

Министерство образования и науки Российской Федерации
Федеральное государственное бюджетное
образовательное учреждение
Высшего профессионального образования
Алтайский государственный технический университет
им. И.И.Ползунова



НАУКА И МОЛОДЕЖЬ – 2015

**XII Всероссийская научно-техническая конференция
студентов, аспирантов и молодых ученых**

СЕКЦИЯ

ИНФОРМАЦИОННЫЕ ТЕХНОЛОГИИ

подсекция

ПРОГРАММНАЯ ИНЖЕНЕРИЯ

Барнаул – 2015

УДК 004

XII Всероссийская научно-техническая конференция студентов, аспирантов и молодых ученых "Наука и молодежь – 2015". Секция «Информационные технологии». Подсекция «Программная инженерия». / Алт. гос. техн. ун-т им. И.И.Ползунова. – Барнаул: изд-во АлтГТУ, 2015. – 70 с.

В сборнике представлены работы научно-технической конференции студентов, аспирантов и молодых ученых, проходившей 23 апреля 2015 г.

Редакционная коллегия сборника:

Кантор С.А., заведующий кафедрой «Прикладная математика» АлтГТУ – руководитель секции, Крючкова Е.Н., профессор, зам. зав. кафедрой ПМ, Сорокин А.В., доцент каф. ПМ, ответственный за НИРС на кафедре ПМ

Научный руководитель подсекции: к.ф.-м.н., профессор, Кантор С.А.

Секретарь подсекции: к.т.н., доцент, Сорокин А.В.

Компьютерная верстка: Сорокин А.В.

© Алтайский государственный технический университет им. И.И.Ползунова

СОДЕРЖАНИЕ

Смирнова А.Ю., Крючкова Е.Н. Автоматическое аннотирование научных текстов	5
Горгосов А.А., Троицкий В.С. Автоматическое создание некоторых сложных элементов геологических чертежей в CorelDraw	7
Шипулин Т.С., Троицкий В.С. Анализатор операционного журнала RS-BANK.....	8
Казанцев М.В., Голубятников В.П. Дискретные модели некоторых нелинейных динамических систем.....	9
Савченко В.В., Крючкова Е.Н. Семантическая обработка текстов на естественном языке .	10
Равицкий М.К., Крайванова В.А. Задача восстановления порядка действий в описаниях алгоритмов на ЕЯ.....	13
Игошин А.А., Крайванова В.А. Семантическая сеть технологий для сайта Nirsproject.ru ...	15
Мягков В.В., Крючкова Е.Н. Разработка алгоритма анализа схожести структуры веществ по микрофотографиям.....	17
Гавриченков Е.А., Крючкова Е.Н. Обработка микрофотографий срезов образцов сварки взрывом	21
Самойлов С.С., Старолетов С.М. Проектирование и реализация обучающей игры жанра «programminggame»	23
Хатников А.С., Андреева А.Ю. Разработка подхода к организации передачи информации с мобильного приложения с использованием Tin Can API.....	26
Ворона О.И., Леонов С.Л. Анализ параметров дифференциальной диагностики.....	28
Фаст А.С. Организация удаленного управления на компьютерах с операционной системой семейства MS Windows	30
Никитин А.А., Старолетов С.М. Искусственный интеллект в моделировании поведения игровых персонажей на основе подхода «behavior tree»	33
Есипенко С.П., Крючкова Е.Н. Применение алгоритма фильтрации частиц в задаче отслеживания позы человека.....	37
Бобков К.А., Крючкова Е.Н. Анализ геометрической структуры изображения в задаче распознавания.....	41
Ненайденко А.С., Боровцов Е.Г. Опыт прогнозирования успеваемости студентов на основе статистических данных	44
Лаптев М.А., Старолетов С.М. Визуальная среда разработки интерактивных новелл	47
Чураков М., Старолетов С.М. Система электронного учета посещаемости занятий на основе RFID считывателя Arduino	50
Некрасов Д.В., Старолетов С.М. Фреймворк для построения систем обнаружения движения и трекинга объектов.....	51

Колесников Н., Мелехов Д., Старолетов С.М. Автоматизированная система врача-ветеринара.....	54
Балакин С., Нгуйен Д., Сучкова Л.И., Старолетов С.М. Клиент-серверная система помощи бездомным животным	55
Макаров А.В., Чуешев А.В., Крайванова В.А. Симулятор процессов жизнедеятельности пасечного хозяйства	57
Чуешев А.В., Крайванова В.А., Андреева А.Ю. Исследование возможностей LMS ILLIAS по интеграции с мобильным приложением.....	59
Шахворостова Д.Н., Овчаров А.А., Винокуров С.В., Сорокина Е.П., Рогозин К.И., Сорокин А.В. Пример использования физических апплетов в дистанционном обучении физике	62
Овчаров А.А., Соловьенко А.А., Винокуров С.В., Рогозин К.И., Сорокин А.В. Тестирующая система по физике	67

АВТОМАТИЧЕСКОЕ АННОТИРОВАНИЕ НАУЧНЫХ ТЕКСТОВ

Смирнова А.Ю. – магистрант, Крючкова Е.Н. – к.ф.-м.н., профессор
Алтайский государственный технический университет (г.Барнаул)

В основе современного мира лежит информация, большая часть которой представлена в электронном виде. На сегодняшний день Всемирная паутина содержит миллиарды документов, и это количество увеличивается с экспоненциальной скоростью. Естественно, человек физически не в состоянии обрабатывать и усваивать информацию в таких объемах. Здесь на помощь человеку приходят различные инструменты для обработки информации, например, поисковые системы. Однако хотелось бы также иметь инструмент, позволяющий быстро классифицировать источник информации как релевантный. Так родилась идея аннотирования. Оно позволяет максимально сократить размер информационного потока за счет выделения основных идей и мыслей в тексте. Таким образом, человеку не нужно обрабатывать огромные массивы информации лишь для того, чтобы принять решение о том, имеет ли она отношение к предмету его интереса, что значительно повышает продуктивность его деятельности. Разумеется, учитывая объемы информации, хотелось бы автоматизировать этот процесс.

Обычно, под аннотацией понимают краткое изложение содержания документа с общим представлением о его теме. Существует множество критериев, по которым можно выделить различные виды аннотаций [1, 2]:

- по степени детализации (индикативная / информативная);
- по способу получения (генеративные / экстрактивные);
- по количеству источников (на основе одного / нескольких документов);
- по внешнему представлению (короткий текст / набор ключевых слов / заголовков / ...);
- по степени зависимости от контекста (общие / тематически-ориентированные);

Методы получения аннотаций в свою очередь можно разделить на генерирующие и извлекающие [2].

В настоящее время основные исследования сосредоточены в области экстрактивных аннотаций. Их основной особенностью является то, что они состоят из фрагментов исходного текста, причем фрагменты берутся в том же виде и порядке, в котором они были приведены изначально. Среди методов их получения выделяются методы на основе машинного обучения и методы на основе графов [2]. Однако при составлении таких аннотаций может быть использован и статистико-вероятностный подход. Он нашел свое применение, например, в решении задачи автоматического аннотирования новостного потока [3], где аннотация строилась на основе тематического представления текста в виде тематических узлов. Такой подход дает весьма хорошие результаты при составлении новостной ленты в различных поисковых системах, однако не подходит для составления аннотаций к научным текстам. Это обусловлено, в первую очередь, иной структурой научной работы и ее размерами.

Извлекающие методы, в целом, показывают очень высокие результаты, однако обладают одним существенным недостатком – они используют цельные предложения из текста, вследствие чего аннотация получается громоздкой и не всегда охватывает все важные моменты, упомянутые в тексте. Именно поэтому в данной работе предполагается разработка генерирующего метода аннотирования.

Генерирующие методы применяются не столь широко, потому что включают в себя сложные подзадачи генерации связного текста и семантического анализа. Однако, в том или ином виде, в некоторых конкретных случаях их удалось разрешить. В большинстве своем они применяются лишь в очень узких предметных областях и не являются универсальными. В качестве яркого примера можно привести работу [4], в которой приведен вариант решения

задачи аннотирования больших массивов числовых данных в приложении к газовым турбинам.

Проблема генерации текста на естественном языке на данный момент является сложной и актуальной проблемой. Ситуация усугубляется тем, что большинство исследований в этой области ведутся за рубежом, и, учитывая специфику, неприменимы для русского языка. В данной работе мы не будем затрагивать этот вопрос.

Целью нашей работы является создание и последующая реализация алгоритма, позволяющего на основе имеющегося текста получить краткую аннотацию к нему на уровне ключевых слов, объектов и связей между ними. Основным отличием предлагаемого алгоритма от уже существующих является попытка использования уже имеющихся знаний о мире при анализе текста. Предполагается использовать подход к составлению аннотаций, позволяющий выделить основные моменты текста путем анализа связей семантического графа.

Для любого текста может быть построен семантический граф. В качестве вершин в нем выступают слова, а в качестве ребер – связи между ними. В системе АОТ[5], к примеру, используются 35 различных связей, начиная от связи «Признак» для связи существительного и прилагательного, и заканчивая связью «Способ» для связи глагола и наречия. Выделяется огромное множество видов связей между существительными (адресат, имя, значение и т.д.).

На основе имеющейся информации можно построить новый граф, в вершинах которого будут расположены понятия, а ребра между вершинами будут обозначать наличие связи между понятиями в тексте. Каждой вершине поставим в соответствие число – частотность, с которой понятие встречается в тексте. Для уменьшения количества вершин будем объединять вершины по признаку синонимии. Также, при наличии большого количества связей в какой-либо из вершин, можно пытаться объединять связанные вершины по принципу гипонимии. Каждому ребру можно присвоить вес, соответствующий количеству упоминаний связи в тексте.

Полученный граф несет в себе следующую важную информацию:

- Вершины с большим весом являются предметом обсуждения в тексте.
- Сильно связанные компоненты показывают ключевые связи между понятиями и, с большой вероятностью, отражают основную мысль текста.

На следующем шаге алгоритма предполагается восстановить связи между вершинами сильно связанных компонентов и подобрать наиболее частотные определения к вершинам с наибольшим весом. В результате получим описание основной идеи текста на понятийном уровне.

В дальнейшем планируется подробнее рассмотреть варианты связей между словами и их влияние на основную идею. На основе этой информации можно ввести более совершенную систему весов для построенного графа и сократить объемы анализируемой информации. Предполагается, что данный подход значительно повысит качество генерируемых аннотаций.

Список литературы

1. Nenkova A. Automatic Summarization: Foundations and Trends in Information Retrieval / A. Nenkova, K. McKeown – 2011. Vol. 5, Nos. 2-3 (2011), pp. 103-233.
2. Осминин П.Г. Современные подходы к автоматическому реферированию и аннотированию // Вестник Южно-Уральского государственного университета. Серия: Лингвистика. —2012 - №25.
3. Лукашевич, Н. Исследование тематической структуры текста на основе большого лингвистического ресурса [Текст] /Н. Лукашевич, Б. Добров //Труды Международной конференции «Диалог-2000». М. 2000.

4. Chasing the content of textual summaries of large time-series data sets / J.Yu, E. Reiter, J. Hunter, Chris Mellish // Natural Language Engineering. 2007. Vol. 13, №1. P. 25-49.
5. Автоматическая Обработка Текста. [Электронный ресурс]. – Режим доступа: <http://aot.ru/>, свободный.

АВТОМАТИЧЕСКОЕ СОЗДАНИЕ НЕКОТОРЫХ СЛОЖНЫХ ЭЛЕМЕНТОВ ГЕОЛОГИЧЕСКИХ ЧЕРТЕЖЕЙ В CORELDRAW

Горгосов А.А. – студент, Троицкий В.С. – к.ф.-м.н., доцент
Алтайский государственный технический университет (г. Барнаул)

Проектирование и проведение геологоразведочных работ связано с выполнением большого числа разнообразных чертежей, схем, карт и прочих графических документов. Для этого существуют специализированные среды (AutoCad и др.), но для освоения они весьма сложны и их применение для простой графики лишено смысла. Исполнителям гораздо проще работать с CorelDraw, тем более, что исходный материал представляется заказчиками обычно в виде cdr-файлов.

Так как CorelDraw это, прежде всего, инструмент графического дизайна, то выполнение в нем некоторых элементов геологических чертежей является весьма трудоемким процессом. Например, наиболее распространенный элемент чертежа, который весьма сложно выполнить в CorelDraw, это "уступ". Это элемент, состоящий из двух кривых и набора радиально направленных отрезков между ними. Количество, длина и расположение отрезков может быть различным. Иногда, его изображают в виде одной кривой и набора отрезков (коротких и в виде буквы "Т").

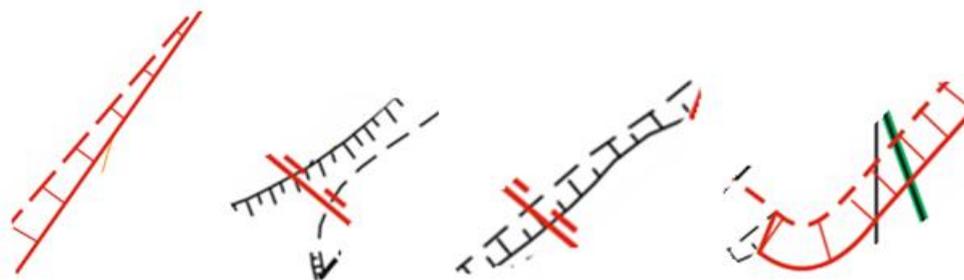


Рисунок 1 – Примеры чертежа

Чтобы изобразить этот элемент, необходимо нарисовать как минимум две кривых (делается достаточно легко и просто) и множество радиальных отрезков. Число этих отрезков велико (десятки, а чаще всего несколько сотен), соответственно, трудозатраты на их создание очень велики.

Нами разрабатывается программное обеспечение для решения этой проблемы. Программа позволит на основе двух опорных кривых, созданных пользователем, создать "уступ", автоматически дорисовав все радиальные отрезки, с заданными характеристиками. Средство разработки - язык Visual Basic, встроенный в CorelDraw.

Планируется развитие ПО с целью автоматизации других трудновыполнимых в этом редакторе элементов геологических чертежей.

АНАЛИЗАТОР ОПЕРАЦИОННОГО ЖУРНАЛА RS-BANK

Шипулин Т.С. – студент, Троицкий В.С. – к.ф.-м.н., доцент
Алтайский государственный технический университет (г. Барнаул)

Автоматизированная банковская система RS-Bank наиболее распространенная в России система комплексной автоматизации деятельности банка. Одним из элементов обеспечения информационной безопасности является механизм журнализации действий пользователей. В RS-Bank он называется операционный журнал.

Все действия пользователей отражаются в специальном журнале, где последовательно сохраняются записи о каждой выполненной операции каждого пользователя. Запись осуществляется последовательно в том порядке, в котором вносятся изменения в таблицы базы данных. В качестве событий, которые регистрируются в журнале, могут выступать как некоторые синтетические процедуры, например вход оператора в систему, выход из системы, запуск отчета или вызов пункта меню, так и простые изменения записей в той или иной таблице базы данных.

Физически операционный журнал из себя представляет обычную таблицу базы данных. Она содержит поля для хранения даты и времени операции, кем и с какой рабочей станции она была выполнена, вид этой операции и набор дополнительных реквизитов.

Операционный журнал является основным инструментом в расследовании различных спорных ситуаций и конфликтов. Штатные средства RS-Bank позволяют восстановить все действия конкретного оператора, или все операции по конкретному объекту системы за заданный промежуток времени. Это делается при помощи механизма фильтрации записей журнала. Для расследования тех или иных конфликтов этого вполне достаточно. Но существуют ситуации, когда нужно понять, что происходило в заданный период времени в системе в целом. Т.е. нужно знать какие операторы работали в системе, какие операции они проводили, сколько таких операций было выполнено, были ли ошибки во время выполнения операций, и много ли их было и т.п. Причем, желательно эту информацию получить в наглядном виде, лучше в виде графиков, диаграмм или схем. К сожалению, штатные средства RS-Bank не позволяют это сделать за приемлемое время и без значительных трудозатрат. В основном, это связано с линейной структурой операционного журнала, плохо согласующейся с реальной работой операторов которую нам нужно отобразить. Операторы работают параллельно, и в каждый момент времени в системе выполняется множество операций. Конечно, в операционном журнале отмечены моменты начала выполнения этих операций и их окончания, ошибки и прочее, но отследить это при помощи штатных фильтров практически невозможно.

Для решения этой проблемы мы разработали программу, позволяющую восстановить картину параллельной работы операторов RS-Bank по линейной структуре журнала операций. Основная идея заключается в объединении отдельных записей журнала операций в специальные структуры, называемые "контейнер". Каждый "контейнер" объединяет записи операционного журнала для некоторого оператора RS-Bank и умеет "проиграть" их, эмулируя работу этого оператора. Если все контейнеры пользователей будут собраны в список, то появляется возможность "проиграть" действия всех работающих операторов одновременно, эмулируя работу системы в целом за любой временной промежуток. В дальнейшем программа будет доработана для визуализации результатов работы в наглядном виде (в виде графиков, диаграмм или схем).

ДИСКРЕТНЫЕ МОДЕЛИ НЕКОТОРЫХ НЕЛИНЕЙНЫХ ДИНАМИЧЕСКИХ СИСТЕМ*

Казанцев М.В. – аспирант, Голубятников В.П. – д.ф.-м.н., профессор
Новосибирский государственный университет (г. Новосибирск)

Периодические режимы функционирования генных сетей и их математических моделей представляют немалый интерес в задачах биоинформатики. Для поиска циклических траекторий в задаваемых этими моделями динамических системах широко используются методы численного моделирования. Мы рассматриваем один из способов уменьшить область поиска циклических траекторий за счет построения дискретных моделей генных сетей специального вида.

Мы рассматриваем n -мерные динамические системы

$$\dot{x}_1 = f_1(x_n) - x_1, \quad \dot{x}_i = f_i(x_{i-1}) - x_i, \quad i \in \{2 \dots n\}, \quad x_j \geq 0$$

в качестве моделей циклических генных сетей. f_i – неотрицательные гладкие или кусочно-постоянные функции, определенные на положительной полуоси. Мы предполагаем, что часть из них является невозрастающими (L-функции), а часть --- неубывающими (Г-функции). Они соответствуют положительным и отрицательным обратным связям в генной сети. Рассмотрим стационарную точку x^* данной системы. Мы разбиваем инвариантную окрестность Q точки x^* на 2^n параллелепипедов гиперплоскостями, параллельными координатным гиперплоскостям и содержащими x^* . Построенным параллелепипедам присваиваются двоичные индексы $\varepsilon \in B^n$, $\varepsilon_i = 0$ если $0 \leq x_i \leq x_i^*$, $\varepsilon_i = 1$ если $x_i^* < x_i$.

Рассмотрим ориентированный граф G , вершины которого соответствуют параллелепипедам, полученным при разбиении Q , а направления дуг между ними соответствуют направлениям векторного поля, заданного динамической системой. В дальнейшем будем называть граф G **графом доменов**. Исходящую степень его вершины будем называть **потенциалом** соответствующего ей домена, а совокупность всех вершин с одинаковым потенциалом --- **потенциальным уровнем**. Граф доменов обладает рядом свойств, которые позволяют сократить область поиска циклов в динамической системе. В частности, циклическая траектория в динамической системе существует только в том случае, если существует цикл в G , проходящий через те же домены, что и эта траектория. В то же время, цикл в графе доменов может существовать только в том случае, если все входящие в него вершины принадлежат одному потенциальному уровню.

Нами установлено, что при заданной размерности системы n и одинаковой четности числа Г-функций среди f_i графы доменов этих систем изоморфны. Данное свойство позволяет при исследовании графов доменов систем высокой размерности рассматривать только две различные системы (с четным и нечетным числом Г-функций), поскольку любой другой граф доменов той же размерности изоморфен графу одной из них.

Разрабатываемое программное обеспечение *StateTransitionDiagramVisualizer* позволяет строить и визуализировать графы доменов динамических систем произвольной размерности. Предусмотрена возможность задания характера функций f_i (L или Г). Программа автоматически разделяет домены на потенциальные уровни и предоставляет возможность

* Работа выполнена при финансовой поддержке Российского фонда фундаментальных исследований (проект 15-01-00745).

наглядно изучать существующие на них циклы. Разработка ведется с использованием языка программирования Java. Предполагается дальнейшее развитие данного программного обеспечения, чтобы сделать возможным изучение более сложных классов динамических систем (в частности, с унимодальными f_i).

СЕМАНТИЧЕСКАЯ ОБРАБОТКА ТЕКСТОВ НА ЕСТЕСТВЕННОМ ЯЗЫКЕ

Савченко В.В. – аспирант, Крючкова Е.Н. – к.ф.-м.н., профессор
Алтайский государственный технический университет (г. Барнаул)

Введение

В условиях стремительно растущего объема информации актуальна задача поиска в больших текстовых коллекциях [1]. Одним из вариантов поиска является семантический поиск, т.е. поиск по смыслу содержащейся в тексте информации [2,3,4]. Это, так или иначе, приводит к необходимости семантической обработки текстов. Одним из вариантов такой обработки можно назвать семантическое сравнение текстов.

Среди наиболее популярных систем семантического поиска можно выделить Google, SearchMonkey, Ask Jeeves, Freebase и AskNet. Однако они имеют определенные недостатки, например, применение семантики лишь для незначительного улучшения результатов поиска, ограничение на длину запроса, снижение качества поиска с увеличением поискового запроса.

Изменчивость синтаксических конструкций и вариативность лексики естественных языков, разнообразие стилей изложения материала – существенно усложняют решение данной задачи.

Семантическая обработка

В данной работе представлен результат разработки метода семантического сравнения текстовых коллекций на русском языке. Ключевая особенность предлагаемой системы – снятие ограничений на величину текстовых коллекций.

Исходными данными являются текстовые коллекции и запрос пользователя (другая текстовая коллекция). Исходя из предположения, что большая текстовая коллекция неоднородна по своему содержанию и в качестве результата сравнения интересна ее определенная часть, текст можно разделить на фрагменты. Как правило, такие фрагменты – это страницы, абзацы или наборы из нескольких предложений. Фрагменты будем называть «окнами». Поэтому фактически задача сводится к поиску одного или нескольких окон.

Условимся обозначать граф как $G=(V, U)$, где V – множество вершин, U – множество ребер. Для каждого окна запроса и поисковых коллекций строится граф семантических связей G_{sem} – "семантический граф". Семантический граф представляет собой ориентированный граф, вершинами которого являются слова русского языка, представленные в нормальной форме, а ребра характеризуются весом и типом семантической связи. Эквивалентными считаются ребра, имеющие один и тот же тип семантической связи и соединяющие соответствующие вершины в том же порядке. Для построения семантического графа каждое предложение из окна коллекции обрабатывается семантическим анализатором. В данной работе используется семантический анализатор RML [5].

Предложения окна обрабатываются последовательно, для каждого очередного предложения строится семантический граф G_{new} . Все ребра семантического графа G_{new}

имеют вес равный единице. На каждой итерации семантический граф G_i предыдущей итерации складывается с графом G_i^{new} обрабатываемого предложения согласно формуле:

$$G_{i+1} = G_i + G_i^{new}, \quad i = 1, \dots, n, \quad (1)$$

Далее необходимо подсчитать величину коэффициента соответствия (релевантности) семантического графа запроса и семантического графа окна. Простой поиск наибольшего общего подграфа, даже с учетом совпадения не только вершин, но и типов ребер, не приведет к цели, так как один и тот же смысл содержится в текстах разного стилового оформления, вершины графа запроса и графа окна коллекции могут быть не только синонимами, но и каким-либо образом связанными по смыслу словами, могут содержать обобщающие сведения или только частичную информацию.

Для поиска связанных по смыслу слов был использован словарь, полученный методом автоматической обработки краткого толкового словаря, в котором представлен перечень слов в нормальной форме [6]. Каждому слову сопоставлен набор слов, связанных с ним ассоциативной, синонимичной и т. д. связью. Таким образом, словарь представляет собой направленный граф $G_{word} = (V_{word}, U_{word})$, где вершины V_{word} - это слова в нормальной форме, а ребра U_{word} имеют действительные весовые коэффициенты от 0 до 1. Назовем граф G_{word} графом справочника. За коэффициент связности слов a_k и a_m - вершин семантического графа запроса $G_{request}$ и семантического графа окна коллекции G_{sem} , возьмем произведение весов от таких же слов до общего предка в графе справочника. При совпадении слов данный коэффициент будет равен 1, иначе будет принадлежать промежутку $[0;1]$, т.к. веса ребер графа справочника действительные числа от 0 до 1. Пусть $i_0 i_1 \dots i_q$ - путь от вершины a_k до вершины t , $j_0 j_1 \dots j_n$ - путь от вершины a_m до вершины t , где t - ближайший общий предок. Тогда коэффициент связности:

$$P(a_k, a_m) = P(a_m, a_k) = \prod_{i=1}^q p_{i-1,i} * \prod_{j=1}^n p_{j-1,j}, \quad (2)$$

где $p_{i-1,i}$ - вес ребра соединяющего вершины $i - 1$ и i , $p_{j-1,j}$ - вес ребра соединяющего вершины $j - 1$ и j .

Замечание: Граф справочника является упрощенной моделью наших знаний о реальном мире, а общий предок слов a_k и a_m в этом графе представляет собой некоторое обобщение соответствующих понятий. Нет смысла искать общего предка слов во всем графе справочника, т.к. произведение ребер на определенном удалении не превысит заданного θ , а это означает, что слова a_k и a_m связаны слабо. Следовательно, нас интересует некое θ окружение искомых слов. В противном случае считаем, что слова никак не связаны по смыслу, и коэффициент соответствия между ними равен 0. Фрагмент графа G_{word} представлен на рисунке 1.

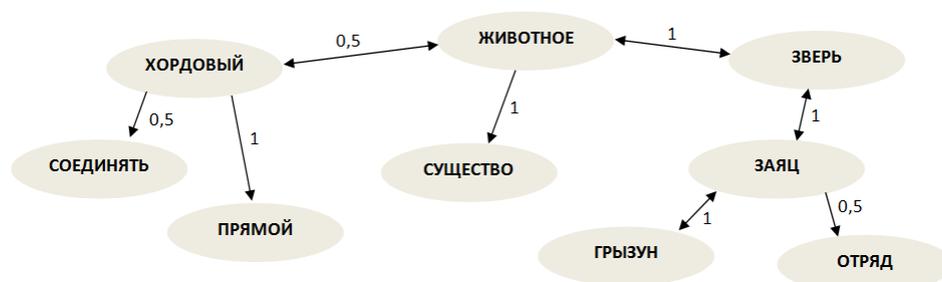


Рисунок 1 – Граф справочника

На следующем этапе ищем наилучшее совпадение семантического графа запроса с графом окна. Значение коэффициента соответствия при совпадении графов рассчитываем по формуле:

$$S = \sum_{k=1}^n D1_k * D2_k * L1_k * L2_k, \quad (3)$$

где n – количество совпавших ребер графа запроса и окна, k – индекс соответствующего ребра, $D1_k$, $D2_k$ – произведение весов ребер в графе справочника от слова запроса и слова окна до общего предка соответственно, $L1_k$ – вес ребра k семантического графа запроса, $L2_k$ – вес ребра k семантического графа окна.

Вариантов совпадения подграфов много, но нас интересует максимальное значение коэффициента соответствия. Определив максимальное значение по всем окнам текстовой коллекции, мы получим общее значение – коэффициент релевантности запроса и текстовой коллекции.

Экспериментальная проверка

Для оценки полученной системы были отобраны текстовые коллекции большого объема, удовлетворяющие одному и тому же запросу к поисковой системе google.ru. Поисковый запрос строился исходя из содержания определенных коллекций. В зависимости от настроек системы были получены различные значения коэффициента релевантности. Однако для коллекций, по содержанию которых строился значение коэффициента минимум на порядок превосходило значение коэффициента других коллекции, не похожих по содержанию.

Затем для проверки алгоритма были обработаны электронные версии журналов из списка ВАК находящиеся в открытом доступе: Биомедицинская химия, Биомедицина, География и природные ресурсы, Вестник Воронежского государственного университета (Серия: География. Геоэкология). Общий объем текстовой базы составил немногим более двух гигабайт данных. При этом экспертная оценка подтвердила корректность алгоритма.

Временные затраты на определение коэффициента релевантности (сравнение семантических графов) приведены на рисунке 3.

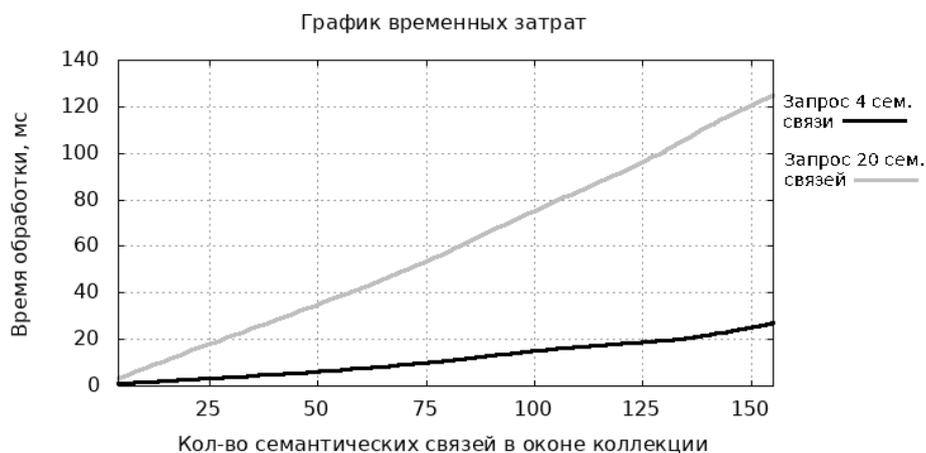


Рисунок 3 – Время обработки

График временных затрат подтверждает полиномиальную сложность алгоритма, что в свою очередь делает разработанную систему вполне приемлемой для семантического поиска в больших текстовых коллекциях. Кроме того, определенные этапы обработки, например, этап определения коэффициента релевантности могут выполняться параллельно, что приведет к увеличению производительности на многоядерных и многопроцессорных системах.

Список литературы

1. Hannah Bast, Marjan Celik Efficient Fuzzy Search in Large Text Collections // ACM Transactions on Information Systems, 2010.
2. Mathieu d'Aquin, Enrico Motta Watson, more than a Semantic Web search engine // IOS Press Amsterdam, 2011.
3. Elbedweihy K., Wrigley S.N., Ciravegna F., Reinhard D., Bernstein A. Evaluating Semantic Search Systems to Identify Future Directions of Research // Second International Workshop on Evaluation of Semantic Technologies, 843, 2012, page 25-36.
4. Tsoumakas G., Laliotis M., Markantonatos N., Vlahavas I. Large-Scale Semantic Indexing of Biomedical Publications at BioASQ // BioASQ Workshop, 2013.
5. Автоматическая Обработка Текста / [Электронный ресурс] Режим доступа: <http://www.aot.ru/>
6. Крайванова В.А., Кротова А.О., Крючкова Е.Н. Построение взвешенного лексикона на основе лингвистических словарей // Материалы Всероссийской конференции с международным участием ЗОНТ-2011, Т.2, Новосибирск, 2011. - с. 32-38

ЗАДАЧА АВТОМАТИЧЕСКОЙ ИЛЛЮСТРАЦИИ АЛГОРИТМОВ ПО ИХ ОПИСАНИЯМ НА ЕСТЕСТВЕННОМ ЯЗЫКЕ

Равицкий М.К. – магистрант, Крайванова В.А. – к.ф.-м.н., доцент
Алтайский государственный технический университет (г. Барнаул)

На сегодняшний день человечество накопило огромный объем знаний, представленных в виде текстов на естественном языке. К сожалению, рост объема информации приводит к снижению ее понятности и доступности. Кроме того, текст имеет крайне высокий уровень абстракции, что требует от читателя значительных усилий для восприятия информации. Это свойство особенно ярко выражено в бюрократических и научных документах, когда простота текста приносится в жертву точности формулировок. В связи с этим актуальной становится задача выделения в тексте фрагментов определенного типа (алгоритмов, свойств объектов, зависимостей), позволяющие читателю быстро понять суть текста и его содержание без прочтения большого объема информации, что позволяет сэкономить массу времени на восприятие требуемой информации.

Целью данной работы является разработка механизма автоматического построения графических схем по описанию алгоритмов на естественном языке.

Предлагаемое решение задачи автоматического графического аннотирования текста, основано на последовательном выполнении следующих этапов.

1. Разбиение текста на предложения.
2. Синтаксический анализ текста.
3. Анализ результатов синтаксического разбора для снижения многозначности.
4. Выбор кандидатов в варианты использования и актеры.

5. Анализ семантики текста, на основе выбранных кандидатов и очистка актеров на основе полученных результатов.
6. Составление шаблонов поведения для каждого актера
7. Построение графического представления на основе полученных результатов.

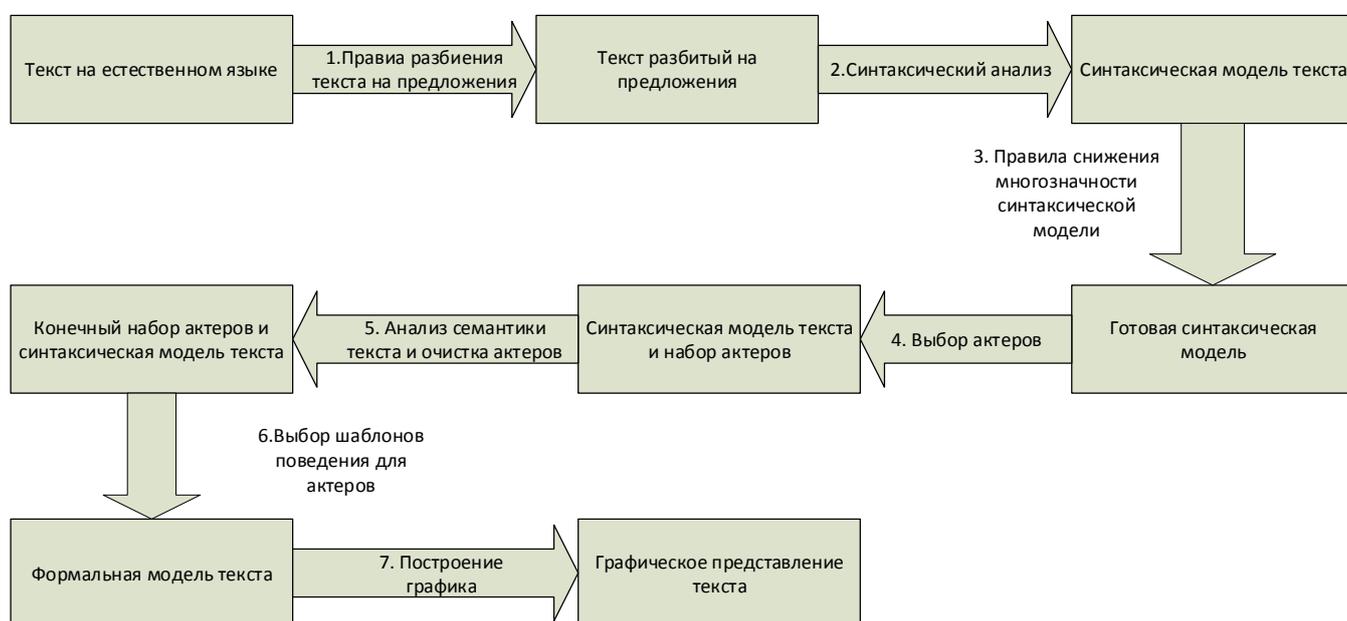


Рисунок 1 – Общая схема работы алгоритма

Каждый этап, исключая этап 2, представляет собой набор применяемых алгоритмических правил. Этапы 1 и 2 являются общими для всех видов аннотирования.

Этап 2 представляет собой преобразование текста на естественном языке в синтаксическую модель, представляющую собой связанный граф. Данный этап является для нас черным ящиком, так как для синтаксического анализа текста было выбрано готовое и достаточно качественное решение Dialingparser.

Этап 3 является этапом очистки текста от личных местоимений в третьем лице. Для этого анализируется синтаксическая модель, полученная на этапе 2.

На 4 этапе происходит анализ синтаксической модели и выбор кандидатов в актеры. Выбор кандидатов происходит следующим образом, выбираются все предложения в тексте, в которых присутствует элемент $\langle v, u, r_{sub} \rangle$, где v – глагол, u – существительное или местоимение, а r_{sub} – субъект действия. Кандидатом в варианты использования будем называть тот подграф предложения, в котором будут содержаться все вершины достижимые из v . Кандидат в актеры будет представлять собой некоторое слово, которое будет являться существительным.

Этап 5 представляет собой очистку актеров на основе полученных результатов анализа семантики текста. На данном этапе происходит анализ связей между актерами по следующему правилу: если актер представляет собой вершину не связанную с другим актером с помощью варианта использования, то он исключается из набора актеров.

На 6 этапе происходит уточнение элементов модели шаблонов, полученных на предыдущем этапе.

7 этап представляет собой визуализацию полученной модели в виде use-casedиаграммы.

Данная работа открыла нам дальнейшие перспективы развития данной темы. Кроме того, в перспективе будет рассматриваться возможность использования тезаурусов для более

тщательного анализа кандидатов в актеры, что позволит обеспечить более качественную выборку кандидатов.

Список литературы

1. Toldova S., Roytberg A., Ladygina A. Evaluating anaphora and conference resolution for Russian: RU-EVAL-2014
2. Krayvanova V. Automatic selection of verbs-makers for segmentation task of process descriptions in natural language texts: AIST(2014)
3. Use Case Diagram <http://plantuml.sourceforge.net/usecase.html> <http://www.cs.utexas.edu/ml/papers/text-kddexplore-05.pdf>

СЕМАНТИЧЕСКАЯ СЕТЬ ТЕХНОЛОГИЙ ДЛЯ САЙТА NIRSPROJECT.RU

Игошин А. – студент, Крайванова В.А. – к.ф.-м.н., доцент
Алтайский государственный технический университет (г. Барнаул)

Эффективность коммуникации - один из ключевых параметров, определяющих работоспособность любой системы в целом. Современная система высшего профессионального образования в сфере ИТ-индустрии - не исключение. Для улучшения обратной связи с работодателями, а также для повышения уровня информированности всех участников процесса подготовки кадров разрабатывается портал Nirspjroject.ru. Сфера ИТ - одна из наиболее динамично развивающихся отраслей, охватывающая широчайший спектр технологий. NIRSPjroject.ru - социальная сеть для обеспечения процесса подготовки кадров в сфере ИТ в Алтайском крае. Назначение портала – это организация взаимодействия между студентами, работодателями и преподавателями; привлечение студентов к реализации реальных проектов; канал влияния работодателей на образовательные программы; формирование портфолио студентов.

Для того, чтобы пользователи портала Nirspjroject.ru могли легко ориентироваться и быстро находить интересующие их материалы, традиционной системы поиска, по ключевым словам недостаточно. Цель работы - разработать интеллектуальную систему поиска и рекомендаций для портала Nirspjroject.ru.

Рекомендательные системы – сравнительно новый класс ПО, в чью задачу входит изучение вкусов пользователя путем анализа его действий и оценок.

Разрабатываемая система направлена на решение следующих задач. Описание навыков и потребностей пользователей. Описание области реализации проектов и требований к участникам. Описание сфер для событий и достижений. Интеллектуальный поиск. Составление рекомендаций для пользователей.

Для решения поставленных задач необходимо разработать структуру данных, которая будет хранить множество понятий (языки программирования, которыми владеет пользователь; компетенции, необходимые участникам проекта; основные темы конференции и т. д.) и связей между ними. Данное множество обладает двумя следующими свойствами. Множество понятий большое, и может динамически расширяться. Элементы множества связаны различными типами отношений (ассоциация, общее-частное и т.д.)

В современной практике для хранения семантической информации применяются три основных структуры данных: таксономии, онтологии и семантические сети. Таксономия — это технология систематизации областей знания, имеющих иерархическое строение. Для проекта не подходит, т.к. в проекте нет строгого иерархического строения. Онтология — формальное описание предметной области, применяется для того, чтобы определить общую

терминологическую базу предметной области. Данная структура не подходит для проекта т.к. предметная область часто пополняется новыми понятиями.

Семантическая сеть — информационная модель предметной области, имеющая вид ориентированного графа. Данная модель наиболее адекватна задачам проекта.

Узлы разработанной семантической сети - это конкретное понятие или технологии, а дуги определяют семантические отношения между этими технологиями.

Узлы семантической сети описываются следующими характеристиками: название (словосочетание на естественном языке); краткая аннотация; ссылка на официальный сайт.

Связи между узлами сети направленные, и описываются следующими полями: исходящий узел; целевой узел; тип связи (динамически расширяемое множество. Основные типы связей: “включает”, “использует”, “поддерживает”, “ассоциирован”.); сила связи (числовой параметр из интервала [0..1]).

На портале Nirsproject.ru используются следующие типы основных объектов: пользователь, проект, достижение, событие. Следующие свойства этих объектов описываются наборами узлов семантической сети:

- для пользователя: список навыков и список потребностей (интересов);
- для проекта: список узлов, описывающих тему, и список требований к участникам проекта;
- для события: список тем события, список требований к участникам;
- для достижения: список тем достижения;

Так как проект реализуется на базе реляционной базы данных, была разработана схема хранения семантической сети в базе. Логическая схема фрагмента базы для хранения сети представлена на рисунке 1.

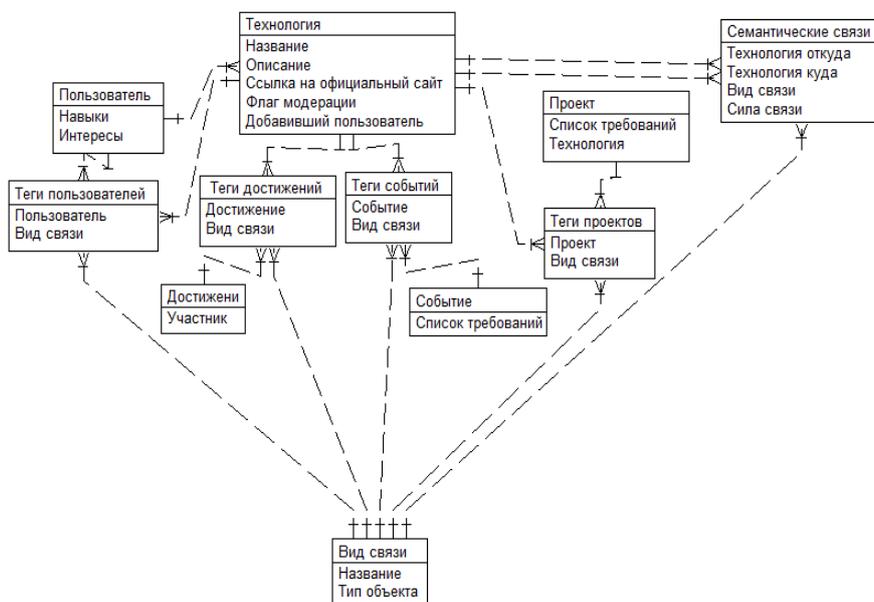


Рисунок 1 – схема БД

Для поддержания семантической сети технологий в актуальном состоянии, на портале Nirsproject.ru пополнение сети будет производиться пользователями, с последующим модерированием. Добавление новой технологии происходит следующим образом. Если пользователь не нашёл технологию в предлагаемом списке, то он заполняет заявку на добавление новой технологии, заполняя все поля. Пользователь дополнительно может указать, с какими технологиями связана новая технология. После заполнения всех полей заявка записывается в БД с флагом модерации «false». Затем модератор проверяет

правильность данных и следит за отсутствием повторов. Если что-то не верно, модератор исправляет неточности описания и устраняет повторы. После этого модератор изменяет флаг модерации на «true».

Предложенная модель обладает следующими преимуществами. Разработанная структура гибкая - предусмотрена возможность для описания большого количества ситуаций. Структура расширяемая - предусмотрено пополнение не только списка технологий, но и связей между ними, а также увеличение разнообразия типов связей. В структуре предусмотрено пополнение участниками сообщества. Возможность автоматического пополнения связей-ассоциаций.

Модель находится в стадии реализации с использованием PHP-фреймворка Symfony2 и ORM Doctrine.

Список литературы

1. Морвиль П., Розенфельд Л., Информационная архитектура в интернете, 3-е издание, 2010, 608 с. ил.
2. Башмаков А. И., Башмаков И.А. Интеллектуальные информационные технологии: Учеб. Пособие. 2005. – 304 с. ил.
3. Официальный сайт Symfony2 [Электронный ресурс] - Режим доступа: <http://symfony.com/>
4. Официальный сайт ORM Doctrine [Электронный ресурс] - Режим доступа: <http://www.doctrine-project.org/>

РАЗРАБОТКА АЛГОРИТМА АНАЛИЗА СХОЖЕСТИ СТРУКТУРЫ ВЕЩЕСТВ ПО МИКРОФОТОГРАФИЯМ

Мягков В.В. – магистрант, Крючкова Е.Н. – к.ф.-м.н., профессор
Алтайский государственный технический университет (г. Барнаул)

Распознавание сложных объектов является проблематичной задачей не только для систем искусственного интеллекта, но и для экспертов. Ручной анализ микрофотографий - это трудоемкая или даже невозможная, в случае необходимости анализа большого количества изображений, задача. Основной сложностью при автоматизированном изучении свойств текстур (под понятием текстуры мы понимаем некоторым образом организованный участок поверхности), является тот факт, что очень сложно разработать универсальный метод распознавания. Иными словами можно сказать, что под любой вид текстуры можно подобрать метод распознавания, который, при качественной настройке будет выдавать практически стопроцентный результат, тогда как на другом виде текстуры этот метод работать не будет. В связи с этим актуальную проблему представляют исследования, направленные на адаптацию методик параметризации структур для оценки их схожести. Всё это вызывает необходимость разработки алгоритма сравнения микрофотографий.

Большинство работ, посвященных сравнению структуры веществ на микрофотографиях, использует в своей основе численное представление характеристик структуры с помощью различных фрактальных или текстурных признаков [1,2]. Работы же с применением других методов и подходов компьютерного зрения практически отсутствуют или содержат в своей основе простейшие подходы на основе сравнения признаков цвета и яркости [3].

Целью данной работы является разработка алгоритма для определения пар микрофотографий, содержащих наиболее похожие по структуре вещества.

Для нахождения наиболее похожих по структуре веществ был реализован метод, основанный на использовании алгоритма SURF(Speeded Up Robust Features)[4]. С помощью алгоритма SURF выбираются те пары микрофотографий, на которых структуры изображенных веществ признаны наиболее похожими. Структура, изображенная на микрофотографии В, считается наиболее похожей на структуру, изображенную на микрофотографии А, если микрофотография В содержит наибольшее, среди остальных микрофотографий, количество достоверных совпадений дескрипторов ключевых точек[4] с дескрипторами микрофотографии А. Назовем достоверными такие сопоставления дескрипторов ключевых точек, для которых выполняется условие:

$$\text{distance} \leq 0.3,$$

где distance - расстояние(мера схожести) между дескрипторами.

Результат сопоставления изображений представлен на рисунке 1.

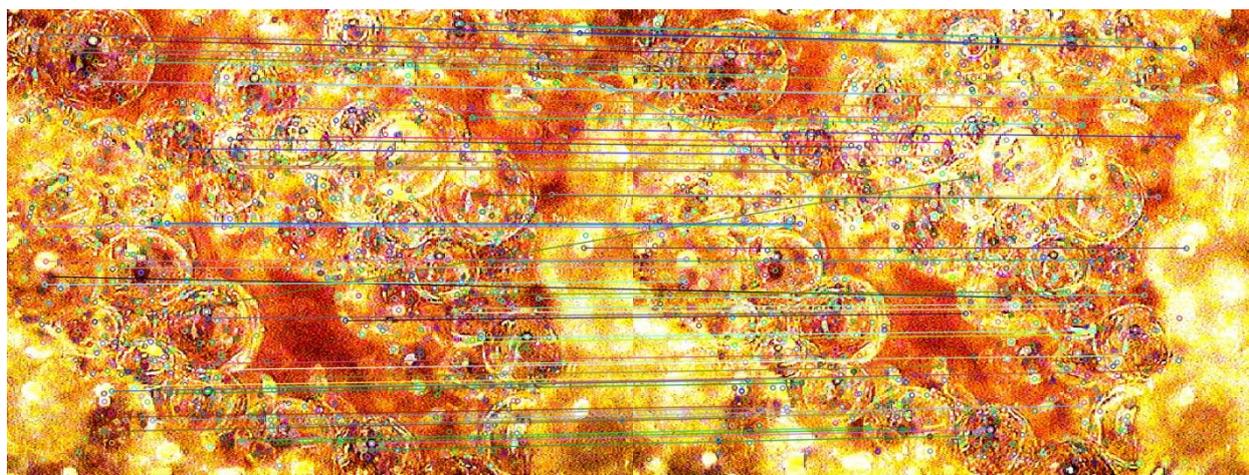


Рисунок 1 – результат работы алгоритма SURF

После работы алгоритма SURF происходит уточнение схожести структуры веществ и выбор похожих частей микрофотографий наибольшей площади. Для этого оба изображения делятся на квадраты со стороной $a = \{50, 100, \dots, n\}$, где n - меньшее измерение изображения. Для каждого из полученных квадратов рассчитывается характеристика заполнения объема:

$$K = \frac{cnt}{a},$$

где cnt - количество ключевых точек внутри квадрата. Полученные квадраты сортируются по убыванию значения K . Для квадратов с наибольшим значением K у обоих изображений рассчитывается характеристика идентичности изображений:

$$T = \frac{dcnt}{mcnt},$$

где $dcnt$ - количество достоверных сопоставлений ключевых точек, $mcnt$ - общее количество сопоставлений ключевых точек.

Далее, сторона квадрата a увеличивается и процесс повторяется.

Для проверки алгоритма программа была запущена на тестовом примере, содержащем 35 микрофотографий и были построены графики, отражающие степень схожести структур веществ. По оси X графика откладывается размер стороны квадрата a , а по оси Y – значение характеристики T . В результате работы алгоритма получается квадрат с наибольшим значением характеристики T и наибольшей стороной a .

Для изображений, структура которых визуально кажется идентичной, графики подтверждают большой уровень их схожести. Из графика видно, что сторона квадрата для разбиения может быть выбрана достаточно большой. Выбранные квадраты с наибольшим значением характеристики идентичности T представлены на рисунке 2. График анализа представлен на рисунке 3.

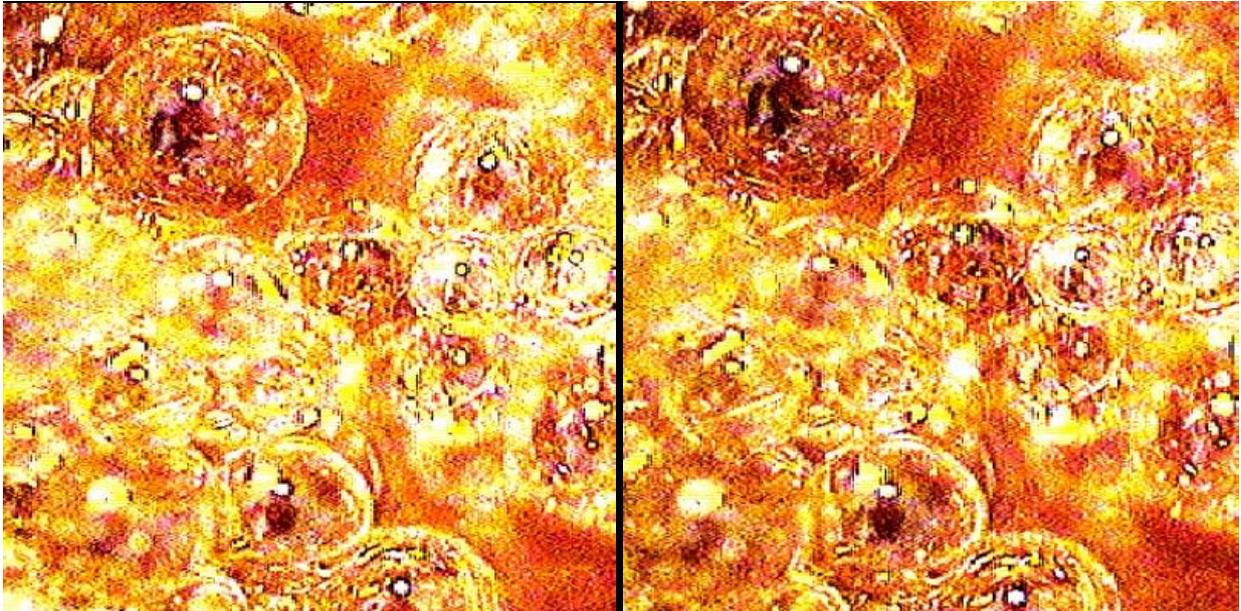


Рисунок 2 – Выбранные квадраты двух похожих по структуре изображений

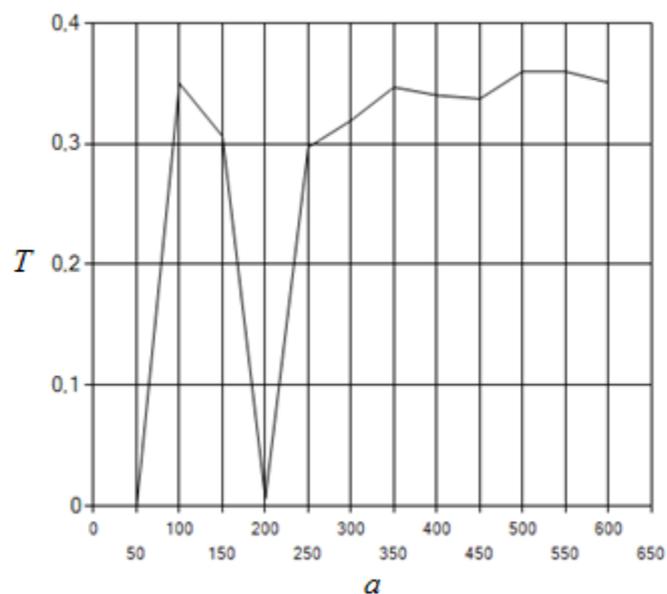


Рисунок 3 – график анализа похожих по структуре изображений

Для менее похожих по структуре веществ, выбранная сторона квадрата будет небольшой и характеристика идентичности T будет значительно уменьшаться после достижения локального максимума. График представлен на рисунке 4. Выбранные квадраты с наибольшим значением характеристики идентичности T представлены на рисунке 5.

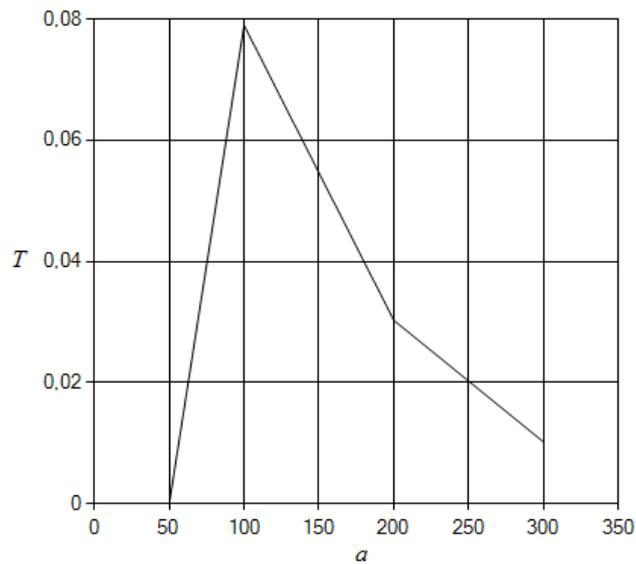


Рисунок 4 – график анализа похожих по структуре изображений

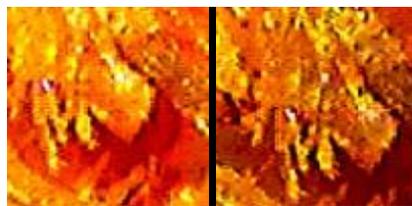


Рисунок 5 – Выбранные квадраты двух похожих по структуре изображений

Для непохожих по структуре веществ, выбранная сторона квадрата будет весьма небольшой, характеристика T будет нулевой или близкой к нулю. График представлен на рисунке 6. Выбранные квадраты с наибольшим значением характеристики идентичности T представлены на рисунке 7.

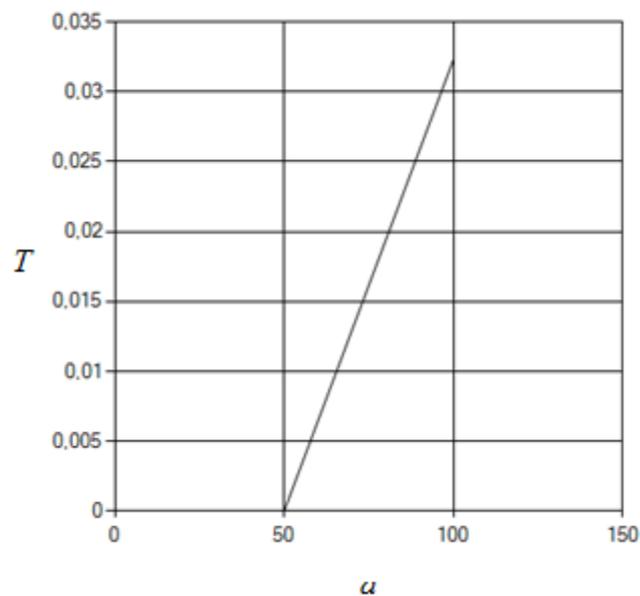


Рисунок 6 – график анализа непохожих по структуре изображений

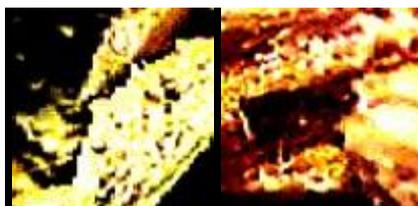


Рисунок 7 – Выбранные квадраты двух непохожих по структуре изображений

В результате исследования был разработан алгоритм для анализа схожести структуры веществ по микрофотографиям. Также был проведен анализ результатов сравнения структуры веществ и построены соответствующие графики.

Список литературы

1. Котов К.А. Метод распознавания сложных объектов на снимках микробиологических препаратов / К.А. Котов // Известия Тульского государственного университета. – 2011. – № 3. – С. 140-143.
2. Бортников А.Ю. Текстурно-фрактальный анализ микроскопических срезов образцов композиционных материалов, наполненных техническим углеродом/ А.Ю. Бортников, Н.Н. Минакова// Известия Томского политехнического университета. – 2006. – № 6. – С. 64-67.
3. Чернова И.Ю. Компьютерная обработка микрофотографий шлифов карбонатных пород с целью изучения микроструктур и коллекторских свойств продуктивных пластов./ И.Ю. Чернова, Р.И. Кадыров // Гис. – 2010. – № 54. – С. 32-40.
4. Bay H. SURF: Speeded Up Robust Features/ H. Bay, T. Tuytelaars, L. Gool// Springer LNCS – 2006. – Vol. 3951, part 1. – P. 404-417.

ОБРАБОТКА МИКРОФОТОГРАФИЙ СРЕЗОВ ОБРАЗЦОВ СВАРКИ ВЗРЫВОМ

Гавриченко Е.А. – студент, Крючкова Е.Н. – к.ф.-м.н., профессор
Алтайский государственный технический университет (г. Барнаул)

В различных областях жизни часто встречается проблемы распознавания изображений, их обработки или изменения. Многие из этих случаев слишком сложны для человеческого глаза из-за размеров, количества объектов и т.п. В таких случаях необходимо решение с помощью вычислительной техники, и решения подобных задач являются весьма нетривиальной задачей. Подобными задачами занимается раздел информационных технологий – компьютерное зрение.

В данной работе были исследованы результаты процесса сварки взрывом, а именно поиск количества контуров взрыва среза образцов и определения их суммарной площади.

Исходную цветную фотографию мы преобразуем в матрицу, состоящую из 0 и 1, которые соответствуют белому и черному цветам соответственно. Для этого была введена специальная переменная: если все три RGB-значения были меньше этого порога, то пиксель мы посчитали черным, иначе – белым. Для преобразования фотографии из графического формата в матрицу была использована библиотека Open CV для языка C++ [2][3].

В данной библиотеке уже есть своя функция поиска контуров, однако она находила множество ненужных нам, по этой причине мы отказались от ее использования.

Для определения границ взрыва был использован метод поиска в ширину (BFS - breadth-first search). Если количество клеток данной области было меньше заданной пороговой величины, то мы считаем, что данная область не является границей (возможно, это какие-то огрехи фотографии, незначащие мелочи при сварке и т.п.).

В процессе обработке может оказаться, что некоторые контуры срезов определяются разорванными неоднородности попадания луча света на пиксель, ошибки преобразования, каких-либо посторонних мелких вкраплений и т.п. Поэтому мы попробовали соединить те контуры, расстояния между которыми меньше заданного порога. В этих целях для каждой пары точек каждой пары контуров было посчитано расстояние, и если оно меньше данного порога, то мы его добавляли в специальный массив линий. Потом среди этих линий находим те, расстояния между которыми наибольшее. Их мы и посчитали продолжением границы, следовательно, добавляем их пиксели в список пикселей границы, а все точки между ними вычеркнули из него.

Таким образом, была написана программа, которая может находить границы контуров образцов сварки взрывом. На рисунке 1 представлены результаты работы программы. В дальнейшем возможно нахождения других характеристик и улучшение текущего алгоритма.



Рисунок 1 – Исходная фотография, фотография с выделенными контурами и результаты численного анализа

Список литературы

1. <http://robocraft.ru/page/opencv/> - режим доступа, открытый
2. The OpenCV user guide - October 01, 2014
3. The OpenCV tutorials - October 01, 2014

ПРОЕКТИРОВАНИЕ И РЕАЛИЗАЦИЯ ОБУЧАЮЩЕЙ ИГРЫ ЖАНРА «PROGRAMMINGGAME»

Самойлов С.С. – студент, Старолетов С.М. – к.ф.-м.н., ст. преподаватель
Алтайский государственный технический университет (г. Барнаул)

Актуальность

Игра жанра «programminggame»-это компьютерная игра, в которой человек не участвует напрямую. Вместо этого он пишет управляющую программу, которая в свою очередь проходит уровень или сражается с себе подобными программами.

Разрабатываемое приложение AITank– это своего рода игра для программиста, каждый уровень которой направлен на изучение какой-либо области в программировании. Игрок не может непосредственно управлять танком. Он может только перед началом уровня написать для него стратегию поведения (искусственный интеллект). Возьмём, например, уровень, в котором танку игрока за минимальное время надо пройти лабиринт. Без основ теории графов это будет сложной задачей, и в описании уровня будет предложен материал по нахождению кратчайшего пути в графе. Вот в такой игровой форме можно представить задачи по некоторым аспектам математики, искусственного интеллекта, теории графов, теории автоматов и другим. Решение задачи гораздо нагляднее и желаннее, чем при традиционном способе обучения.

Новизна

В 1961 году была разработана первая игра для программиста – «Бой в памяти», где программы сражались за оперативную память, пока одна из них не захватывала её полностью. Дальше были шахматные турниры между программами, а в 90-х появилась первая игра для программистов, где между собой сражались роботы.

Отдельно стоит сказать о Colobot, которая была выпущена в 2001 году и позиционировалась как стратегия, обучающая программированию. Она имела трёхмерную графику, достаточно богатый мир, компанию и обучающие задания. Там можно строить здания и создавать разные виды роботов, для каждого из которых можно написать стратегию действий на языке, похожем на Java. Надо сказать, что богатый трёхмерный мир усложняют работу программиста-игрока, заставляя его отвлекаться от основной задачи (например, нахождение минимального пути в графе) и заниматься такими вещами как вычисление места, где можно проехать и наблюдение за тем, чтобы не кончилось топливо.

Ещё один известный пример игры для программиста это Robocode. Здесь уже соревнования ведутся между игроками, а мир двумерный, и представляет из себя прямоугольник без каких-либо препятствий внутри. Для создания стратегии танка предоставляется довольно большое и сложное API. По Robocode проводились мировые чемпионаты, но он мало полезен для обучения программированию.

Нечто похожее можно увидеть на различных олимпиадах по программированию. Например, на олимпиадах студентов ACM-ICPC проходит CodeGameChallenge. Каждый год создаются новые игры: стратегия, космическое сражение, хоккей или другие, а студенты пишут искусственные интеллекты, которые соревнуются между собой. Но всё это опять же малопригодно для обучения.

Таким образом на настоящий момент нет игры для программистов, направленной на изучение конкретной темы в программировании. В Colobotслишком богатый мир, чтобы абстрагироваться от него и изучать конкретную тему, а Robocode и дополнения к олимпиадам программирования направлены на соревнования между более опытными программистами.

Описание реализации

В рамках курсового проекта по дисциплине «Архитектурное проектирование и паттерны программирования» были разработаны на языке программирования Java два модуля для AITank. Первый модуль — это ядро системы, в котором создаётся уровень и действуют танки в соответствии со своими стратегиями и характеристиками. Одну из таких стратегий и предстоит написать игроку, для этого не предполагается никакого встроенного текстового редактора, так как с этой задачей может хорошо справиться любая современная IDE для Java. Чтобы написать стратегию танка нужно реализовать один интерфейс с одним методом:

```
TankSolutionaction(TankSolution solution, Tank me, MapReadOnly map,  
ListReadOnly<Tank> tanks, ListReadOnly<Prize> prizes, ListReadOnly<Bullet> bullets);
```

На вход этой функции передаётся пустое решение, танки игрока, карта, все остальные танки, призы и летящие снаряды. Вернуть нужно заполненное решение, которое включает в себя скорость отдельно левой и правой гусеницы, скорость поворота каждого из двух башен и указание на то, нужно ли производить выстрел отдельно для каждого из двух орудий. Удалить вражеского танка из списка или, например, понизить его здоровье до нуля при реализации этого интерфейса не представляется возможным. Зато можно просмотреть все данные о любом танке, снаряде или призе, вплоть до оставшегося времени перезарядки орудий. Реализация этого интерфейса будет вызываться несколько раз в секунду для каждого танка пока игрок не выполнит или не провалит задание. Этот модуль ничего не «знает» о том, как его будут визуализировать.

Поэтому в рамках того же курсового проекта был разработан визуализатор с помощью фреймворка LibGDX, который изображён на рисунке 1. Можно двигать и масштабировать камеру, менять скорость воспроизведения и выбирать один из танков или призов. Здоровье, количество снарядов и время до перезарядки орудий показываются слева для танка игрока и справа для выбранного.



Рисунок 1 – Визуализатор

Далее будет разработан графический интерфейс пользователя с использованием фреймворка JavaFX, примерное изображение которого приведено на рисунке 2. Здесь

РАЗРАБОТКА ПОДХОДА К ОРГАНИЗАЦИИ ПЕРЕДАЧИ ИНФОРМАЦИИ С МОБИЛЬНОГО ПРИЛОЖЕНИЯ С ИСПОЛЬЗОВАНИЕМ TIN CAN API

Хатников А.С. – студент, Андреева А.Ю. – к.ф.-м.н., доцент
Алтайский государственный технический университет (г. Барнаул)

Описание предметной области

В связи с тем, что дистанционное обучение в последние годы приобретает всё большую популярность, возникает необходимость в стандартизации подходов к созданию курсов дистанционного обучения. В связи с этим Министерство Обороны США и Департамент политики в области науки и технологии Администрации Президента США в ноябре 1997 объявили о создании инициативы ADL (Advanced Distributed Learning). Целью создания данной инициативы является развитие стратегии, проводимой министерством обороны и правительством в области модернизации обучения и тренинга, а также для объединения высших учебных заведений и коммерческих предприятий для создания стандартов в сфере дистанционного обучения.

31 октября 2012 года в Лас Вегасе прошла одна из самых известных ежегодных конференций в сфере электронного обучения – DevLearn 2012. На ней ADL совместно с Rustici Software было проведено несколько докладов по новой спецификации в попытке подтолкнуть представителей отрасли использовать новый стандарт вместо устаревшего SCORM. Там же в Вегасе команда Tin Can API оборудовала отдельный, крупный участок на выставочном павильоне, чтобы донести до участников конференции ключевые моменты нового стандарта и оказать помощь заинтересовавшимся.

Tin Can API — это спецификация программ в сфере дистанционного обучения, которая позволяет обучающим системам общаться между собой путём отслеживания и записи учебных занятий всех видов. Информация об учебной деятельности сохраняется в специальную базу — Learning Record Store (LRS). LRS может являться как частью СДО, так и быть самостоятельной системой.

LRS (Learning Record Store) – группа Tin Can предлагает ввести новый объект, в котором будет храниться вся информация, собранная о пользователе из разных сред обучения (LMS, мобильный телефон, планшет, живой класс). Задумка в том, чтобы не быть постоянно привязанными к одной LMS, и чтобы можно было использовать сколько угодно разных систем управления обучением и других инструментов. При этом вся информация и логи сохраняются в отдельном сетевом объекте, которым является Learning Record Store. Собранная в LRS информация может быть запрошена одной из LMS, инструментами для генерации отчетов или другими LRS-ами.

Возможности Tin Can API

- Мобильное обучение (Mobile Learning) - оптимизация в работе с мобильными устройствами. Более детальное отслеживание успехов ученика и возможность продолжать собирать информацию о его продвижении даже при отсутствии интернет-соединения.
- Симуляторы – дает возможность следить за продвижением пользователя в полноценных десктопных программах-симуляторах и передавать собранную информацию в систему управления обучением.
- Серьезные игры (Serious Games) - Tin Can API позволяет включить в учебную программу курса обучающие игры.
- Отслеживание живой активности - в Tin Can API мы не привязаны к виртуальному обучению, у нас есть возможность отслеживать любые события, которые кажутся нам частью процесса обучения. Tin Can предлагает совместить цифровое обучение с

обучением в реальном мире с помощью самостоятельного занесения информации учителями и учениками.

- Отслеживание событий без связи с интернетом - Tin Can API позволяет отслеживать активность и продвижение пользователя даже при отсутствии постоянного интернет соединения, сохраняя информацию на устройстве до возобновления связи с сетью.
- Безопасность и аутентификация - Tin Can призван поднять планку безопасности и аутентификации. Предлагаемые Tin Can решения всё ещё далеки от идеала, но нам, по крайней мере, обещают дать возможность обезопасить пути коммуникации между представляемыми ученику обучающими материалами и репозитарием логов обучения (LRS).

11 лет назад, когда всё происходило в браузерах, SCORM соответствовал требованиям и устраивал пользователей. Но сейчас технологии двигаются в разных направлениях, и привязка к браузеру не дает реализовать некоторые возможности. Tin Can API позволяет отказаться от использования браузера как единственного инструмента для доставки отслеживаемых обучающих материалов. Потребность в этом возникла давно, так как по сути SCORM не умеет отслеживать информацию в самостоятельных, десктопных приложениях. С появлением рынка мобильных приложений и невозможностью отследить продвижение ученика в них проблема снова была поднята в профессиональных кругах. Tin Can призван решить проблему: в него добавлена возможность согласования информации между нативными мобильными приложениями, симуляторами, серьезными играми и LRS-репозиториями.

Постановка задачи

Целью данной работы является разработка программного продукта, который позволяет передавать информацию между мобильными образовательными ресурсами и сохранять ее в базу данных (LRS).

Актуальность

Ежегодная конференция DevLearn проходит в Лас-Вегасе и является самым большим и уважаемым событием в Северной Америке в сфере E-Learning.

В 2012 году в конференции участвовало около 2000 человек из 42 стран. Были представлены 85 вендоров с разнообразными продуктами и решениями для компаний.

На конференции было официально заявлено, что новый стандарт Tin Can, отныне становится основным стандартом AICC, на который равняется весь рынок eLearning. Из 85 вендоров, представленных на выставке, 30 заявили о поддержке в своем продукте TIN CAN API.

Список литературы

1. Официальный сайт «Tin Can API». [Электронный ресурс]: - Режим доступа: <http://www.tincanapi.com/>, свободный.
2. Статья на habrahabr «Tin Can API – замена устаревшему стандарту SCORM». [Электронный ресурс]: - Режим доступа: <http://habrahabr.ru/post/156067/>, свободный.
3. Проект Tin Can. [Электронный ресурс]: Режим доступа: <http://scorm.com/ru/tincanoverview/> - свободный.
4. По следам конференции DevLearn 2012 года. [Электронный ресурс]: - Режим доступа: <http://armikael.com/elearning/devlearn-conference-2012.html>, свободный.
5. DevLearn 2012 Conference and Expo. [Электронный ресурс]: - Режим доступа: <http://www.elearningguild.com/DevLearn/content/2272/devlearn-2012-conference-and-expo--home/>, свободный.

АНАЛИЗ ПАРАМЕТРОВ ДИФФЕРЕНЦИАЛЬНОЙ ДИАГНОСТИКИ

Ворона О.И. – аспирант, Леонов С.Л. – д.т.н, профессор
Алтайский государственный технический университет (г.Барнаул)

Постановка задачи

Существует несколько способов исследования заболеваний легких и один из них это компьютерная томография. Рентгеновская компьютерная томография [1] – способ визуализации внутренней структуры с получением отдельных срезов вплоть до ее объемной реконструкции. Томограммы используются рентгенологами для дифференциальной диагностики заболеваний. Но кроме традиционных подходов качественного анализа изображений возможно исследовать денситометрическую плотность пикселей изображений, формируя количественные параметры как отдельных срезов, так и их набора.

Целью исследования является анализ поведения вычисляемых параметров томографических изображений при изменении входных данных. Анализ заключается в оценке чувствительности и устойчивости алгоритмов определения этих параметров.

Предметная область

Дифференциальная диагностика шаровидных образований легких (ШОЛ) основана на вычислении следующих параметров:

1. Средняя средних по срезам плотностей S_p .
2. Среднее по срезам среднеквадратичное отклонение СКО.
3. Коэффициент корреляции между средними плотностями и их аппроксимацией параболой S_p
4. Коэффициент корреляции между СКО и его аппроксимацией параболой СКО.
5. Средняя по срезам фрактальная размерность FR.
6. Среднеквадратичное отклонение фрактальной размерности dFR.

Значения среднего, СКО и коэффициентов корреляции рассчитываются по известным формулам. Единственной особенностью является корректировка плотности изображения по двум опорным точкам: денситометрической плотности воздуха и крови в аорте.

Фрактальная размерность вычисляется следующим образом [2]:

Рассчитывается мера, которая является усредненным значением модулей разности яркости пикселей. Если последовательность мер является фракталом, то она должна зависеть от расстояния между пикселями ϵ по степенному закону :

$$M(\epsilon) = A^* \epsilon^x = A^* \epsilon^{3-d}, \quad (1)$$

где A – константа;

d – фрактальная размерность.

После вычисляется логарифм от (1) и методом наименьших квадратов находится само значение d .

После того, как рассчитаны все необходимые параметры, проводится дифференциальная диагностика.

Анализ параметров

Для расчета всех этих параметров используются значения денситометрической плотности пикселей, выделенной врачом-рентгенологом области (пример двух выделенных

областей представлен на рисунке 1). На выделяемую область накладывается ряд ограничений по размеру и расположению. Кроме того, для получения параметров изображения требуется анализ нескольких срезов ШОЛ (файлов изображений). Поэтому при получении параметров для каждого среза зона интереса обычно не изменяется ни по размерам, ни по положению на срезе. Поэтому является достаточно важными вопросами оценка стабильности (устойчивости) алгоритмов получения параметров изображения при изменении:

- размеров области интереса как по горизонтали, так и по вертикали;
- положения зоны интереса внутри ШОЛ;
- смещению зоны интереса от среза к срезу;
- изменения размеров зоны от среза к срезу.

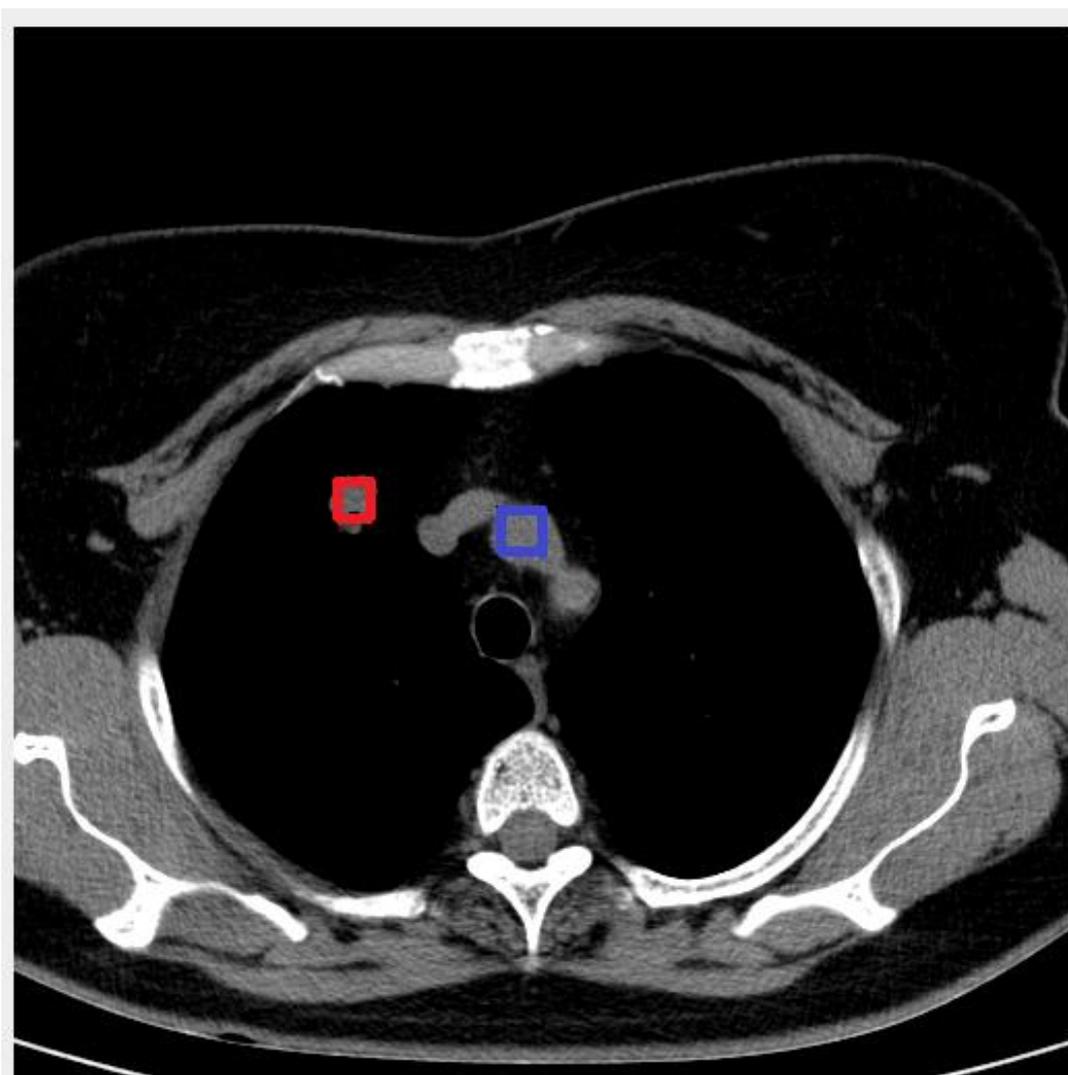


Рисунок 1 – Компьютерная томограмма с выделенной областью интереса

Ответы на эти вопросы позволят оценить устойчивость задачи вычисления параметров томографического изображения и выработать практические рекомендации для повышения точности их расчета дифференциальной диагностики в целом.

Список литературы

1. Прокоп М., Галански М. Спиральная и многослойная компьютерная томография. Том.1. – М.: МЕДпресс-информ, 2009. - 416 с.
2. Молодкин И.В., Шайдук А.М., Коновалов В.К., Леонов С.Л. Статистические свойства диагностических параметров при денситометрии шаровидных образований в легком// Известия Алтайского государственного университета. 2014 №1. С. 218-222.

ОРГАНИЗАЦИЯ УДАЛЕННОГО УПРАВЛЕНИЯ НА КОМПЬЮТЕРАХ С ОПЕРАЦИОННОЙ СИСТЕМОЙ СЕМЕЙСТВА MS WINDOWS

Фаст А.С. – студент

Алтайский государственный технический университет (г. Барнаул)

Постановка задачи. Рассмотрим проблему организации удаленного доступа для компьютеров с операционной системой семейства MS Windows версии XP и старше. Система должна обеспечивать удаленное управление только в направлении «компания – клиент», но не «клиент – клиент». Предлагаемое программное обеспечение состоит из двух частей: программа, которая устанавливается у клиентов компании, и программа, устанавливаемая непосредственно в офисе компании. Работа с программой не должна существенно занимать большое количество памяти и ресурсов процессора, при этом работники в офисе должны видеть изображение на мониторе клиента, управлять курсором мыши на стороне клиента, управлять клавиатурой, а также иметь возможность автоматического обновления программы. Данные, передающиеся по сети, должны быть зашифрованы.

Анализ рынка ПО для организации удаленного доступа. На сегодняшний момент на рынке представлено множество программ для удаленного управления, но существуют две основные проблемы при организации удаленного управления компьютером в коммерческих условиях: высокая стоимость программы при использовании её в коммерческих целях (например TeamViewer, Radmin, Remote Manipulator System, Ammyu Admin) и необходимость оплачивать лицензию на компьютерах клиентов. Другая проблема – необходимость наличия внешнего ip-адреса у компьютера клиента, что возможно далеко не всегда. Также есть возможность организации частной сети поверх интернета, например, с помощью Hamachi или ей подобных, но стоит отметить, что все подобные методы требуют отдельной настройки на каждом компьютере клиента, что невозможно ввиду большого их количества, а также ввиду возможной необходимости настраивать все заново (большинство клиентов – бухгалтеры, на их компьютерах могут отсутствовать необходимые полномочия для подобного рода настроек). Отдельно стоит отметить Remote Desktop Protocol компании Microsoft, но для использования его штатными средствами системы WindowXP необходимо скачивать дополнительные обновления, и наличие внешнего ip. Таким образом, на рынке не найдено готового решения, не требующего настройки на стороне клиента, внешнего ip-адреса, и бесплатного для коммерческого использования. Соответственно, необходимо создать собственную систему, которая будет использоваться на одном предприятии и не будет коммерчески зависеть от других компаний.

Основная схема работы. В основную структуру системы положена клиент-серверная архитектура. Клиентские приложения разделены на два вида: управляющие и управляемые,

т.е. первые запускаются на компьютере, который осуществляет управление другой машиной, на которой в свою очередь запущен клиент второго вида. Сделано это для того, чтобы третьи лица не могли пользоваться системой. Для запуска управляющего приложения необходимо ввести логин и пароль, которые должен подтвердить сервер.

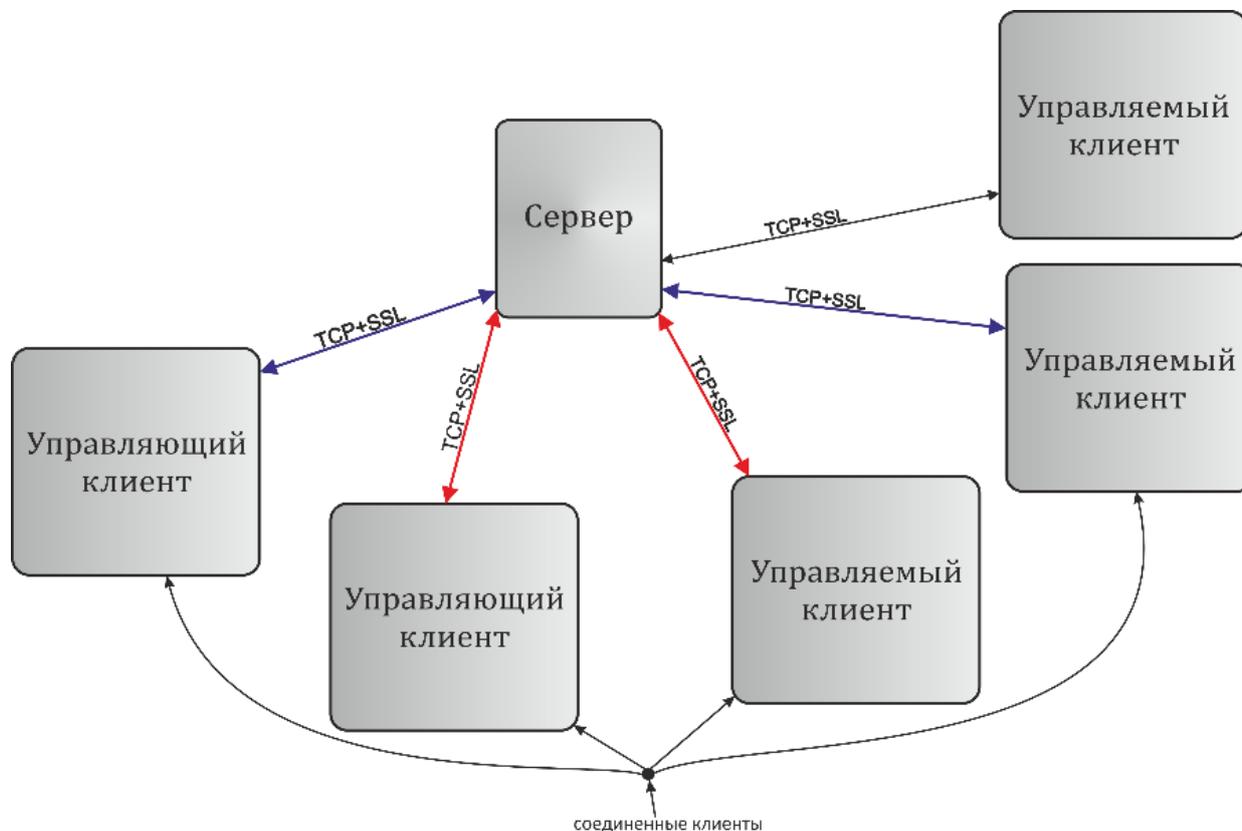


Рисунок 1 – Архитектура системы

При запуске клиента происходит соединение с сервером и аутентификация клиента. Управляемое клиентское приложение при каждом запуске генерирует уникальные ID и пароль, которые в зашифрованном виде передаются на сервер. После запуска управляющего клиента необходимо ввести ID и пароль того клиента, с которым необходимо соединиться, после подтверждения введенных данных сервер начнет сеанс связи, в течении которого сервер осуществляет трансляцию данных с одного клиента на другой; управляющее приложение передает координаты мыши, различные жесты мыши, а также нажатия клавиш на клавиатуре, управляемый компьютер принимает данные и повторяет те же действия, в свою очередь отправляя изменившееся изображение экрана.

В программе управляемого клиента реализована защита от залипаний клавиш, гарантируется, что по завершении программы, ни одна из клавиш не останется зажатой.

Также управляющий компьютер можно перевести в «режим наблюдения», в этом случае человек может только наблюдать за тем, что происходит на другом компьютере и при желании может включиться в процесс работы в любой момент.

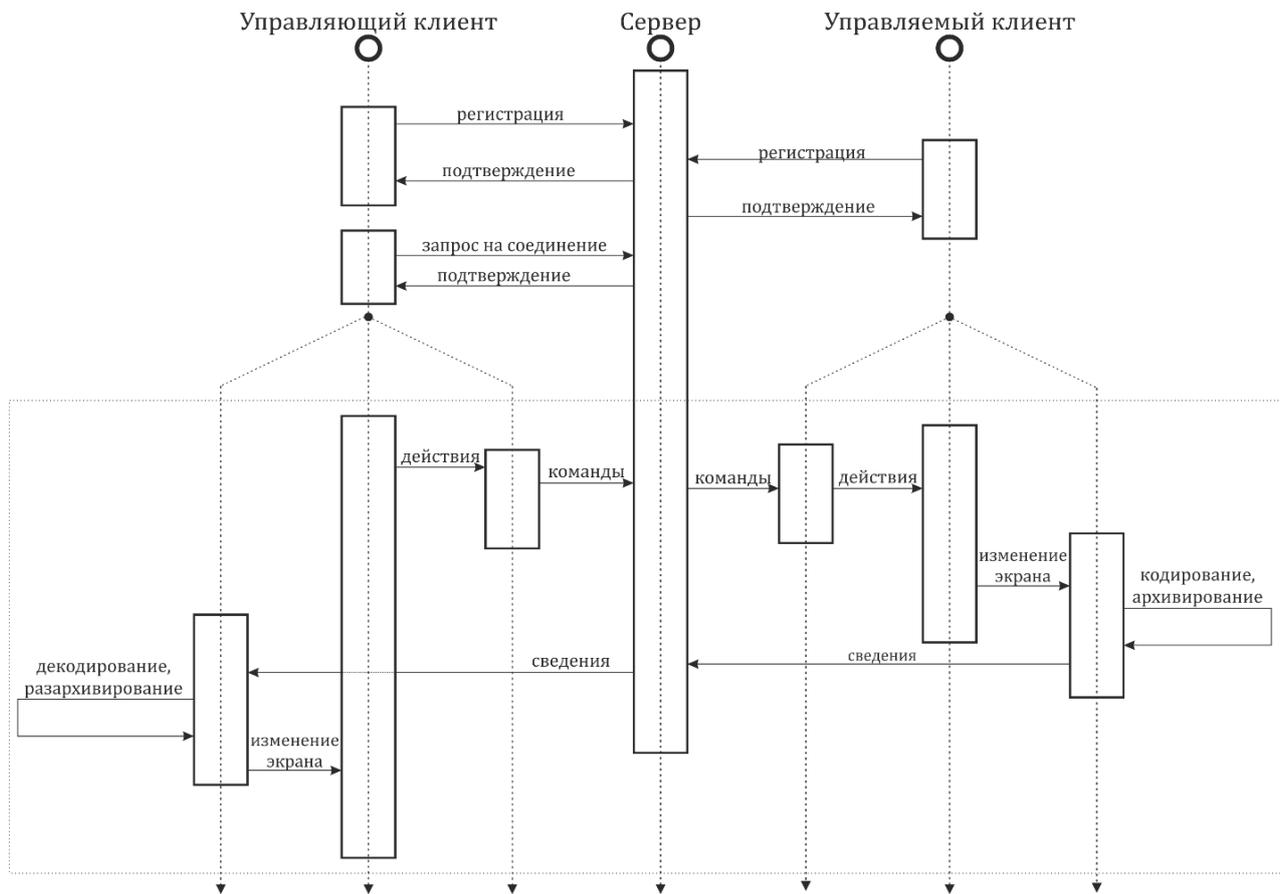


Рисунок 2 – Диаграмма линии жизни

Управляемый клиент. Управляемый клиент осуществляет непрерывный захват изображений с экрана, анализ части экрана, изменившейся с момента прошлого захвата. Затем полученное изображение сжимается, архивируется и отправляется управляющему компьютеру.

Одновременно с этим он получает от управляющего компьютера действия, которые необходимо совершить, в их числе нажатия на клавиатуру, в том числе и с комбинацией клавиш нажатия. Одиночные и двойные клики мышью, передвижения мыши с зажатой клавишей, то есть все необходимые для повседневной работы. После получения команды, клиент получает необходимые её параметры и выполняет её, используя функции `mouse_event` и `keybd_event` библиотеки `User32.dll`, включенной в системы семейства Windows.

Управляющий клиент. Управляющий клиент получает изображения с управляемого клиента и отображает их на экране, учитывая различное соотношение размеров экрана. Также он непрерывно захватывает координаты курсора мыши (пересчитывает их ввиду возможно разного соотношения сторон экранов, для корректной передачи координат), различные жесты и события, описанные выше.

Оба клиента корректно обрабатывают непредвиденный разрыв соединения, в этом случае обе программы перезапускаются с теми же данными, которые использовали в предыдущем сеансе для того, чтобы можно было быстро начать новый сеанс. В случае же корректного завершения любой из программ оба клиента разрывают соединения и готовы для использования в следующих сеансах.

Изображения сжимаются с помощью формата JPEG, а также с помощью ZIP-архивирования, в управляющем клиенте происходит разархивирование.

Сервер ведет учет всех подключенных клиентов и обязательные логи для каждого сеанса, включающие время начала сеанса, его конца, а также ip-адреса клиентов и величина трафика, прошедшего через сервер в данном сеансе. Сделано это для того, чтобы была возможность контролировать использование программы, а также для оценки затрачиваемого трафика, а в случае изменения технологий, для сравнения с предыдущими результатами.

Средства реализации задачи. В качестве основного фреймворка для создания системы был выбран .Net Framework версии 3.5. Во-первых, заказчик и его клиенты работает с ОС семейства Windows (в основном - XP), а данный фреймворк позволяет создавать сложные системы в том числе и под эту платформу. Также фреймворк обладает рядом встроенных библиотек для работы с сетевыми соединениями и изображениями.

В качестве протокола передачи данных используется TCP, ввиду его гарантированной доставки пакетов и отсутствия переупорядочивания пакетов, например как у протокола UDP. Следствием этого является эффект «разгона» программы, т.е. спустя небольшое время после начала сеанса система начинает работать быстрее, это одна из особенностей передачи через протокол TCP большого количества пакетов в одном и том же направлении.

Для надежной защиты персональных данных поверх TCP используется протокол SSL(самоподписанный сертификат X509).

Сервер работает под управлением системы MSWindowsServer 2012 R2, серверная часть также написана с использованием Net Framework 3.5.

Вывод. Построенная система успешно функционирует на предприятии ООО «КОМПТЭС+», отвечает требованиям простоты и скорости работы.

Список литературы

1. Грин Дж. Изучаем C# [Текст] /Э. Стиллмен, Дж. Грин.-Санкт-Петербург: Питер, 2012. - 696 с.
2. Скит Дж. C#: программирование для профессионалов./Дж. Скит. – Москва – Вильямс, 2011 – 511с.
3. Моримото Р. Microsoft Windows Server 2012. Полное руководство/ Р. Моримото, М. Ноэл, Г. Ярдени, О. Драуби, Э. Аббейт, К. Амарис. - Москва – Вильямс, 2013 - 1456 с.
4. Кузьменко Н. Компьютерные сети и сетевые технологии/Н. Кузьменко. - Санкт-Петербург: Наука и техника, 2013 – 368 с.

ИСКУССТВЕННЫЙ ИНТЕЛЛЕКТ В МОДЕЛИРОВАНИИ ПОВЕДЕНИЯ ИГРОВЫХ ПЕРСОНАЖЕЙ НА ОСНОВЕ ПОДХОДА «BEHAVIOR TREE»

Никитин А.А. – студент, Саролетов С.М. – к.ф.-м.н, ст. преподаватель
Алтайский государственный технический университет (г. Барнаул)

В мире программирования есть много нерешенных проблем, одна из них – это искусственный интеллект. Искусственный интеллект может применяться для задач распознавания, компьютерного зрения, создания моделей виртуальных объектов. В рамках данной работы будет рассмотрена задача моделирования поведения игровых персонажей.

Существующими подходами для решения такой задачи могут являться конечные автоматы, шаблоны в виде скриптов. Однако наиболее актуальной концепцией, на наш взгляд, является подход с использованием так называемых деревьев поведений.

Модель деревьев поведения появились впервые в игре Halo 2 в 2005 году [3], согласно [3] это был наиболее развитый искусственный интеллект. Затем деревья поведения начали активно использоваться в робототехнике, и в это время данная модель была теоретически сформулирована, были введены основные понятия и описаны базовые алгоритмы из этой концепции, которые приведены в работе [1] Для ознакомления приведем основные понятия и алгоритмы.

Дерево поведения — это способ задания стратегии взаимодействия игрового объекта с миром игры. Дерево поведения — это связный ациклический граф, в котором имеется три типа вершин: лист, декоратор и композит. Каждый раз, когда игровому объекту необходимо принять решение, он посылает сигнал корню дерева поведения на исполнение. Этот сигнал проходит по некоторым вершинам дерева и в конечном итоге возвращает некоторое значение запустившему сигнал объекту. Сигнал может передвигаться только из родительской вершины в дочернюю, при этом за то, что сигнал попадет в дочернюю вершину отвечает тип и вид родительской вершины. Когда сигнал попадает в вершину, то он заставляет исполняться некоторый код, в зависимости от типа и вида вершины. Так попадая в лист, сигнал заставляет исполняться пользовательский код, определенный только для данной вершины, после чего лист возвращает значение родительской вершины, на основании которого родительская вершина производит некоторые действия по управлению сигналом и возвращаемым значением своей родительской вершине и т. д. Возвращаемых значений всего четыре: «успешно», «не успешно», «запущено» и «ошибка».

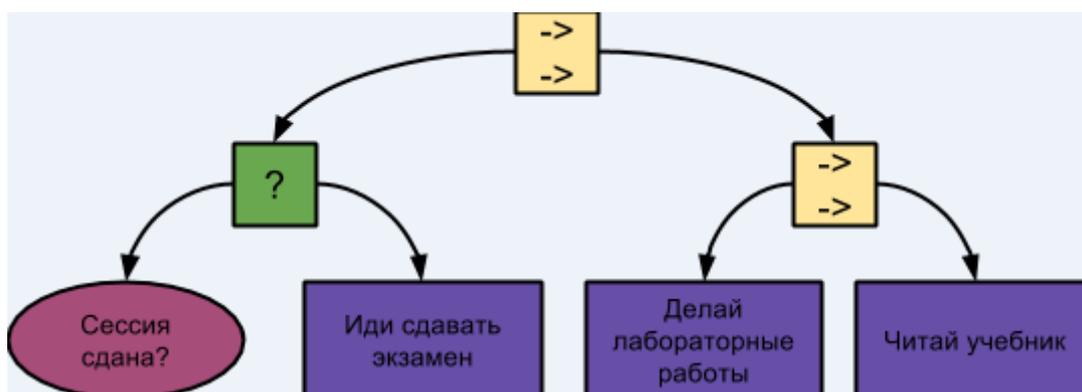


Рисунок 1 – Дерево поведения студента во время сессии

Лист не имеет дочерних вершин. Для каждого листа программист определяет некоторый программный код. Существует два вида листов: условие и действие. Когда лист-условие принимает сигнал, то исполняет код, который возвращает «успешно», «не успешно» или «ошибка», в то время как лист-действие исполняет код, который возвращает «успешно», «ошибка» или «запущено».



Рисунок 2 – Лист-условие и лист-действие

Декоратор имеет ровно одну дочернюю вершину. Декоратор перенаправляет сигнал своей дочерней вершине, когда же дочерняя вершина возвращает некоторое значение, то декоратор изменяет это значение в зависимости от своего вида и передает родительской вершине. Виды декораторов могут быть базовыми (например, декоратор-инвертор, декоратор, перенаправляющий сигнал заданное число раз, декоратор, всегда возвращающий «успех»), и могут быть определены программистом.

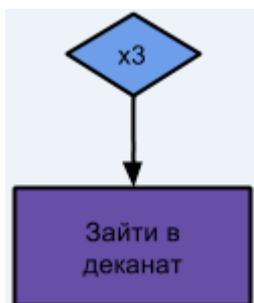


Рисунок 3 – Декоратор, повторяющий три раза сигнал дочерней вершине

Композит имеет неопределенное число дочерних вершин и управляет передачей сигнала дочерним вершинам в зависимости от вида композита. Например, композит-селектор передает сигнал по порядку дочерним вершинам до тех пор, пока одна из дочерних вершин не вернет «успешно», «запущено» или «ошибка», при этом возвращая такое же значение своей родительской вершине. Если же все дочерние вершины возвратили результат «не успешно», то селектор возвращает результат «не успешно». Другим примером является параллельный композит, который передает сигнал всем дочерним вершинам одновременно. Такой композит имеет два пороговых значения S и F. Если количество дочерних вершин, вернувших значение «успешно» больше S, то параллельный композит вернет результат «успешно», если количество дочерних вершин, вернувших значение «не успешно» больше F, то параллельный композит вернет результат «не успешно», иначе - «запущено».



Рисунок 4 – Композит-селектор

Для того, чтобы пользовательский код в листах мог влиять и определять поведение игрового объекта используется подход «классной доски». При таком подходе все листовые вершины дерева имеют доступ на чтение и запись к общему набору переменных. Данный подход реализуется с помощью ассоциативного массива.

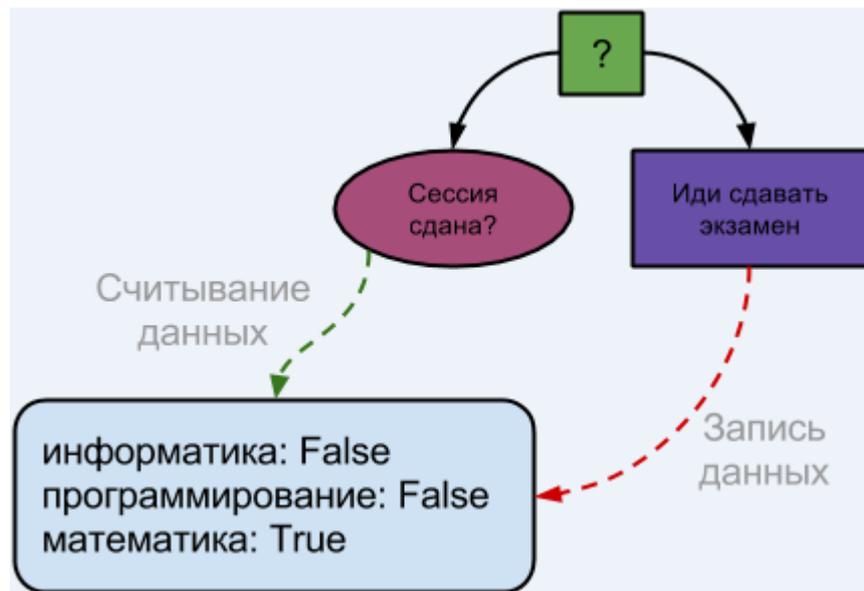


Рисунок 5 – Использование данных в подходе "классной доски"

Преимуществом дерева поведения является его иерархичность, то есть любое поддерево дерева поведения — это тоже дерево поведения, что позволяет разделить большое дерево на логические части и далее работать с каждой частью отдельно.

Данный подход применяется в крупнейших современных фреймворках для разработки игровых приложений Unreal Engine и Unity3D для создания поведения автономных игровых объектов. Нами реализована библиотека для проектирования на уровне кода деревьев поведения в рамках курсовой работы на языке Java. С помощью этой библиотеки реализован виртуальный мир с участием двух персонажей, которым назначена стратегия избегать препятствий.

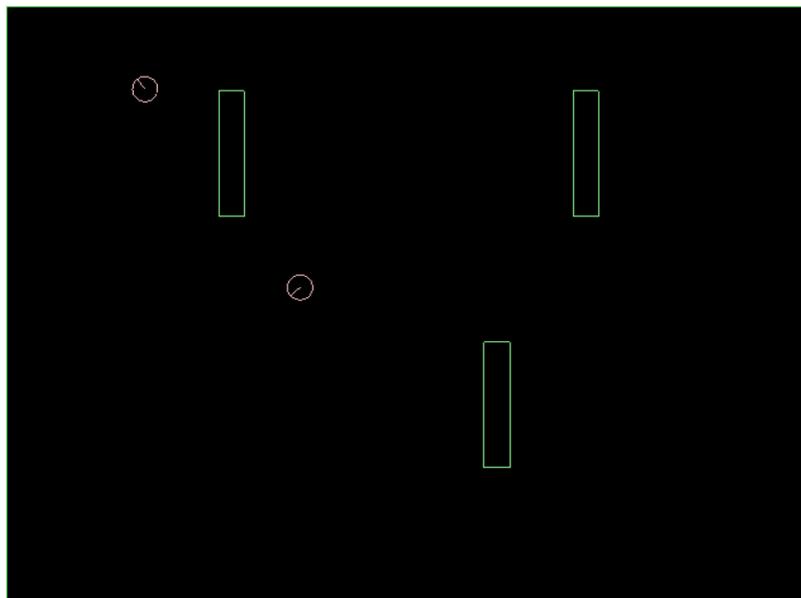


Рисунок 6 – Объекты передвигаются в случайном направлении, избегая препятствий

В развитии каждой системы наступает период, когда в ней есть только один недостаток – человеческий фактор. В нашем случае дерево поведения создается человеком и не изменяется на протяжении работы игрового приложения. Теперь появляется необходимость избавиться от человеческого влияния на структуру дерева поведения, то есть сделать эту

структуру динамически самоорганизовывающейся на основе влияния внешних факторов игрового мира. Таким образом в дальнейших исследованиях предполагается рассматривать деревья поведения в контексте самообучающейся системы.

Список литературы

1. Marzinotto A. Towards a Unified Behavior Trees Framework for Robot Control / A. Marzinotto [и др.] .- Swedish Research Council and the European Union Project, 2013 .- 8 с.
2. Ogren P. Modularity of UAV Control Systems using Computer Game Behavior Trees / P. Ogren .- Swedish Defence Research Agency, Stockholm, 2012 .- 8 с.
3. Delmer S. Behavior Trees for Hierarchical RTS AI / S. Delmer .- Plano: The Guildhall at SMU, 2013.- 10 с.

ПРИМЕНЕНИЕ АЛГОРИТМА ФИЛЬТРАЦИИ ЧАСТИЦ В ЗАДАЧЕ ОТСЛЕЖИВАНИЯ ПОЗЫ ЧЕЛОВЕКА

Есипенко С.П. – аспирант, Крючкова Е.Н. – к.ф.-м.н., профессор
Алтайский государственный технический университет (г. Барнаул)

Отслеживание позы человека необходимо в самых разных областях человеческой деятельности. Например, такая задача ставится при создании умных систем слежения, беспилотных летательных аппаратов и самоуправляемых автомобилей. Множество применений есть и в медицине: слежение за состоянием больного человека, поиск патологий опорно-двигательной системы, обнаружение травм и другие. Кроме того, качественное отслеживание позы человека нужно для систем захвата движения (motion capture) в пакетах компьютерной анимации.

Постановка задачи

Задача отслеживания позы человека может формулироваться разным способом. В данной работе рассматривался относительно простой вариант задачи: для данной видеопоследовательности с изображением человека, и заданной начальной конфигурации его конечностей в пространстве, определить движение конечностей на протяжении всего видео. На практике, начальное положение конечностей не известно, что делает задачу отслеживания позы человека еще труднее, но в данной работе мы будем рассматривать упрощенный вариант. К тому же, даже в такой постановке отслеживание движения конечностей остается непростой задачей.

Теперь сформулируем задачу строго. Дана видеопоследовательность, представляющая собой последовательность кадров v_t . Также известно начальное положение человека, который представлен множеством цилиндров. В данной работе будем рассматривать только три части тела: туловище (торс), плечо (часть руки от туловища до локтевого сустава) и предплечье. Изначально даны координаты и углы поворота трех цилиндров, соответствующих данным частям тела. Конечно, моделирование человеческого тела цилиндрами является достаточно грубым, но с другой стороны, требует небольшого числа параметров.

Необходимо определить положение цилиндров в пространстве соответствующее реальному положению человека для всех кадров видеопоследовательности. Качество решение задачи, другими словами, соответствует ли положение цилиндров, реальной позе человека будет определяться визуально. В данной работе, определялся первый кадр, на котором, по мнению человека, цилиндры расположены неправильно. Необходимо отметить, что такой подход очень субъективен, но вполне уместен для начальной оценки предлагаемого решения.

Предлагаемое решение

Обзор различных подходов к решению данной задачи представлен в работе [1]. В данной работе будет рассмотрен вероятностный подход, реализованный с помощью алгоритма фильтрации частиц (Particle Filter). Выбор вероятностного подхода обусловлен тем, что движение человека имеют сложную природу и поэтому их сложно описать явно. Выбор алгоритма фильтрации частиц связан с тем, что он может работать с нелинейными процессами, к которым относится движение человека.

Отметим, что на данном этапе применялся базовый вариант фильтра частиц, впервые описанный в работе [2], известный также как алгоритм CONDENSATION (Conditional Density Propagation).

Подробное описание алгоритма фильтрации частиц выходит за рамки данной работы. Рассмотрим лишь краткую схему его работы, но прежде введем основные понятия.

Состояние системы S_t включает в себя положение торса, пространственные углы всех конечностей, а также первые производные всех этих параметров (скорости). Отметим, что размеры всех цилиндров приняты за константы, а конечности организованы в виде дерева так, что их положение зависит от положения конца родителя, то есть для них решается задача прямой кинематики (direct kinematics) [3]. Таким образом, зная вектор состояния мы можем полностью восстановить положение конечностей-цилиндров в пространстве. Кроме того, только параметры входящие в вектор состояния системы зависят от времени, и, фактически, являются переменными, значения которых необходимо оценивать для каждого кадра.

Наблюдение N_t содержит информацию о внешнем состоянии системы, которую мы можем получить в момент времени t . В нашем случае, наблюдением является проекция изображения с кадра v_t на конечности, соответствующие состоянию S_t .

Функция перехода $f_t(S_t, ns_t)$ задает динамику системы, и позволяет перейти от текущего состояния S_t к следующему состоянию S_{t+1} с некоторой погрешностью (шумом) ns_t .

Другими словами, $S_{t+1} = f_t(S_t, ns_t)$. Эта функция неявно задает $p(S_{t+1} | S_t)$ — функцию плотности распределения вероятности состояния S_{t+1} при данном состоянии S_t .

Функция правдоподобия l_t показывает насколько наблюдения N_t соответствует состоянию S_t . Отметим, что наблюдение может содержать погрешности, а сама функция правдоподобия может неточно описывать указанную зависимость. Поэтому, функция правдоподобия неявно задает $p(N_t | S_t)$ — функцию плотности распределения вероятности наблюдения N_t при данном состоянии S_t .

Теперь рассмотрим общую схему работы фильтра частиц.

Искомую функцию плотности $p(S_t | N_{1:t})$ неявно зададим множеством взвешенных состояний

$$WS_t = \left\{ (S_t^{(i)}, W_t^{(i)}), i=1 \dots N, \sum_{i=1}^N W_t^{(i)} = 1 \right\}.$$

Необходимо на текущей итерации t , по заданному множеству взвешенных состояний WS_t найти множество взвешенных состояний WS_{t+1} . Для этого из текущего множества WS_t выбирается N векторов состояний, с вероятностью выбора каждого вектора, пропорциональной его весу, причем выбранные состояния могут повторяться.

Затем для каждого выбранного состояния определяется состояние для следующего момента времени с помощью функции перехода.

Для полученных состояний высчитываются веса, с помощью функции правдоподобия. После этого все веса нормируются.

Сформированное таким образом множество векторов состояний, с соответствующими весами принимается за искомое множество WS_{t+1} .

Выше представлена очень краткая схема фильтра частиц. Ознакомиться подробнее с данной техникой можно в работе [2]. А с применением для задачи отслеживания положения человека в работе [3].

В рамках данной работы в функции перехода использовался явный переход, когда к переменным прибавляются их первые производные (скорости), а потом ко всем переменным, в том числе и к скоростям, добавлялась случайная величина с нормальным распределением. Для каждой переменной задавались свои параметры распределения, подобранные экспериментально.

Стоит отметить, что для упрощения вычислений, вместо цилиндров для получения наблюдения использовалось сечение цилиндров вдоль главной оси. Его проекция на кадр представляется в виде четырехугольника, что значительно упрощает обратную проекцию с изображения кадра на конечность. С помощью обратной проекции на сечение цилиндра мы получали изображение конечностей для заданного состояния на заданном кадре.

Функция правдоподобия подбиралась экспериментально. Лучше всего, на данный момент, работает оценка корреляции между изображением конечности на текущем кадре с изображением конечности на начальном кадре. Окончательная оценка правдоподобия складывается из оценок правдоподобия для отдельных конечностей с определенными весами. Отметим, что описанная функция правдоподобия является достаточно слабой и в дальнейшем будет дорабатываться.

Если говорить о полученных результатах, то в целом, можно отметить, что фильтр частиц работает для данной задачи, хотя и требует значительной доработки. Для тестовых видеопоследовательностей удалось добиться правильных результатов, в среднем для 3-5 секунд видео. На рисунке 1 представлены примеры работы алгоритма. На нем изображены десять состояний с наибольшим весом для первых четырех секунд видео. Уже на четвертой секунде видно, что одно из состояний с большим весом явно не соответствует реальному положению конечностей.

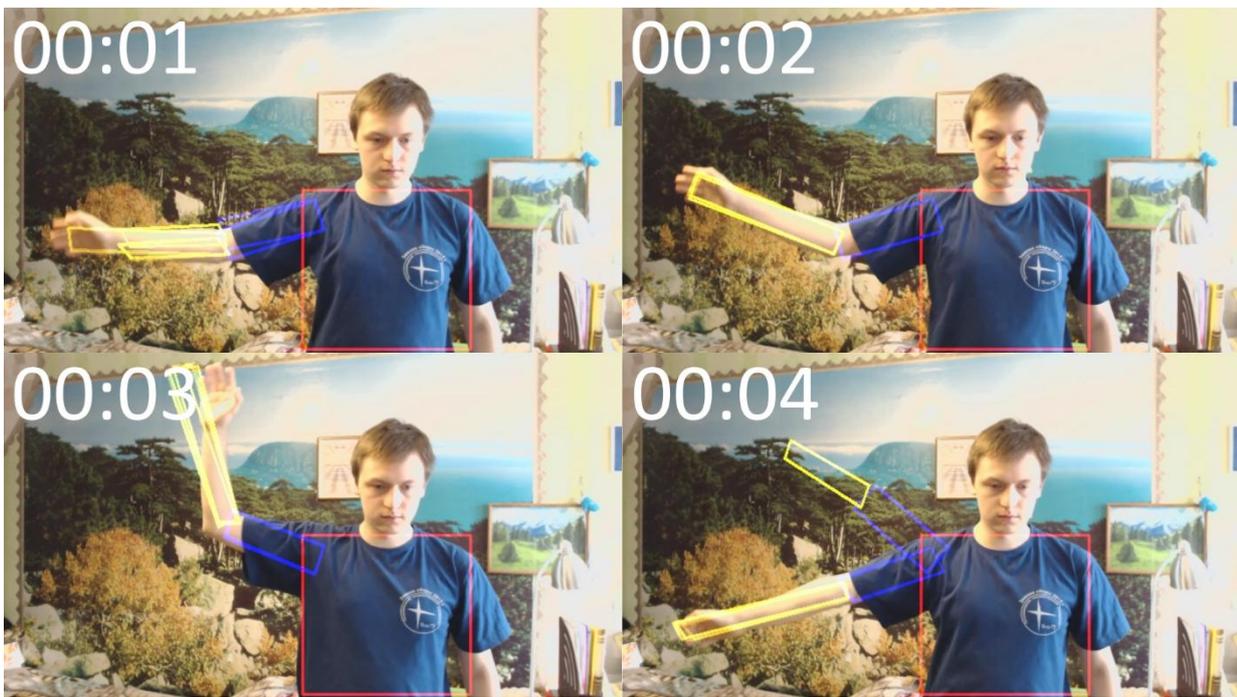


Рисунок 1 – Пример работы алгоритма

Конечно, полученные результаты достаточно скромные. Однако, стоит учесть, что они получены для базовой реализации фильтра частиц с простейшей функцией правдоподобия. Переход к более точным реализациям алгоритма фильтрации частиц, таким как UPF [4] или MPD-PF [5], а также к более качественной функции правдоподобия, может значительно улучшить полученные результаты.

Список литературы

1. Есипенко С.П., Крючкова Е.Н. Проблема скелетной реконструкции произвольных биологических и механических объектов в системах компьютерного зрения [Электронный ресурс] / С.П. Есипенко, Е. Н. Крючкова // Научно-образовательный журнал АлтГТУ «Горизонты образования», 2014, вып.16, Приложение: 11-я Всероссийская научно-техническая конференция «Наука и молодежь-2014». Секция "Информационные технологии". Подсекция "Программное обеспечение вычислительной техники и автоматизированных систем". – 2014. – С. 11-13. – Режим доступа: <http://edu.secna.ru/media/f/povt2014.pdf>
2. Isard M., Blake A. CONDENSATION — Conditional Density Propagation for Visual Tracking [Текст] / M. Isard, A. Blake // International journal of computer vision 29 (1). – 1998. – С. 5-28.
3. Sidenbladh H. Stochastic tracking of 3D human figures using 2D image motion [Текст] / H. Sidenbladh, M. Black, D. Fleet // ECCV 2000. – 2000. – С. 702-718.
4. Rui Y., Chen Y. Better proposal distributions: object tracking using unscented particle filter [Текст] / Y. Rui, Y. Chen // Computer Vision and Pattern Recognition, 2001. CVPR 2001. Proceedings of the 2001 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, vol.2. – 2001. – С. 786-793.
5. Wang F. Particle Filtering with Multi Proposal Distributions [Текст] / F. Wang, Q. Zhao, Y. Zhang, L. Zhang // International Journal of Communications, Network and System Sciences, 1. – 2008. – С. 22-28.

АНАЛИЗ ГЕОМЕТРИЧЕСКОЙ СТРУКТУРЫ ИЗОБРАЖЕНИЯ В ЗАДАЧЕ РАСПОЗНОВАНИЯ

Бобков К.А. – аспирант, Крючкова Е.Н. – к.ф.-м.н., профессор
Алтайский государственный технический университет (г. Барнаул)

Введение

Компьютерное зрение на сегодня является одним из самых быстро развивающихся направлений исследований. Распознавание образов – это центральная часть в области машинного зрения, а распознавание фигур является его частным случаем, т.к. множество задач классификации объектов можно свести к классификации их фигур. В итоге классификация объектов по типу или определенным признакам предоставляет нам возможность проводить дальнейший анализ, поступающей в программу визуальной информации, что в свою очередь позволяет программе принимать решения и действовать в зависимости от окружающей обстановки.

Одним из ярких примеров использования компьютерного зрения, а также систем распознавания и классификации визуальной информации, является роботизация и автоматизации управления автомобилем без участия человека.

В данной работе мы рассмотрим методы классификации объектов на основе анализа их контуров [1,2]. Для классификации фигур по средством описания контура используют такие методы, как: комплексные координаты, функция расстояния, касательный угол, кривизна контура, а также дескрипторы Фурье [3]. Все представленные выше методы, кроме Фурье дескрипторов, входят в класс «одномерные функции представления фигур». Рассмотрим каждый метод более подробно.

Функция расстояния

Функция расстояния R_n для контура $P_n = (x_n, y_n)$, $n = [1, N]$ вычисляется как расстояния каждой точки (x_n, y_n) от неподвижной точки $C(x_0, y_0)$.

В качестве точки C обычно выбирают центроид представленной фигуры. Основным из недостатков данного метода является его неустойчивость к вращению, перемещению или масштабированию.

Комплексные координаты

Представим изображения в виде функции $f(x, y)$, и $P_n = (x_n, y_n)$, $n = [1, N]$ является множеством точек на контуре фигуры. В таком случае $z_n = x_n + iy_n$ называется комплексной координатой. Данные комплексные числа можно представить как дескрипторы фигуры, либо использовать их как входные параметры для преобразования Фурье. Недостатком данного метода также является его не инвариантность к аффинным преобразованиям, но данную проблему можно избежать, учитывая центр тяжести, где центр тяжести $g = (x_g, y_g)$.

Касательный угол

Используя данный подход, будем считать каждый контур кривой линией, таким образом, можно рассчитать угол касательной к каждой его точке. Данный метод обладает двумя существенными недостатками: чувствительность к шумам на изображении, а также

прерывность. Для избежание прерывности используется кумулятивная угловая функция, которую можно записать как Φ , где Φ – касательный угол к случайной выбранной точке на контуре. Также для нивелирования проблемы чувствительности к шумам на изображении можно использовать различные методы шумоподавления.

Фурье дескрипторы

Фурье дескрипторы получаются путем применения Фурье преобразования к представленным выше одномерным функциям представления фигуры. Фурье дескрипторами называются нормированные коэффициенты Фурье разложения.

$$a_k = \frac{2}{T} \int_0^T c(t) \cos(kwt) dt,$$

$$b_k = \frac{2}{T} \int_0^T c(t) \sin(kwt) dt,$$

$$c_k = \sqrt{a_k^2 + b_k^2}$$

Предположим, что контур объект обозначается непрерывной и периодичной функцией $c(t)$, и что здесь a_k – реальная часть; b_k – мнимая часть; c_k – Фурье дескриптор.

Выбор подходящего метода

Фурье дескрипторы устойчивы к перемещению, масштабированию и вращению объекта и, следовательно, идеально подходят для представления фигуры. Исходя из вышеперечисленных плюсов Фурье дескрипторов было принято решение использовать их для описания контуров. Для применения данного метода первым шагом мы производим преобразование изображения для выделения его контуров. Для этих целей нами был выбран Canny фильтр, т.к. данный фильтр хорошо зарекомендовал себя в данной области.

Иерархическое представление геометрии изображения

Определение характеристики только внешнего контура не обеспечивает полноценное описание объекта для дальнейшей его классификации. Следовательно, нами было предложено решение данной проблемы: построение иерархической древовидной структуры, в которой содержится информация о геометрическом строении представленного объекта, т.е. информация о иерархии контуров. Каждый узел дерева представляет собой структуру, включающую в себя – Фурье дескриптор для данного контура, значение высоты и ширины, массив подузлов.

Таким образом, построив иерархическую структуру, мы можем с большей точностью описать наш объект и в дальнейшем классифицировать его не только по сопоставлению внешнего контура, но и по сравнению внутренней структуры. Построение иерархии контуров позволяет нам проводить различную пост обработку полученного дерева, одним из возможных вариантов пост обработки является свертывание в группы различных похожих объектов (по Фурье дескриптору, а также по площади контура) на одном уровне иерархии. Данный фильтр позволяет сократить алгоритмическую сложность при сравнении и поиску схожих иерархических структур при сравнении объектов.

Предложенный подход, в совокупности с пост обработкой дерева, позволяет нам сравнивать различные объекты посредством сопоставления характеристических деревьев.

Ниже приведен пример построения дерева объекта, представленного на рисунке 1.

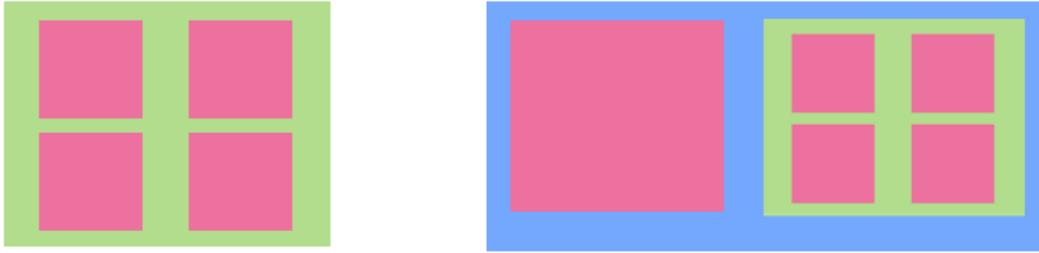


Рисунок 1 – Простое изображение, с четко выраженной иерархической структурой

Для представленного выше изображения наша система построила следующее дерево (Рисунок 2).

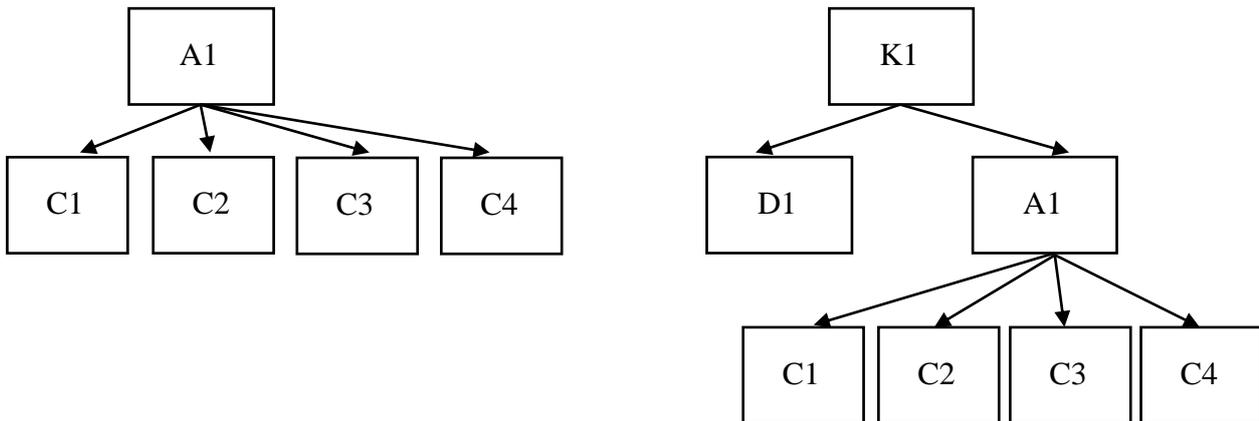


Рисунок 2 – Дерево геометрии для предложенного изображения

Стоит отметить, что построенные деревья схожи, т.к. присутствуют схожие структурные части, следовательно мы можем сопоставить деревья и вычислить процент схожести объектов.

Вывод

Проанализировав уже изученные различные методы описания и классификации структур по их контурам, мы столкнулись с проблемой определения сложных композитных объектов, состоящих из нескольких частей.

Так как стандартные подходы, в основном, учитывали только внешний контур объекта, что не позволяло с достаточной точностью провести его классификацию, нами был предложен подход, основывающийся не только на описании контура посредством вычисления Фурье дескриптора, но и учитывающий геометрическую и иерархическую структуру объекта. Наш выбор Фурье дескрипторов, как универсальных характеристик контура, обусловлен тем, что данный подход является устойчивым к аффинным преобразованиям.

Таким образом, предложенный нами подход позволяет с достаточной точностью классифицировать композитные объекты.

Список литературы

1. Zhang D., Lu G. Review of shape representation and description techniques // Pattern Recognition. – Oxford: Elsevier, 2004. – С.1-19.

2. Yang Mingqiang, Kpalma Kidiyo and Ronsin Joseph, A Survey of Shape Feature Extraction Techniques, Pattern Recognition Techniques, Technology and Applications, Peng-Yeng Yin (Ed.) (2008) С.43-90.
3. Richard Szeliski Computer Vision: Algorithms and Applications, 2013. – С.133-140.

ОПЫТ ПРОГНОЗИРОВАНИЯ УСПЕВАЕМОСТИ СТУДЕНТОВ НА ОСНОВЕ СТАТИСТИЧЕСКИХ ДАННЫХ

Ненайденко А.С. – аспирант, Боровцов Е.Г. – к.т.н., профессор
Алтайский государственный технический университет (г. Барнаул)

Одной из самых приоритетных задач в сфере высшего образования является повышение качества подготовки, конкурентоспособности, а также востребованности и эффективности выпускаемых вузом специалистов.

Одним из методов определения количественных характеристик уровня подготовки студентов в образовании является квалиметрия.

Важным элементом системы менеджмента качества образования в Алтайском государственном техническом университете является модульно-рейтинговая система квалиметрии учебной деятельности студентов (МРСК), которая внедрена в вузе с 2005-2006 учебного года. МРСК позволяет получить следующие виды рейтинговых оценок: входной рейтинг студента (среднее арифметическое вступительных экзаменов абитуриента); рейтинг по конкретной дисциплине, изучаемой в текущем семестре; комплексный (семестровый) рейтинг, включающий рейтинги по всем дисциплинам текущего семестра; обобщенный рейтинг, учитывающий входной рейтинг и успехи за все время с начала обучения в вузе и до момента аттестации; обобщенный рейтинг по циклам дисциплин (ЕН, ГСЭ, ОПД, СД); выходной рейтинг, определяемый с учетом результатов государственного экзамена и защиты дипломного проекта.

За 10 лет функционирования модульно-рейтинговой системы, накопилось огромное количество статистической информации об успеваемости студентов, которая представляет научный интерес, и может быть исследована с целью извлечения знаний. Одним из примеров таких знаний является получение прогнозов успеваемости студентов, которые могут быть использованы для выявления как перспективных студентов, так и тех студентов, которым требуется дополнительная помощь в освоении образовательной программы.

Очевидно, что при обучении на успеваемость студента может влиять огромное множество различных факторов. Эти факторы могут быть как количественного так и категориального вида.

Авторами данной работы, в качестве апробации использования аппарата нейронных сетей, для построения модели прогнозирования успеваемости студентов, были выбраны следующие факторы: входной рейтинг студента, результаты промежуточных аттестаций, а также результаты сессий за первый год обучения и своевременность их сдачи. Прогнозируемой величиной, в данном случае, является комплексный рейтинг студента после второго года обучения. Таким образом анализируется влияние входных знаний студента, приобретенных в школе, и знаний, полученных за первый год обучения, на успеваемость после второго курса.

Для составления прогноза были взяты данные об успеваемости студентов факультета информационных технологий всех специальностей и направлений очной формы обучения 2006 – 2012 годов поступления. Из обучающей выборки предварительно были исключены те случаи, когда студент был отчислен по каким-либо причинам, не связанным с неудовлетворительным выполнением им плана учебного процесса (перевод в другой вуз, отчисление по собственному желанию и т.д.). В общей сложности анализируемая выборка составила 1322 человека.

Полученная модель прогнозирования основывается на трехслойной полносвязной искусственной нейронной сети обратного распространения ошибки. В качестве алгоритма обучения выбран алгоритм Левенберга-Маркварда. Первый (входной) слой сети содержит 9 входов (x_1 – входной рейтинг студента, x_2 – балл первой аттестации 1 семестра, x_3 – балл второй аттестации 1 семестра, x_4 – результаты первой сессии, x_5 – признак сдачи сессии в срок, x_6 - x_9 – аналогичные параметры как и x_2 - x_5 , только для второго семестра обучения), промежуточный слой – 6 нейронов, на выходе – одно значение (Рисунок 1). Если прогнозное значение равно 0, либо близко к нему, то такой студент, вероятнее всего, будет отчислен. Выбор числа нейронов скрытого слоя равным 6 был обусловлен тем, что с увеличением числа нейронов в этом слое средняя квадратичная ошибка (MSE) уменьшалась незначительными темпами, но при этом скорость обучения и сложность сети росли.

Для повышения точности прогноза, перед обучением сети, все числовые данные были нормированы таким образом, чтобы интервалы их значений лежали в пределах от 0 до 1 (т.е. максимальной оценке в 100 баллов соответствует 1).

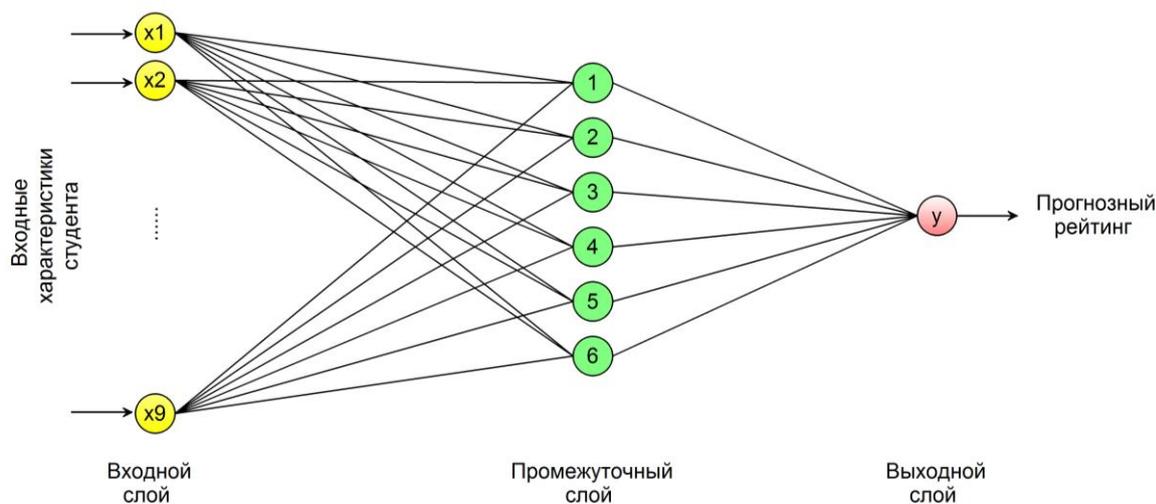


Рисунок 1 – Упрощенная схема модели нейронной сети

Построение, настройка, обучение нейронной сети, а также все вычисления производились в пакете MATLAB, с помощью встроенного тулбокса - Neural Network Toolbox.

Для обучения нейронной сети на входы были поданы данные об успеваемости за 2006-2011 годы поступления. Обучающая выборка составила 1133 записи. График обучения сети представлен на Рисунке 2, а результаты обучения в Таблице 1.

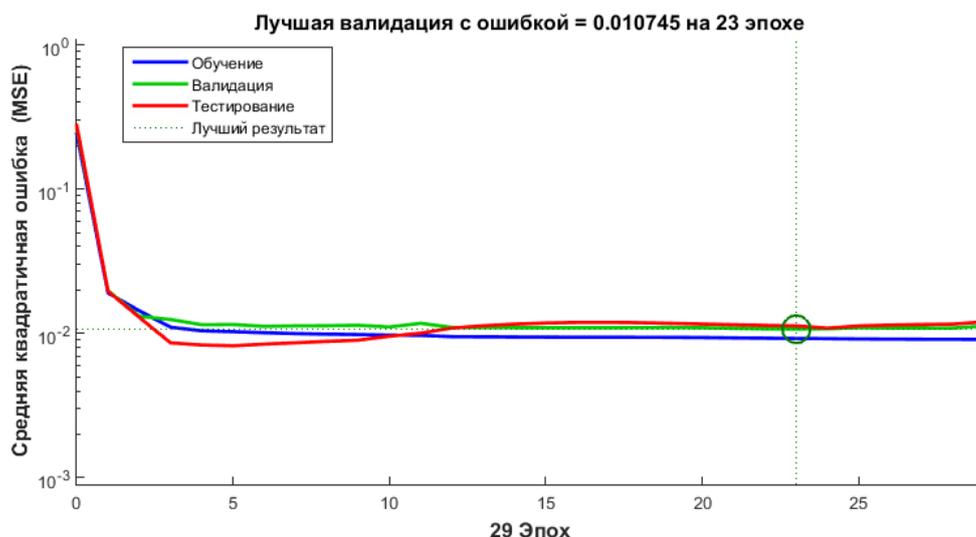


Рисунок 2 – График обучения сети

Таблица 1 – Результаты обучения сети

	Кол-во записей выборки	Ошибка MSE	Коэф. корреляции
Обучение	850	0,00924595	0,903110
Валидация	113	0,0107454	0,910030
Тестирование	170	0,0112912	0,901332

В качестве тестирующей выборки были использованы данные об успеваемости студентов 2012 года поступления (189 записей в выборке). Результаты тестового прогнозирования представлены в Таблице 2, а гистограмма распределения ошибки - на Рисунке 3.

Таблица 2 – Результаты тестового прогнозирования

	Значение
MSE	0,00458524
Коэф. корреляции	0,931297

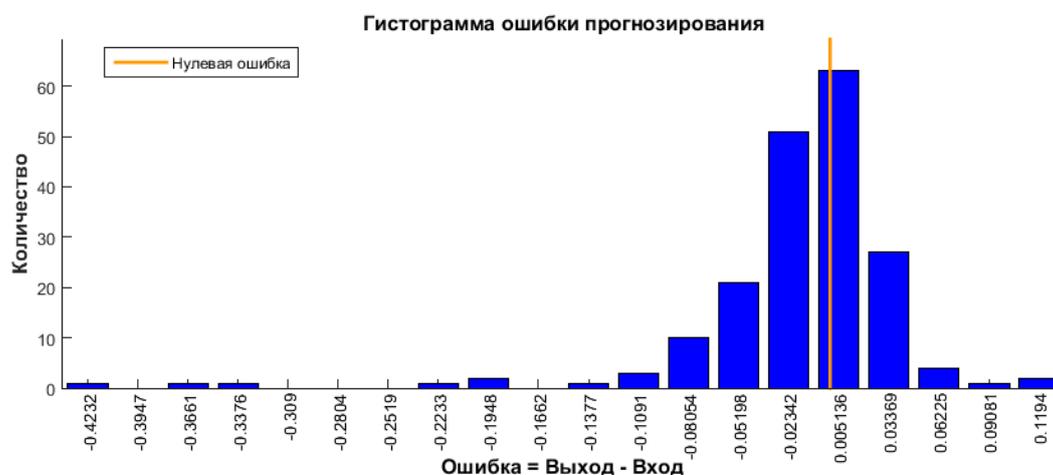


Рисунок 3 – Гистограмма ошибки прогнозирования

Учитывая то, что данные перед обучением и тестовым прогнозированием были нормированы, необходимо перейти обратно к 100-бальной шкале. Таким образом, средняя ошибка в баллах составит около 7 баллов.

Анализируя результаты проделанной работы, можно сделать следующие выводы:

1. Построенная нейросетевая модель прогнозирования успеваемости студентов хорошо справляется с поставленной задачей и выдает прогнозы в среднем с ошибкой менее 7 баллов.
2. Между входным рейтингом студента и его успеваемостью на первом курсе существует высокая корреляция с показателем комплексного рейтинга студента после второго года обучения (коэффициент корреляции составляет свыше 0,93).

Список литературы

1. Хайкин С. Нейронные сети: полный курс. М.: ООО «И. Д. Вильямс», 2006. 1104 с.
2. NeuroProject / Обучение/ Учебник – Методы прогнозирования [электронный ресурс]. URL: http://www.neuroproject.ru/forecasting_tutorial.php (дата обращения 21.04.2015).

ВИЗУАЛЬНАЯ СРЕДА РАЗРАБОТКИ ИНТЕРАКТИВНЫХ НОВЕЛЛ

Лаптев М.А. – студент, Старолетов С.М. – к.ф.-м.н., ст. преподаватель
Алтайский государственный технический университет (г. Барнаул)

Интерактивная новела (книга-игра) — литературное произведение, которое позволяет читателю участвовать в формировании сюжета. Чаще всего читателю предлагается стать главным героем книги и, в зависимости от принимаемых решений, перемещаться между страницами или главами. Таким образом, книга-игра читается не последовательно, а в той очередности, в которой читатель проходит страницы или главы.

За рубежом этот жанр опробован уже давно и потом широко распространён. В России интерактивная литература пока мало известна, но имеет огромный потенциал, так как предоставляет читателю гораздо больше свободы по сравнению с обычной книгой. В книге-игре имеются ветвления, поэтому классическая бумажная книга-игра разбивается на пронумерованные параграфы или страницы, а читатель в зависимости от принимаемых решений перемещается между ними. Соответственно книгу-игру невозможно читать последовательно и читателю необходимо с каждым переходом пролистывать книгу и отыскивать нужный параграф. Но и для этой проблемы есть решение. Очень удобно перенести книгу–игру в электронный формат, тогда переход из одного конца книги в другой, легко осуществить с помощью одного клика. Кроме того, переход к электронному формату предоставляет ряд других преимуществ. Книга теперь не занимает место на полке, не может порваться или испортиться. Для производства электронной книги не нужно вырубать деревья. А само содержание теперь может быть дополнено анимацией, звуками и видео. Таким образом, книга-игра почти полностью перевоплощается в компьютерную игру. Наиболее перспективным направлением компьютерных игр сейчас являются игры для смартфонов. А среди смартфонов более 80% аппаратов используют операционную систему Android. Поэтому, наиболее актуальным будет написание книги-игры в формате игры для Android. В связи с вышесказанным было принято решение написать приложение-визуальный

редактор, которое позволяло бы создавать интерактивные книги и собирать их в различные форматы: pdf, html и самое главное – в приложение для ОС Android.

В общем случае, интерактивная новелла представима ориентированным графом с начальной вершиной. Каждому параграфу соответствует вершина графа, а каждому переходу – дуга. В рамках курсовой работы по дисциплине “Архитектурное проектирование и паттерны программирования” нами был спроектирован и реализован прототип визуального редактора как десктопное приложение на языке C#.

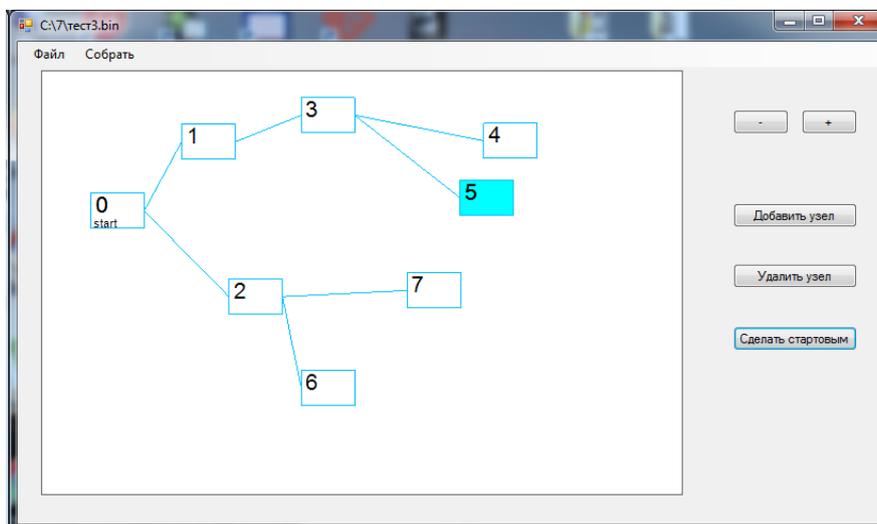


Рисунок 1 – Визуальный редактор связей между параграфами

На представленной форме можно добавлять, удалять параграфы, а также перемещать их по своему усмотрению в пределах окна. Добавление текста, изображений и переходов осуществляется в окне редактирования параграфа.

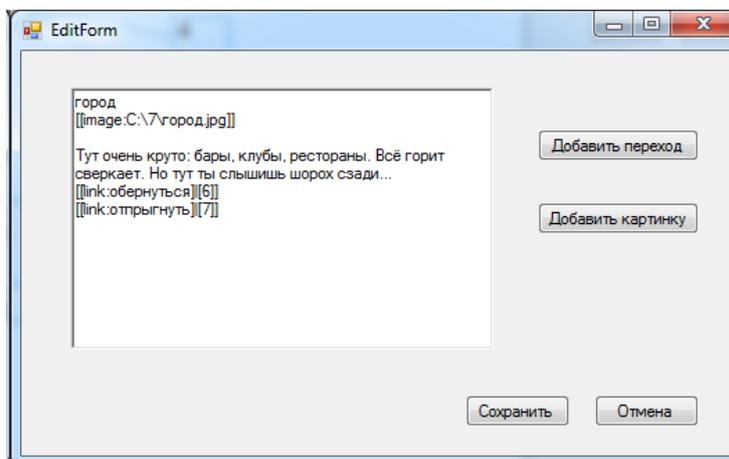


Рисунок 2 – Редактирование переходов в параграфе

По окончании наполнения книги, она собирается в один из форматов (pdf, HTML, Android-приложение). В текущей версии реализовано сохранение в pdf формат, пример приведен на рисунке:

0
Коль поедешь ты направо
За зелёную дубраву,
В Лукоморье попадёшь,
Знай, что дуб с котом не ложь.
Там Яга живёт старушка
В ветхой маленькой избушке,
На куриных двух ногах,
В путников вселяет страх.

А налево повернёшь,
Здесь проедешь через рожь,
Дальше двести вёрст лесами -
Коль любím ты небесами,
То увидишь царский терем.
Плотно в нём закрыты двери
Грідница полна людей.
Хоть суров, но не злодей
Весь в парче и в горностаях
Царь Никита восседает.



У тебя 2 дороги:
налево
направо

Рисунок 3 – Пример сгенерированной книги с условиями и переходами

Список литературы

1. Издательство Сергея Селиванова. Что такое книга-игра? [Электронный ресурс]. Режим доступа - <http://bookalive.ru/>
2. Что такое "книги-игры"? [Электронный ресурс]. Режим доступа - <http://quest-book.ru/gamebooks/>
3. Виртуальные детективы [Электронный ресурс]. Режим доступа - <http://book.net.ua/>
4. Интервью Браславского Дмитрия Юрьевича [Электронный ресурс]. Режим доступа - <http://quest-book.ru/forum/viewtopic.php?t=2494>

СИСТЕМА ЭЛЕКТРОННОГО УЧЕТА ПОСЕЩАЕМОСТИ ЗАНЯТИЙ НА ОСНОВЕ RFID СЧИТЫВАТЕЛЯ ARDUINO

Чураков М. – студент, Старолетов С.М. – к.ф.-м.н., ст. преподаватель
Алтайский государственный технический университет (г. Барнаул)

Все чаще мы встречаем RFID (Radio Frequency Identification) [1] технологии в жизни. RFID - системы автоматической идентификации объектов с помощью специального оборудования: меток, считывателей, чипов, карт. Технология радиочастотной идентификации использует энергию электромагнитного поля для чтения и записи информации на небольшое устройство - RFID метку. Информация на ней может перезаписываться и дополняться.

Примерами являются наклейки в магазинах на товарах, в электронных пропусках, в брелоках для домофонов, в библиотеках. В АлтГТУ используются электронные пропуска на технологии RFID. Все они привязаны к каждому студенту. Но кроме как на вход и выход из университета они не используются. С помощью RFID меток можно решить, например, следующие задачи в университете:

- доступ в аудитории только по пропускам преподавателей;
- RFID наклейки на книги и выдача книг по электронному пропуску;
- ведение электронного журнала посещаемости студентами занятий.

Целью настоящей работы является исследование RFID технологии и применение ее в университете для ведения электронных журналов посещаемости.

Задачи работы:

- сборка устройства считывателя;
- разработка проекта программного обеспечения для достижения цели.

Для реализации работы потребовались: считыватель RFID меток, который выводит данные с пропуска в COM – порт, программа, которая будет считывать данные с порта и база данных, в которой будет храниться уникальный номер карты и данные студента.

Чтобы считать такую метку, нужен RFID считыватель, наподобие тех, которые используются в турникетах. В виду ограниченного бюджета, было принято решение собрать считыватель в домашних условиях. Для его создания использовалось популярное устройство Arduino и RFID модуль.

Arduino[2] – это устройство на основе микроконтроллера ATmega328, которое может взаимодействовать с различными модулями, с компьютером, со смартфоном и прочими устройствами.



Рисунок 1 – Arduino Uno

Назначение Arduino в данной работе - взаимодействие с RFID модулем на чтение данных с метки и отправки данных в COM – порт. Программа для устройства пишется на

упрощенном языке Си – Processing или Wiring. Для написания программы микроконтроллера, использовалась бесплатная IDE Arduino.

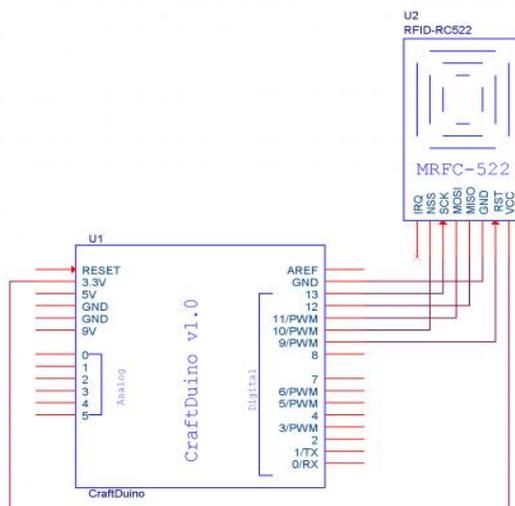


Рисунок 2 – Схема подключения RFID модуля к Arduino Uno

База данных создавалась в Microsoft SQL Server 2008 R2.

Основное приложение написано на языке программирования C++ в QtCreator. Для работы с COM – портом использовалась библиотека QSerialPort. Для доступа к базе данных применялись библиотека QSql. Программа собирает данные о каждой проведенной карточке с указанием даты и своим личным номером (для определения аудитории).

В конечном варианте исполнения, в программе будут реализованы функции, такие как:

- Наличие или отсутствие конкретного студента на занятиях;
- Анализ посещаемости занятий каждым студентом, группой.
- Анализ посещаемости определенных предметов.

Текущая версия сборки оборудования и ПО позволяет работать только напрямую с компьютером. В дальнейшем будет добавлен Ethernet - модуль для передачи данных с нескольких Arduino устройств на сервер.

Список литературы

1. Arduino RFID RC-522 datasheet [Электронный ресурс.] – Режим доступа <http://playground.arduino.cc/Learning/MFRC522>
2. Arduino Uno Datasheet [Электронный ресурс]. – Режим доступа: http://www.arduino.cc/en/uploads/Main/Arduino_Uno_Rev3-schematic.pdf

ФРЕЙМВОРК ДЛЯ ПОСТРОЕНИЯ СИСТЕМ ОБНАРУЖЕНИЯ ДВИЖЕНИЯ И ТРЕКИНГА ОБЪЕКТОВ

Некрасов Д.В. – студент, Старолетов С.М. – к.ф.-м.н., ст. преподаватель
Алтайский государственный технический университет (г. Барнаул)

В последнее время системы видеонаблюдения получили широкое распространение. Это связано с тем, что со временем такие системы упали в цене, и всё больше компаний стали рассматривать эти системы, как инструмент обеспечения безопасности своих объектов.

Для обеспечения безопасности в системе видеонаблюдения должны быть решены две

основные задачи - обнаружение движения и сопровождение (трекинг) объектов. Для решения этих задач существует множество алгоритмов, каждый из которых подходит для какого-то конкретного случая [1, 2]. Иерархия методов обнаружения движения приведена на рисунке 1.

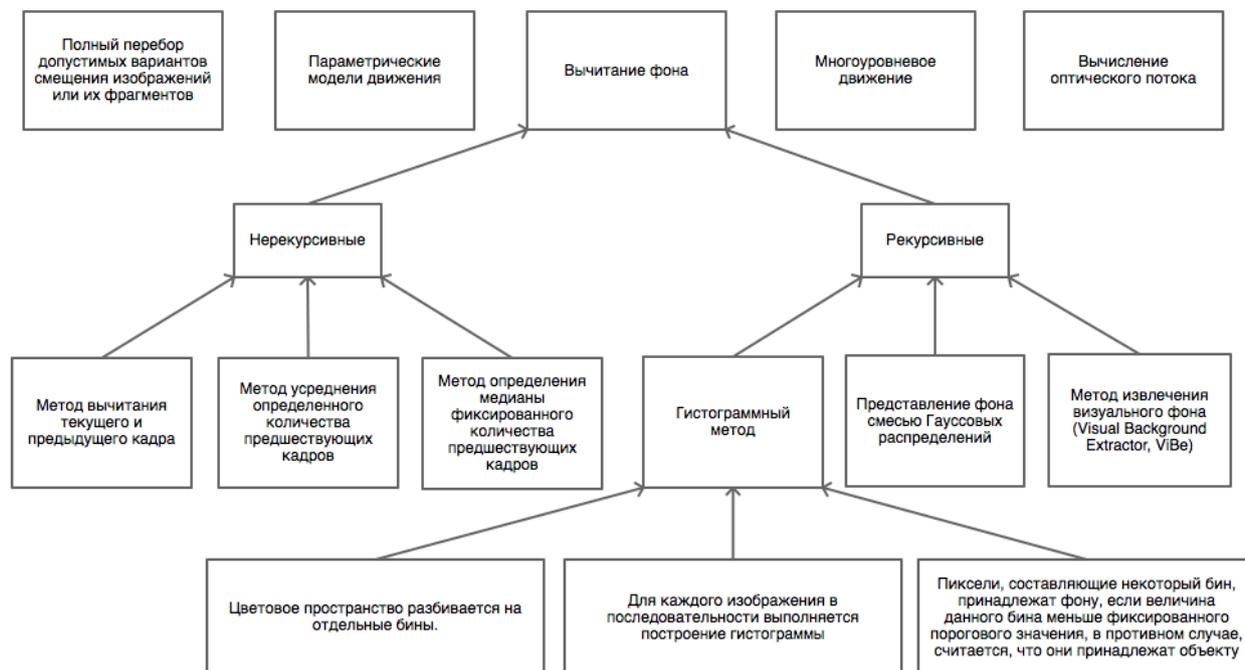


Рисунок 1 – Иерархия методов обнаружения движения

В системах видеонаблюдения камера статична и не меняет ракурс, фон является малоподвижным, незначительно меняющимся между кадрами. Поэтому в таких системах для обнаружения движения наиболее эффективным является метод вычитания фона [3]. Нам необходимо выделить движущиеся объекты на каждом кадре. В результате мы должны получить прямоугольник, ограничивающий объект, или попиксельную маску объекта. В общем виде метод вычитания фона заключается в следующем:

1. Возьмём чистое изображение без объектов – фон.
2. Вычтем фон из новых изображений с объектами.
3. Сравним разницу для каждого пикселя с определённым порогом.
4. Если разница больше порога, то считаем пиксель принадлежащим переднему плану.
5. Получаем бинарную маску переднего плана (фон - 0, передний план - 1)

Для каждой конкретной задачи необходимо использовать некоторые модификации этого метода. Например, если фон незначительно менялся от кадра к кадру (колыхание деревьев), необходимо применить фильтрацию бинарной маски (медианный фильтр или фильтр Гаусса). Более того, с течением долгого времени фон может меняться значительно (например, смена времен года). В таком случае необходимо предусмотреть смену фона. Могут быть случаи, когда нам не удастся получить чистое изображение без объектов. В таком случае можно взять среднее значение по всем кадрам и считать это фоном.

Следующей задачей является трекинг объектов. Для того, чтобы сопровождать объект, нам необходимо сначала выделить его на кадре. Для этого можно воспользоваться методом вычитания фона, либо выделить его вручную, указав область, в которой этот объект находится. Кроме того, можно воспользоваться готовым решением для обнаружения

конкретных объектов (например, детектор людей). Задача сопровождения объекта заключается в определении положения этого объекта на последующих кадрах. В результате мы должны получить последовательность координат объекта в видео фрагменте. Такая последовательность называется след объекта. Задача трекинга объектов осложняется тем, что внешний вид объекта меняется с течением времени. Кроме того, сцена может состоять из нескольких объектов, которые могут перекрывать друг друга. Существует много способов для сопровождения объектов. Основные из них:

1. Ассоциация обнаружений между кадрами.
2. Сопоставление шаблонов.
3. Стая точек.
4. Сдвиг среднего.
5. Комбинация методов.

Наиболее часто используемым методом трекинга является ассоциация, или сопровождение через обнаружение. Допустим, у нас есть метод, который позволяет выделять движущиеся объекты на многих кадрах. Задача сопровождения заключается в том, что нам нужно определить, какому объекту следующего кадра соответствует объект на текущем кадре. Для этого существует множество способов. Простейший способ - поиск ближайшего объекта. Кроме того, для объекта можно вычислить некоторые признаки (например цвет, текстуру форму и т. д.) и искать на следующих кадрах объекты с похожими признаками. Для более эффективной работы метода также можно следить за тем, чтобы вектор скорости данного объекта менялся не сильно относительно предыдущего кадра.

Если объект, который мы хотим сопровождать, выделить вручную, то можно построить упрощённую модель этого объекта и искать на следующем кадре область, похожую по своим характеристикам, на область, занимаемую нашим объектом на первом кадре. В этом случае у нас отсутствует этап обнаружения движения, что может ускорить работу алгоритма. Кроме того, этот метод работает при динамическом изменении фона, что иногда может быть полезно.

Для построения модели объектов существует несколько способов. Мы можем выделить объект, выделенная область будет шаблоном, и затем мы будем искать объекты на последующих кадрах по этому шаблону. Также можно найти некоторое количество характерных точек на объекте и искать движение этих точек. Кроме того, мы можем посчитать некоторые характеристики (например гистограмму цветов) и искать на последующих кадрах область, похожую по этим характеристикам на область объекта.

Как мы видим, в задачах обнаружения движения и сопровождения объектов существует множество различных реализаций, различных характеристик и параметров, которые можно варьировать в зависимости от решаемой задачи. Поэтому хотелось бы спроектировать и реализовать инструмент, с помощью которого можно за короткий промежуток времени решать задачи, связанные с обнаружением движения и трекинга объектов. Для этого было предложено реализовать фреймворк, с помощью которого можно было бы решать такие задачи. Фреймворк – это программная платформа, определяющая структуру программной системы, облегчает разработку и объединение разных компонентов большого программного проекта. Фреймворк отличается от понятия библиотеки тем, что библиотека может быть использована в программном продукте просто как набор подпрограмм близкой функциональности, не влияя на архитектуру программного продукта и не накладывая на неё никаких ограничений. В то время как каркас диктует правила построения архитектуры приложения, задавая на начальном этапе разработки поведение по умолчанию, каркас, который нужно будет расширять и изменять согласно указанным требованиям[4].

Фреймворк будет позволять загружать видео, размечать на нём области, на которых будет производиться обнаружение движения. Для обнаружения движения можно воспользоваться уже реализованным во фреймворке методом вычитания фона, самостоятельно задавая и изменяя некоторые параметра алгоритма, в зависимости от поставленной задачи. Также, алгоритм можно изменить под себя, написав собственную реализацию некоторых методов. Кроме того, во фреймворке будет представлена некоторая реализация алгоритмов сопровождения объектов (через обнаружение и с помощью выделения объектов вручную). Здесь также можно будет изменять характеристики алгоритма. Часть алгоритмов будет реализована с использованием библиотеки OpenCV.

Во фреймворке будут предусмотрены интерфейсы, реализуя которые пользователь может выполнять какие-либо действия с обнаруженными объектами, например распознавать эти объекты, записывать информацию о них в базу данных и т. д. В рамках летней практики нами было реализовано распознавание жестов с использованием каскада Хаара, реализованного в OpenCV [5].

Список литературы

1. Кустикова В.Д. Разработка мультимедийных приложений с использованием библиотек OpenCV и IPP. Отслеживание движения и алгоритмы сопровождения ключевых точек. Н. Новгород, Нижегородский государственный университет им . Н.И. Лобачевского, 2013. – 94 с.
2. Форсайт Д. А., Понс Ж. Компьютерное зрение. Современный подход. М., Вильямс, 2004. – 928 с.
3. Конушин А. Видео-лекции спецкурсов ВМК МГУ “Введение в компьютерное зрение” и “Дополнительные главы компьютерного зрения” – Режим доступа: <https://www.lektorium.tv/course/22847?id=22847>
4. Фреймворк – 2015 – Режим доступа: <https://ru.wikipedia.org/wiki/Фреймворк>
5. Обучение OpenCV каскада Хаара – 2014 – Режим доступа: <http://habrahabr.ru/post/208092/>

АВТОМАТИЗИРОВАННАЯ СИСТЕМА ВРАЧА-ВЕТЕРИНАРА

Колесников Н., Мелехов Д. – студенты, Старолетов С.М. – к.ф.-м.н., ст. преподаватель
Алтайский государственный технический университет (г. Барнаул)

В любом городе имеется большое количество домашних животных, следовательно, существуют и ветеринарные клиники для их обслуживания и лечения. В Барнауле имеется несколько таких клиник, и по заказу одной из них нами разрабатывается программное обеспечение для автоматизации деятельности.

В клинике имеется ПК, на котором принимаются и обрабатываются заявки на ветеринарное обслуживание и ведется электронный журнал истории болезни. В настоящее время все данные хранятся в таблице Excel, что затрудняет поиск, автоматическую обработку и синхронизацию данных между врачами. Данную таблицу было решено заменить десктопным C# приложением.

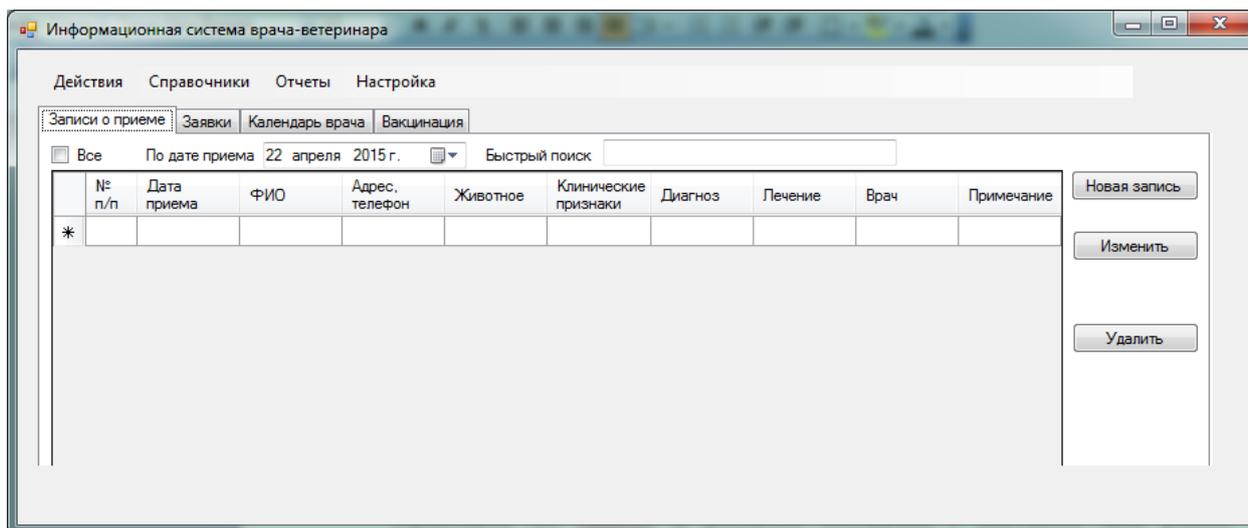


Рисунок 1 – прототип приложения для ПК

Также в клинике работают врачи по заявкам и используют планшеты под управлением ОС Android. В результате обсуждения с заказчиком было решено реализовать мобильное приложение для ведения выездной истории болезни животного. А также разработать средства синхронизации данных с десктопной версией.

Решаемые задачи:

- Учет животных и их историй болезни.
- Регистрация диагнозов.
- Учет заявок.
- Календарь заявок для врача на день.
- Учет вакцинаций.
- Поиск информации.
- Учет материалов и выполненных работ.
- Отчеты.
- Синхронизация между версией для ПК и мобильной версией.

Список литературы

1. Android. – Режим доступа: <http://www.android.com>

КЛИЕНТ-СЕРВЕРНАЯ СИСТЕМА ПОМОЩИ БЕЗДОМНЫМ ЖИВОТНЫМ

Балакин С., Нгуйен Д. – студенты,
 Сучкова Л.И. – к.т.н., профессор, Старолетов С.М. – к.ф.-м.н., ст. преподаватель
 Алтайский государственный технический университет (г. Барнаул)

В городах всего мира очень остро стоит проблема бездомных животных. Много организаций и простых людей пытаются внести свой вклад в решение этой проблемы. Кто-то просто берет животных к себе на время, кто-то помогает деньгами или едой, другие же создают приюты. Но приюты не всегда могут помочь всем и сразу, а для людей, которые могут помочь, не всегда удобно каждый раз ездить в приют. Так же в приюте далеко не

всегда автоматизирована система учета домашних животных. Все это побудило нас создать клиент-серверное приложение для организации помощи бездомным животным.

Сегодня в эпоху информационных технологий мобильные приложения играют важную роль на рынке программного обеспечения. В последнее время наметилась устойчивая тенденция роста в данном сегменте. Мобильные приложения являются разнообразными, богатыми, ежедневно меняющимися вот уже в течение нескольких лет. Наиболее успешной в этом плане оказалась операционная система Android, выпущенная компанией Google. И сегодня эта операционная система имеет много конкурентоспособных преимуществ перед другими мобильными платформами. Главными её достоинствами являются доступность, распространенность, простота использования, а также невысокая цена. Всё это позволило Android охватить очень большую аудиторию пользователей.

Благодаря этим достоинствам мы можем быстро решить большое количество проблем, включая и нашу задачу о распространении информации о бездомных животных. Поэтому в нашем проекте мы решили использовать Android-платформу для разработки клиентской части.

Для автоматизации приюта целесообразно использовать web-приложение, которое может использоваться для поиска и передачи животных всеми желающими, а также для наполнения информации и решения задачи учета денежных средств администраторами после прохождения аутентификации.

Постановка задачи

Необходимо создать клиент-серверное приложение, которое позволит автоматизировать работу приютов для бездомных животных, а также позволит обычным пользователям узнавать о животных, которые там находятся. Было решено, что наше приложение будет состоять из следующих компонентов:

1. Мобильное приложение, функционирующее на операционной системе Android.

Основные функции приложения:

- Просмотр всех животных из приютов.
- Просмотр животных из социальных сетей Facebook, Vkontakte.
- Отправка анкеты о найденном животном.
- Возможность фотосъемки животного и отправка в анкету фотографии.
- Возможность указания местоположения животного на GoogleMap.
- Возможность сделать звонок в приют прямо из приложения.
- Сортировка животных по критериям.
- Полезная информация.

2. Серверная часть, функционирующая на Java Spring MVC.

Основные функции серверной части:

- Добавлять, удалять, редактировать данные о животных.
- Добавлять, удалять, редактировать данные о приютах.
- Добавлять, удалять, редактировать данные о передержках.
- Добавлять, удалять, редактировать данные о сотрудниках.
- Добавлять, удалять, редактировать данные о волонтерах.
- Добавлять, удалять, редактировать данные о спонсорах.
- Добавлять, удалять, редактировать данные о финансах.
- Отслеживание прохождения финансовых потоков в приюте.

3. База данных, а также средства для отправки и приема данных между серверной частью и мобильным приложением.

Заключение

Система в настоящий момент реализуется командой разработчиков, с использованием гибких методологий разработки и подхода TDD.

Список литературы

1. Android. – Режим доступа: <http://www.android.com>
2. Spring. Web MVC framework. Режим доступа: <http://docs.spring.io/spring/docs/current/spring-framework-reference/html/mvc.html>

СИМУЛЯТОР ПРОЦЕССОВ ЖИЗНЕДЕЯТЕЛЬНОСТИ ПАСЕЧНОГО ХОЗЯЙСТВА

Макаров А.В., Чуешев А.В. – студенты, Крайванова В.А. – к.ф.-м.н., доцент
Алтайский государственный технический университет (г. Барнаул)

Медоносная пчела давно стала необходима человеку. В доисторические времена и в глубокой древности человеку нужен был только мед, а затем воск. С ростом знаний появилась потребность и в других продуктах пчеловодства. У меда и других пчелопродуктов имеется особая способность запускать саморегулирующиеся процессы в организме, восстанавливать необходимое равновесие и действовать с профилактической целью. Мед – единственный продукт, который усваивается полностью. Через него в организм человека передается природный потенциал, набор макро- и микроэлементов, ферментов, активных веществ, которые просто не заменимы на стадиях развития и роста.

Однако, основное значение пчел не в том, что они производители меда, воска, прополиса и других продуктов пчеловодства, а в том, что они — основные опылители почти всех цветковых растений. Пчелы играют важную роль в опылении цветущих растений, являясь самой многочисленной группой в экосистемах, связанных с цветами. Они выполняют практически 80% всей опылительной работы и лишь 20% всех растений опыляют шмели и бабочки. В настоящее время существует реальная угроза вымирания пчел. Исчезновение пчел с нашей планеты может усилить уже начавшийся продовольственный кризис.

Вместе с тем, пчелиный рой является одной из сложнейших и интереснейших форм организации общественной жизни животных. В связи с этим являются актуальными следующие цели: популяризация пчеловодства и создание компьютерных моделей жизни этих насекомых. Для их достижения предлагается разработать симулятор жизнедеятельности пасечного хозяйства на основе компьютерного моделирования процессов и событий жизнедеятельности пчел. Существующие аналоги [1,2] обладают весьма узким функционалом и не решают проблему в комплексе.

Целевая аудитория проектов – это как начинающие пчеловоды, так и опытные пасечники. Для первых это средство изучения азов ведения пасеки в интерактивной виртуальной среде. Для вторых – возможность провести различные эксперименты с помощью виртуального инструмента, моделируя пасеку по своему усмотрению, создавая те или иные условия (погода, местоположения сбора пыльцы, источники воды и т.д.).

Этот проект поможет пчеловодам углубить знания по профессиональной области, получить удобный инструмент для ведения пасечного учета, эффективнее организовать пасеку и уход за ней. Тем самым, повысится не только выживаемость пчеловодного хозяйства, но и увеличится количество производимой семьями продукции (мёда, цветочной пыльцы, прополиса, маточного молочка, пчелиного яда, пчелопакетов, новых пчелосемей, маток и т.д.).

Симулятор процессов жизнедеятельности пасечного хозяйства - это широкотематический биологический симулятор, где пользователю предстоит выполнять работу (задачи) профессии (деятельности) пасечника, изучать процессы и события, происходящие на смоделированной пасеке. Модель проекта предполагает каркас для последующих модификаций и получения необходимых комбинаций факторов жизнедеятельности пчелиных семей в различных условиях и на разных уровнях:

- пасека (со всеми внешними погодными и внутренними условиями);
- отдельно взятый улей (со внутренним микроклиматом);
- пчелиная семья (координация жизнедеятельности семьи);
- отдельно взятая пчела (изучение повадок и жизнеспособности).

Симулятор генерирует виртуальную область пространства на основе входных данных:

- регион пасеки;
- климатические условия сезонов года (минимальная температура, средняя температура, максимальная температура, выпадение осадков, ветра и пр.);
- местоположения точек сбора пыльцы (удаленность, размерность, доходность и пр.);
- растительность на пасеке (сад, деревья, трава, кустарники и пр.);
- количество и расположение ульев (расстояние друг от друга, расположение на основе сторон света, ветра и пр.);
- форма и состав ульев (размерность, тип рамок, количество вмещаемых рамок и пр.);
- данные о семьях по годам (матка, расплод, болезни, паразиты и пр.);
- текущее время года на пасеке;
- учетные данные (пасечники, помощники и др.).

После генерации модели следует стадия обработки всех процессов жизнедеятельности пасеки в зависимости от выбранного режима:

- полная обработка (обработка всех процессов и событий) в реальном времени;
- уровневая обработка (на основе каркаса и выбора подуровня) в реальном времени;
- вероятностная обработка (на основе текущих данных строится вероятностная модель последующих событий) в определенном пользователем диапазоне времени.

В ходе или по итогам стадии обработки процессов, происходящих на пасеке, пользователь может принимать непосредственное участие в наблюдении, изменении, создании и удалении объектов на хозяйстве, а также их свойств.

Перспективы развития программного продукта:

- широкие возможности применения и большой потенциал приложения, как на научной основе, так и на развлекательной;
- централизованная информация о пчеловодстве, благодаря работе со многими профессиональными пчеловодами;
- высокая обратная связь пользователей программы позволит ускорить процесс разработки конечного программного продукта;
- внимание общественности к реально существующей проблеме;
- достижения в области науки и получения нового опыта пчеловодства.

Возможные пути развития:

- создание единой базы данных и знаний по профессиональной деятельности (ведение пасеки, болезни пчел и др.) в виде сайта или книги в содействии с сообществом пчеловодов;
- объединение пасечников в единую онлайн-сеть для общения, взаимопомощи и обмена своими наработками, информацией о пасеке и сгенерированными моделями;
- создание отдельного программного продукта для ведения пасечного учета;

- предоставление пользователями программного продукта результатов моделирования общественности для познавательной и научной деятельности.

Список литературы

1. BeeOrganizer <http://beeorganizer.ru>
2. Программы для пчеловодов // Компьютер и пчелы, <http://p4elovod.com/programmy>
3. Рут А. Энциклопедия пчеловодства / Рут А. - «Художественная литература» - 1993 г.
4. Феферман А.Е. Достижения науки и передовой опыт в пчеловодстве / Феферман А.Е. - Россельхозиздат - 1966 г.
5. Материалы международной научно-практической конференции «Современное пчеловодство. Проблемы, опыт, новые технологии», г. Ярославль 2010
6. Ловчиновская М.Я. Болезни пчел / Ловчиновская М.Я. - Колос - 1970 г.
7. Смирнов К.И. Достижения науки и передовой опыт в пчеловодстве / Смирнов К.И. - Всесоюзное совещание пчеловодов-опытников - 1965 г.

ИССЛЕДОВАНИЕ ВОЗМОЖНОСТЕЙ LMS ILIAS ПО ИНТЕГРАЦИИ С МОБИЛЬНЫМ ПРИЛОЖЕНИЕМ

Чушев А.В. – студент, Андреева А.Ю. – к.ф.-м.н, доцент, Крайванова В.А. – к.ф.-м.н, доцент
Алтайский государственный технический университет (г. Барнаул)

Введение

Современная скорость технологического прогресса предъявляет повышенные требования к знаниям и умениям человека, к его способностям получать новую необходимую информацию и успешно применять ее в практике. Использование систем дистанционного обучения (СДО) позволяет повысить эффективность образовательного процесса за счет интенсификации и персонификации обучения. Следующим шагом в усовершенствовании образовательных технологий становится так называемое мобильное обучение -использованиемобильных и портативных ИТ-устройств, таких как смартфоны, планшетные компьютеры, ноутбуки, в преподавании и обучении.

Изменения в сфере образования, в способах получения новых знаний и умений неизбежны. В таких условиях мобильное обучениевидится хорошим способом повысить эффективность доставки образовательного контента, что позволит студентам получать новые знания от лучших преподавателей из разных стран, находясь в любой точке мира, постоянно контролировать свои результаты и делать упор на необходимые умения, используя специальную виртуальную среду.

Таким образом, к современным СДО предъявляется важно требование – быть не просто сетевой системой доставки образовательного контента,но и возможность эффективного использованияпортативныхИТ-устройств, в том числе в оффлайн-режиме.

Цель исследования выбор инструментов реализации мультиплатформенного решения для мобильного обучения в АлтГТУ.

Причины выбора СДО ILIAS

В качестве системы для интеграции с мобильными платформами была выбрана СДО ILIAS. Конечно, данный продукт наибольшее распространение получил на территории Германии, и не имеет такого влияния как СДО Moodle в нашей стране, однако данная система отвечает всем современным требованиям и имеет ряд особенностей и преимуществ над

другими (включая преимущества над СДО Moodle), которые необходимы нам для реализации поставленной цели.

Прежде всего, система ILIAS является SCORM-сертифицированной, что позволяет без лишних усилий унифицировать многие процессы, получить независимость от платформы и предусмотреть возможность обмена курсами с другими системами на основе SCORM-пакетов. Данная система имеет интерфейс, адаптированный для мобильных устройств, что является одним из важнейших требований для реализации поставленной цели. Кроме того, переобучение пользователей не станет проблемой, поскольку система русифицирована, обладает встроенным редактором, интуитивно-понятным интерфейсом, позволяет пользователю иметь собственный репозиторий, персональную среду обучения, вести общение с преподавателем для более качественного изучения материала.

Выбирая СДО ILIAS, следует учесть вышеописанные преимущества, а также обратить внимание на тот факт, что система не интегрирована с мобильными платформами на территории РФ в отличие от стран ЕС, в которых сторонние разработчики уже активно развивают свои приложения в данной сфере. Все вышеперечисленное позволяет надеяться на хорошие темпы роста использования данной системы, а соответственно и популярность мобильного приложения в нашей стране.

Способы интеграции мобильных приложений с LMS ILIAS

В 2011 году на технической конференции в Берне разработчиками ILIAS были предложены некоторые способы по интеграции данной СДО с мобильными приложениями [2]:

- использование адаптивного интерфейса системы.
- ILIAS Mobileplugin с созданием шаблонов.
- фреймворк jQuery mobile [5].
- разработка нативных приложений.
- фреймворк Phonegap (Cordova) [6].

На первый взгляд наиболее предпочтительные методы это использование ILIAS MobilePlugin и разработка нативных приложений для каждой из известных платформ. Установка плагина на сервер LMS, который на сегодняшний день реализует порядка 80% функционала системы, позволит программисту переопределить стили и шаблоны для представления информации на экранах мобильных устройств, но работа пользователя без прямого подключения к СДО (offline-работа) будет невозможна. С такой же проблемой придется столкнуться при использовании адаптивного интерфейса системы в связке с сенсорно-ориентированным web-фреймворком для разработки кроссплатформенных web-приложений jQueryMobile, если целью разработчика является реализация приложения для offline-работы пользователя.

Нативные приложения, несмотря на возможность реализации работы без подключения к системе, требуют больших финансовых вложений и времени в силу особенностей разработки под каждую из мобильных платформ (Android, iOS, WindowsPhone). Таким образом, на первый план в качестве возможного варианта (при условии реализации offline-работы пользователя) выходит использование фреймворка Phonegap (Cordova). Phonegap (Cordova) – это бесплатный open-source фреймворк, который позволяет разрабатывать кроссплатформенные приложения, используя web-технологии, при этом возможна реализация основных функциональных особенностей платформы [6].

Реализация функциональных возможностей платформы достигается путем установки и подключения к проекту приложения специально написанных плагинов, что позволяет работать с такими компонентами устройства, как файловая система, акселерометр, камера и т.д. Из всего вышесказанного использование данного фреймворка видится хорошим способом интеграции LMS ILIAS с мобильным приложением, учитывая при этом возможность работать без прямого подключения пользователя к системе обучения.

Разработка ПО

Выбранная система дистанционного обучения позволяет экспортировать учебные курсы и тесты в форматах XML и SCORM. В качестве первого шага по интеграции мобильного приложения с LMS ILIAS была поставлена задача - реализовать приложение для просмотра учебных курсов, включая учебные модули и главы, экспортируемых из СДО в виде XML-пакетов.

В качестве платформы для реализации был выбран фреймворк Phonegap (Cordova), а для создания GUI – фреймворк jQueryMobile. Использование Phonegap позволило сохранить функциональные особенности устройства, реализовать кроссплатформенность и offline-работу пользователя. Для работы с учебными курсами и модулями скачанные пользователем XML-пакеты необходимо разместить в директории «XMLEngine», которая представляет собой рабочее пространство приложения.

Наиболее сложным моментом при разработке данного ПО было написание компонента, позволяющего «разбирать» пакеты для последующего вывода информации на экран. Для решения возникшей сложности необходимо было изучить структуру самого пакета, а так же подключить необходимые плагины Phonegap к проекту для распаковки и работы с файловой системой устройства. В качестве плагинов были выбраны Fileplugin и Zipplugin, как самые распространенные при работе с фреймворком Phonegap и обладающие отличной документацией [3,4].

Приведем структуру XML-пакета для краткого описания работы компонента для вывода учебных курсов на экран мобильных устройств. В целом XML-пакет представляет собой файл-манифест и ресурсы (текстовая, графическая информация). Файл манифест описывает всю структуру пакета и местоположение подманифестов, которые отвечают за описание структуры и местоположение учебных модулей курса. Таким образом, работа компонента сводится к правильной распаковке XML-пакетов, извлечению файла-манифеста и подманифестов для вычисления местоположения учебных модулей и глав курса. Найденная учебная информация в автоматическом режиме преобразуется для просмотра на мобильных устройствах.

Функционал разработанного приложения:

- Разархивирование XML-пакетов пользователя с сохранением их в кэш.
- Просмотр списка учебных курсов.
- Просмотр списка учебных модулей курса.
- Переход на выбранный учебный модуль.
- Просмотр списка глав учебного модуля.
- Переход на выбранную главу учебного модуля с просмотром информации.
- Сохранение последних посещенных модулей.
- Отслеживание незаполненных учебных модулей, оповещение об этом пользователя.

Разработанное ПО обладает двумя недостатками.

Во-первых, учебные модули пользователю необходимо вручную экспортировать и переносить на мобильное устройство, что повышает риск возникновения ошибок в случае, если пользователь расположил учебный курс не в той директории.

Во-вторых, использование XML-пакетов не позволяет оповещать СДО о достижении каких-либо успехов в изучении учебных материалов.

Направления развития разработанного ПО

В качестве дальнейшего развития проекта необходимо заменить использование XML-пакетов на SCORM-пакеты, для синхронизации с СДО и реализовать доступ к системе обучения через приложение. Кроме того, использование дополнительных web-технологий таких, как AngularJs и ngCordova, позволит разрабатывать приложение согласно концепции

MVC, что напрямую скажется на удобстве написания новых компонентов для приложения и его локализации.

Список литературы

1. Басов И.И., Шнайдер В.Г. Дистанционное преподавание: общие проблемы и личный опыт // APRIORI. Серия: Гуманитарные науки. 2014. №2. URL: <http://cyberleninka.ru/article/n/distantcionnoe-prepodavanie-obschie-problemy-i-lichnyy-opyt> (дата обращения: 16.06.2015).
2. IliasMobile. – Режим доступа: <http://www.slideshare.net/fschmid5001/2011-iliac-conference-berne-iliac-mobile?related=1>
3. Cordova-plugin-file. – Режимдоступа: <https://github.com/apache/cordova-plugin-file>
4. Cordova-plugin-zip. – Режимдоступа: <https://github.com/MobileChromeApps/cordova-plugin-zip>
5. JQuery Mobile API. – Режим доступа: <http://api.jquerymobile.com/>
6. PhoneGap Documentation. – Режимдоступа: <http://docs.phonegap.com/en/edge/>

ПРИМЕР ИСПОЛЬЗОВАНИЯ ФИЗИЧЕСКИХ АППЛЕТОВ В ДИСТАНЦИОННОМ ОБУЧЕНИИ ФИЗИКЕ

Шахворостова Д.Н., Овчаров А.А., Винокуров С.В., Сорокина Е.П. – студенты,
Рогозин К.И. – к.х.н., доцент, Сорокин А.В. – к.т.н., доцент
Алтайский государственный технический университет (г. Барнаул)

Современные клиентские цифровые устройства (персональные компьютеры, планшеты и смартфоны) и поддерживаемое ими программное обеспечение, основанные на использовании информационных и коммуникационных технологий (ИКТ), обладают большими возможностями как конструирования и воспроизведение умопостигаемых образов, а также их операционным интерактивным управлением в реальном времени в соответствии с принимаемыми пользователями решениями в предложенном контексте. По нашему мнению, внедрение современных ИКТ в учебный процесс сейчас определяется не техническими и технологическими возможностями, а способностью специалистов в конкретной области знания их освоить и предложить обучаемым учебный контент (вербальное и иное содержание предметной области) в виде специализированных геймифицированных (игровых) когнитивных (познавательных) продуктов, позволяющих приобрести требуемые компетенции (результаты обучения). Следствием большой сложности создания подобных продуктов, а также необходимостью использования различных линеек клиентских цифровых устройств (Windows, Apple, Android), является принципиальная невозможность конструирования их от начала до конца одним творческим коллективом. Продуктивным в этом случае является путь интеграции существующих опробованных программных средств, адаптированных под технические средства, с унификацией способов доставки и репрезентации учебного контента.

В данной публикации описывается пример реализации такого подхода. Ядром для конструирования является обучающая среда MOODLE, являющаяся свободно распространяемый продукт, удовлетворяющая перечисленным выше условиям. Модель физического процесса симулируются с помощью апплетов написанных на языке Java, созданных в Колорадском государственном университете (США). Обучающая среда, обладая большим набором интерактивных средств, предполагающих деятельность студентов, позволяет организовать (включая и контрольно-измерительные мероприятия) и сделать

однородным весь процесс обучения, включая и предлагаемый пример. Физические апплеты служат для создания конкретной физической ситуации, визуализирующей модель процесса, подлежащего усвоению студентами.

Используемая нами обучающая среда развернута на серверах МГУ им. М.В. Ломоносова и размещена по адресу <http://distant.msu.ru/course/view.php?id=181>.



Рисунок 1 – Титул веб-страницы

На указанной веб-странице размещены учебные материалы, достаточные для выполнения учебных заданий: курс лекций, справочник, презентации лекций, контрольно-измерительные материалы и вспомогательные файлы. Контрольно-измерительные материалы представляются в виде тестов. Предлагаемый в публикации пример

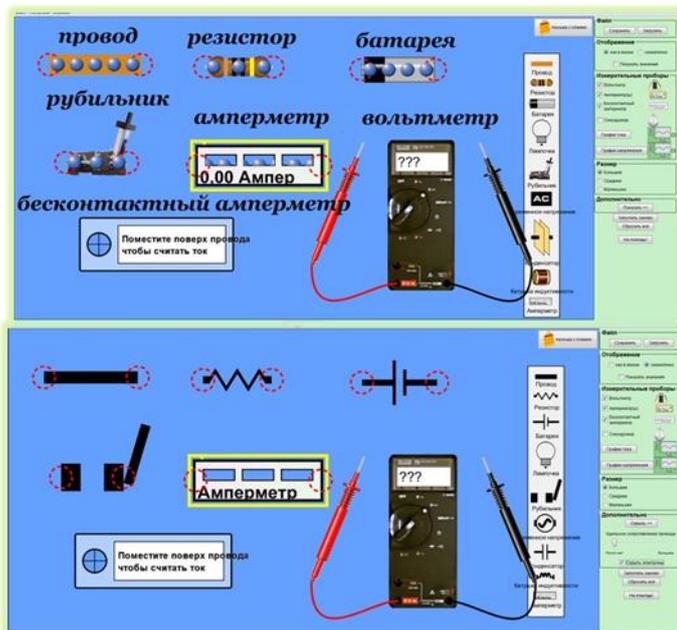


Рисунок 2 – Общий вид панели инструментов

является тестом «Электрический ток. Моделирование», который сопровождается несколькими файлами, загружаемыми студентами.

Тест состоит из девяти заданий, каждое из которых может быть выполнено в любой последовательности. Предложенная нами и реализованная стратегия прохождения теста состоит в том, в нем отсутствует ограничение по времени на его прохождение,

предоставляется три попытки, которые должны быть реализованы в течение 3 календарных дней.

Раздел «Электродинамика» (электрический ток) в физике является одним из наиболее сложных для студентов. Он не может быть освоен без зрительных образов различных сторон и частей возможных явлений. Последние должны быть доступны для деятельности обучающимся. Электрические цепи собираются с помощью семи элементов, изображенных на рисунках. Первый способ представления авторы назвали «как в жизни». Второй способ называется «схематично». С использованием данных элементов студенты могут собрать или загрузить созданными заранее разработчиками прилагаемые для изучения и последующего контроля схемы, одна из которых приведена ниже.

Само задание визуализировалось средой MOODLE в представленном ниже виде.

Загрузите апплет из архива, который находился на стартовой странице этого теста.
После чего соберите сами (РАЗМЕР СРЕДНИЙ)

после чего загрузите из того же архива схему "4_ветви"

ВЫПОЛНИТЕ ЗАДАНИЕ при данных номиналах ЭДС и сопротивлений

E1=10 В (ПЛЮС СПРАВА) ; r1=0,5 Ом; E2=8 В СПРАВА); r2=0,6 Ом; E3=9 В (ПЛЮС СЛЕВА); Ом; E4=11 В (ПЛЮС СЛЕВА) ; r4=0,8 Ом; R1=8 Ом; R3=11 Ом; R4=12 Ом.

А теперь поговорим о ВТОРОМ правиле Кирхгофа
Я определил, что

Для КОНТУРА aegca: Выберите... I1R1 Выберите... I1r1
I3R3 Выберите... I3r3= Выберите... E1 Выберите... E3

или в числах Выберите... Выберите... Выберите... Выберите...
Выберите... = Выберите... Выберите... Выберите... Выберите...

Проверить

Рисунок 3 – Вид одного учебного задания

В данном случае оно основано на использовании второго правила Кирхгофа, утверждающего, что «в любом замкнутом контуре электрической цепи алгебраическая сумма произведения тока на сопротивление равна алгебраической сумме ЭДС, действующих в этом же контуре».

$$\sum_i I_i R_i = \sum_k E_k$$

Задание содержит 22 вопроса, ответ на которые нужно выбрать из предлагаемых наборов. Число уникальных вариантов ответов более чем 10^{25} , что исключает возможность выполнения задания путем случайного выбора. Это заставляет студентов сконцентрироваться на сути задания, которое представляет непростую интеллектуальную задачу. Само задания состоит из двух частей.

Первая их них предполагает составление правильной расстановки символом выражении, являющимся выражением второго правила Кирхгофа для данной учебной ситуации. Знаки для падений напряжения зависят от направлений обхода контура и тока,

текущего на конкретном участке цепи. Последнее находится визуально по движению электронов, либо по знаку перед значениями силы тока на амперметре.

Знаки перед электродвижущими силами источников тока зависят от способа их включения в цепь и направлением обхода контура. Вторая часть, требует выбора не только знаков, но и конкретных численных значений физических величин, измеряемых с помощью виртуальных средств измерения – амперметров и вольтметра. Названные особенности подробно описаны в комментариях, которые могут быть загружены студентами до либо во время выполнения заданий. Рассматриваемый тест, был студентам АлтГТУ им. И.И. Ползунова в осеннем семестре 2014/2015гг. включал 9 подобных заданий. В тестировании приняло участие 73 человека.

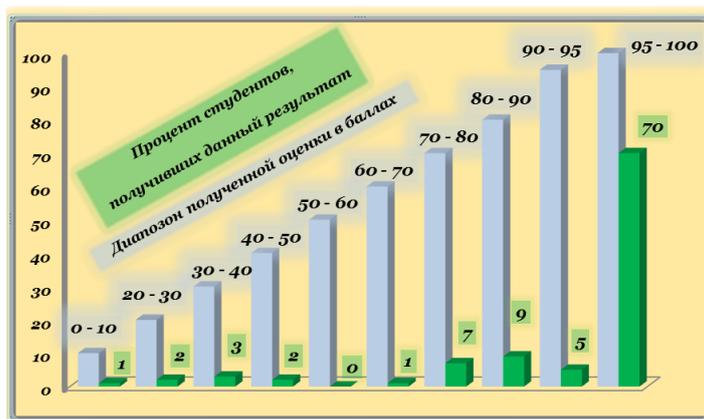


Рисунок 4 – Результаты теста

Несмотря на большую трудоемкость и сложность выполнения заданий для нас был неожиданным результаты прохождения теста, которые приведены на гистограмме. Из приведенных данных следует, что большинство студентов (83%) набрали более 70 баллов из 100. Большое количество (23) максимальных результатов (100 баллов), говорит о том, что задания составлены корректно, а предложенный способ работы является не только интересным обучаемым, но и эффективным. Для этих студентов являлось принципиально важным получение подобного результата. Ниже мы приводим время и результаты по попыткам одного из студентов – Владимира Борзенко.

Таблица 1. Пример прохождения теста

Попытка	Затраченное время	Результат
1.	13 час. 56 мин	48,58
2.	56 мин. 42 сек	95,60
3.	5 мин. 34 сек.	98,76
4.	44 сек	100,00

Уменьшение времени на прохождение каждой последующей попытки объясняется тем, что студенты выполняли один и тот же набор заданий, сформированный из банка (каждый вид заданий имел до 20 вариантов) случайным выбором, и нами реализованы настройки прохождения теста, при которых сохранялись правильные ответы предыдущих попыток. Студентам требовалось лишь исправить допущенные ранее ошибки. Поэтому само тестирование становилось серьезной интеллектуальной игрой, в которой студенты получали не только результат в баллах, но и требуемые компетенции.

Вся работа по созданию заданий была проведена группой студентов под руководством доцента кафедры физики К.И. Рогозина. Студенты учатся на разных специальностях АлтГТУ им. И.И. Ползунова, и, по-видимому, никогда станут преподавателями, тем более физики. Вот мнение одной из них – Дарьи Шахворостовой.

«Я по профессии - будущий энергетик, поэтому, когда я узнала, что буду заниматься заданиями по законам Кирхгофа, сильно заинтересовалась, так как это основы основ моей профессии. Но с другой стороны, я думала, что предстоит большая работа, представляя размеры возможных цепей, расчетов, проверок. Так как изучение данных законов требует особой усидчивости, особенно при решении задач, ведь стоит потерять один знак «-» или «+», и задача будет решена совершенно неправильно. Но, не тут - то было, работа оказалась очень легкой и интересной, очень жаль, что я не знала описанных в статье методов, когда изучала физику. Разобравшись в работе апплетов, можно сильно упростить процесс изучения различных физических законов.

Задания, которые мы представили в тестах, были не очень сложные, но достаточно трудоемкие. Большое количество ответов не давало возможности перейти порог, наткнувшись наугад, поэтому как бы студентом не хотелось, но «прохалаявить» не получилось. Открывая задание, студенты видели не только условие задачи, но и физические апплеты, которые воссоздавали конкретную электрическую цепь, от которой они отталкивались в ходе решения. Поняв суть работы апплета, любой двоечник, ответит на вопросы теста на «отлично». Не маленькие электрические цепи собираются с помощью всего лишь семи элементов, изображенных на рисунках. Собирая цепь, ученик получает, как теоретический опыт, так и практический.

Любой закон в физике бывает не так-то просто понять с первого раза, а особенно без зрительных образов. Физические апплеты делают нагляднее и проще изучение любого физического явления. Тем самым упрощая процесс обучения не только обучающихся, а также учителей и преподавателей.



Рисунок 5 – Команда студентов

Когда был запущен тест в работу, мне было очень интересно, какие мы получим результаты. И, наша работа была оправдана достаточно неплохими результатами. Применение данных физических апплетов на практике оказалось вполне результативным. Я думаю, остались довольны проделанной работой не только разработчики, но и студенты».

Список литературы

1. Konstantin Rogozin, Sergey Kuznetsov, Diana Kondrashova & Irina Lisina. Cognitive Test as a Tool for Physics Learning//Proceedings of the World Conference on Physics Education: February, 2014 Ankara, Turkey – ISBN 978-605-364-658-7, pp. 607-615.

2. Konstantin Rogozin Physics Learning Instruments of XXI Century / Proceedings of the World Conference on Physics Education: February, 2014, Ankara, Turkey – ISBN 978-605-364-658-7, pp. 913-923.
3. Рогозин К.И., Кузнецов С.И., Маерков А.О., Пшенова У.И. Инструменты и средства веб-поддержки традиционных курсов обучения физике // Высшее образование сегодня – 2014, №9 – С. 29 – 32.
4. Рогозин К.И., Боровцов Е.Г., Веряев А.А., Шаповалов А.А. М-learning ресурсный центр для общеобразовательных школ// ИННОВАЦИИ – 2014 – № 10 (192) – С. 30-34.
5. Лапёнок М. В., Макеева В. В. Технология реализации индивидуальной образовательной траектории учащегося школы с использованием электронных образовательных ресурсов. Педагогическое Образование в России – 2012 - Выпуск № 6 .

ТЕСТИРУЮЩАЯ СИСТЕМА ПО ФИЗИКЕ

Овчаров А.А., Соловьенко А.А., Винокуров С.В. – студенты,
 Рогозин К.И. – к.х.н., доцент, Сорокин А.В. – к.т.н., доцент
 Алтайский государственный технический университет (г. Барнаул)

Рост объема знаний в современном мире носит экспоненциальный характер с периодом удвоения их объема, по разным оценкам, от 2 до 10 лет, тогда как накопление знаний индивидом в течение жизни при традиционной системе образования имеет линейный характер ориентировочно до 27 летнего возраста.



Рисунок 1 – Знания созданные и усвоенные

В соответствии с приведенным выше графиком (за единицу знаний взят их объем, созданный в 2000 г.), разрыв между знаниями, которые могут быть усвоены с помощью традиционных методов обучения и теми, что необходимо использовать, никогда не будет ликвидирован, а будет только расти. Это означает, что наряду с традиционными формами обучения и используемыми в них педагогическими методиками необходимо внедрять иные, которые позволяют изменить традиционный линейный характер накопления знаний, ориентированных на физиологические возможности индивидуумов, с учетом их психических возможностей и накопленного опыта получения новых элементов результатов обучения. По нашему мнению такие педагогические техники и методики, а также формы обучения должны базироваться на использовании клиентских цифровых

устройств – персональных компьютеров, планшетов и смартфонов. Об этом говорят результаты опроса проведенного нами среди студентов АлтГТУ, приведенные ниже.



Рисунок 2 – Результаты опроса студентов АлтГТУ им. И.И. Ползунова

Гаджеты (MID – устройства, в англоязычной литературе «Mobil Internet Devices») наиболее важны, поскольку они у современных студентов входят в необходимый набор предметов, которые постоянно находятся в пользовании и потому доступны и привычны. Поэтому первоочередной задачей является создание программных продуктов, которые с одной стороны были легко осваиваемы организаторам учебного процесса (преподавателям) и понятны с точки зрения имеющегося пользовательского опыта обучаемым, а с другой стороны позволяли реализовывать поставленную задачу.

Тот же опрос показал, что более 70% студентов играют разное время (от 10 минут в день до нескольких часов) в различные игры на своих цифровых устройствах. Поэтому представляется целесообразным построить процесс обучения в виде интеллектуальной игры, в результате которой получали не только результат в баллах, но и требуемы результаты обучения. Игра – это естественное состояние с помощью которой человек познает мир и отрабатывает навыки принятия решений в рамках стоящих текущих задач.

В реализуемой нами разработке контрольно-измерительные материалы формируются через создаваемую специализированную тестирующую систему по физике для школьников. Суть системы – предоставление преподавателям школ инструмента для создания вопросов разных типов: сбор формул из паззлов-картинок (лёгкий уровень), а также конструирование формул вручную (уровень сложнее, так как для построения формулы необходимо знать абсолютно точное её написание и угадать не получится). Также планируется ввести другие виды тестов с возможностью расширения системы (предоставление тестов не только для предмета физики, но и для психологии, математики и т.д.). Преподаватель готовит набор вопросов, затем объединяет их в тесты по своему усмотрению, назначает уровень сложности вопросам и устанавливает, какое количество баллов заработает ученик, в случае положительного ответа на очередной вопрос. Система разрабатывается наиболее гибким образом для преподавателя, так что порядок вопросов в тесте можно назначать как вручную, так и автоматической генерацией. Разрабатываемая тестирующая система позволяет создавать отдельные профили учеников с указанием их собственного логина и пароля, формировать списки учеников, объединяя их по группам (классам). Данный подход необходим для фиксирования результатов прохождения теста и выставления оценки по итогу. Важной особенностью является то, что система кроссплатформенная, а значит, может работать на большинстве современных операционных систем: не только семейства Windows, но и на Unix (Linux, FreeBSD, MacOS), что для школ является актуальным на данный момент. На нынешней стадии разработки система уже позволяет создавать тесты на

конструирование формул и корректно предоставляет прохождение с последующим выставлением балла.

Тестирующая среда позволяет разрабатывать тесты из 3-х видов заданий и проходить их. Тестирующая среда позволяет разрабатывать тесты из 3-х видов заданий и проходить их. Задания являются интерактивными и графическими. Ниже приведены примеры заданий.

1. Конструирование формулы. Учитель конструирует сложную формулу из простых элементов. Элемент формулы представляет из себя букву, цифру, атрибут или знак арифметической операции. При прохождении задания ученик пытается воспроизвести правильную формулу из некоторого количества элементов, среди которых встречаются неправильные. В нашей программе осуществлена возможность быстро и удобно сконструировать формулу, используя специальные кнопки для добавления знаков операций, а также отмены последнего действия и очистки элементов. Мы расцениваем такие задания как сложные. Правильность сборки формулы проверяется исходя из того, что множители и слагаемые можно переставлять как угодно, уменьшаемое, вычитаемое, делимое и делитель переставлять нельзя, большие и маленькие буквы считаются разными (Рисунок 3).

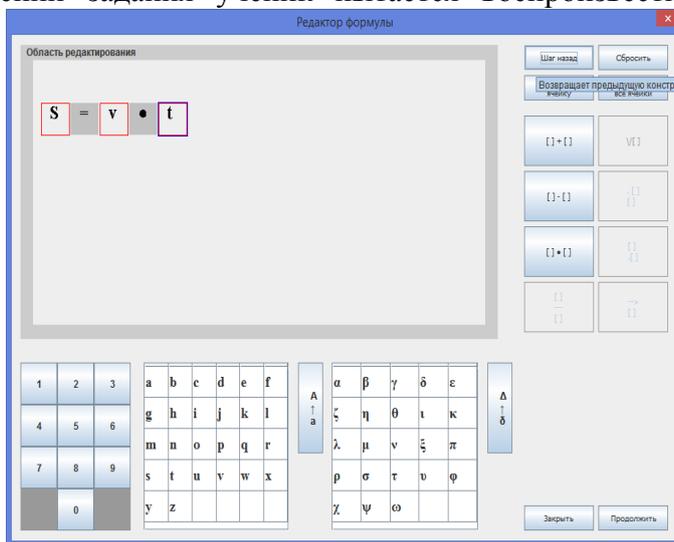


Рисунок 3 – Редактор формул

Таким образом, перед учеником ставятся следующие задачи: а) Вспомнить правильную формулу; б) Правильно понять структуру формулы и расставить знаки операций; в) Выбрать из множества элементов только те, которые принадлежат формуле; г) Расставить элементы в формуле

2. Размещение картинок в области. Учитель добавляет области на экран, в которые требуется разместить выбранные картинки. При правильном размещении картинок, можно собрать формулу или мозаику. Ученик пытается воспроизвести исходную картинку, перетаскивая кусочки из набора в нужные области. Среди кусочков могут встретиться те, которые не относятся к набору правильных. В некоторых заданиях в области может быть расположена только один кусочек (задача на расстановку кусочков), в других заданиях в области могут быть расположены несколько кусочков (задача на классификацию).

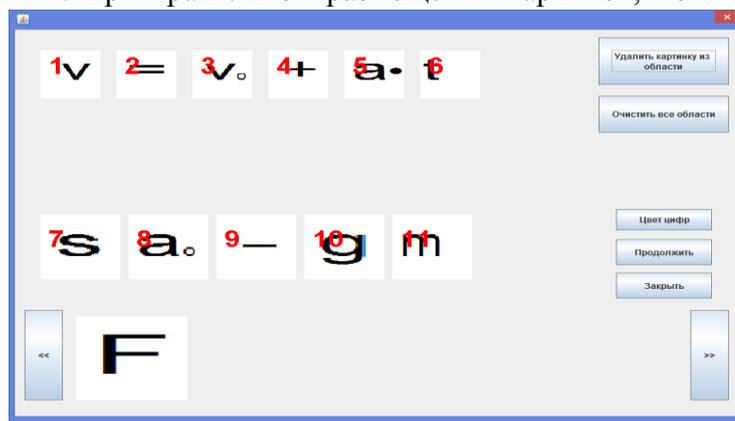


Рисунок 4 – Редактор сборки рисунков

Мы расцениваем такие задания как средние по сложности. Проверка задания выполняется исходя из того, чтобы картинки располагались в нужных областях, и на экране не было пустых областей (Рисунок 4).

Таким образом, перед учеником ставятся следующие задачи: а) Выбрать из набора кусочков те, которые относятся к тематике задания; б) Вспомнить, как должны быть расположены кусочки; в) Расположить кусочки в области.

3. Установить соответствие. Учитель добавляет кусочки картинок на экран и устанавливает соответствие между ними. Кусочки картинок расположены в два вертикальных ряда. Каждой картинке из левого вертикального ряда соответствует одна картинка из правого ряда. Соответствие показывается линией от одной картинке к другой (Рисунок 5). Для прохождения задания ученику необходимо установить соответствие также, как это сделал учитель. Задание считается выполненным, если все соответствия установлены между картинками.

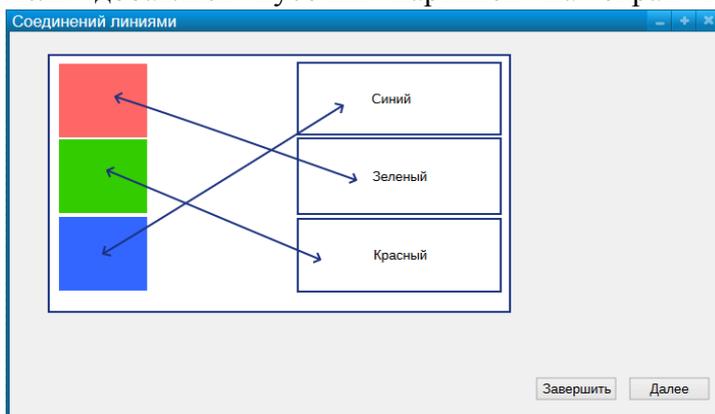


Рисунок 5 – Редактор соответствий

Список литературы

1. Рогозин К.И., Кузнецов С.И., Маерков А.О., Пшенова У.И. Инструменты и средства веб-поддержки традиционных курсов обучения физике// Высшее образование сегодня – 2014, №9 – С. 29 – 32.
2. Рогозин К.И., Боровцов Е.Г., Веряев А.А., Шаповалов А.А. М-learning ресурсный центр для общеобразовательных школ// ИННОВАЦИИ – 2014 – № 10 (192) – С. 30-34.