В качестве дополнения сделана система уровней и начисление опыта, выбор способностей и развитие персонажа в соответствии полученному уровню.

В программе реализовано создание элементов экипировки и способностей персонажа. Реализована система характеристик персонажа и их расчет в соответствии с экипированными предметами.

Окно программы включает в себя: ячейки для экипировки, ячейки для оружия, описание характеристик персонажа. При нажатии на ячейку появляется список предметов, из которого можно поместить выбранный предмет в ячейку. При наведении на предмет вызывается окно с информацией о предмете.

Литература

1. Хокинг Дж. Unity в действии. Мультиплатформенная разработка на C# / Пер. с англ. И. Рузмайкиной. — СПб.: Питер, 2016.

Володькин М.Д, Зайцев Д.В.
рук. Зяблищева О. В., к.т.н,
Ковровская государственная технологическая академия им. Дегтярева В. А.
ул. Маяковского, 19, Ковров, Россия,
matvolodkin@gmail.com, wk.zaytsev.dv@gmail.com

Мобильный клиент для мониторинга строительства

В нашей работе предпринята попытка создания мобильного клиента для мониторинга строительства. Цель данного проекта заключается в том, чтобы облегчить контроль пользователя за строительством объекта, а также максимально структурировать все необходимые и важные данные в одном месте для быстрого доступа к ним. Например, экономическая статистика, полный план объекта, фото и видео-отчетность.

Проект разработан под ОС: Android в IDE Android Studio[1], на языке программирования Java[2], с использованием языка свободного описания структур документов XML для настройки конфигурации, а также используется язык SQL[3] для обеспечения связи между приложением и базой данных. На данный момент в приложении реализован фронтенд, базовые механизмы перехода между экранами при помощи кнопок и интерактивного бара, настроена БД для данных пользователя. Реализованы следующие экраны:

- Главный экран приложения, благодаря которому пользователь может узнать сводную информацию по проекту, а также перейти на основные экраны, с помощью быстрых клавиш (рис. 1)

- Экран, содержащий план объекта (рис. 2)

- Экран менеджмента. С помощью этого экрана пользователь может связаться с кем либо, а также перейти к просмотру видеоряда камер (рис. 3)

- Экран статистики по строительству позволяет просмотреть пользователю завершённые и незавершённые этапы строительства, общий процент завершённого строительства, а также фото- и видеоотчет (рис. 4)

- Экран управления видеонаблюдением со списком камер (рис. 5)

- Экран экономической статистики, на данном экране пользователь может изучить такие данные как финансовые расходы и общая сумма затрат (рис. 6)

Потенциальным заказчиком нашего приложения является строительная компания, она будет подключать приложение клиенту в качестве дополнительной услуги. Так, например, человек решил на строительство, он заключает договор с строительной компанией, которая подключает и настраивает данное приложение пользователю на мобильный телефон. Теперь пользователь может контролировать процесс удаленно и не ездить в зону строительства, что удобно, а также способствует уменьшению нагрузки как на логистические линии, так и финансовые расходы клиента. Благодаря приложению пользователь может подключиться к видеокамерам в режиме реального времени и проконтролировать процесс строительства, также он может связаться с каким-либо членом команды, просмотреть финансовую отчетность, статистику по объекту: примерную дату окончания строительства, какие этапы строительства уже пройдены и завершены, а какие только предстоит выполнить, каждый выполненный этап подкрепляется фото или видео - отчетом по проделанной работе. В приложении также можно детально изучить полный план объекта, и узнать точные размеры какого-либо помещения.

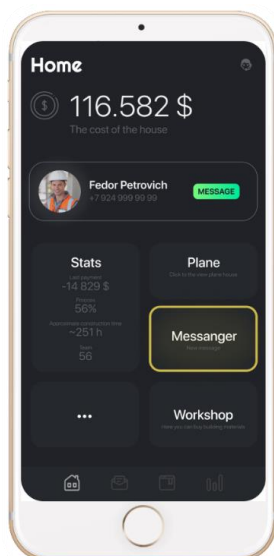


Рис. 1. Главный экран

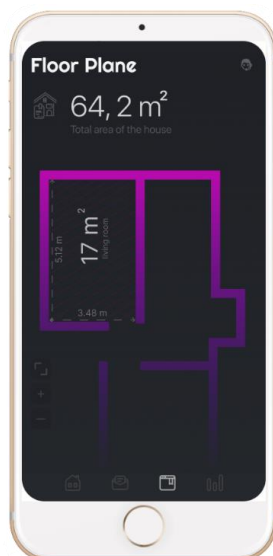


Рис. 2. Экран с планом дома

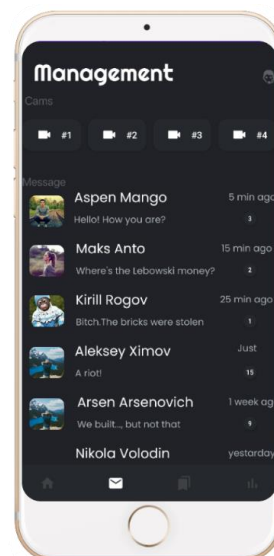


Рис. 3. Экран менеджмента

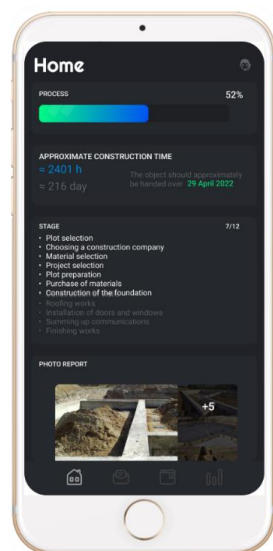


Рис. 4. Экран статистики

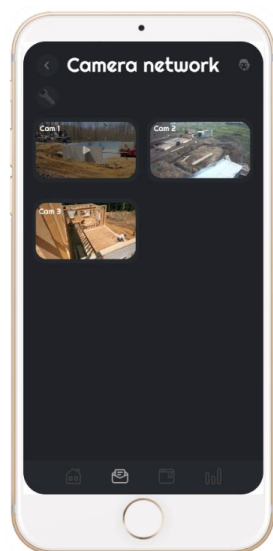


Рис. 5. Экран управления

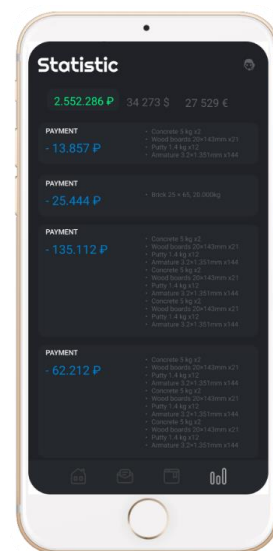


Рис. 6. Экономическая статистика

В наши планы входит дальнейшей развитие и поддержка данного проекта. Планируется разработка бэкэнда приложения: функционал экранов камер, статистики, мессенджера; разработка desktop приложения для администрирования, благодаря которому, можно будет гибко настраивать и подключать новых пользователей.

Литература

1. Документация по IDE Android Studio «Documentation for app developers» – <https://developer.android.com/docs>.
2. «Java Documentation» – <https://docs.oracle.com/en/java/>.
3. «MySQL Documentation» – <https://dev.mysql.com/doc/>.

Кашин Н.П.

*Муромский институт (филиал) федерального государственного образовательного учреждения высшего образования «Владимирский государственный университет имени Александра Григорьевича и Николая Григорьевича Столетовых»
602264, г. Муром, Владимирская обл., ул. Орловская, 23*

Комплексирование векторных данных с учетом топологий

В исследовательской работе поставлена задача разработать алгоритм комплексирования векторных данных. Под комплексированием понимается объединение, сочетание геопространственных объектов.

Исходными данными для работоспособности алгоритма являются две карты одной и той же местности, но сделанные в разный период времени или с разным масштабом.

Алгоритм объединяет некоторый набор векторных данных на одной карте с другой картой. То есть копировать набор данных с одной карты на другую.

Алгоритм основывается на принципе вычисления барицентрических координат. Барицентрические координаты - три числа, в сумме равные единице, определяющие положение точки в треугольнике, равные массам, которые следует поместить в вершинах треугольника так, чтобы определяемая точка сделалась центром тяжести этих масс (рис. 1).

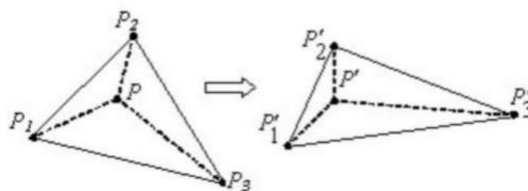


Рисунок 1 – Пример работы барицентрических координат

Расставим базовые точки, которые будут являться вершинами будущих треугольников. На растровых изображениях их можно найти с помощью алгоритма SIFT [1]. Для нахождения базовых точек векторных данных, необходимо провести их растеризацию.

Для автоматического построения треугольников на базовых точках была использована триангуляция Делоне [2]. Триангуляцией Делоне называется такая триангуляция, в которой для любого треугольника верно, что внутри описанной около него окружности не находится точек из исходного множества.

В качестве среды разработки используется географическая информационная система QGIS [3]. Она имеет открытый исходный код и подробную документацию.

Алгоритм реализуется с помощью языка программирования PyQGIS [4]. Данный язык позволяет запускать созданный скрипт в консоли системы QGIS, что ускорит и упростит процесс реализации алгоритма.

Литература

1. David G. Lowe. Object recognition from local scale-invariant features // Proceedings of the International Conference on Computer Vision. — 1999. — Т. 2. — С. 1150—1157.
2. Алгоритм триангуляции Делоне методом заметающей прямой// URL: <https://habr.com/ru/post/445048/> (дата обращения: 16.02.2022)
3. Документация QGIS// URL: <https://qgis.org/ru/docs/index.html> (дата обращения: 10.02.2022)
4. Документация PyQGIS// URL: <https://qgis.org/pyqgis/> (дата обращения: 14.02.2022)

Козлов А.С.

*Научный руководитель: к.т.н., доц. Е. Е. Канунова
Муромский институт (филиал) федерального государственного образовательного
учреждения высшего образования «Владимирский государственный университет
имени Александра Григорьевича и Николая Григорьевича Столетовых»
602264, г. Муром, Владимирская обл., ул. Орловская, 23
oid@mivlgu.ru*

Программа «Пазлы»

Современный уровень развития программных и аппаратно-программных систем, а также систем управления, связан со значительным повышением их сложности из-за присущих им требований к гибкости, открытости, эргономичности и перенастраиваемости. Наиболее перспективным считается объектно-ориентированный подход (ООП) проектирования. Достоинством данного подхода является возможность использования итеративно-поступательного цикла создания программного обеспечения и перенос акцента проектирования с разработки алгоритмов функционирования системы на построения системы абстракций и их взаимодействия.

В докладе рассматриваются вопросы разработки программы «Пазлы». В современном обществе большую популярность имеет проведение досуга за персональными компьютерами или ноутбуками. Рассматриваемая программа нацелена на предоставление пользователю полезного и занимательного времяпрепровождения.

Существует множество аналогов разработанной программы, таких как веб-сервисы «Пазлы Онлайн» и «PuzzleHeap» или приложение «Чудо Пазлы». Основной недостаток программ-аналогов – это отсутствие возможности генерации головоломки из собственного изображения.

В докладе описываются принципы разработки и использования программы «Пазлы».

Приложение реализовано в среде Microsoft Visual Studio 2019 [1] с использованием технологии Windows Forms [2]. Приложение имеет простой пользовательский интерфейс, удобный набор элементов управления, а также имеет возможность сохранения шаблона головоломки в файл.

Первое окно программы включает: кнопки импорта и экспорта файла, область предварительного просмотра изображения, набор элементов генерации пазла.

Второе окно включает: область решения головоломки, область просмотра исходного изображения, элемент, отображающий процент решенной головоломки.

В программе реализовано два метода генерации пазлов, реализация которых также будет озвучена в докладе.

Последовательность работы программы «Пазлы» следующая:

1. Импорт изображения с помощью диалогового окна;
2. Выбор метода генерация головоломки;
 - a. Экспорт изображения с шаблоном мозаики;
 - b. Генерация пазла;
3. В случае генерации - открытие нового окна программы с «кусочками» пазла и областью сборки.

Литература

1. msdn.microsoft.com - Библиотека MSDN.
2. docs.microsoft.com/ru-ru/dotnet/desktop/winforms - Руководство по классическим приложениям (Windows Forms .NET)

Королев Л.Я., Витушкин А.С.

*Руководитель: к.т.н. доцент каф. ИС Екатерина Евгеньевна Канунова
Муромский институт (филиал) федерального государственного образовательного
учреждения высшего образования «Владимирский государственный университет
имени Александра Григорьевича и Николая Григорьевича Столетовых»
602264, г. Муром, Владимирская обл., ул. Орловская, 23
E-mail: porisov33@gmail.com*

Распознавание цвета объектов и их сортировка

В настоящее время почти везде можно столкнуться с искусственным интеллектом. Существующие на сегодня интеллектуальные системы имеют достаточно узкие области применения. Например, программы, способные обыграть человека в шахматы, не могут отвечать на вопросы. В нашем случае мы будем реализовывать искусственный интеллект, который будет сортировать объекты по цвету. Эта задача поможет людям понять, как работает ИИ изнутри, так как он используется повсеместно, но мало кто задумывается, как он работает.

В докладе рассматриваются вопросы разработки такого искусственного интеллекта. С помощью данного проекта, можно показать, как в легкой и доступной форме можно продемонстрировать способности современного компьютера.

В докладе описываются принципы использования ИИ.

Искусственный интеллект будет работать под управлением одноплатного компьютера Raspberry Pi, использующего систему Raspbian. В качестве языка программирования будет использоваться Python. Система Raspbian имеет встроенный текстовый редактор для написания скриптов на Python, его мы и будем использовать.

В качестве компьютерного зрения для распознавания объектов было решено взять библиотеку OpenCV. Данная библиотека поддерживает платформу, на которой будет реализовано компьютерное зрение.

Для сортировки объектов будет использоваться сервопривод. Данное устройство легко поддается командам, написанных на языке Python, а также хорошо совместимо с одноплатными компьютерами.

Этап разработки состоит из трех частей:

1. Написание скрипта, который будет различать объекты по цвету;
2. Настройка управления сервоприводом для сортировки объектов;
3. Разработка конструкции для визуального представления работы ИИ.

В результате работы были получены навыки работы с одноплатным компьютером, библиотекой OpenCV, реализовано простейшее компьютерное зрение, управляемое ИИ.

Литература

1. docs.opencv.org/4.x/d6/d00/tutorial_py_root.html – документация по библиотеке OpenCV на Python.
2. www.raspberrypi.com/documentation/ - документация по использованию системы Raspbian.

Костюхина А.М.

*Руководитель: к.т.н. доцент каф. ИС Екатерина Евгеньевна Канунова
Муромский институт (филиал) федерального государственного образовательного
учреждения высшего образования «Владимирский государственный университет
имени Александра Григорьевича и Николая Григорьевича Столетовых»
602264, г. Муром, Владимирская обл., ул. Орловская, 23
E-mail: kostyuhinastasia@yandex.ru*

Разработка автоматизированной системы для подачи музыкальных звонков в образовательных учреждениях

Автоматизация процессов жизнедеятельности – неотъемлемая часть современного общества. Человечество развивает и совершенствует технические устройства, чтобы освободить себя от рутинных и однотипных действий, которые являются необходимыми.

В МБОУ СОШ № 8 с начала 2020 учебного года была установлена система для подачи музыкальных звонков. Трансляционная система состоит из усилителя и 20 динамиков, которые располагаются в рекреациях школы. Для обеспечения работы системы необходимо подавать звуковой сигнал с компьютера на каждые урок и перемену. В данный момент это делается вручную. Каждый день в школе проходит по 8 уроков, поэтому было принято решение автоматизировать процесс подачи звонков.

Разработанная программа позволяет добавить расписание уроков, выбрать для каждого звонка свою мелодию. Также в программе предусмотрена возможность оповещения о чрезвычайной ситуации. Расписание может быть импортировано в файл xml, для быстрой загрузки в программу ранее созданных расписаний.

Приложение разрабатывалось на объектно-ориентированном языке программирования C# [1, 2] в интегрированной среде разработки Microsoft Visual Studio 2019. Для разработки пользовательского интерфейса выбрана технология Windows Forms [3].

В результате выполнения проекта были получены навыки самостоятельной работы по формализации поставленной задачи, разработке, тестированию и отладке программ на ЭВМ. Разработана программа для автоматической подачи музыкальных звонков в образовательных учреждениях.

Данная программа может быть полезна всем образовательным учреждениям, в которых установлена система для подачи музыкальных звонков.

Литература

1. Мэтт Вайсфельд. Объектно-ориентированное мышление.: Издательство: Питер, 2013. ISBN: 978-5-496-00793-1, 978-0321861276;
2. Шилдт, Герберт. Полный справочник по C#.: Пер. с англ. — М.: Издательский дом “Вильямс”, 2004. – 752 с.: ил. – Парал. Тит. Англ.
3. Общие сведения о Windows Forms // URL: [https://msdn.microsoft.com/ru-ru/library/8bхху49h\(v=vs.110\).aspx](https://msdn.microsoft.com/ru-ru/library/8bхху49h(v=vs.110).aspx) (Дата обращения: 16.03.2022);

Крайнов С.А.

Научный руководитель: к.т.н. Е.Е. Канунова

*Муромский институт (филиал) федерального государственного образовательного учреждения высшего образования «Владимирский государственный университет имени Александра Григорьевича и Николая Григорьевича Столетовых»
602264, г. Муром, Владимирская обл., ул. Орловская, 23
oid@mivlgu.ru*

Разработка объектно-ориентированной программы «Экранный тренажёр»

Современный уровень развития программных и аппаратно-программных систем, а также систем управления, связан со значительным повышением их сложности из-за присущих им требований к гибкости, открытости, эргономичности и перенастраиваемости. Наиболее перспективным считается объектно-ориентированный подход (ООП) проектирования. Достоинством данного подхода является возможность использования итеративно-поступательного цикла создания программного обеспечения и перенос акцента проектирования с разработки алгоритмов функционирования системы на построения системы абстракций и их взаимодействия.

В связи с бурным развитием информационных технологий, у многих пользователей персональных компьютеров возникла потребность быстро вводить текстовую информацию используя компьютерную клавиатуру. Это и послужило появлению разнообразных программ-тренажеров, для получения усовершенствования навыка набора текста.

Клавиатурные тренажеры позволяют не только выучить расположение клавиш, но и правильно размещать руки, что значительно ускорит печать. Слепой метод, которому обучают современные клавиатурные тренажеры, позволяет читать бумажный оригинал и одновременно, не глядя на клавиатуру, набирать текст.

Клавиатурные тренажеры рассчитаны на самостоятельное обучение, поэтому в программе должны быть включены методические указания по организации работы, так как, чтобы научиться быстрому набору, необходимо не только знать расположение клавиш. Здесь важна и правильная постановка рук, и верное размещение клавиатуры на столе.

Важным элементом обучения компьютерной машинописи является содержание упражнений. Есть упражнение с генерацией букв, в этом упражнении пользователь научится координироваться в расположении клавиш. Следующее упражнение - текст, который генерируется случайными символами или полное грамотное слово. Пользователь может выбрать на каком языке будут генерироваться буквы и в каком регистре. И последнее упражнение - печать полных предложений, где пользователь будет печатать текст с использованными навыками сменой языка, после упражнения будут отображены скорость печати и процент качества без ошибок.

Клавиатурный тренажер - это эффективная программа для освоения слепой машинописи. В докладе описываются принципы разработки и использования разработанной программы «Клавиатурный тренажер». Программа проста в использовании и будет интересна для всех пользователей.

Литература

1. msdn.microsoft.com - Библиотека MSDN.
2. docs.microsoft.com/ru-ru/dotnet/desktop/winforms - Руководство по классическим приложениям (Windows Forms .NET)

Крестин М.О.

*Научный руководитель: к.т.н., доцент каф. ИС Р.А. Симаков
Муромский институт (филиал) федерального государственного образовательного
учреждения высшего образования «Владимирский государственный университет
имени Александра Григорьевича и Николая Григорьевича Столетовых»
602264, г. Муром, Владимирская обл., ул. Орловская, 23
noonemaddy@gmail.com*

Исследование систем управления базами данных на соответствие стандарту SQL

Язык SQL совместно с базами данных стал неотъемлемой частью многих сфер деятельности в современном мире. С момента первого упоминания об этом языке в 1974 г. и по сей день различные СУБД продолжают развиваться независимо друг от друга: MySQL, PostgreSQL, Oracle, Microsoft SQL Server, Firebird, Ред База данных и т.д. Все эти компании являются конкурентами на IT рынке, но их объединяет то, что все они предоставляют интерфейс через язык SQL.

С развитием СУБД стали появляться различные диалекты SQL. Появление многочисленных диалектов SQL и их разрастание должно было привести к проблемам совместимости и прочим противоречиям. Деятельность по стандартизации языка SQL началась одновременно с появлением его первых коммерческих реализаций. В 1982 году комитету по базам данных Американского национального института стандартов (ANSI) было поручено разработать спецификацию стандартного языка реляционных баз данных. За 39 лет компания ANSI выпустила 9 стандартов, каждый из которых дополнил предыдущий. Последний стандарт был выпущен в 2019 году.

Когда вопрос встает о возможностях СУБД, то пользователи могут полагаться только на заявления поставщиков о том, на основе какого стандарта реализована версия СУБД, так как для проведения независимого исследования потребуются большие трудозатраты. В связи с этим актуальность независимого автоматизированного исследования на соответствие различных систем управления базами данных и SQL стандартов высока.

Проведя исследование рынка, выяснилось, что доступных предложений в части программного обеспечения, которые бы могли позволить провести такое исследование, нет. Поэтому были выделены два основных пути подготовки к исследованию:

1. Изучить SQL стандарты, подготовить набор тест-кейсов и выполнить тестирование каждой СУБД вручную. Для реализации такого варианта не требуется дополнительное программное обеспечение, но работы будут сопровождаться большими трудозатратами, так как придется вручную выполнять все тест-кейсы в большом количестве СУБД. Так как специалист будет выполнять «рутинную» работу, то возрастет риск ошибки.

2. Изучить SQL стандарты, разработать открытое приложение, которое будет иметь возможность взаимодействовать с СУБД, подготовить набор тест-кейсов и структурировать его, выполнить тестирование каждой СУБД при помощи разработанного приложения. Для реализации такого варианта потребуется время на разработку приложения, но трудозатраты на проведение тест-кейсов заметно сократятся. Также будет снижен риск ошибки. В дальнейшем приложение может быть независимо оценено и доработано, в том числе с участием вендоров СУБД.

После определения положительных и отрицательных моментов был выбран второй вариант подготовки к исследованию. Он имеет преимущества перед первым, потому что необходимо будет один раз подготовить и структурировать тест-кейсы и можно будет использовать данный набор для тестирования на множестве СУБД и это менее затратно по времени, чем выполнять вручную набор тест-кейсов в каждой СУБД, также разработанное приложение в дальнейшем можно будет развивать и модифицировать для проведения дополнительного тестирования.

Опираясь на исследования рынка и выбор дальнейшего варианта подготовки к исследованию, было решено разработать приложение, которое будет выполнять проверку СУБД на соответствие тому или иному SQL стандарту и позволит компаниям более требовательно подойти к выбору СУБД для своих сервисов и приложений, а разработчикам даст более точное

представление о возможностях СУБД. Разработанная программа также может пригодиться при тестировании новых версий СУБД. Важно учесть также что, разрабатываемая программа должна быть кроссплатформенной, для того чтобы специалисты могли использовать её на любой операционной системе. Также программа должна быть с открытым кодом, а структурированный файл должен быть понятен и доступен для изменения.

Исходя из требований были определены следующие задачи:

- изучить SQL стандарт и выделить тест-кейсы;
- спроектировать структуру данных для хранения текс-кейсов;
- изучить механизмы взаимодействия различных СУБД с внешними приложениями;
- спроектировать приложение;
- разработать приложение.

При реализации приложения должны быть учтены следующие требования:

- приложение должно запускаться на различных ОС, то есть быть кроссплатформенным;
- приложение должно быть с открытым кодом;
- добавление новых тест-кейсов в структурированный файл стандарта должно происходить без изменения кода программы.

Разрабатываемое приложение будет взаимодействовать с различными СУБД и нести в себе структурированный набор тест-кейсов из SQL стандарта. По результатам работы приложение должно предоставить пользователю отчет о проведенном тестировании. Для взаимодействия с СУБД будет использована технология JDBC. Java Database Connectivity (JDBC) — это стандартный API для взаимодействия с реляционными базами данных. JDBC имеет набор классов и интерфейсов, которые могут использоваться для Java-приложения и общаться с базой данных.

Для удовлетворения требований было решено использовать следующие программные средства:

- язык программирования Java для разработки основного приложения;
- технология JDBC для взаимодействия с различными СУБД;
- среда разработки IntelliJ IDEA;
- язык структурированных запросов SQL для написания тест-кейсов;
- структура текстового представления JSON для создания структуры хранения тест кейсов;
- язык динамической разметки html – для построения отчета.)

Перед реализацией приложения необходимо разработать алгоритм работы. Работа приложения должна начинаться с предоставления пользователю списка SQL стандартов доступных для исследования, то есть тех, которые структурировали и поместили в JSON файл. После выбора одного из стандартов приложение перебирает структурированный файл и на его основе создает список тест-кейсов. В дальнейшем, когда все тест-кейсы записаны в программу, предоставляется выбор доступных для исследования СУБД. За выбором СУБД приложение предлагает выполнить последовательность действий по вводу конфигурационной информации конкретной базы данных или создать новую, если это допустимо. Затем приложение пытается соединиться с базой данных и в случае удачного соединения начинает последовательно выполнять тест-кейсы. За выполнением каждого тест-кейса следует запись результата в результирующий список, на основе которого в конце строится отчет.

После изучения и подготовки тест-кейсов их необходимо занести в структурированный файл, для того чтобы приложение могло успешно их читать. Структура должна быть информативной, но при этом не перегруженной, чтобы пользователь мог самостоятельно добавлять новые тест-кейсы. Изучив один из стандартов, стала понятна приблизительная структура будущего файла. Анализ был выполнен на примере раздела «Типы данных», данный раздел содержит подразделы «Числовые типы данных», «Строковые типы данных», «Логические» и так далее, каждый подраздел содержит в себе описание подтипов, например числовой тип данных содержит в себе описание типов integer, decimal, float и так далее. Для того чтобы проверить может ли СУБД работать с одним из типов данных, необходимо выполнить как минимум два SQL запроса:

1. создать таблицу с полями необходимого типа;
2. добавить в созданную таблицу данные.

На текущем этапе проведения исследования на соответствие СУБД со стандартами SQL был выбран подход к проведению, выбраны технологии и начата реализация. Выполнена разработка приложения, позволяющего сократить время на проведения тестирования. Разработанное приложение было выполнено с соблюдением всех требований:

- язык программирования java для кроссплатформенности;
- технология jdbc для взаимодействия с различными СУБД одинаковыми методами;
- структурированный json файл с понятной структурой для того, чтобы в дальнейшем любой пользователь мог создавать свои тест-кейсы;
- отчет в html формате, для отражения результатов тестирования.

На текущий момент ведется подготовка тест-кейсов на основе SQL стандарта 2011 года. Проведение данного исследования поможет в развитии систем управления базами данных и повысить уровень осведомленности о возможностях СУБД у пользователей.

В дальнейшем планируется модернизация приложения, добавление поддержки тестирования новых версий СУБД.

Литература

1. Крис Фиайли. SQL: Руководство по изучению языка. — М.: Peachpit Press, 2003. — 456 с.
2. Герберт Шилдт. Java. Полное руководство, 10-е издание = Java. The Complete Reference, 10th Edition. — М.: «Диалектика», 2018. — 1488 с. С. Давыдов, А. Ефимов. IntelliJ IDEA. Профессиональное программирование на Java. — СПб.,: БХВ, 2005. — 800с.
3. Официальный сайт JSON// URL: <http://json.org/json-ru.html>
4. SQL стандарт 2011 IWD 9075-2:2011, Foundation (SQL/Foundation). - Edited by: Jim Melton, 2011 - 1483 с.

Лялин М.С.

*Муромский институт (филиал) федерального государственного образовательного учреждения высшего образования «Владимирский государственный университет имени Александра Григорьевича и Николая Григорьевича Столетовых»
602264, г. Муром, Владимирская обл., ул. Орловская, 23*

Использование бикубической интерполяции для моделирования персистентной гомологии

Изображение можно разложить на отдельные матрицы (баркоды) при помощи персистентной обработки. Каждая матрица может обозначать определенную структуру на изображении. Также из данных матриц можно восстановить исходное изображение путем сложения матриц.

Данные матрицы баркодов можно изменять, при этом восстановленное изображение будет другим. Если изменить матрицы, которые внутри себя не имеют других структур, и восстановить изображение, то при повторном разложении на баркоды длины всех баркодов должны остаться неизменными. Матрицу баркода нужно изменять таким образом, чтобы элемент с максимальным значением яркости оставался прежним.

Для изменения матрицы достаточно применить функцию вида $y = f(x)$, где x и y являются вещественными числами от 0 до 1. Функция изменения баркода применяется следующим образом:

1. Сохраняется минимальный ненулевой элемент и максимальный элемент матрицы.
2. Все точки от минимальной до максимальной яркости матрицы необходимо привести к диапазону от 0 до 1, где 0 соответствует минимальной яркости, а 1 максимальной.
3. Для каждой точки матрицы применить функцию преобразования баркода.
4. Восстановить значения каждой точки к диапазону от минимальной до максимальной яркости.

Для корректного преобразования функция должны соблюдаться несколько условий:

1. Производная данной функции при любом значении x от 0 до 1 не должна быть меньше нуля. Иными словами, при каждом наборе из двух входных значений x_1 и x_2 , при котором x_1 больше x_2 , функция должна возвращать набор значений y_1 и y_2 , где y_1 больше или равна y_2 .
2. Результат $f(0)$ должен быть равен 0, а результат $f(1)$ должен быть равен 1, чтобы точки, с которых началось и которыми закончилось построение баркодов, были неизменными.

Это необходимо для сохранения баркода, чтобы при повторном их построении не появилось лишних структур.

В данном случае в качестве функции преобразования используется функция, основанная на одномерной бикубической интерполяции, однако стоит также отметить, что может подойти любая другая функция, которая не противоречит вышеизложенным правилам. Для использования данной функции необходимо определить начальные точки на координатной двумерной плоскости. Обязательно должны быть определены две точки с координатами (0, 0) и (1, 1) Начальные точки должны соответствовать правилу, при котором производная построенного по ним графика не будет отрицательной в каком-либо месте.

Далее строится график с бикубической интерполяцией по указанным ранее точкам. По данному графику для каждой точки x можно взять y , который будет результатом функции $f(x)$. Применяв данную функцию преобразования для каждой точки баркода нового изображения, длины всех его баркодов должны быть идентичны длинам баркодов исходного изображения. Это будет указывать на устойчивость персистентной гомологии при деформации изображения. Данную особенность можно использовать для сопоставления похожих изображений.

Марихов А.И.
канд. техн. наук, доцент О.В. Зяблицева,
Ковровская государственная технологическая академия имени В. А. Дегтярева
Владимирская область, г. Ковров, ул. Маяковского 19
almarihov@icloud.com

Моделирование игрового процесса с взаимодействием нескольких объектов в игре «Пушки»

Программирование игр в настоящее время является одним из интересных и перспективных направлений программирования.

Целью работы является моделирование игрового процесса на примере создания игры «Пушки»

Игра была создана на языке программирования C++. При запуске программы открывается экран игры (рис.1), состоящий из двух управляемых объектов – пушек, стоящих друг на против друга. Между ними находятся две горы, об которые разбиваются снаряды, которые игроки запускают друг в друга.

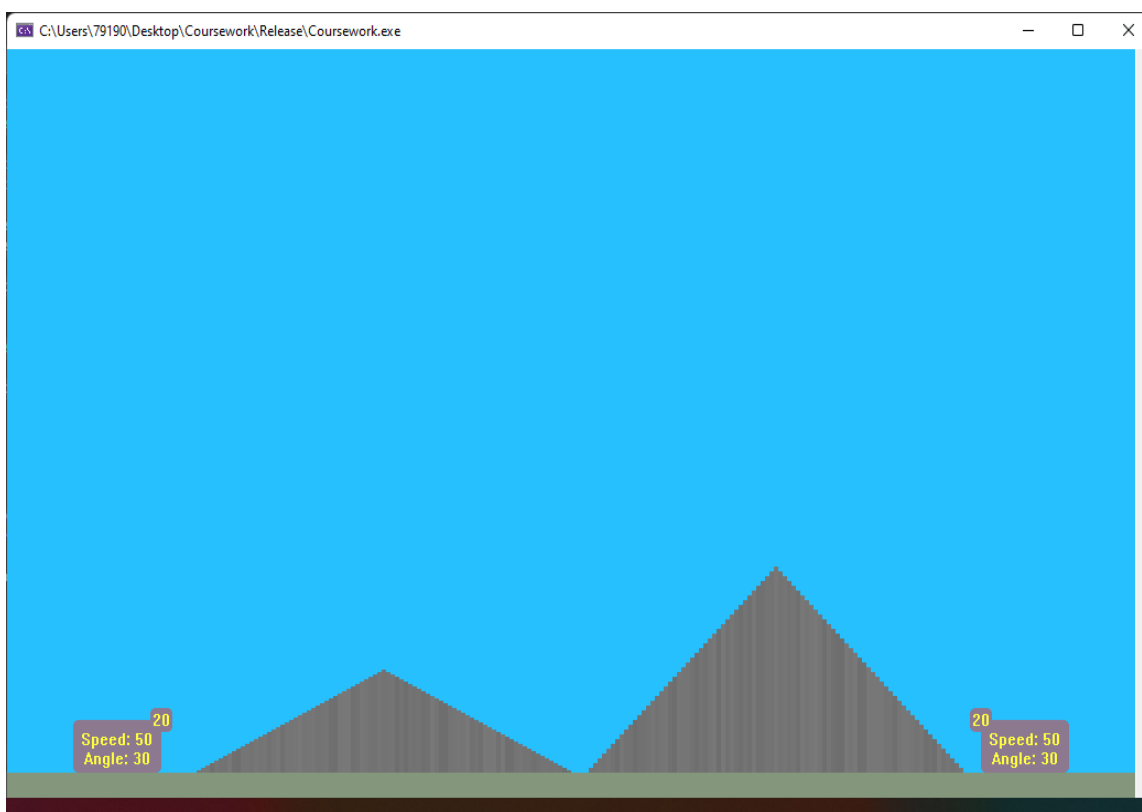


Рис.1 Экран игры

Игра предназначена для двух игроков. Каждый игрок способен одновременно управлять показателями скорости и высоты подъема пушки. Пушка имеет 20 снарядов, которые каждый игрок может запускать одновременно друг в друга и, в случае столкновения снарядов, оба снаряда разрушаются. В случае, если один игрок потеряет свои снаряды, второй игрок сразу же выигрывает. Также тот игрок, который первый поразит своего противника, выигрывает.

Как только игра заканчивается, в центре экрана игры появляется надпись (рис.2) об окончании игры и о том, кто выиграл в этот раз. С помощью кнопки «срассе» можно обновить снаряды и сыграть снова. Также, чтобы закончить работу программы, можно нажать на кнопку «esc».

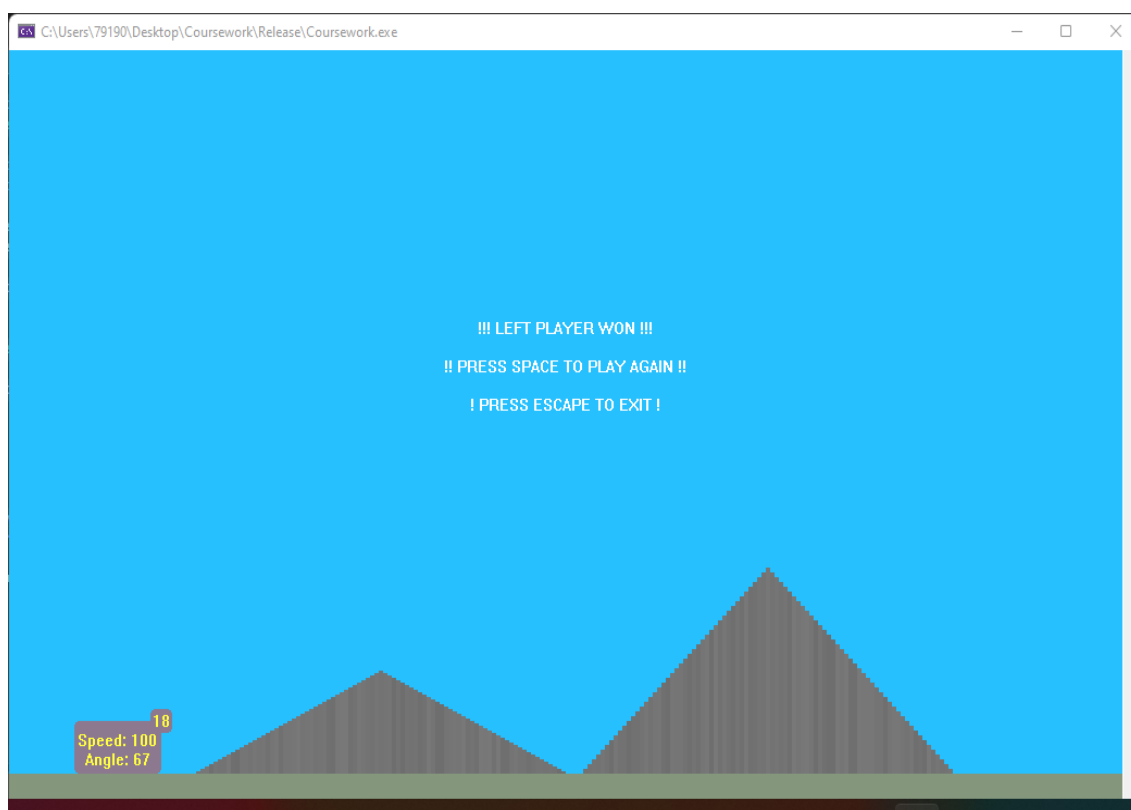


Рис.2 Экран окончания игры

Горячие клавиши, которые используются в игре:

Управление левой пушкой:

W / S — увеличить / уменьшить высоту подъема пушки

A / D — уменьшить / увеличить скорость

LSHIFT — стрелять

Управление правой пушкой:

↑ / ↓ — увеличить / уменьшить высоту подъема пушки

← / → — уменьшить / увеличить скорость

RSHIFT — стрелять

Таким образом, была создана игра «Пушки», написанная на языке программирования C++ стандарта 2015 года, которая позволяет двум игрокам решить кто из них лучше владеет стратегическим складом ума.

Литература

1. Березин, Б. И. Начальный курс C и C++: учебное пособие / Б. И. Березин, С. Б. Березин. – Москва: Диалог-МИФИ, 2012. – 280 с.: ил.,табл. – Библиогр.: с. 277. – ISBN 5-86404-075-4. – Текст: электронный. –URL: <http://biblioclub.ru/index.php?page=book&id=448000>.
2. Корчуганова, М. Р. Объектно-ориентированное программирование на C++: электронное учебное пособие / М. Р. Корчуганова, К. С. Иванов, Л. В. Бондарева; Кемеровский государственный университет, Кафедра вычислительной математики. – Кемерово: Кемеровский государственный университет, 2015. – 196 с.: ил. – Библиогр. в кн. – ISBN 978-5-8353-1832-2. – Текст: электронный. –URL: <http://biblioclub.ru/index.php?page=book&id=481559>.

Минеев Р.Р.

*Муромский институт (филиал) федерального государственного образовательного учреждения высшего образования «Владимирский государственный университет имени Александра Григорьевича и Николая Григорьевича Столетовых»
602264, г. Муром, Владимирская обл., ул. Орловская, 23
Email: minirorus7@gmail.com*

Исследование тенденций в алгоритмах сжатия данных без потерь

Сжатие без потерь — это метод сжатия, при котором не теряются данные в процессе сжатия. Сжатие без потерь «упаковывает» данные в файл меньшего размера, используя своего рода внутреннее сокращение для обозначения избыточных данных. Например, если размер исходного файла составляет 1,5 МБ, сжатие без потерь может уменьшить его примерно до половины этого размера в зависимости от типа сжимаемого файла. Это делает методы сжатия без потерь удобным для передачи файлов по сети, так как файлы меньшего размера передаются быстрее. Сжатие без потерь также удобно для хранения файлов, поскольку они занимают меньший объем [1].

При дальнейшем развитии алгоритмов сжатия данных стоит несколько задач:

1. Необходимо представить входной поток данных в виде, более удобном для кодирования существующими алгоритмами.
2. Закодировать входной поток таким образом, что выходной поток будет занимать места меньше и сможет декодироваться обратно в исходную строку.
3. Алгоритм должен быть производительным и выполнять операции кодирования и декодирования быстро [2].

Обзор существующих алгоритмов и анализ принципов работы данных алгоритмов:

1. Преобразование Барроуза — Уилера (Burrows-Wheeler transform, BWT, историческое название - блочно-сортирующие сжатие [3]). Данный алгоритм меняет порядок символов во входной строке таким образом, что повторяющиеся подстроки образуют на выходе идущие подряд последовательности одинаковых символов.

2. LZSS (Lempel-Ziv-Storer-Szymanski) [4] - алгоритм, являющийся модификацией стандартного метода LZ77. Основная разница между исходным LZ77 и LZSS состоит в том, что в методе LZ77 запись ссылки на словарь может быть длиннее, чем строка, которую она замещает (то есть запись такой ссылки делает сжатый фрагмент длиннее, чем несжатый). В методе LZSS подобные ссылки опускаются в случае, если длина строки меньше некоторой настройки («break even»). LZSS применяет однобитный флаг для обозначения того, является ли следующий фрагмент сжатого потока литералом (байтом) или ссылкой в словарь (парой значений длина и смещение) [5].

3. Алгоритм LZRW1 (Lempel-Ziv Ross Williams) [6] является модификацией LZSS и разработан с целью обеспечения максимальной скорости компрессии и декомпрессии. Степень сжатия LZRW1 равна примерно 1,5-2.

4. В ноябре 2021 года мировому сообществу в открытом виде был представлен новый алгоритм сжатия изображений без потерь. Quite OK Image Format [7] (QOI) – это алгоритм компрессии RGB и RGBA изображений, по размеру результирующего изображения близок к PNG, но в скорости превосходит его в 20-50 раз для компрессии и в 3-4 раза для декомпрессии [8].

5. Finite State Entropy (FSE) [9] – алгоритм энтропийного кодирования, похожий на алгоритм Хаффмана, и на арифметическое кодирование. При этом он взял лучшее от них обоих: работает так же быстро, как кодирование Хаффмана, и со степенью сжатия как у арифметического кодирования. FSE принадлежит семейству кодеков ANS.

6. Метод rANS [10] также, как и FSE, является членом семейства кодеков ANS. Он позволяет достичь практически оптимального сжатия при очень высокой скорости работы. В этом rANS ничем не уступает FSE, так как оба алгоритма построены на общей теоретической базе. Однако алгоритм rANS значительно проще в реализации, чем FSE.

Подводя итоги, можно сделать выводы о производительности и необходимости использования данных подходов при реализации алгоритмов сжатия данных без потерь. Для повышения степени сжатия можно использовать алгоритм преобразования входного потока, чтобы увеличить возможность более сильного сжатия путем сохранения повторяющихся блоков данных. С данной задачей хорошо справляется алгоритм BWT. Для повышения скорости сжатия в алгоритмах LZRW1 и QOI используется механизм запоминания блоков данных в hash-таблицы по различным формулам hash-функций, а в алгоритмах rANS и FSE (так же, как и в одном из условий QOI) используется запись блока в виде его смещения от предыдущих блоков данных и нет необходимости записывать данные целиком.

Литература

1. Recommendation T.45 (02/00): Run-length colour encoding. International Telecommunication Union. 2000.
2. Jacob Ziv, Abraham Lempel. A Universal Algorithm for Sequential Data Compression IEEE Transactions on Information Theory, May 1977. - pp. 337—343.
3. Huffman, D. «A Method for the Construction of Minimum-Redundancy Codes». Proceedings of the IRE. 40 (9), 1952. - 1098–1101c. doi:10.1109/JRPROC.1952.273898.
4. J. Duda, K. Tahboub, N. J. Gadil, E. J. Delp, The use of asymmetric numeral systems as an accurate replacement for Huffman coding, Picture Coding Symposium, 2015.
5. Bell T.C. 1986. Better OPM/L test compression. IEEE Trans. Commun. COM-34. 12, 1176-1182.
6. Williams, R.N., «An Extremely Fast Ziv-Lempel Data Compression Algorithm», Data Compression Conference 1998 (DCC'98), 8–11 April 1998, Snowbird, Utah, pp.362-371
7. Dominic Szablewski, The Quite OK Image format, Specification Version 1.0, 2022.01.05.
8. Witten, Ian H.; Neal, Radford M.; Cleary, John G. «Arithmetic Coding for Data Compression» (PDF). Communications of the ACM. 30 (6): 1987. - 520–540. doi:10.1145/214762.214771
9. <http://fastcompression.blogspot.com/2013/12/finite-state-entropy-new-breed-of.html> [электронный ресурс] Дата обращения: 02.03.2022
10. <https://fgiesen.wordpress.com/2015/12/21/rans-in-practice/> [Электронный ресурс] Дата обращения: 02.03.2022

Митрофанова К.Р.

Научный руководитель: к.т.н., Комкова С.В.

*Муромский институт (филиал) федерального государственного образовательного учреждения высшего образования «Владимирский государственный университет имени Александра Григорьевича и Николая Григорьевича Столетовых»
602264, г. Муром, Владимирская обл., ул. Орловская, 23*

Исследование алгоритмов сегментации зрачка аномального глаза с использованием улучшения изображения в пространственной области

Развитие системы здравоохранения в Российской Федерации на сегодняшний день заняло одно из приоритетных действующих направлений Правительства нашей страны, и это не случайно. Здоровье нации является предпосылкой социального благополучия и нормального экономического функционирования государства. Обеспечение населения доступной медицинской помощью представляет важнейшую задачу для любой страны независимо от модели хозяйствования и развития финансовой системы.

В условиях современного развития медицинской науки и практики, функциональная диагностика является одной из стремительно развивающихся областей. Функциональные методы исследования, согласно современным рекомендациям, являются основой для установления правильного диагноза.

Биометрия — это автоматизированные методы распознавания человека на основе одной или нескольких физических или поведенческих характеристик.

Среди этих биологических характеристик рисунок радужной оболочки привлекает все большее внимание, потому что это один из самых точных и надежных методов идентификации человека. Кроме того, узоры радужной оболочки обладают высокой степенью случайности и уникальности даже у однояйцевых близнецов и остаются постоянно стабильными на протяжении всей жизни взрослого человека.

На производительность систем распознавания радужной оболочки существенно влияет точность сегментации, особенно на неидеальных изображениях радужной оболочки.

Система распознавания радужной оболочки состоит из четырех основных этапов: сегментация радужной оболочки, нормализация радужной оболочки, извлечение признаков и сопоставление признаков. Сегментация радужной оболочки является первым и, вероятно, самым важным шагом в системе распознавания радужной оболочки глаза, поскольку производительность системы сильно зависит от этого шага, и ошибки могут привести к неправильной классификации во время идентификации. Этот шаг включает в себя изоляцию радужной оболочки от других компонентов изображения глаза. Как правило, внутренняя граница радужной оболочки определяется границей зрачка, а внешняя граница радужной оболочки — это граница между белой склерой и радужной оболочкой. Многие приложения требуют точного определения характеристик зрачка, таких как распознавание радужной оболочки, обнаружение взгляда и диагностика заболеваний.

Целью работы является исследование алгоритмов сегментации зрачка аномального глаза с использованием улучшения изображения в пространственной области, а также разработка и реализация программы для выявления результатов, с помощью которых стало бы возможно повысить точность определения зрачка аномального глаза.

В качестве языка программирования и среды разработки был выбран MATLAB. Данный язык является высокоуровневым интерпретируемым языком программирования, включающим основанные на матрицах структуры данных, широкий спектр функций, интегрированную среду разработки, объектно-ориентированные возможности и интерфейсы к программам, написанным на других языках программирования.

Моргаева В.А.

*Научный руководитель: к. т. н., доц. каф. ИС Подгорнова Ю.А.
Муромский институт (филиал) федерального государственного образовательного
учреждения высшего образования «Владимирский государственный университет
имени Александра Григорьевича и Николая Григорьевича Столетовых»
602264, г. Муром, Владимирская обл., ул. Орловская, 23
E-mail: vika_ozhegina@mail.ru*

Распознавание кисты молочной железы с помощью алгоритма случайного леса

Киста является достаточно распространенной патологией молочной железы. Она имеет пузыревидную форму и наполнена жидким содержимым. Данное образование относится к доброкачественным, но оно опасно возможностью развития воспалительного процесса и зарождением внутри капсулы онкологического заболевания.

Своевременная постановка диагноза пациенту, поможет минимизировать риск развития злокачественных образований на фоне доброкачественных изменений.

Маммография является одним из самых распространенных способов диагностики заболеваний молочной железы [1]. Однако не всегда на маммографических снимках можно точно определить диагноз и назначить правильное лечение пациенту.

Для повышения точности постановки корректного диагноза используются компьютерные системы для обработки и анализа заболеваний молочной железы на маммографических снимках.

Каждый алгоритм распознавания новообразований содержит несколько основных этапов:

1) Предварительная обработка изображения: этот шаг важен для некоторых методов, с целью улучшения изображения и уменьшения шума с минимальным искажением функций изображения. В некоторых САПР отсутствует этап предварительной обработки[2].

2) Сегментация изображений. Основная цель сегментации - выделение области интереса (ROI), соизмеримой с желаемыми свойствами.

3) Расчет признаков: на этом этапе извлекаются различные признаки в соответствии с характеристиками поражений из изображения. Так как существует большое количество признаков, используемых для классификации образований, необходимо провести анализ и выбрать из них наиболее эффективные для поставленной цели. В последующем эти признаки используются для различения доброкачественных и злокачественных образований. От эффективности отобранных признаков во многом зависит результат корректной классификации областей.

4) Классификация: в соответствии с выбранными характеристиками подозрительные области классифицируются на доброкачественные и злокачественные на основе различных методов классификации.

Предварительная обработка изображения выполняется с помощью адаптивного выравнивания гистограммы, медианной фильтрации для улучшения контрастности и подавления шума на изображении. Затем для полученного изображения выполняется сегментация методом Оцу для выделения области интереса.

После выполнения сегментации изображения выполняется расчет признаков построение дерева решений и на его основе выполняется классификация новообразования с помощью алгоритма случайного леса.

Для классификации изображения используются признаки, рассчитанные на основе гистограммы направленных градиентов (HOG), а также матрицы смежности уровней серого (GLCM).

На основе признаков, вычисленных для обучающих изображений, выполняется построение дерева решений.

Основная идея любого алгоритма дерева решений заключается в следующем:

- выбор лучшего атрибута, используя Мету выбора атрибутов (ASM), чтобы разделить записи;
- установка выбранного атрибута узлом решения и разбейте набор данных на меньшие подмножества.

Построение дерева, повторяя этот процесс рекурсивно для каждого потомка, пока не будет выполнено одно из условий:

- все кортежи принадлежат одному и тому же значению атрибута;
- больше нет оставшихся атрибутов;
- больше нет экземпляров;

На основе построенных деревьев решений выполняется классификация алгоритмом случайного леса.

Случайный лес — это множество решающих деревьев. В задаче регрессии их ответы усредняются, в задаче классификации принимается решение голосованием по большинству. Все деревья строятся независимо по следующей схеме:

Выбирается подвыборка обучающей выборки размера (м.б. с возвращением) – по ней строится дерево (для каждого дерева — своя подвыборка).

Данная модель использует две ключевые концепции, которые и делают этот лес случайным:

- случайная выборка образцов из набора данных при построении деревьев.
- при разделении узлов выбираются случайные наборы параметров [3].

Для построения каждого расщепления в дереве просматриваем некоторое количество (определяется случайным образом) случайных признаков (для каждого нового расщепления — свои случайные признаки) [4].

Выбирается наилучший признак и расщепление по нему (по заранее заданному критерию). Дерево строится, как правило, до исчерпания выборки (пока в листьях не останутся представители только одного класса), но в современных реализациях есть параметры, которые ограничивают высоту дерева, число объектов в листьях и число объектов в подвыборке, при котором проводится расщепление.

Для оценки эффективности результата работы алгоритма распознавания были взяты маммографические снимки из международной базы данных MIAS. В результате тестирования алгоритма было выявлено, что 3 маммограммы из 35 были классифицированы не правильно, остальные были классифицированы корректно. Чувствительность алгоритма составила 88,89%, специфичность 94,12%, а точность алгоритма 91,43%.

Литература

1. Заридзе Д.Г. Динамика заболеваемости злокачественными новообразованиями и смертности от них в России. / Д.Г. Заридзе, А.Д. Каприн, И.С. Стилиди// Вопросы онкологии. 2018. Т.64. №5. С. 578-591.
2. Afsaneh Jalalian, Syamsiah Mashohor , Rozi Mahmud , Babak Karasfi , M. Iqbal B. Saripan , Abdul Rahman B. Ramli. Foundation and methodologies in computer-aided diagnosis systems for breast cancer detection. // EXCLI Journal. 2017. №16. С. 113-137.
3. Дьяконов А. Случайный лес (Random Forest) – URL: <https://dyakonov.org/2016/11/14/%D1%81%D0%BB%D1%83%D1%87%D0%B0%D0%B9%D0%BD%D1%8B%D0%B9-%D0%BB%D0%B5%D1%81-random-forest/> (Дата обращения: 15.03.2022).
4. Штукатуров С. Реализация и разбор алгоритма «случайный лес» на Python – URL: <https://tproger.ru/translations/python-random-forest-implementation/> (Дата обращения: 18.03.2022).

Назарова С.О., Иванова О.О.
Научный руководитель – преподаватель Т.Н. Чернявская,
преподаватель Л.Н. Виноградова
Государственное бюджетное профессиональное образовательное учреждение
Владимирской области "Муромский колледж радиоэлектронного приборостроения"
602267, г. Муром Владимирской обл., ул. Комсомольская, д.55
e-mail: mtrp@narod.ru

Симуляция эпидемии коронавируса на Arduino. Исследование скорости темпа распространения коронавируса.

В наше время ковид является одной из самых глобальных проблем всего человечества. Коронавирусная инфекция распространена повсеместно и регистрируется в течение всего года с пиками заболеваемости зимой и ранней весной, когда эпидемическая значимость ее колеблется от 15% до 34%. Инфекция распространяется воздушно-капельным и контактным путем. Источником инфекции являются больные с клинически выраженной или скрытой формой заболевания. Инкубационный период заболевания, провоцируемого коронавирусной инфекцией, зависит от формы и длится от 3 до 14 дней.

Нашей целью является исследование скорости темпа распространения коронавируса.

Для создания данной модели нам потребовалась платформа Arduino, она будет являться вычислительным ядром, и дисплей. Так же мы переместили модель на специальную программу под названием Processing. Мы запрограммировали нашу модель специальным кодом, где задали необходимые значения.

Наше исследование проходило в два испытания. В первом случае жители не использовали средства самозащиты(рис.1). Заражение шло быстрым темпом. Во втором случае жители были самоизолированы и использовали средства самозащиты(рис.2). Скорость заражения значительно уменьшилась при тех же параметрах заражения.

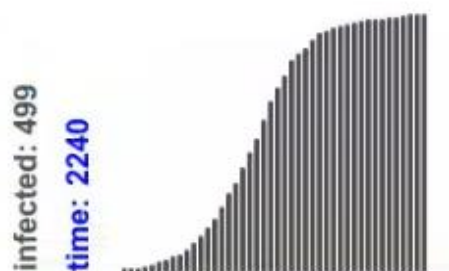


Рис. 1 – До самоизоляции



Рис. 2 – После самоизоляции

В ходе нашего исследования мы выяснили, что самоизоляция, а также создание вакцины и массовой вакцинации значительно сокращает темп распространения коронавируса. Следуя указаниям и мерам предосторожности, описанным выше, вы будете способствовать предотвращению распространения коронавирусной инфекции.

Литература

1. Wikipedia – свободная энциклопедия: Processing [Электронный ресурс]. — Режим доступа: <https://ru.wikipedia.org/wiki/Processing>
2. Все о коронавирусе [Электронный ресурс]. — Режим доступа: <https://xn--80aesfpebagmfb1c0a.xn--p1ai/about-covid/#world-statistics>
3. Arduino IDE: установка и настройка [Электронный ресурс]. — Режим доступа: <http://wiki.amperka.ru/articles/arduino-ide-install>

Никишов Д.А.

*Научный руководитель: к.т.н., доцент, доцент каф. ИС С.А. Щаников
Муромский институт (филиал) федерального государственного образовательного
учреждения высшего образования «Владимирский государственный университет
имени Александра Григорьевича и Николая Григорьевича Столетовых»
602264, г. Муром, Владимирская обл., ул. Орловская, 23
E-mail: daniilnikisov74@gmail.com*

Разработка программы для моделирования аппаратных ускорителей искусственных нейронных сетей на базе мемристивных устройств

Аппаратный ускоритель искусственного интеллекта (ИИ) — это специализированное устройство, архитектура которого позволяет выполнять алгоритмы искусственных нейронных сетей (ИНС) быстрее по сравнению с универсальными вычислителями. В настоящее время одним из наиболее перспективных подходов к созданию аппаратных ускорителей ИИ является использование массивов мемристивных устройств, на базе которых создаются устройства аналогового матрично-векторного умножения [1]. Такие устройства обладают высокой производительностью и низким энергопотреблением, что позволяет их использовать в составе бортовых, носимых и встраиваемых систем.

Развитие нового типа аппаратных ускорителей ИИ на базе мемристивных устройств вызывает необходимость создания программных средств (модулей, библиотек) для систем автоматизированного проектирования и компьютерного моделирования. В частности, необходимо разработать программу для имитационного моделирования, которая позволит проводить моделирование процессов преобразования информации в рамках процедуры матрично-векторного умножения с применением массивов мемристивных устройств с учетом нестабильностей электрофизических характеристик электронных компонентов [2].

Теоретической базой разработанной программы являются общий подход, методы и алгоритмы определения, обеспечения и оптимизации точности и допусков ИНС [3,4], которые имеют следующие преимущества и особенности по сравнению с известными до настоящего времени в России и за рубежом аналогами:

- позволяют определять, обеспечивать и оптимизировать количественные показатели точности и допусков ИНС, нормируемые российскими и международными стандартами, в условиях дестабилизирующих воздействий [5];
- успешно реализуются в виде пакетов прикладных программ, баз данных и библиотек с открытым исходным кодом.

В данном докладе приводится проект программы для моделирования аппаратных ускорителей ИНС на базе мемристивных устройств. Работа выполнена при поддержке Субсидии Министерства науки и высшего образования РФ (проект №13.2251.21.0098, соглашение №075-15- 2021-1017).

Литература

1. Борданов И. А. Современное состояние в области аппаратной реализации искусственных нейронных сетей на базе мемристоров / И. А. Борданов, С. А. Щаников, С. Н. Данилин // Телекоммуникации. – 2020. – №. 8. – С. 35-48.
2. Simulation of Inference Accuracy Using Realistic RRAM Devices / A. Mehonic, D. Joksas, W. H. Ng et al. // Front. Neurosci. – 2019. – Vol. 13.
3. Щаников С.А. Методология программно-аппаратного моделирования нейроморфных вычислительных систем на базе мемристивных устройств // Российские нанотехнологии. 2021. Т. 16. № 6. С. 816-824.
4. Shchanikov S.A. Methodology for Hardware-in-the-Loop Simulation of Memristive Neuromorphic Systems // Nanobiotechnology Reports. 2021, Vol.16, No. 6, pp. 782-789.
5. Galushkin A.I., Danilin S.N., Shchanikov S.A. The Research of Memristor-Based Neural Network Components Operation Accuracy in Control and Communication Systems // 2015 International Siberian Conference on Control and Communications, SIBCON 2015 – Proceedings. 2015. PP. 1-6. doi:10.1109/SIBCON.2015.7147034

Панкратов Д.А.

*Научный руководитель: к. т. н., доц. каф. ИС Подгорнова Ю.А.
Муромский институт (филиал) федерального государственного образовательного
учреждения высшего образования «Владимирский государственный университет
имени Александра Григорьевича и Николая Григорьевича Столетовых»
602264, г. Муром, Владимирская обл., ул. Орловская, 23*

Создание платформера-головоломки с голосовым управлением на игровом движке Unity

Индустрия компьютерных игр (также индустрия интерактивных развлечений) — сектор экономики, связанный с разработкой, продвижением и продажей компьютерных игр. В неё входит большое количество специальностей, по которым работают десятки тысяч человек по всему миру [7].

Первые компьютерные игры появились еще на заре компьютерной эры, во времена ламповых ЭВМ, так в 1952 году А. С. Дуглас как часть своей университетской диссертации создал программу «ОХО», которая являлась простейшей реализацией игры «крестики-нолики» [7].

Целью данной работы является создание игры в жанре платформер-головоломка с голосовым управлением на игровом движке Unity.

В ходе работы будут решены следующие задачи:

- анализ аналогов;
- реализация голосового управления;
- программная реализация логики игры;
- разработка уровней игры.

Платформер – это жанр компьютерных игр, где подавляющей частью игрового процесса является перемещение по платформам путём прыжков, а также собирание предметов, зачастую необходимых для завершения уровня, или же наделяющие протагониста особыми навыками, как привило исчезающими со временем. Например, неуязвимостью, скоростью, увеличенной высотой прыжка и т.д.

Головоломка – название обширного жанра компьютерных игр, целью которых является решение логических задач, требующих от игрока задействования логики, стратегии, интуиции или в некоторых случаях удачи. Существует огромное количество разновидностей головоломок. В некоторых из них игроку необходимо в определённой последовательности и форме собрать случайные блоки. Головоломки-лабиринты не зависимо от масштабности и уровня сложности предлагают пройти по запутанным дорогам, отыскав, в большинстве случаев, единственно правильный путь [4].

Unity (Unity Technologies) - межплатформенная среда разработки компьютерных игр, которая отличается большим количеством поклонников и обилием обучающих материалов, а также наличием огромной библиотеки ассетов и плагинов, с помощью которых можно значительно ускорить процесс разработки. Это одни из самых популярных движков среди разработчиков игр. Unity использует open-source проект Mono, который является одной из многочисленных реализаций платформы Microsoft .NET [1].

Unity поддерживает скрипты на C#, созданные в соответствии с одним из двух основных подходов: традиционным и широко используемым объектно-ориентированным подходом и информационно-ориентированным подходом, который теперь тоже поддерживается в Unity в отдельных случаях благодаря нашему высокопроизводительному многопоточному стеку информационно-ориентированных технологий (DOTS) [1].

C# - объектно-ориентированный язык программирования. На сегодняшний момент это один из самых мощных, быстро развивающихся и востребованных языков в ИТ-отрасли. В настоящий момент на нем пишутся самые различные приложения: от небольших десктопных программ до крупных веб-порталов и веб-сервисов, обслуживающих ежедневно миллионы пользователей [2].

Литература

1. Документация по Unity // URL: <https://docs.unity3d.com/ru/530/Manual/> (дата обращения 04.04.2022)
2. Документация по C# // URL: <https://docs.microsoft.com/ru-ru/dotnet/csharp/> (дата

обращения 04.04.2022)

3. Сверхбыстрое распознавание речи без серверов на реальном примере // URL: <https://habr.com/ru/post/237589/> (дата обращения 04.04.2022)

4. Лабиринты: классификация, генерирование, поиск решений // URL: <https://habr.com/ru/post/445378/> (дата обращения 04.04.2022)

5. Голосовой ввод в Unity // URL: <https://docs.microsoft.com/ru-ru/windows/mixed-reality/develop/unity/voice-input-in-unity> (дата обращения 04.04.2022)

6. Работа с микрофоном в Unity. Часть 1 // URL: <https://www.iterekhin.ru/2014/12/unity-sound-part1.html> (дата обращения 04.04.2022)

7. Игровая индустрия: Геймдев (Gamedev)// URL: <https://hsbi.hse.ru/articles/igrovaya-industriya-geymdev/> (дата обращения 04.04.2022)

Пронищев П.В.

*Научный руководитель: д.т.н., доцент каф. ИС Д.Е. Андрианов
Муромский институт (филиал) федерального государственного образовательного
учреждения высшего образования «Владимирский государственный университет
имени Александра Григорьевича и Николая Григорьевича Столетовых»
602264, г. Муром, Владимирская обл., ул. Орловская, 23
pronpro@mail.ru*

Разработка информационной системы учета и систематизации материальных средств специальной номенклатуры

В современных условиях многообразия информационных систем несмотря на широкий спектр разрабатываемого программного обеспечения часто возникает потребность в узкоспециализированных разработках для решения специальных задач, обусловленных потребностью конкретных организаций. Зачастую с проблемами отсутствия специализированных информационных систем сталкиваются государственные организации и учреждения, в том числе и силовые ведомства.

В настоящее время в Воздушно-космических силах Российской Федерации оперативный учет материальных средств ведется по морально устаревшей системе с использованием бумажных носителей информации (книг, журналов). Компьютерный учет реализован в виде хранения учетных данных в электронных таблицах Excel в разрозненном состоянии, что затрудняет систематизацию и обработку информации. Отсутствие специализированного программного обеспечения для ведения учета материальных средств номенклатуры Воздушно-космических сил существенно замедляет оперативную обработку учетной информации.

Возможности современных Вооруженных сил в оснащении компьютерной техникой позволяют использовать информационные системы, предназначенные для решения проблем оперативного учета.

Вывод. Для оптимизации учета нефинансовых активов в Воздушно-космических силах Российской Федерации необходима специализированная информационная система, учитывающая потребности современных требований. Данная разработка позволит упростить доступ и систематизировать учетные данные материальных средств специальной номенклатуры, отображать движение, состояние и категоричность материальных средств, автоматизировать разработку учетно-отчетной документации. В рамках работы складов – реализуется функционал складской логистики – движение и хранение изделий рассматривается в комплексе с учетом делопроизводства и бухгалтерии, формируется необходимая система складской навигации. Реализуется алгоритм отправки материальных средств в сторонние организации, воинские части и ремонтные предприятия. Информационная система предоставляет функционал для расчета основных характеристик материальных средств – остаток ресурса по эксплуатации, сроку годности, остаток гарантийного ресурса.

Цель работы

Создание информационной системы учета, позволяющей вести оперативный учет и в кратчайшие сроки формировать необходимые отчетные формы.

Задачи

- проанализировать аналогичные системы;
- произвести анализ средств разработки;
- разработать структуру базы данных;
- определить методы защиты информации;
- определить способы тестирования информационной системы.

Анализ систем аналогов

В рамках решения задач оперативного учета авиационного имущества неоднократно предпринимались попытки разработки специализированных систем, но ввиду сложности специфики и отсутствия поддержки со стороны государственных структур завершены не были и в эксплуатацию не вводились. Патентный поиск подтвердил, что в Воздушно-космических силах

Российской Федерации разработки систем учета и систематизации технического имущества не ведутся. Реализованное программное обеспечение отсутствует.

Анализ средств разработки

Для реализации программы используются следующие инструменты:

1. Реляционная модель данных – применяется для представления данных предметной области. Представляет собой табличный способ представления информации. Строки, содержащиеся в таких таблицах представлены в виде записей с уникальным идентификатором. Реляционные базы данных применяются для хранения и обработки взаимосвязанных данных.

2. Язык T-SQL – инструмент манипулирования данными. Представляет собой структурированный язык запросов, предназначенный для работы с данными в базе.

3. Язык C# на базе фреймворка .NET – инструмент для реализации программного продукта. Является языком программирования, подтвержденным современной объектно-ориентированной концепцией программирования.

Разработка структуры базы данных

Разработка структуры программы учета и систематизации материальных средств специальной номенклатуры подразумевает выделение нескольких уровней моделирования с целью осуществления перехода от предметной области к формируемой базе данных:

- непосредственно предметная область, которая включает характеристики материальных средств и особенности учета;

- модель предметной области, формирование которой подразумевает выделение существенных составляющих учета и систематизации, подлежащих включению в проект;

- логическая модель данных (Рисунок 1) – представляет прототип будущей базы данных, построение которой производится в структуре информационных обобщений характеристик объектов учета без привязки к СУБД, решения, принятые на данном уровне, определяют границы формируемой базы данных;

- физическая модель данных – формируется с использованием средств выбранной СУБД MS SQL Server, реализация производится средствами реляционной модели;

- собственно база данных и приложение-оболочка, которые являются результатом построения вышеперечисленных моделей.

решения, принятые на каждом этапе моделирования, будут оказывать значительное влияние на последующие и конечный результат разработки.

Центральной частью разработки информационной системы является концептуальная схема, которая представляет собой описание логической структуры данных. Схема строится на основании изучения предметной области и характеристик объектов.

Таким образом, в ходе разработки программного обеспечения необходимо обеспечить как физическую независимость от способа хранения и методов доступа, так и логическую возможность изменения одного приложения без изменения остальных, работающих с этими же данными.

Использование системы управления базами данных конечным пользователем должно исключать программирование, реализуя только встроенные функции и методы прикладной программы.

Целями разработки базы данных, реализующей алгоритм управления движением и систематизацией материальных средств, являются хранение большого объема данных, быстрый поиск требуемой информации, добавление, удаление и изменение хранимой информации, вывод ее в удобном виде.

Определяя подход к построению базы данных, реализуется возможность пользовательского ввода данных, необходимых для решения задач по управлению движением и систематизации материальных средств.

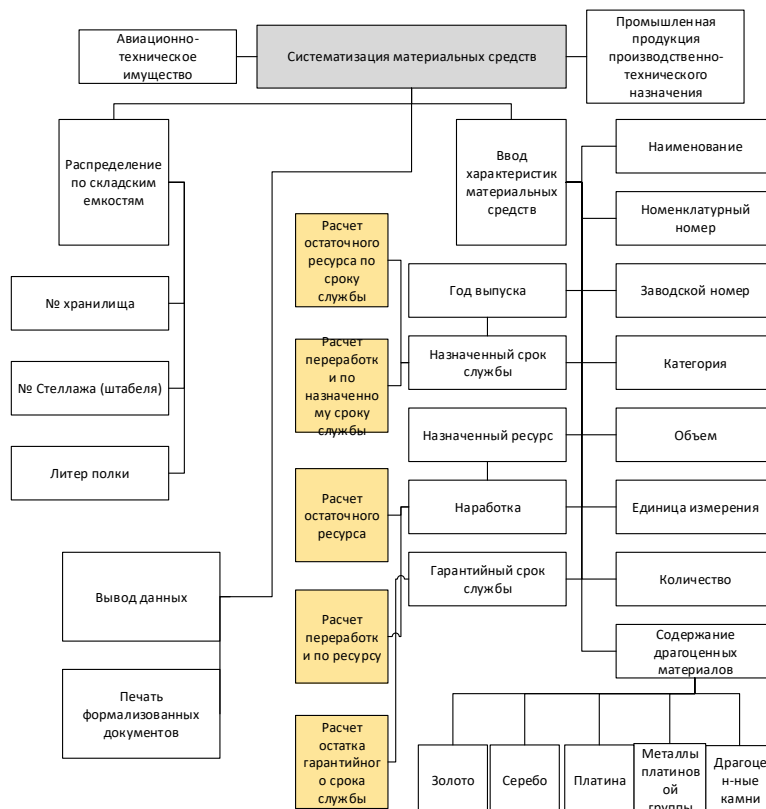


Рисунок 1. Логическая модель данных

Исходя из рамок, определенных при моделировании базы данных системы учета материальных средств специальной номенклатуры, обозначим основные требования к разработке:

- минимальная избыточность – подразумевает отсутствие дублирования учетных данных;
- непротиворечивость – исключение возникновения противоречий при изменении данных;
- целостность данных – означает, что в базе данных должны храниться только правильные данные, соблюдающие логические условия;
- независимость данных – подразумевает, что программа-оболочка не зависит от хранимых данных, способа их хранения;
- возможность добавления, удаления и актуализации данных;
- безопасность – защиту баз данных от несанкционированного доступа, преднамеренного или непреднамеренного нарушения целостности структуры и хранимой информации;
- высокая производительность – быстроедействие должно максимально обеспечивать взаимодействие разработки с пользователем в диалоговом режиме.;
- соблюдение стандартов.

Методы защиты разрабатываемой базы данных

Специфика используемых данных подразумевает особый подход к защите хранимой информации. С целью исключения утечки и повреждения применяются следующие подходы к защите:

- защита базы данных паролем;
- разграничение прав доступа к объектам базы данных;
- защита записей в таблицах базы данных.

Тестирование информационной системы

Тестирование информационной системы учета и систематизации материальных средств специальной номенклатуры будет производиться специалистами авиационно-технических служб в рамках учебно-методических занятий и сборов.

Рыкова Е.В.

Научный руководитель: к.т.н. Комкова С.В.

*Муромский институт (филиал) федерального государственного образовательного учреждения высшего образования «Владимирский государственный университет имени Александра Григорьевича и Николая Григорьевича Столетовых»
602264, г. Муром, Владимирская обл., ул. Орловская, 23*

Диагностика глаукомы с использованием области CDR и NRR на изображениях сетчатки глаза

Глаукома - это хроническое прогрессирующее заболевание, которое при отсутствии лечения может привести к слепоте. Глаукома представляет собой целый ряд состояний, при которых поток водянистой влаги в глазу блокируется, вызывая повышение внутриглазного давления (ВГД). Нарушение притока крови к диску зрительного нерва также может вызвать глаукому, даже без аномально высокого ВГД. Даже при тщательном мониторинге и снижении ВГД примерно 25% пациентов теряют зрение. Глаукома является второй по распространенности причиной слепоты во всем мире.

Наиболее распространенный тип глаукомы, первичная открытоугольная глаукома (ПОУГ), имеет слабые симптомы на ранних стадиях своего течения и, следовательно, часто остается незамеченным. Поздняя диагностика глаукомы значительно увеличивает стоимость лечения и повышает риск нарушения зрения. Повышенный уровень ВГД считался наиболее точным средством диагностики; однако в настоящее время признано, что высокое ВГД является фактором риска развития глаукомы, но не является точной диагностикой. Поэтому существует необходимость в точной системе диагностики.

Раннее выявление глаукомы играет ключевую роль в предотвращении необратимого повреждения глаз.

Целью работы является исследование алгоритмов сегментации сетчатки глаза с использованием, а также разработка и реализация программы для диагностики глаукомы с использованием области CDR и NRR на изображениях сетчатки глаза.

В качестве языка программирования и среды разработки был выбран MATLAB. Данный язык является высокоуровневым интерпретируемым языком программирования, включающим основанные на матрицах структуры данных, широкий спектр функций, интегрированную среду разработки, объектно-ориентированные возможности и интерфейсы к программам, написанным на других языках программирования.

Толмачев А.С.

*Научный руководитель: к.т.н., доцент, доцент каф. ИС С.А. Щаников
Муромский институт (филиал) федерального государственного образовательного
учреждения высшего образования «Владимирский государственный университет
имени Александра Григорьевича и Николая Григорьевича Столетовых»
602264, г. Муром, Владимирская обл., ул. Орловская, 23
Email: ne_magl@mail.ru*

Обзор алгоритмов поиска автономного пути в неизвестной среде

Поиск пути — одна из важнейших задач в области искусственного интеллекта. Алгоритмы поиска пути в неизвестной среде являются важной составляющей в робототехнике, которая активно используется как в современной науке, так и в повседневной жизни. Автономный робот способен работать самостоятельно без руководства человека, что не только позволяет оптимизировать многие производственные процессы предприятий, но и дает возможность исследовать территории, которые недоступны или опасны для человека. Так создание и модернизация автономных роботов может значительно упростить жизнь человека

Задача поиска автономного пути включает в основном три принципа.

1. Путь должен пролегать от начальной точки к конечной точке.
2. Путь должен обеспечивать движение агента с обходом препятствий.
3. Путь должен быть оптимальным.

Однако, когда речь заходит о неизвестной среде, алгоритмы поиска пути не могут давать гарантии оптимальности пути. Его место занимает другой принцип – алгоритм должен быть быстрым. Алгоритм должен иметь высокую частоту обновления для быстрой адаптации к новым препятствиям.

В докладе представлен обзор существующих алгоритмов и анализ принципов работы таких алгоритмов, как алгоритм D*-lite, метод потенциальных полей, метод быстро исследуемых случайных деревьев, муравьиный алгоритм и нейронные сети. В результате анализа алгоритмов поиска пути можно выяснить, для каких карт лучше всего подходит тот или иной алгоритм. Для больших и сложных карт подойдут алгоритм на основе муравьиной колонии и быстро исследуемое случайное дерево. RRT быстрее муравьиной колонии, однако муравьиная колония может оптимизировать свой путь. Для больших, но не сложных карт может подойти метод потенциальных полей, однако проблема локальных минимумов все еще остается и требует решения. Для малых карт алгоритм D-lite* отлично подойдет за счет быстрой скорости. Алгоритм поиска, основанный на нейронных сетях, универсален и способен находить путь к цели в не зависимости от размера и сложности карты.

Литература

1. Лю В., Методы планирования пути в среде с препятствиями (обзор) // Математика и математическое моделирование. 2018. № 01. С. 15–58
2. Казаков К.А., Семенов В.А. Обзор современных методов планирования движения. Труды ИСП РАН, том 28, вып. 4, 2016, стр. 241-294
3. Heo, Sn., Chen, J., Liao, Yc. et al. Auto-splitting D* lite path planning for large disaster area. Intel Serv Robotics (2022). URL: <https://doi.org/10.1007/s11370-022-00416-8>
4. S.M.LaValle, The RRT Page [электронный ресурс], URL: <http://lavalle.pl/rrt/index.html>
5. Метод потенциалов// Интеллектуальные алгоритмы поиска в решении логистических задач [электронный ресурс], URL: <https://sites.google.com/site/ds6gerenko/lekci/ostovnye-dereva/prokladka-kommunikacij-mezdu-prepatstviami-1/metodpotencialov>
6. Hagelbäck J., Использование потенциальных полей в сценарии стратегии реального времени [электронный ресурс], URL: <https://habr.com/ru/post/262181/>
7. Xiaoyun L., Zhian Z., Peifang D., Dynamic Path Planning of Unknown Environment Based on Deep Reinforcement Learning [электронный доступ], URL: <https://doi.org/10.1155/2018/5781591>

Хлызова В.Г.

*Научный руководитель: д.т.н., профессор каф. ИСПИ Хорошева Е.Р.
Федеральное государственное бюджетное образовательное учреждение высшего
образования «Владимирский государственный университет имени Александра
Григорьевича и Николая Григорьевича Столетовых» (ВлГУ)
Россия, 600000, Владимирская область, г. Владимир, ул. Горького, дом № 87
E-mail: valeriakhlyzoba@gmail.com*

Хронологическая архитектура системы управления цифровыми документами

Хронологическая архитектура является завершающим этапом моделирования архитектуры информационных систем, формализует этапы создания информационной системы и дает ответ на вопрос «В какой последовательности и в какие сроки?» [1].

В данной статье рассматривается построение хронологической архитектуры системы управления цифровыми документами (СУЦД), предназначенной для выпуска, хранения и верификации документов любого вида внутри конкретной организации. Актуальность и важность создание хронологической архитектуры системы управления цифровыми документами на этапе моделирования архитектуры системы определяется необходимостью наглядного представления этапов разработки продукта, для дальнейшего выбора сроков завершения каждого из этапов и планирования разработки системы в целом.

Внедрение системы управления цифровыми документами позволит снизить стоимость выпуска и верификации документов, а также стандартизировать выпуск документов путем создания и последующего использования образцов (шаблонов) для документов.

СУЦД должна поддерживать следующие функции:

1. создание новых организаций в системе;
2. создание внутри организации виртуальных кошельков;
3. создание и публикация шаблонов для выпуска документов (поддержка и контроль версионирования шаблонов);
4. выпуск документов по выпущенным шаблонам;
5. авторизация в системе приглашенных пользователей;
6. управление ролями пользователей и виртуальными кошельками внутри организации;
7. создание правил для прохождения верификации документа;
8. верификация документа;
9. сбор статистики по выпущенным документам и пройденным верификациям.

Система управления цифровыми документами состоит из административной части (web-сайта) и мобильного приложения. В качестве инструментов разработки административной части были выбраны Angular (платформа для разработки клиентской части приложения) и коа (относительно новая платформа для разработки преимущественно серверной части приложения). Мобильное приложение разработано с помощью React Native – фреймворк для разработки кроссплатформенных мобильных приложений.

Для разработки информационной системы управления цифровыми документами выделено 5 основных этапов:

1. выбор стратегии и анализ требований;
2. выбор технологий;
3. разработка и тестирование;
4. внедрение;
5. эксплуатация.

На первом этапе должен быть составлен список требований к системе заказчиками, и на их основе сформулировано техническое задание, а также созданы макеты страниц (для этого можно использовать онлайн-сервис для разработки интерфейсов - Figma).

На втором этапе на основе составленных требований осуществляется выбор технологий. В качестве технологии разработки системы управления цифровыми документами был выбран подход SSI. Для реализации системы за основу можно взять опенсорсные решения, такие как

Aries Framework — это комплексная реализация протоколов Aries. Его цель - предоставить универсальную библиотеку для создания приложений SSI.

Следующий этап - непосредственно разработка системы. Для управления процессом разработки можно использовать баг-трекингую систему -Jira. Последние этапы — это внедрение и последующая эксплуатация системы. В начале этапа внедрения необходимо написать программную документацию для последующего использования системы.

Хронологическая архитектура построена в соответствии с ARIS-методологией и представлена на рисунке 1.

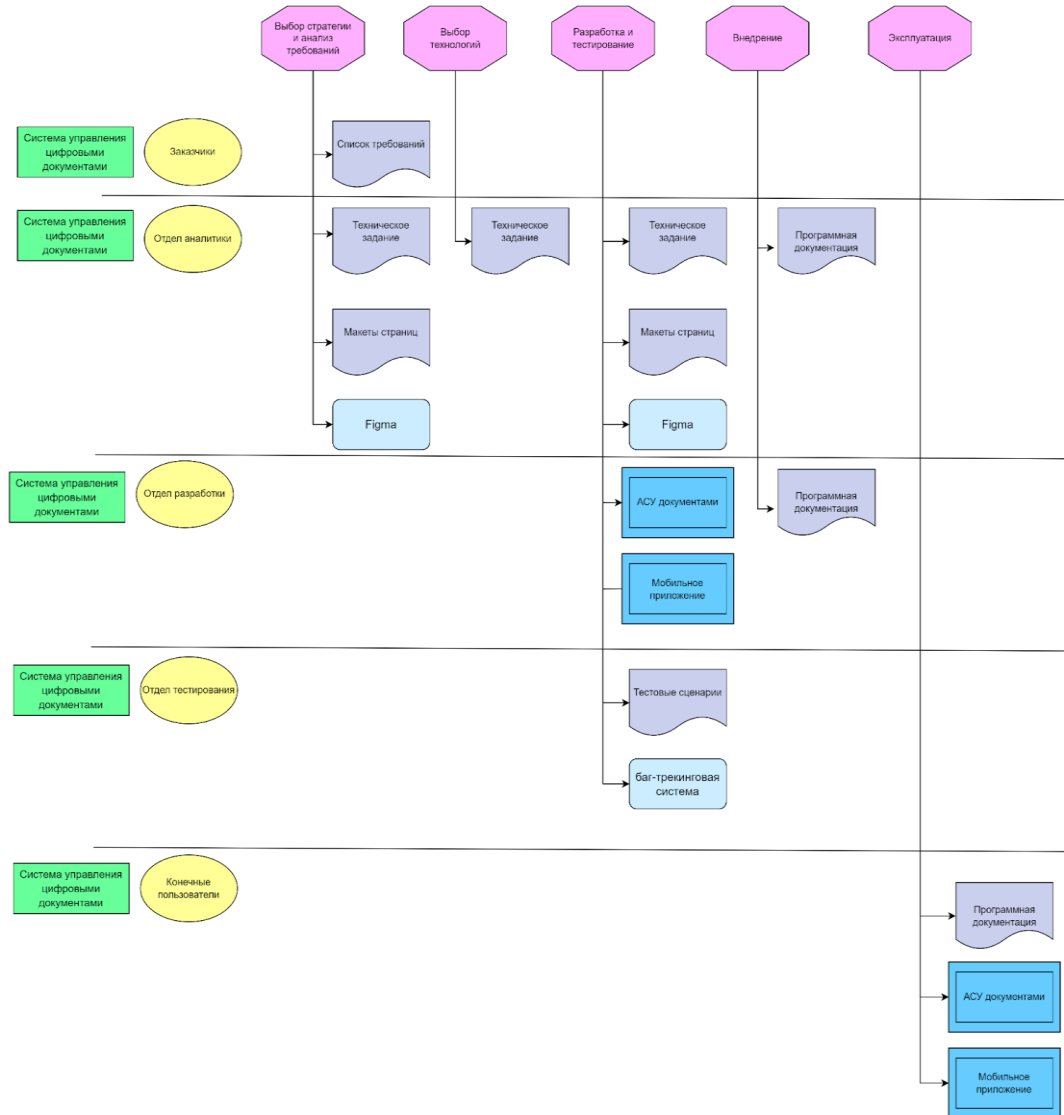


Рис. 1. Хронологическая архитектура системы управления цифровыми документами

После построения хронологической архитектуры системы управления цифровыми документами, мы можем распределить нагрузку для каждого участника разработки, определить основные документы и инструменты, которые потребуются на каждом этапе разработки системы, эффективно управлять проектом создания СУЦД.

Литература

1. Забегалин Е.А. Сборник методических рекомендаций по определению и моделированию архитектуры автоматизированных информационных систем в консалтинговых проектах. Департамент управленческого консалтинга компании IBS. Москва, 2006.

Хорев А.А.

Научный руководитель: к.т.н. С.В. Комкова

*Муромский институт (филиал) федерального государственного образовательного учреждения высшего образования «Владимирский государственный университет имени Александра Григорьевича и Николая Григорьевича Столетовых»
602264, г. Муром, Владимирская обл., ул. Орловская, 23*

Разработка мобильного приложения по поиску пропавших животных

На заре становления мобильные устройства не были компактных размеров, их использование не представлялось возможным без доступа к электросети, а высокая стоимость такого устройства сильно ограничивала его распространение среди населения. Мобильные технологии с тех пор сделали большой шаг вперед, однако качественный рост технологий в этой сфере произошел за последнее десятилетие. Сегодня мобильные устройства компактны, удобны, многофункциональны, производительны и их роль в жизни каждого человека сильно выросла. Такое распространение смартфонов открыло широкие возможности мобильной разработки.

Проблема бездомных животных и поиска домашних животных актуальна во всем мире. Эта проблема имеет свои определенные последствия: бездомные животные являются переносчиками болезней и угрозой человеческой безопасности. В случае с домашними животными, если они попадают из дома на улицу, то, вероятнее всего они не способны к ней адаптироваться. Разрабатываемое мобильное приложение должно будет упростить и ускорить процесс нахождения домашних животных и поиска новых хозяев для бездомных животных.

Необходимо создать мобильное приложение, в котором будут реализованы следующие функции:

- Вход и регистрация пользователя;
- Личный кабинет;
- Размещение объявлений о потерянном или замеченном животном;
- Размещение объявлений о передержке, выгуле домашнего животного и т.п.

Будущий программный продукт будет использовать базу данных, которая включает в себя таблицы представлений, которые будут созданы в ходе создания и разработки мобильного приложения.

В качестве интегрированной среды разработки программного продукта был выбран Python и среда разработки Microsoft Visual Studio Code. Он поддерживает подсветку синтаксиса, IntelliSense, рефакторинг, отладку, навигацию по коду, поддержку системы контроля версий Git и другие возможности.

Шаблов Д.В., Семёнов А.С.

*Научный руководитель: преподаватель Крашенинникова Е.Н.
ГБПОУ ВО "Муромский колледж радиоэлектронного приборостроения"
602267, Владимирская область, г. Муром, ул. Комсомольская, д.55
krash.ab.nik@mail.ru*

Инженерная графика – путь к успеху. Мотивационно-юмористический короткометражный фильм

При обучении может наблюдаться такая ситуация, когда понимание важности и нужности освоения предмета заслоняется пресловутым «надо». От этого страдает качество получения знаний и умений. Видя такую проблему, мы захотели попробовать решить её необычным и получившим сейчас популярность, способом-с помощью видео. В качестве значимого предмета выступает инженерная графика. Известно, что она оказывает колоссальное влияние на нашу жизнь, принимает самую важную роль во многих сферах жизни человека. А некоторые области деятельности невозможно даже представить без инженерной графики. При этом не многие задаются вопросом, почему инженерная графика так важна, в чем её успех как науки и как с её помощью человек достигнет своего успеха. Для того чтобы лучше в этом разобраться, попробовать развить интерес к её познанию у себя и других, мы решили создать короткометражный фильма-ролик. Он является совмещением познавательной, научной, интересной информации, добытой нами в ходе работы и небольшого оттенка юмора для осознания, что освоение сложного предмета может быть легким и весёлым. Главной целью нашего видео является создание мотивации у людей, студентов, наших товарищей заниматься инженерной графикой с интересом, попробовать самим поучаствовать в процессе обучения других новым, ненавязчивым методом.

В нашем видео мы расскажем и покажем, как и где зародилась инженерная графика, как она повлияла на историю всего человечества и как последние несколько тысяч лет инженерная графика идёт к своему успеху и как с её помощью достичь успеха. Вы заинтригованы? Тогда смотрите

Литература

- 1.Мясоедова Н.В., Леонова Л.М., Притыкин Ф.Н., Кошелева Л.И. Инженерная графика (геометрическое и проекционное черчение)/ Омск: ОмГТУ,2005.-1.-С. 2-3, 16-19-52с.
- 2.Никулин Е.А. Компьютерная геометрия и алгоритмы машинной графики.-СПб.: БХВ-Петербург, 2003.-560с.-3000 экз.-ISBN 5-94157-264-6.
- 3.Цифровое постпроизводство//The Essential Reference Guide for Filmmakers = Краткий справочник кинематографиста.-Rochester.: Eastman Kodak, 2007.-С.167-185.-214с.

Щербаков М.А.

*Научный руководитель: к. т. н., доц. каф. ИС Подгорнова Ю.А.
Муромский институт (филиал) федерального государственного образовательного
учреждения высшего образования «Владимирский государственный университет
имени Александра Григорьевича и Николая Григорьевича Столетовых»
602264, г. Муром, Владимирская обл., ул. Орловская, 23*

Разработка системы поддержки принятия решений для создания и анализа инвестиционного портфеля

В настоящее время люди все больше стали погружаться в мир инвестиций, поскольку большинство из них не имеют достаточного капитала для вложения в недвижимость или мелкий бизнес, они начинают инвестировать в фондовый рынок [1].

Заработать на фондовом рынке можно тремя способами:

- 1) купить ценную бумагу по низкой цене и продать её через некоторое время по более выгодной;
- 2) собрать портфель из дивидендных акций и получать доход каждый сезон в виде дивидендов компаний;
- 3) собрать портфель из недивидендных акций в фиксированном процентном соотношении по стоимости каждой бумаги и фиксировать прибыль той акции, стоимость которой превысила указанный ранее процент.

Актуальность работы заключается в том, что в последние годы инвестирование в фондовый рынок становится все более популярным и является конкурентной альтернативой привычному виду вложения денег – банковским вкладам

Целью данной работы является разработка системы поддержки принятия решений для создания и анализа инвестиционного портфеля.

Для достижения поставленной цели будут решены следующие задачи:

- 1) анализ предметной области;
- 2) обзор аналогов разрабатываемой системы поддержки принятия решений;
- 3) разработка алгоритмов создания и анализа инвестиционного портфеля;
- 4) проектирование информационной системы;
- 5) реализация программы.

«Тинькофф Инвестиции» - это брокер, который является сервисом для вложения денежных средств, запущенный в октябре 2016 года. Пользователям доступны онлайн операции с акциями, облигациями, фондами и драгоценными металлами [5].

Python — это высокоуровневый язык программирования, который используется в различных сферах IT, таких как машинное обучение, разработка приложений, web, парсинг и другие [2].

PyQt5 - это набор Python библиотек для создания графического интерфейса на базе платформы Qt5 от компании Digia. Он доступен для Python 2.x и 3.x. Этот учебник использует Python 3. Библиотека Qt является одной из самых мощных библиотек GUI (графического интерфейса пользователя) [3].

Литература

1. Статья по фондовому рынку // URL: <https://bcs.ru/blog/kak-zarabotat-na-akciyah-i-cennyh-bumagah> (дата обращения 04.04.2022)
2. Все о Python // URL: <https://all-python.ru/osnovy/yazyk-programmirovaniya.html> (дата обращения 04.04.2022)
3. PyQt5: первые программы // URL: <https://pythonworld.ru/gui/pyqt5-firstprograms.html> (дата обращения 04.04.2022)
4. Инвестиции для новичков // URL: <https://journal.open-broker.ru/investments/zachem-nuzhny-investicii/> (дата обращения 04.04.2022)
5. Документация по Tinkoff Invest // URL: <https://www.tinkoff.ru/invest/> (дата обращения 04.04.2022)