

Министерство образования Республики Беларусь
Учреждение образования
«Белорусский государственный университет
информатики и радиоэлектроники»

КОМПЬЮТЕРНЫЕ СИСТЕМЫ И СЕТИ

**57-я научная конференция
аспирантов, магистрантов и студентов**

Сборник тезисов докладов

19–23 апреля 2021 года
Минск, БГУИР

УДК 004.021+004.2+004.4+004.6+004.7+004.9
ББК 32.97+32.81

57-я конференция аспирантов, магистрантов и студентов учреждения образования «Белорусский государственный университет информатики и радиоэлектроники», 19-23 апреля 2021 г., БГУИР, Минск, Беларусь: сборник тезисов докладов. – Мн. – 2021. – 155с.; ил.

В сборнике опубликованы тезисы докладов, представленных на 57-й научной конференции аспирантов, магистрантов и студентов БГУИР. Материалы одобрены оргкомитетом и публикуются в авторской редакции.

Для научных и инженерно-технических работников, преподавателей, аспирантов, магистрантов и студентов вузов.

СОДЕРЖАНИЕ

СЕКЦИЯ «ВСТРАИВАЕМЫЕ ВЫЧИСЛИТЕЛЬНЫЕ СИСТЕМЫ РЕАЛЬНОГО ВРЕМЕНИ».....	7
1. ЦЕЛОЧИСЛЕННЫЙ СОПРОЦЕССОР ВЕЙВЛЕТ-ПРЕОБРАЗОВАНИЯ ДЛЯ ПЛИС.....	8
2. РЕАЛИЗАЦИЯ ПОДМНОЖЕСТВА ИНСТРУКЦИЙ СОФТ ПРОЦЕССОРА АРХИТЕКТУРЫ RISC-V RV32I ДЛЯ ПЛИС ТИПА FPGA.....	10
3. ПРОЦЕССОР SNA-3 НА БАЗЕ FPGA	12
4. СИСТЕМА НАБЛЮДЕНИЯ ЗА СТАТИЧЕСКИМИ ОБЪЕКТАМИ В УСЛОВИЯХ ПЛОХОЙ ВИДИМОСТИ	13
5. СВЁРТОЧНЫЕ НЕЙРОННЫЕ СЕТИ ДЛЯ ОБРАБОТКИ РЕЧИ.....	14
СЕКЦИЯ «ПРОГРАММНОЕ ОБЕСПЕЧЕНИЕ ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ»	16
6. ПРОГРАММНО-АППАРАТНЫЙ МОДУЛЬ АУДИОФИКСАЦИИ	17
7. СИСТЕМА АВТОМАТИЧЕСКОГО РЕЗЕРВНОГО КОПИРОВАНИЯ И ОБРАБОТКИ ДАННЫХ НА ЯЗЫКАХ ПРОГРАММИРОВАНИЯ JAVA, JS	19
8. СОВРЕМЕННЫЕ ПОДХОДЫ В ОНЛАЙН-ОБРАЗОВАНИИ.....	22
9. АНАЛИЗ ПОВЕДЕНИЯ ЧЕЛОВЕКА ПО МИМИКЕ ЛИЦА.....	23
10. МОБИЛЬНОЕ ПРИЛОЖЕНИЕ ДЛЯ ВЕДЕНИЯ ДНЕВНИКА ПУТЕШЕСТВИЙ...	24
11. ПРОГРАММНОЕ СРЕДСТВО ОБРАБОТКИ ЗВУКОВЫХ СИГНАЛОВ	25
12. ПРЕДСТАВЛЕНИЕ СИГНАЛОВ В МОДОВОЙ ДЕКОМПОЗИЦИИ	26
13. УСТРОЙСТВО СКАНИРОВАНИЯ ЦВЕТА И МОБИЛЬНОЕ ПРОГРАММНОЕ СРЕДСТВО АНАЛИЗА ЦВЕТОВОЙ ПАЛИТРЫ И ПОДБОРА ЦВЕТА.....	28
14. КЛАССИФИКАЦИЯ ЭЛЕКТРОННОЙ ПОЧТЫ	30
15. СРАВНЕНИЕ LR И GLR СИНТАКСИЧЕСКИХ АНАЛИЗАТОРОВ	31
16. УДАЛЕННАЯ СИСТЕМА УПРАВЛЕНИЯ ДАННЫМИ	33
17. МОБИЛЬНОЕ ПРОГРАММНОЕ СРЕДСТВО УПРАВЛЕНИЯ ПРЕДМЕТАМИ КОЛЛЕКЦИИ НА ОСНОВЕ РАДИОЧАСТОТНЫХ МЕТОК.....	34
18. АППАРАТНО-ПРОГРАММНЫЙ КОМПЛЕКС ПО МОНИТОРИНГУ И АНАЛИЗУ СОСТОЯНИЯ ОКРУЖАЮЩЕЙ СРЕДЫ.....	36
19. НЕРАЗРУШАЮЩЕЕ ТЕСТИРОВАНИЕ ЗАПОМИНАЮЩИХ УСТРОЙСТВ НА БАЗЕ ДВОЙНЫХ АДРЕСНЫХ ПОСЛЕДОВАТЕЛЬНОСТЕЙ	37
20. ПРОГРАММНОЕ СРЕДСТВО АВТОМАТИЗАЦИИ НАЙМА ПЕРСОНАЛА ИТ-КОМПАНИИ	39
21. СИСТЕМА КОНТРОЛЯ ДОСТУПА С ПРОПУСКАМИ НА ОСНОВЕ ТОКЕНИЗИРОВАННЫХ EMV КАРТ	40
22. МОДЕЛЬ МАШИННОГО ОБУЧЕНИЯ ДЛЯ СОРТИРОВКИ ТВЕРДОГО БЫТОВОГО МУСОРА	41

23. ПРОГРАММНОЕ СРЕДСТВО МОНИТОРИНГА ФИЗИЧЕСКОГО РАЗВИТИЯ ДЛЯ МОБИЛЬНОЙ ПЛАТФОРМЫ ANDROID	43
24. НАБОР СРЕДСТВ ДЛЯ УДАЛЁННОГО УПРАВЛЕНИЯ МИКРОКОНТРОЛЛЕРАМИ ARDUINO	45
25. АНАЛИЗ МНОГОМЕРНЫХ ДАННЫХ НА ПРИМЕРЕ ДАННЫХ ПО ОНКОЛОГИЧЕСКИМ ЗАБОЛЕВАНИЯМ ЛЁГКИХ: ПОДБОР ИНФОРМАТИВНЫХ ПРИЗНАКОВ	47
26. ПРИМЕНЕНИЕ ИСКУССТВЕННОЙ НЕЙРОННОЙ СЕТИ ДЛЯ ОПРЕДЕЛЕНИЯ ТИПА МЕСТНОСТИ	48
27. АВТОМАТИЗИРОВАННАЯ СИСТЕМА УПРАВЛЕНИЯ РЕСТОРАНОМ.....	49
28. АЛГОРИТМ ПЕРЕМЕЩЕНИЯ РОБОТА “ПАУКА”	51
29. О ПРОБЛЕМЕ ОПРЕДЕЛЕНИЯ ЭМОТИВНОЙ ОЦЕНКИ ТЕКСТА	53
30. ПОВЫШЕНИЕ ЭФФЕКТИВНОСТИ РАБОТЫ КОМАНДЫ РАЗРАБОТЧИКОВ...54	
31. ПРОГРАММНОЕ СРЕДСТВО МОДЕЛИРОВАНИЯ КАРТЫ НЕИСПРАВНОСТЕЙ ЗАПОМИНАЮЩИХ УСТРОЙСТВ С ЗАДАННЫМИ ПАРАМЕТРАМИ	55
32. ПРОГРАММНОЕ СРЕДСТВО ОРГАНИЗАЦИИ УПРАВЛЕНИЯ ТЕЛЕФОННЫМИ НОМЕРАМИ НА ПЛАТФОРМЕ JAVA	57
33. ПРОГРАММНОЕ СРЕДСТВО УПРАВЛЕНИЯ ОЧЕРЕДЯМИ НА ПЛАТФОРМЕ JAVA.....	59
34. ПРОГРАММНОЕ СРЕДСТВО РЕАЛИЗАЦИИ ФИЛЬТРА КАЛМАНА.....	60
35. ОБЗОР РЕШЕНИЙ ЗАДАЧИ ИДЕНТИФИКАЦИИ АВТОРА ТЕКСТА	61
36. ПРИМЕНЕНИЕ ГЕНЕТИЧЕСКИХ АЛГОРИТМОВ ДЛЯ КРИПТОАНАЛИЗА ПОДСТАНОВОЧНЫХ ШИФРОВ.....	62
37. ПРОБЛЕМЫ И ПЕРСПЕКТИВЫ СТАНДАРТИЗАЦИИ КИБЕРФИЗИЧЕСКИХ СИСТЕМ ОБРАЗОВАТЕЛЬНОГО НАЗНАЧЕНИЯ	64
38. ПРИМЕНЕНИЕ PID РЕГУЛИРОВАНИЯ ДЛЯ УПРАВЛЕНИЯ СУДОМОДЕЛЬЮ	65
39. ПРОГРАММНОЕ СРЕДСТВО ОТРИСОВКИ ТРЕХМЕРНЫХ СЦЕН С ИСПОЛЬЗОВАНИЕМ ТЕХНОЛОГИИ ТРАССИРОВКИ ЛУЧЕЙ	67
40. НАХОЖДЕНИЕ КРАТЧАЙШИХ ПУТЕЙ В ГРАФЕ МЕТОДОМ МАТРИЧНОГО УМНОЖЕНИЯ.....	68
СЕКЦИЯ «ВЫСШАЯ МАТЕМАТИКА»	69
41. КОМПЬЮТЕРНАЯ ПРОГРАММА, ПРОГНОЗИРУЮЩАЯ РОСТ ИНФЛЯЦИИ ..	70
42. ПОЕДИНОК «ГРУЗОВИК – ПЕШЕХОД»: ОПТИМАЛЬНЫЕ ПО БЕЗОПАСНОСТИ ТРАЕКТОРИИ	71
43. ТЕОРИЯ ЧИСЕЛ В АССИМЕТРИЧНОМ ШИФРОВАНИИ.....	73
44. МАТЕМАТИЧЕСКАЯ МОДЕЛЬ ОСЦИЛЛЯТОРА НА ТРЕНИИ.....	74
45. КОМПЬЮТЕРНОЕ ЗРЕНИЕ И ЕГО ОСНОВНЫЕ ЗАДАЧИ	76
СЕКЦИЯ «ЭЛЕКТРОННЫЕ ВЫЧИСЛИТЕЛЬНЫЕ МАШИНЫ»	78
46. ИНФОРМАЦИОННО-КОММУНИКАТИВНАЯ МОДЕЛЬ И УПРАВЛЕНИЕ УЧЕБНЫМ ПРОЦЕССОМ	79

47. ПРИЗНАКИ И КРИТЕРИИ НЕСТАНДАРТНОГО ПОВЕДЕНИЯ ПОЛЬЗОВАТЕЛЯ	81
48. ПРИМЕНЕНИЕ ПОЛЯРИЗАЦИИ ДЛЯ ПОВЫШЕНИЯ ДОСТОВЕРНОСТИ РЕЗУЛЬТАТОВ ЛАЗЕРНОГО ТРИАНГУЛЯЦИОННОГО СКАНИРОВАНИЯ.....	82
49. ПРОГРАММНО-АППАРАТНЫЙ КОМПЛЕКС «БЕСПИЛОТНЫЙ ЛЕТАТЕЛЬНЫЙ АППАРАТ С ДИСТАНЦИОННЫМ УПРАВЛЕНИЕМ»	84
50. МЕХАНИЗМ ПОИСКА ДОКУМЕНТОВ В ИНФОРМАЦИОННО-ПОИСКОВОЙ СИСТЕМЕ	86
51. СОЗДАНИЕ ПЕЧАТНОГО СЕРВИСА С ВОЗМОЖНОСТЬЮ ОПЛАТЫ ЭЛЕКТРОННЫМИ ДЕНЬГАМИ.....	87
52. СЕГМЕНТАЦИЯ СПУТНИКОВЫХ СНИМКОВ С ИСПОЛЬЗОВАНИЕМ НЕЙРОННОЙ СЕТИ U-NET	88
53. АНАЛИЗ ЭФФЕКТИВНОСТИ НЕЙРОННОЙ СЕТИ S2S ДЛЯ УДАЛЕНИЯ ШУМОВ НА ИЗОБРАЖЕНИИ	89
54. СОВРЕМЕННЫЕ МЕТОДЫ АНАЛИЗА СЕТЕВОГО ТРАФИКА	91
55. КОМПЛЕКС КОНТРОЛЯ РАБОТЫ СЕРДЕЧНО-СОСУДИСТОЙ СИСТЕМЫ ЧЕЛОВЕКА.....	92
56. ЗАЩИТА ГРАФИЧЕСКИХ ОБРАЗОВ С ПОМОЩЬЮ СИСТЕМ ВОДЯНЫХ ЗНАКОВ	93
57. ИСПОЛЬЗОВАНИЕ МЕТОДА ПЕРЕДАЧИ ОБУЧЕНИЯ ПРИ РАЗРАБОТКЕ КЛАССИФИКАТОРА ДЛЯ ОПРЕДЕЛЕНИЯ ПАТОЛОГИЙ НА ИЗОБРАЖЕНИЯХ ПОЛУЧЕННЫХ МЕТОДОМ ОПТИЧЕСКОЙ КОГЕРЕНТНОЙ ТОМОГРАФИИ ...	94
58. ПОСТРОЕНИЕ КАРТЫ ПОМЕЩЕНИЯ С ПОМОЩЬЮ SLAM.....	96
59. РЕАЛИЗАЦИЯ ВЕБ-СЕРВИСА ДЛЯ РАБОТЫ С КАМЕРОЙ ВИДЕОНАБЛЮДЕНИЯ	97
60. ИССЛЕДОВАНИЕ АВТОМАТИЧЕСКОЙ ВЕКТОРИЗАЦИИ НА ПРИМЕРЕ КОМПИЛЯТОРА MSVC 14.28	99
61. ПРОЦЕССОР ОБРАБОТКИ ИЗОБРАЖЕНИЙ НА ОСНОВЕ НЕРАЗДЕЛИМОГО ДВУМЕРНОГО ПОДХОДА	100
62. ПЛАТФОРМА ДЛЯ СЪЕМКИ СФЕРИЧЕСКИХ ПАНОРАМ С ИСПОЛЬЗОВАНИЕМ RASPBERRY PI	101
63. СИСТЕМА ДИСТАНЦИОННОГО ОБУЧЕНИЯ ТЕХНИЧЕСКИМ ДИСЦИПЛИНАМ	102
64. ПОДХОДЫ В МАШИННОМ ОБУЧЕНИИ ДЛЯ ЗАДАЧ КРЕДИТНОГО СКОРИНГА	104
СЕКЦИЯ «ФИЗИКА»	106
65. КВАНТОВАЯ ПСИХОЛОГИЯ	107
66. ПЕРСПЕКТИВА ИСПОЛЬЗОВАНИЯ КВАНТОВОЙ КРИПТОГРАФИИ КАК ОСНОВНОГО МЕТОДА ЗАЩИТЫ КОММУНИКАЦИИ	108
67. ГОЛОГРАММЫ В СОВРЕМЕННОМ МИРЕ.....	110
68. МАШИННОЕ ОБУЧЕНИЕ В ФИЗИКЕ.....	112

СЕКЦИЯ «ФИЛОСОФИЯ, ЛОГИКА И ЖИЗНЬ»	113
69. ВООБРАЖАЕМАЯ БИОЛОГИЯ: АНАТОМИЯ РУСАЛКИ.....	114
70. КРОВОТЫЕ НОРЫ: ЧТО ЖДЕТ ЧЕЛОВЕКА ПО ТУ СТОРОНУ ВСЕЛЕННОЙ.....	115
71. ЛОГИКА ВНУТРИ КВАНТОВОГО КОМПЬЮТЕРА	116
72. ВЛИЯНИЕ МУЛЬТФИЛЬМОВ НА ПСИХИКУ ДЕТЕЙ	117
73. СВОБОДА ВОЛИ: РЕАЛЬНОСТЬ, ВОЗМОЖНОСТЬ ИЛИ ИЛЛЮЗИЯ?	119
74. «ЭКСПЕРИМЕНТЫ ЛЕЙН» С ТОЧКИ ЗРЕНИЯ ФИЛОСОФИИ ЭКЗИСТЕНЦИАЛИЗМА.....	121
75. СОВРЕМЕННОЕ СЕКТАНТСТВО. МЕТОДЫ И ИНСТРУМЕНТЫ МАНИПУЛИРОВАНИЯ СОЗНАНИЕМ.....	123
76. СОФИСТИКА И ЛОГИЧЕСКАЯ АРГУМЕНТАЦИЯ.....	125
77. SOFT SKILLS – ГЛАВНОЕ ТРЕБОВАНИЕ СОВРЕМЕННОЙ ИТ-ИНДУСТРИИ	127
78. ВСЕГДА ЛИ МЫ МЫСЛИМ?	127
79. ФИЛОСОФСКОЕ ОСМЫСЛЕНИЕ ПРОБЛЕМ ПРОИСХОЖДЕНИЯ ВСЕЛЕННОЙ И ТЕОРИИ БОЛЬШОГО ВЗРЫВА	130
СЕКЦИЯ «ИНФОРМАТИКА»	132
80. АЛГОРИТМ КОЛОРИЗАЦИИ ИЗОБРАЖЕНИЙ НА ОСНОВЕ ГЕНЕРАТИВНО- СОСЯЗАТЕЛЬНЫХ НЕЙРОННЫХ СЕТЕЙ.....	133
81. СГЛАЖИВАЮЩИЕ СВОЙСТВА КЛЮЧЕЙ КОМБИНАЦИОННОГО УСТРОЙСТВА ВСЕВОЗМОЖНЫХ ПЕРЕСТАНОВОК	135
82. ОТСЛЕЖИВАНИЕ ОБЪЕКТОВ НА ВИДЕОЗАПИСИ ПРИ ПОМОЩИ МЕТОДОВ ГЛУБОКОГО ОБУЧЕНИЯ.....	137
83. АРИФМЕТИКА БЕЗ СКОБОК.....	139
84. ТРАНСФОРМАЦИИ ПРОСТРАНСТВА-ВРЕМЕНИ	141
85. ОНТОЛОГИИ И ЯЗЫКИ ДЛЯ ПРЕДСТАВЛЕНИЯ МАТЕМАТИЧЕСКИХ ЗНАНИЙ В SEMANTIC WEB.....	143
86. ЧЕГО МОЖНО ОЖИДАТЬ ОТ SEMANTIC WEB И ДРУГИХ СЕМАНТИЧЕСКИХ ТЕХНОЛОГИЙ В 2021 ГОДУ	144
87. СРАВНИТЕЛЬНЫЙ АНАЛИЗ ИНТЕРПОЛЯЦИОННЫХ МНОГОЧЛЕНОВ ЛАГРАНЖА И НЬЮТОНА.....	146
88. РОБАСТНЫЕ АЛГОРИТМЫ С РАЗРЕЖИВАНИЕМ ДЛЯ ВЕРОЯТНОСТНОГО ТЕМАТИЧЕСКОГО МОДЕЛИРОВАНИЯ	147
89. ГРАФОВЫЕ БАЗЫ ДАННЫХ И ИХ СВОЙСТВА.....	148
90. ОСОБЕННОСТИ РАБОТЫ С ПЛАТОЙ COSMOS+ OPENSDD ПРИ ПЕРЕНОСЕ НА СОВРЕМЕННЫЕ АППАРАТНЫЕ И ПРОГРАММНЫЕ ПЛАТФОРМЫ.....	150
91. КРИПТОГРАФИЧЕСКИЙ АЛГОРИТМ RSA И АЛГОРИТМЫ СОЗДАНИЯ ЦИФРОВОЙ ПОДПИСИ НА ЕГО ОСНОВЕ	152

**СЕКЦИЯ «ВСТРАИВАЕМЫЕ ВЫЧИСЛИТЕЛЬНЫЕ СИСТЕМЫ
РЕАЛЬНОГО ВРЕМЕНИ»**

ЦЕЛОЧИСЛЕННЫЙ СОПРОЦЕССОР ВЕЙВЛЕТ-ПРЕОБРАЗОВАНИЯ ДЛЯ ПЛИС

Бурак А.А.

Белорусский государственный университет информатики и радиоэлектроники
г. Минск, Республика Беларусь

Петровский Н.А. – канд. техн. наук

Рассматривается способ реализации целочисленного сопроцессора вейвлет-преобразования 5/3 для ПЛИС.

Лифтинг-схема, представленная Свелдсом [1] в 1995 году, предлагает эффективный способ построения вейвлетов. Свелдс и Добеши доказали два важных свойства лифтинг схемы:

- Лифтинг сохраняет биортогональность преобразования анализ-синтез;
- Любой вейвлет с конечной импульсной характеристикой можно представить в виде лифтинг схемы.

Лифтинг схема позволяет сократить используемые ресурсы вплоть до 2х раз. Лифтинг схема представляет собой последовательные шаги предсказания и обновления. Для представления пары вейвлетов в виде лифтинг схемы необходимо представить данные вейвлеты в виде полифазной матрицы $E(z)$:

$$E(z) = \begin{bmatrix} H_e(z) & H_o(z) \\ G_e(z) & G_o(z) \end{bmatrix} \quad (1)$$

где $H_e(z)$ и $H_o(z)$ – четные и нечетные полифазные компоненты НЧ фильтра анализа,

$G_e(z)$ и $G_o(z)$ – четные и нечетные полифазные компоненты ВЧ фильтра анализа.

После этого данную полифазную матрицу необходимо разложить на треугольные матрицы следующим образом:

$$E_0 = \begin{bmatrix} K & 0 \\ 0 & 1/K \end{bmatrix} \cdot \prod_{i=1}^m \begin{bmatrix} 1 & U_i(z) \\ 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} 1 & 0 \\ -P_i(z) & 1 \end{bmatrix} \quad (2)$$

где K – постоянный масштабирующий коэффициент;

m – количество шагов предсказания и обновления;

$P_i(z)$ – i -й шаг предсказания;

$U_i(z)$ – i -й шаг обновления.

Лифтинг схема продемонстрирована на рисунке 1.

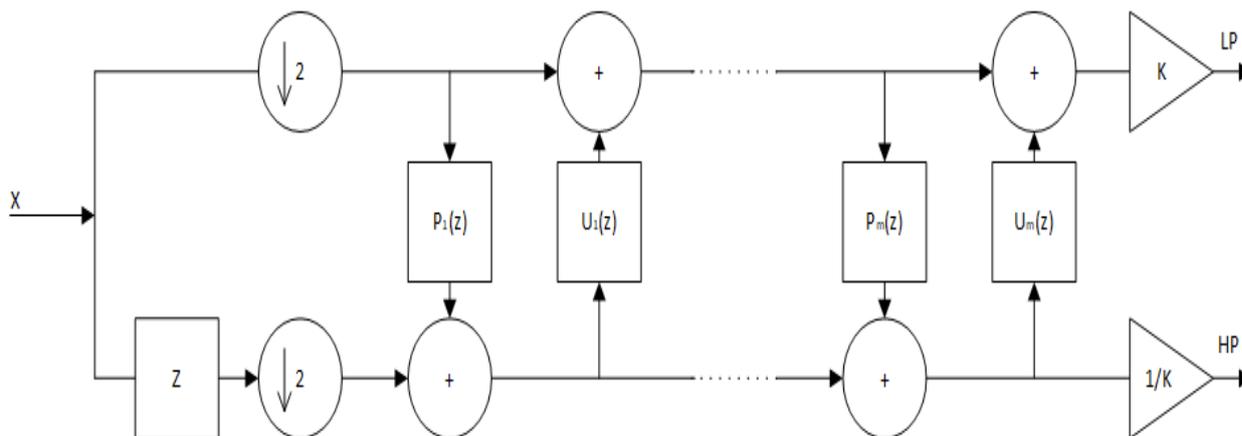


Рисунок 1. Лифтинг-схема

Используя полифазную матрицу анализа, можно найти соответствующие вейвлет-преобразование:

$$\begin{bmatrix} LP(z) \\ HP(z) \end{bmatrix} = E(z) \cdot \begin{bmatrix} X_e(z) \\ X_o(z) \end{bmatrix} \quad (3)$$

где $LP(z)$ – низкочастотные компоненты сигнала;

$HP(z)$ – высокочастотные компоненты сигнала;

$X_e(z)$ и $X_o(z)$ – четные и нечетные компоненты входного сигнала соответственно.

Для вейвлета 5/3 коэффициенты:

- НЧ-фильтра $h = [-1/8, 2/8, 6/8, 2/8, -1/8]$;
- ВЧ-фильтра $g = [-1/2, 1, -1/2]$.

Для данных фильтров получаем 1 шаг предсказания и 1 шаг обновления:

$$P_1 = \frac{1}{2}z^{-1} + \frac{1}{2}$$

$$U_1 = \frac{1}{4} + \frac{1}{4}$$

Получаем формулы для выходных коэффициентов:

$$HP[2n + 1] = X[2n + 1] - \left\lfloor \frac{X[2n] + X[2n + 2]}{2} \right\rfloor \quad (4)$$

$$LP[2n] = X[2n] + \left\lfloor \frac{HP[2n - 1] + HP[2n + 1]}{4} \right\rfloor \quad (5)$$

На основании формул (4), (5) был разработан целочисленный сопроцессор для вычисления одномерного дискретного вейвлет-преобразования, структура которого представлена на рисунке 1.

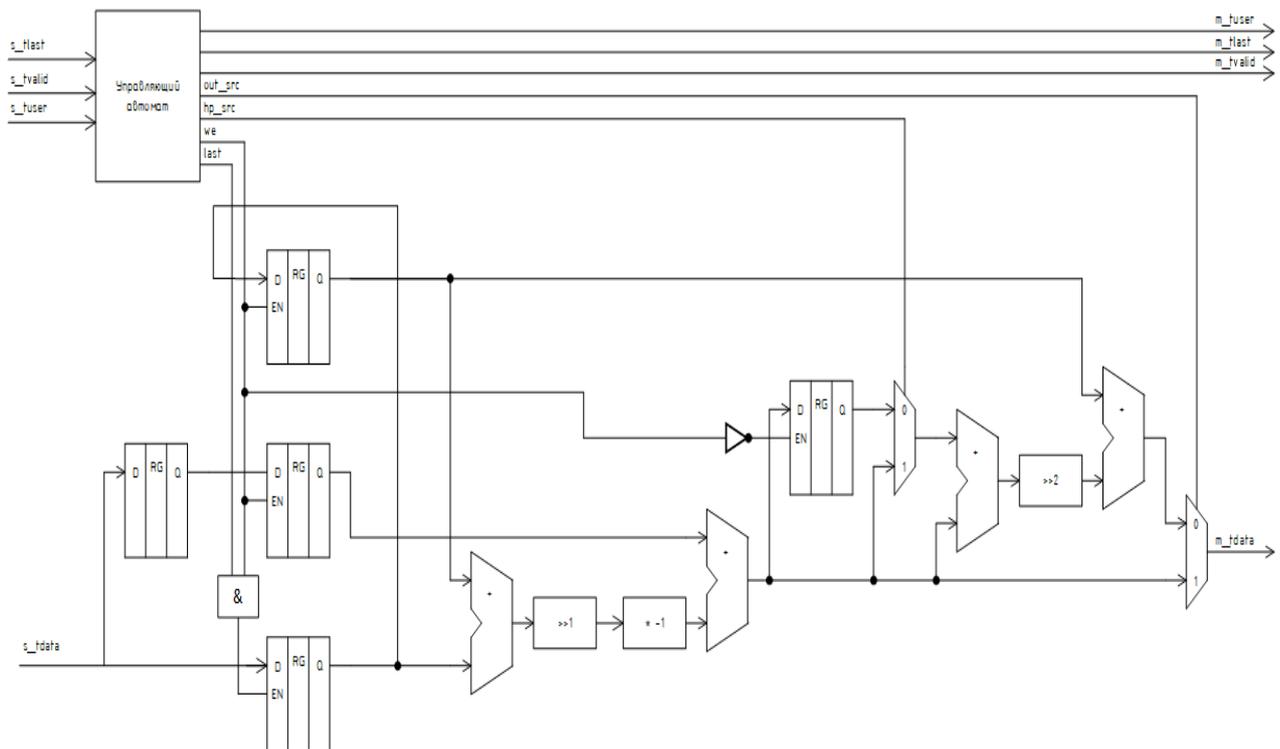


Рисунок 1. Структурная схема целочисленного сопроцессора

Оценка производительности и потребляемых ресурсов проводилась в САПР Vivado 2019.1 для отладочной платы Zybo (кристалл xc7z010clg400-1) со стандартными стратегиями для синтеза, размещения и трассировки проекта. Максимальная частота работы сопроцессора составляет 123 МГц. Потребляемые ресурсы представлены в таблице 1.

Таблица 1 – Потребляемые ресурсы ПЛИС

Ресурс	Затраты	Доступно	Затраты, %
LUT	51	17600	0.29
FF	71	35200	0.2

Список использованных источников:

1. *The Lifting Scheme: A New Philosophy in Biorthogonal Wavelet Constructions* / .W. Sweldens // *Wavelet Applications in Signal and Image Processing*, 1995 – P. 68–79.

РЕАЛИЗАЦИЯ ПОДМНОЖЕСТВА ИНСТРУКЦИЙ СОФТ ПРОЦЕССОРА АРХИТЕКТУРЫ RISC-V RV32I ДЛЯ ПЛИС ТИПА FPGA

Бурак А.А.

Белорусский государственный университет информатики и радиоэлектроники
г. Минск, Республика Беларусь

Станкевич А.В. – канд. техн. наук

Рассматривается реализация RISC-V процессора для ПЛИС типа FPGA. Процессор реализует подмножество команд RV32I за исключением инструкций FENCE, ECALL, EBREAK.

RISC-V — это открытая архитектура со стандартным набором команд (ISA), основанная на концепции RISC (reduced instruction set computer — компьютер с простым набором команд). В отличие от большинства других ISA, спецификация RISC-V ISA доступна для бесплатного использования, реализации в ПЛИС и непосредственно в кремнии.

В спецификации RISC-V ISA определено около 50 инструкций. Так же есть стандартные расширения (C, M, A, F и D). Обязательным для реализации является только подмножество команд I (Integer), в которое входят инструкции по работе с памятью (загрузка в регистр из памяти и сохранение из регистра в память), ветвления, а также инструкции арифметических и логических операций и несколько служебных. Все команды разделены на 6 типов:

- 1) R — операции АЛУ типа регистр – регистр;
- 2) I — операции АЛУ с непосредственным значением в команде;
- 3) S — операции загрузки/сохранения;
- 4) B — условная передача управления;
- 5) U — операции с расширенным непосредственным значением;
- 6) J — безусловная передача управления.

Форматы типов инструкций представлены в таблице 1.

Таблица 1 – Форматы инструкций

Тип	31	25	24	20	19	15	14	12	11	7	6	0
R	funct7		rs2		rs1		funct3		rd		opcode	
I	imm[11:0]				rs1		funct3		rd		opcode	
S	imm[11:5]		rs2		rs1		funct3		imm[4:0]		opcode	
B	imm[12 10:5]		rs2		rs1		funct3		imm[4:1 11]		opcode	
U	imm[31:12]								rd		opcode	
J	imm[20 10:1 11 19:2]								rd		opcode	

В рамках проекта RISC-V были разработаны: 5-ти ступенчатый конвейерный процессор, коммутатор шины данных, контроллер блочной памяти, блок ввода-вывода и универсальный асинхронный приемопередатчик. Структурная схема реконфигурируемой системы с софтверным процессором архитектуры RISC-V на базе системы на кристалле Xilinx Zynq приведена на рисунке 1.

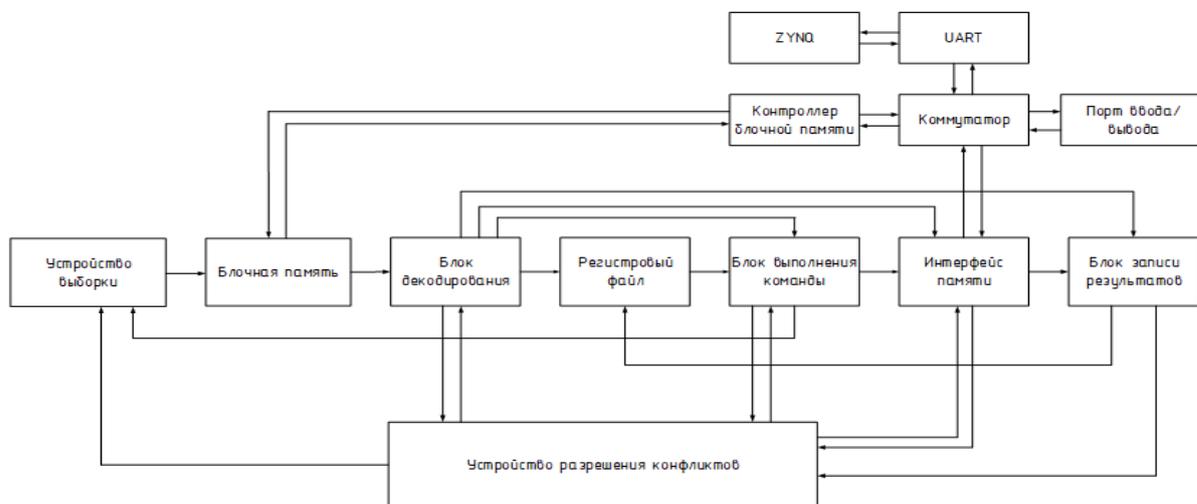


Рисунок 1 – Структурная схема проекта

Обмен данными между процессором и периферийными блоками осуществляется по шине данных. Временная диаграмма операций чтения и записи представлена на рисунке 2, где A0, A1, A2

являются адресами по которым производится чтение данных; D0, D1, D2 соответственно прочитанным данным, A3, A4 и D3, D4 являются соответственно адресами и данными для записи в память; W, HW, B размеры читаемых данных (W – 4 байта, HW – 2 байта, B – байт).

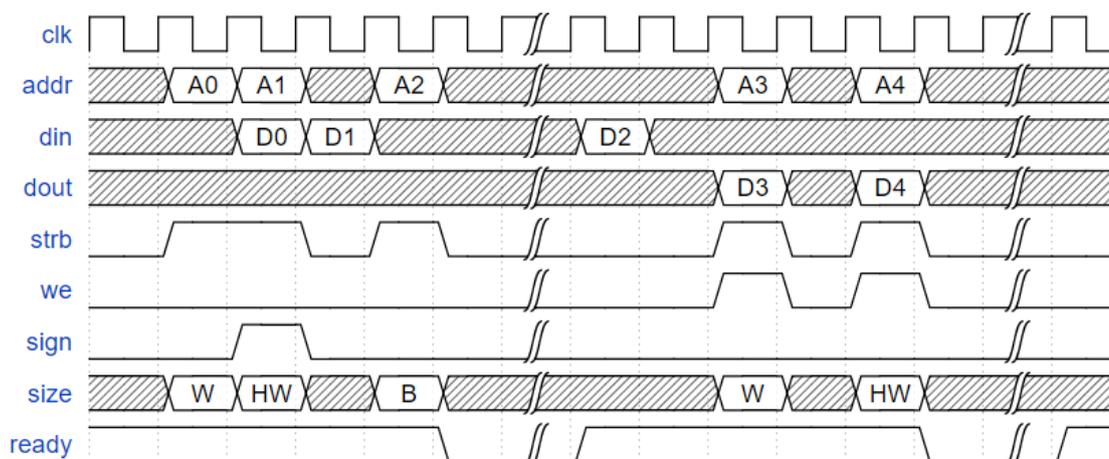


Рисунок 2 – Временная диаграмма операций чтения и записи по шине данных.

Для базовой верификации процессора применялась целенаправленное тестирование. К процессору подключались память команд, содержащая все поддерживаемые команды процессором, а также некоторые их комбинации, как приводящие, так и не приводящие к возникновению конфликтов в конвейере и подключалась внешняя память данных. В случае если после запуска тестовой последовательности содержание памяти данных совпадает с предполагаемым, то процессор можно считать условно рабочим, т.к. правильное выполнение программы маловероятно при наличии ошибок в аппаратуре.

Проект был подготовлен в САПР Vivado 2019.1 для отладочной платы Zybo с кристаллом серии Xilinx Zynq (xc7z010clg400-1). Используя доступные в САПР стратегии синтеза, размещения и трассировки проекта, такие как Flow_PerfOptimized_high и Performance_NetDelay_high, позволило выполнить ограничение на тактовую частоту работы процессора в 100 МГц. Затраченные ресурсы ПЛИС представлены в таблице 2.

Таблица 1 – Форматы инструкций

Ресурс	Затрачено	Доступно	Затрачено, %
LUT	1798	17600	10.22
LUTRAM	19	6000	0.32
FF	1759	35200	5.0
BRAM	8	60	13.3
IO	8	100	8.0
BUFG	1	32	3.13

Можно оценить производительность процессора на тестовом наборе SPECINT2000 в данном наборе содержится примерно 25% команд загрузки, 10% команд сохранения, 11% команд условного перехода, 2% команд безусловного перехода, и 52% команд типа R. Среднее количество необходимых тактов для выполнения 1-й инструкции (CPI) находится по формуле 1.

$$CPI = \sum_i C_i \cdot N_i \quad (1),$$

где C_i — количество тактов, необходимых для выполнения i -го типа инструкций;

N_i — относительное количество i -х инструкций в тестовом наборе.

Подставив значения в формулу 1 и вычислив, получаем $CPI = 2,1$. Производительность может быть оценена по формуле 2.

$$P = \frac{F_{cpu}}{CPI} \quad (2),$$

где F_{cpu} — тактовая частота процессора.

Подставив значения в формулу 2 получаем что производительность процессора равна примерно 47.6 миллионов операций в секунду.

Список использованных источников:

1. Waterman A. The RISC-V Instruction Set Manual Volume I: Unprivileged ISA/ Waterman A., Asanovi'c K.

ПРОЦЕССОР SHA-3 НА БАЗЕ FPGA

Ероховец В.А.

Белорусский государственный университет информатики и радиоэлектроники
г. Минск, Республика Беларусь

Станкевич А.В. – канд. техн. наук)

Современные цифровые системы используют алгоритмы хеширования для обеспечения конфиденциальности личных данных своих пользователей. Вместе с тем, бурный рост производительности вычислительной техники ставит под сомнение безопасность алгоритмов, разработанных 10-15 лет назад. В качестве примера можно рассмотреть криптоалгоритм SHA-2, который широко использовался ранее, а сейчас заменён на современный, более надёжный алгоритм SHA-3.

Алгоритм хеширования SHA-3 представляет собой новую концепцию формирования хеш-значения сообщения. SHA-3 – алгоритм хеширования переменной разрядности. В качестве официального криптоалгоритма для SHA-3 был выбран алгоритм Кессак. Он разработан в 2012 году группой авторов во главе с Йоаном Дайменом. 5 августа 2015 года алгоритм был принят в качестве стандарта FIPS 202. [1]

Алгоритм SHA-3 состоит из нескольких этапов: дополнения сообщения и функции “губки”. Так как блоки, которыми оперирует алгоритм, кратны некоторому числу r , в алгоритм включено дополнение сообщения. Однако, алгоритм, выполняющий дополнение, отличается от алгоритма дополнения SHA-2: к сообщению добавляется единичный бит, после него определенное количество (от 0 до $r-1$) нулевых битов, и в конце еще один единичный бит. $r-1$ нулевых битов может быть добавлено в случае, когда последний блок сообщения имеет длину $r-1$ бит. Тогда этот блок дополняется единичным битом, а следующий блок будет состоять из $r-1$ нулей и единицы. [1]

Основным блоком в структуре алгоритма является функция “губки”. Дополненное входное сообщение разбивается на блоки, которые затем последовательно складываются по модулю 2 со строкой состояния. Массив строк состояний формируется на каждом этапе работы алгоритма. После n итераций “впитывания”, выполняется обратный процесс “отжимания”, в ходе которого части промежуточных результатов извлекаются и составляют итоговое значение. Структура блока представлена на рисунке 1.

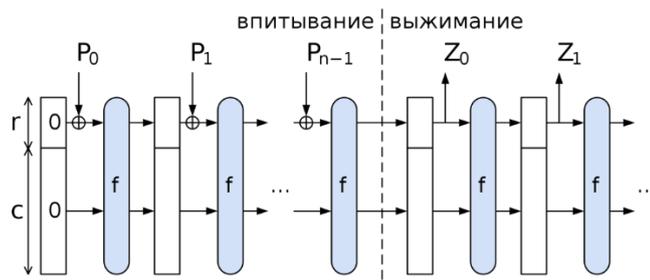


Рисунок 1 – Структурная схема блока, выполняющего функцию “губки”

При аппаратной реализации необходимо учитывать несколько факторов. Среди них произвольная длина сообщения и большой размер массива состояний S . Алгоритм не содержит блоков умножения и деления, что позволяет эффективно реализовывать алгоритм на FPGA. Этот фактор и стал определяющим в выборе данного алгоритма в качестве национального стандарта.

Алгоритм имеет множество реализаций на языках описания аппаратуры. В таблице 1 приведены результаты операции place-and-route на кристаллах семейства Virtex-6.

Таблица 1 – Результаты операции place-and-route некоторых аппаратных реализаций.

Реализация	Тактовая частота (МГц)	Slice	LUT
[2]	188,9	2 220	9 895
[3]	250	144	610
[4]	291,21	1 015	-
[5]	754,32	1 171	4 635

Тактовая частота одной из реализаций процессора SHA-2 составляет 218,9 МГц [6] и практически идентична тактовой частоте процессора SHA-3. Приведенные данные позволяют сделать вывод, что алгоритм хеширования SHA-3 при аппаратной реализации имеет производительность, сопоставимую с производительностью реализаций SHA-2, обеспечивая при этом более высокую криптостойкость.

Список использованных источников:

1. Wikipedia[Электронный ресурс]. – Электронные данные. – Режим доступа: <https://ru.wikipedia.org/wiki/SHA-3/>

2. OpenCores[Электронный ресурс]. – Электронные данные. – Режим доступа: <https://opencores.org/projects/sha3>
3. Compact FPGA Implementations of the Five SHA-3 Finalists. / S. Kerckhof [et al.] // Researchgate, January 2011. P.71-74
4. Novel Arithmetic Architecture for High Performance Implementation of SHA-3 Finalist Keccak on FPGA Platforms. / Kashif Latif [et al.] // Proceedings of the 8th international conference on Reconfigurable Computing: architectures, tools and applications, 2012. – P 496- 502.
5. Beyond the Limits: SHA-3 in just 49 Slice./ V. Arriba // 29th International Conference on Field Programmable Logic and Applications (FPL), 2019. – P. 130-135
6. Design of high-throughput SHA-256 hash function based on FPGA. / S. Suhaili // 6th International Conference on Electrical Engineering and Informatics (ICEEI), 2017. – P.247-251.

СИСТЕМА НАБЛЮДЕНИЯ ЗА СТАТИЧЕСКИМИ ОБЪЕКТАМИ В УСЛОВИЯХ ПЛОХОЙ ВИДИМОСТИ

Жук В. Г.

Белорусский государственный университет информатики и радиоэлектроники
г. Минск, Республика Беларусь

Шемаров А.И. – канд. техн. наук

При наблюдении за объектами требуется четкая картинка без помех, но в случае плохих погодных условия, таких как туман, дождь или снег, внимательное отслеживание объекта становится затруднительным. Для облегчения работы требуется построить систему, которая сможет облегчить данную задачу, выдав четкую картинку статического объекта наблюдения.

Туман, дымка, снег и дождь могут сильно повлиять на системы наблюдения за наземными или морскими объектами, так как при снятии изображения с обычной камеры, наблюдаемые объекты могут быть искажены, либо и вовсе потеряны. Для непосредственного постоянного наблюдения за объектом необходимо проводить пост обработку. Само построение системы необходимо чтобы результаты отслеживания статических объектов были наиболее достоверными. Существует несколько алгоритмов обработки изображений для уменьшения влияние погодных условий [1], но они не подходят напрямую для систем реального времени, где это важно для получения стабильного результата за ограниченное время. Система технического зрения, способная чтобы уменьшить влияние суровых погодных условий с задержкой всего в несколько миллисекунд, могла бы иметь очень большое значение.

Воздействие плохих погодных условий на инфракрасные изображения не так сильно, как на стандартные изображения (инфракрасный спектр охватывает длины волн от 700 нм до 1 мм). Закон рассеяния Рэлея гласит, что более длинные волны меньше рассеиваются частицами в воздух, что означает, что инфракрасный свет рассеивается меньше, чем видимый свет. Следовательно, инфракрасные изображения будут в меньшей степени искажены дымкой и туманом, чем обычные изображения, а при объединении изображений с двух камер: камеры выдающей обычное визуального (RGB) изображение и камеры ближнего инфракрасного (IR) диапазона, может быть достигнута полная система удаления дефектов из-за воздействия плохих погодных условий с сохранением цвета и четкости изображения. Данная система показывает наиболее надежные результаты при постобработке именно за счёт слияния двух типов изображений. Существуют различные методы слияния двух кадров, их следует разделять в зависимости от их принципа работы: слияние на основе характеристических особенностей; попиксельное слияние; слияние на основе объектов.

Поскольку в системах видеонаблюдения крайне важна скорость обработки данных, следовательно, требуется подробнее рассматривать метод, который наиболее быстр. Данным методом, можно считать попиксельное слияние с установкой весовых коэффициентов.

На рисунке 1 показана обобщённая схема алгоритма, в которой он разделен на два основных этапа – подготовка и обработка. В ходе подготовительного этапа формируются данные, необходимые используемые в ходе слияния и формирования карты хроматических компонентов цветовой модели YUV (где Y — яркостная составляющая, U и V — цветоразностные составляющие).

Для повышения визуального качества следует выполнять корректировку инфракрасного изображения. Следует использовать данные таблиц перекодировки и гистограмм, чтобы определения коэффициента коррекции, которые могут позволить сократить вычислительную сложность алгоритма в данном методе.

Наблюдение за динамическими и за статическим объектами имеет несколько разный подход. Так при наблюдении за статическим объектов производится сбор сведений, заключающийся в регистрации признаков и фактов, которые характеризуют каждую единицу исследуемого объекта. Данные о каждой единице наблюдения являются результатом статистического учета и представляют собой первичную информацию.

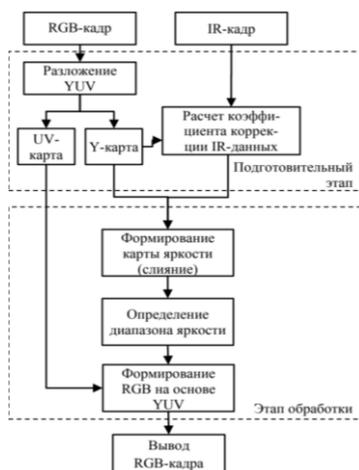


Рисунок 1 – Обобщенная схема алгоритма [2].

Обычные однопроцессорные компьютеры или компьютеры с несколькими процессорами не подходят для интенсивной вычислительной обработки изображений в критически важных для безопасности системах. Используя специализированные вычислительные платформы, такие как графические процессоры (GPU) или цифровые сигнальные процессоры (DSP), можно ускорить вычисления и получить результат за более короткое и предсказуемое время, чем на обычном компьютере.

Другим примером такой платформы является программируемая логическая интегральная схема (ПЛИС). Использование ПЛИС, также способно увеличить скорость обработки и изображения, а также показывает более высокую точность, чем портативный компьютер. Это позволяет проводить обработку и закладывать основу для системы удаления помех и артефактов в режиме реального времени.

Список использованных источников:

1. Karlsson. J. *FPGA-Accelerated Dehazing by Visible and Near-infrared Image Fusion: Independent thesis Advanced level: 11.06.2015* / J. Karlsson // Mälardalen University, 2015. – 48 P.
2. Улучшение визуального качества изображений, полученных в сложных условиях освещенности на основе инфракрасных данных/ А.Г. Зотин [и др] // Программные продукты и системы / Software & Systems, 2016 – С. 109 -119.

СВЁРТОЧНЫЕ НЕЙРОННЫЕ СЕТИ ДЛЯ ОБРАБОТКИ РЕЧИ

Прокопеня А.С.

Белорусский государственный университет информатики и радиоэлектроники
г. Минск, Республика Беларусь

Азаров И.С. – доцент, д. т. Н..

В данной работе рассматривается применение свёрточных нейронных сетей для задачи распознавания речи. Предложен метод распознавания изображения спектрограммы звукового сигнала с помощью свёрточных нейронных сетей. Реализованы алгоритмы для предварительной обработки входных данных. Оценено качество распознавания, проведено сравнение двух предложенных моделей распознавания речи посредством применения свёрточных нейронных сетей.

В работе рассмотрены свёрточные нейронные сети (СНС) для обработки речи, т. к. сети такого вида в настоящее время показывают один из лучших результатов в области распознавания изображений. Такая особенность СНС позволяет рассматривать не временную реализацию звукового сигнала (речи), а ее спектрограмму, которая распознается как изображение. Важной особенностью свёрточных нейронных сетей является устойчивость к изменениям масштаба и смещениям изображения, что, очевидно, имеет место в случае спектрограмм речевых сигналов [2].

Первую модель:

Чтобы реализуемая свёрточная нейронная сеть показывала необходимые результаты, необходимо сформировать несколько директорий, в которых будут храниться данные об аудио сигналах:

1. Директория тренировочных данных.
2. Директория проверочных данных.
3. Директория тестовых данных.

С целью минимизации ошибки, обучение требуется проводить на массиве данных от 1000 до 10000 [2]. В данной модели использовалась выборка данных 100 изображений спектрограмм для тренировки, 50 изображений для проверки. В качестве функций активации в разных слоях использовались «сигмоида» и ReLU. Разберем структуру сети:

- Сверточный слой, выделяющий 32 признака;
- Ядро 3×3;
- Функция активации ReLU;
- Слой подвыборки (MaxPooling) с ядром 2×2;
- Слой преобразования массива в вектор;
- Полносвязный слой из 32 нейронов, функция активации ReLU;
- Слой Dropout против переобучения;
- Полносвязный слой из 10 нейронов; активационная функция «сигмоида».

Результаты исследования модели

Для получения статистических данных были проведены эксперименты в которых создавалось по 5 нейронных сетей для 30,40,50 эпох с количеством сверточных слоев от 1 до 4. Представленная нейронная сеть показала точность около 80 – 98% в тестовой выборке.

Вторая модель:

Вторая модель была спроектирована для задачи распознавания эмоций по голосу. Применение таких нейросетей находит себя во множестве областей профессиональной деятельности человека: разработке вспомогательных роботов, автономных транспортных средств, оборудовании для нейро-обратной связи и т. д. [1].

Для начала необходимо провести предобработку звука. Для этого необходимо преобразовать звуковую волну в цифровой вид. Для этого выполняется процедура дискретизации звуковой волны. После выполнения операции дискретизации на выходе будет получен массив чисел, каждое из которых представляет амплитуду звуковой волны через интервалы 1/8000 секунды.

В качестве особенностей исходного сигнала используются MFCC (Mel-Frequency Cepstral Coefficients, мел-частотные кепстральные коэффициенты) [1]. Для извлечения MFCC необходимо разложить звуковую волну на отдельные составляющие. Далее необходимо вычислить логарифм мощности сигнала в каждом мелокне и произвести дискретное косинусное преобразование.

Построение нейронной сети и ее обучение

Свёрточная нейронная сеть имеет следующую архитектуру:

1. Первый уровень обработки представляет собой два свёрточных слоя и max pooling с 128 фильтрами размера 5, имея на выходе 216×128×2 нейронов.
2. Второй уровень обработки состоит из трех слоев свёртки также с 128 фильтрами. На выходе второго слоя модель имеет 216×128×3 нейронов.
3. Последний уровень обработки представляет собой полносвязный слой, который производит классификацию по 10 эмоциям. Структура нейронной сети представлена на рисунке 1.

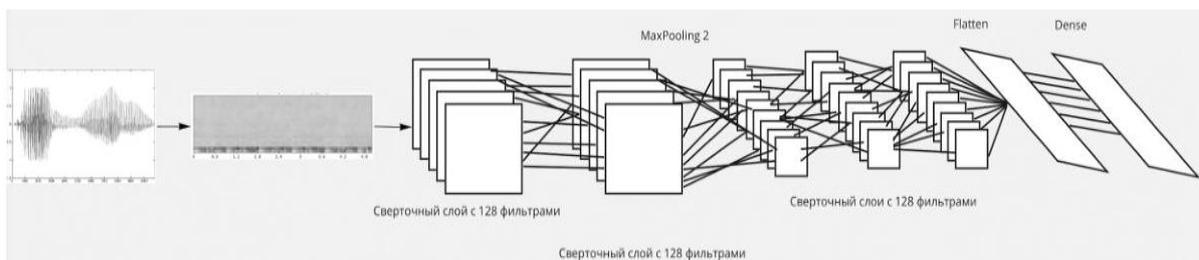


Рисунок 1 – структура свёрточной нейронной сети

Результаты:

В качестве входных данных использовался набор данных от RAVDESS [1], состоящий из аудиозаписей 24 актеров, которые содержат различные эмоции: счастье, злость, грусть, отвращение, спокойствие, удивление. Данный набор данных был разделен на тренировочные и тестовые данные в пропорции 80/20. В результате моделирования точность данной системы составила 73%.

В результате проведенного анализа, а также сравнения выбранных моделей свёрточных нейронных сетей для задачи распознавания речи получены следующие результаты: точность первой модели варьируется от 80%-98%, вторая же модель показывает результат с точностью в 73%. Из этого следует, что, подход к построению свёрточной нейронной сети предложенный в первой модели является более точным.

Список использованных источников:

1. Научная электронная библиотека [Электронный ресурс]. – Режим доступа: <https://www.elibrary.ru/item.asp?id=40872772>. – Дата доступа: 17.04.2021.
2. Белоруцкий Р.Ю., Житник С.В. Распознавание речи на основе свёрточных нейронных сетей // Вопросы радиоэлектроники. 2019. №4. С. 47-52.

**СЕКЦИЯ «ПРОГРАММНОЕ ОБЕСПЕЧЕНИЕ
ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ»**

ПРОГРАММНО-АППАРАТНЫЙ МОДУЛЬ АУДИОФИКСАЦИИ

Авсяник Е.С., Деменковец Д.В.

Белорусский государственный университет информатики и радиоэлектроники
г. Минск, Республика Беларусь

Леванцевич В.А. – старший преподаватель

В работе описано применение аудиодатчиков в охранных системах. Описывается работа модуля на основе алгоритма быстрого преобразования Фурье. Предлагаются варианты применения модуля аудиофиксации.

Системы охраны заводских и жилых помещений могут включать в себя большое количество датчиков: датчики движения, датчики разбития стекла, активные инфракрасные датчики движения и присутствия, фотоэлектрические датчики, ультразвуковые датчики, вибродатчики и так далее.

Видеорегистраторы и камеры обычно работают постоянно, записывая каждый момент времени. Это не очень выгодно с точки зрения энергопотребления и объемов хранимых записей. Альтернативное решение могло бы быть следующим: включать запись только после каких-либо изменений во внешней среде. Таким индикатором может выступать звук. Разбитое стекло, шаги, скрип дверей, звон ключей – все это вызывает достаточный шум. Устройство, реагирующее на изменение звука, может выступать как самостоятельная единица в охранной системе, так и в качестве вспомогательного модуля в более сложных.

Задачей проекта стала разработка аудиофиксатора – устройства, которое сохраняет данные в моменты, когда уровень шума поднимается выше определенного. Основной проблемой является анализ звукового сигнала в реальном времени. Для обработки и работы с сигналом требуется перевести его в дискретную цифровую форму. Полученный после оцифровки сигнал представляет собой массив значений уровня, измеренных через равные промежутки времени. Отсюда следует, что цифровой сигнал в явном виде не содержит никакой информации о его частотной составляющей. Для того, чтобы разложить сигнал на отдельные частотные составляющие необходимо представить его в виде суммы гармонических колебаний. Для разложения сигналов на синусоидальные колебания применяется преобразование Фурье.

Так как стоит задача анализировать сигнал в режиме реального времени, целесообразно будет использовать FFT (fast Fourier transform, БПФ) – алгоритм быстрого вычисления дискретного преобразования Фурье. Быстрое преобразование Фурье представляет собой определенный алгоритм вычисления, который позволяет уменьшить количество производимых действий относительно прямого (по формуле) вычисления ДПФ. В основе алгоритма заложено разбиение заданной последовательности отсчетов дискретного сигнала на несколько промежуточных последовательностей. Следует отметить, что алгоритм БПФ точнее стандартного ДПФ, т. к. при сокращении операций снижаются суммарные ошибки округления [1].

В настоящее время известны несколько алгоритмов быстрого преобразования Фурье, которые являются частными случаями единого алгоритма, базирующегося на задаче разбиения одного массива чисел на два с последующим рекурсивным вычислением каждого массива чисел по дискретному преобразованию Фурье и объединении результатов расчетов.

Алгоритм БПФ представляет собой рекурсивное разделение исходной последовательности отсчетов $s_0(N)$ на две последовательности $s_1(N/2)$ и $s_2(N/2)$, где s_1 содержит отсчеты s_0 , находящиеся на четных позициях, а s_2 – отсчеты, находящиеся на нечетных позициях [2]. Затем для полученных последовательностей вычисляется дискретное преобразование Фурье [3], и полученные последовательности $S_1(N/2)$ и $S_2(N/2)$ половинной длины объединяются в результирующую N -точечную последовательность по формулам:

$$S(k) = S_0(k) + W_N^k \cdot S_1(k), k = 0 \dots \frac{N}{2} - 1, \quad (1)$$

$$S\left(k + \frac{N}{2}\right) = S_0(k) - W_N^k \cdot S_1(k), k = 0 \dots \frac{N}{2} - 1, \quad (2)$$

где W_N^k вычисляется по формуле:

$$W_N^k = \exp\left(-j \cdot \frac{2\pi}{N} \cdot k\right), k = 0 \dots N - 1, \quad (3)$$

Преобразование может быть представлено графически, изображенном на рисунке 1.

Из-за специфической формы графа он получил название «бабочка». Таким образом, на нижнем уровне рекурсии получаются последовательности, состоящие из одного отсчета, которые уже являются результатом преобразования Фурье. Соответственно весь алгоритм вычисления ДПФ сводится рекурсивному разбиению исходной последовательности на её четную и нечетную части и последующему их объединению с помощью преобразования «бабочка».

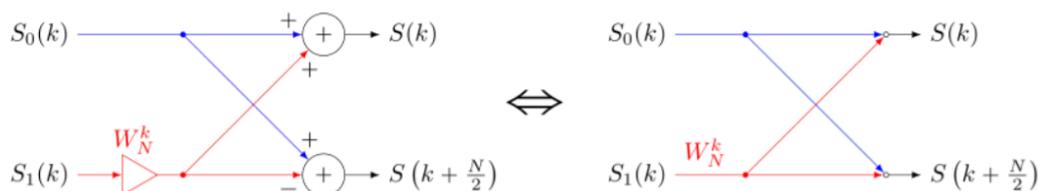


Рисунок 1 – Преобразование «бабочка»

Программно-аппаратный модуль состоит из аналогового датчика звука (микрофон), Wi-fi модуля и микроконтроллера из семейства STM32. Датчик звука передает данные по протоколу SPI. После получения аналогового сигнала происходит аналогово-цифровое преобразование сигнала и последующая отправка данных на фильтрацию внутри микроконтроллера. Данные из фильтра в зависимости от наличия подключения к интернету либо идут на запись SD-карты по протоколу SPI, либо на Wi-fi модуль по 23 протоколу USART. Схема подключения аппаратного модуля представлена на рисунке 2.

Дальше микроконтроллер через Wi-fi модуль по протоколу TCP отправляет данные в Azure IoT Hub. IoT Hub — это управляемая служба, размещенная в облаке, которая выступает в качестве центрального концентратора сообщений для двунаправленной связи между вашим приложением IoT и устройствами, которыми оно управляет.

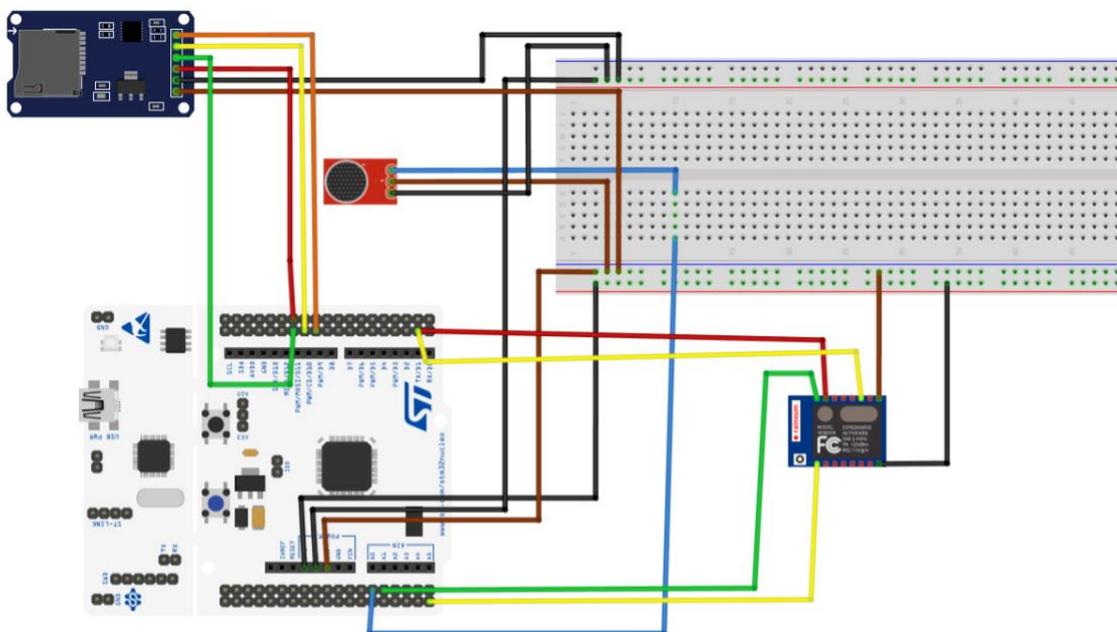


Рисунок 2 – Схема подключения аппаратного модуля

Полученный аудиофиксатор может записывать звуки, после появления определенного шума. Такое устройство может стать частью более сложной системы, например, активируя другие датчики при появлении посторонних звуков. Также аудиофиксатор можно использовать в качестве «аудиоловушки», что предоставляет возможность в последующем идентифицировать человека по голосу.

Список использованных источников:

1. The fast Fourier transform [Электронный ресурс]. – Электронные данные. – Режим доступа: http://www.autex.spb.su/download/dsp/dsp_guide/. – Дата доступа: 24.03.2021.
2. Сглаживание цифровых сигналов [Электронный ресурс]. – Электронные данные. – Режим доступа: <https://cyberleninka.ru/article/n/>. – Дата доступа: 18.03.2021.
3. Оконное сглаживание [Электронный ресурс]. – Электронные данные. – Режим доступа: <http://www.dsplib.ru/content/win/win.html>. – Дата доступа: 21.03.2021.
4. Кондыбаева А.Б. Методы обработки и сжатия аудиосигнала с помощью быстрого преобразования фурье [Электронный ресурс]. – Электронные данные. – Режим доступа: <https://sibac.info/studconf/tech/xxxii/42188>. – Дата доступа: 19.03.2021.

СИСТЕМА АВТОМАТИЧЕСКОГО РЕЗЕРВНОГО КОПИРОВАНИЯ И ОБРАБОТКИ ДАННЫХ НА ЯЗЫКАХ ПРОГРАММИРОВАНИЯ JAVA, JS

Аль-Саррих З.

Белорусский государственный университет информатики и радиоэлектроники
г. Минск, Республика Беларусь

Видничук В.Н. – м.т.н. ассистент кафедры ПОИТ

Облачные технологии фундаментально меняют ожидания в отношении того, как должны распределяться, управляться и потребляться ресурсы вычислений, хранения и сети в глобальном масштабе. Мощные вычислительные системы и системы хранения данных, высокий уровень безопасности, легкая доступность и настраиваемость, надежная масштабируемость и функциональная совместимость, экономичные и гибкие «облачные» вычисления являются самыми востребованными в современном быстрорастущем деловом мире. Клиент, организация или предприятие, внедряющее новую облачную среду, может выбрать подходящую инфраструктуру, платформу, программное обеспечение и сетевые ресурсы для каждого бизнеса, каждый из которых обладает эксклюзивными возможностями.

Целью проекта является повысить эффективность автоматического резервного копирования и обработки данных путём разработки онлайн-системы облачного хранения на языках программирования JavaScript и Java.

Объектом исследования являются процессы резервного копирования и обработки данных, реализуемые в системе облачного хранения данных.

Как правило, компьютеры и программы, составляющие информационную систему, отличаются друг от друга. Некоторые из них имеют собственные ресурсы (файловая система, процессор, принтер, база данных и т.д.), другие могут получить доступ к этим ресурсам. Компьютер (или программа), которая управляет ресурсом, называется сервером ресурса (файловый сервер, сервер базы данных, вычислительный сервер). Клиент и сервер ресурса могут находиться на одном компьютере или на разных компьютерах, подключенных к сети.

Общие принципы взаимодействия в сети с серверами, узлами, предоставляющими определенные функции (сервисы), и клиентами, потребляющими эти функции определяет клиент-серверная архитектура.

Клиент-серверные технологии определяют свои собственные или используют существующие правила взаимодействия между клиентом и сервером, называемые протоколом обмена (протоколом связи).

Ниже описаны преимущества клиент-серверных систем.

Клиент-серверный подход является модульным, при этом компоненты программ сервера компактны и автономны.

Неисправность сервера не влияет на другие компоненты операционной системы.

Автономность компонентов позволяет запускать их на нескольких процессорах на одном компьютере (симметричная обработка) или на нескольких компьютерах в сети (распределенные вычисления).

Задача клиента, как правило, заключается в предоставлении пользовательских услуг и, прежде всего, пользовательского интерфейса, т.е. средств для получения, отображения и обработки введенных пользователем данных, которые служат основой для запроса к серверу.

Любая сеть (включая одноранговую сеть), основанная на современных сетевых технологиях, обычно имеет основанные на двухзвенной архитектуре элементы взаимодействия клиент-сервер. Данный тип сети называется двухзвенной, потому что три основных компонента должны быть распределены между двумя узлами (клиент и сервер).

Двухзвенная архитектура используется в клиент-серверных системах, где сервер напрямую и полностью отвечает на запросы клиентов, используя только собственные ресурсы. То есть сервер не вызывает сторонние сетевые приложения и не обращается к сторонним ресурсам для выполнения какой-либо части запроса.

Минус двухзвенной архитектуры заключается в том, что если упал сервер или отвалилась база данных, то есть испортилось 1 звено – система не способна функционировать и клиенты не способны работать с системой.

Поэтому для предотвращения появления в работе системы аналогичных проблем в архитектуре приложения реализовывают кластер с балансировкой нагрузки, отвечающий за распределение запросов пользователей между серверами. Для распределения запросов в кластере используется один или несколько входных вычислительных узлов, через которые задачи перенаправляются с одной машины на другую.

Благодаря клиент-серверной организации, разрабатываемая нами программа позволит удаленно работать с данными, что очень важно в современных условиях, когда у специалистов есть возможность работать на легко переносимых устройствах. На сервере предусмотрена возможность параллельной обработки запросов. Система также должна обеспечивать возможность просмотра,

редактирования, удаления данных, создания новых записей. Соответствующие изменения должны быть внесены в базу данных с помощью MySQL-запросов на серверной части приложения.

Для разработки программного средства нам необходимо обратить внимание на архитектуру проектируемого программного обеспечения, которую мы проиллюстрируем с помощью диаграммы развертывания. Данная диаграмма используется для представления общей конфигурации и топологии распределенной программной системы и показывает распределение компонентов по отдельным узлам системы. Он также показывает наличие физических путей передачи информации между аппаратными устройствами, участвующими в реализации системы. Схема развертывания используется для визуализации элементов и компонентов программы, которые существуют только во время ее выполнения (выполнения). Он представляет только те компоненты, которые являются исполняемыми файлами или динамическими библиотеками. Компоненты, которые не используются во время выполнения, не отображаются на диаграмме развертывания.

При разработке архитектуры системы использовались услуги Amazon Web Services (AWS) – коммерческого общедоступного облака, поддерживаемого и разрабатываемого компанией Amazon с 2006 года. AWS предоставляет абонентам как услуги по моделированию инфраструктуры (виртуальные серверы, ресурсы хранения), так и услуги на уровне платформы (облачные базы данных, программное обеспечение для подключения к облаку, бессерверные облачные вычисления, средства разработки).

Облако развернуто в нескольких территориально распределенных центрах обработки данных, сгруппированных в группы близости, называемые «регионами». В пределах одного региона реализовано несколько «зон доступности», в пределах которых обеспечивается высокая доступность хостинг-услуг.

По состоянию на 2019 г. в 20 регионах насчитывается 60 зон доступности. Абоненты могут выбрать регион и зону доступности, а также имеют возможность организовать репликацию данных и передачу приложений между зонами доступности. В целях доступности разрабатываемого ПС как можно более большой группе лиц было принято решение арендовать сервера в 4 регионах: США, Западной Европе и 2 в Азиатско-Тихоокеанском регионе.

На рисунке 1 предоставлена диаграмма развертывания, показывающая узлы и связи между компонентами архитектуры.

Ключевая инфраструктурная услуга AWS – служба аренды виртуальных серверов EC2. Для реализации программно-аппаратной части сервиса используется два сервера: «DataBase Server» и «Backend Server». На виртуальные машины данных серверов была поставлена ОС CentOS Linux 7 x86_64. Для реализации пользовательской части ПС использованы 2 идентичных по наполнению сервера «Frontend USA Server» и «Frontend Russia Server». Данное решение используется для балансировки нагрузки между сетевыми устройствами (серверами) с целью сокращения времени обслуживания запросов пользователей. За балансировку нагрузки отвечает средство «Balance Loader».

Запросы от устройств пользователя, гостя или менеджера поступают в первую очередь на «Balance Loader» по протоколу https (протокол безопасной передачи данных, поддерживает технологию шифрования TLS/SSL). Балансировщик нагрузки направляет пользователя на один из двух серверов «Frontend USA Server» или «Frontend Russia Server» в зависимости от адреса, с которого поступил запрос. Это позволит улучшить работу пользователя с пользовательской частью системы.

Балансировщик нагрузки работает с серверами, отвечающими за отображение пользовательской части ПС, и роутером по протоколу TCP (один из основных протоколов передачи данных интернета). Соответственно таблице маршрутизации роутера, которая описывает соответствие между адресами назначения и интерфейсами, пакеты данных поступают на «Backend Server».

«Backend Server» работает с «DataBase Server» по PostgreSQL client-server протоколу и с помощью стандарта JDBC, который позволяет приложению взаимодействовать с СУБД. Стандарт JDBC основан на «концепции так называемых драйверов, позволяющих получать соединение с базой данных по специально описанному URL. Драйверы могут загружаться динамически (во время работы программы). Загрузившись, драйвер сам регистрирует себя и вызывается автоматически, когда программа требует URL, содержащий протокол, за который драйвер отвечает» [1].

Частью архитектуры является NAT, механизм в сетях TCP/IP, позволяющий преобразовывать IP-адреса транзитных пакетов. Этот механизм предотвращает внешний доступ к внутренним хостам, одновременно разрешая внутренний доступ к внешним. Трансляция создается при инициации соединения изнутри сети. Ответные пакеты, поступающие извне, соответствуют созданной трансляции и поэтому разрешены к прохождению. Если для пакетов, поступающих извне, соответствующей трансляции не существует (а она может быть созданной при инициации соединения или статической), они не пропускаются.

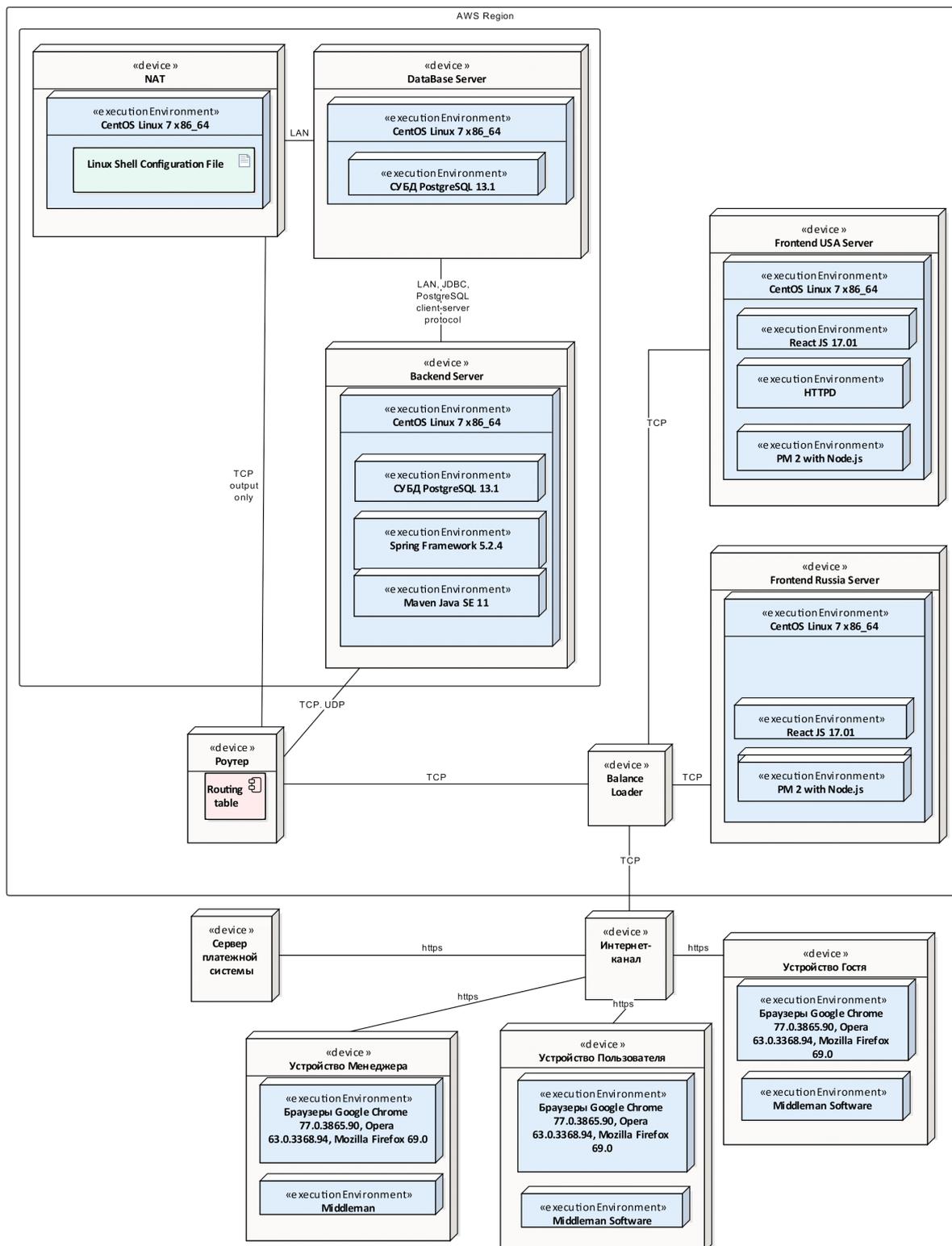


Рисунок 1 – Диаграмма развертывания

В итоге использование в системе двухзвенной клиент-серверной архитектуры с кластером с балансировкой нагрузки позволяет равномерно распределять запросы клиентов между доступными серверами и обеспечивает стабильную работу системы резервного копирования и обработки данных.

Список использованных источников:

1. Java Database Connectivity [Электронный ресурс]. – Режим доступа: https://ru.wikipedia.org/wiki/Java_Database_Connectivity. – Дата доступа: 03.04.2021.

СОВРЕМЕННЫЕ ПОДХОДЫ В ОНЛАЙН-ОБРАЗОВАНИИ

Арзуманян А.С.

Белорусский государственный университет информатики и радиоэлектроники
г. Минск, Республика Беларусь

Парамонов А.И. – канд. техн. наук

Рассматриваются особенности современных технологических решений в онлайн-образовании, которые направлены на повышение эффективности всего процесса обучения. При выполнении обзора делается акцент на определение влияния с точки зрения усвоения материала.

Современное образование стремительно меняется из года в год. Учреждения образования всё чаще стали применять в практике технологические решения для организации всех процессов в образовании. Несомненно, огромное влияние на сферу EdTech оказал Covid-19, тем самым ещё чётче обозначив проблемы на рынке и возможности для его развития.

В настоящее время большое развитие получили такие системы, как LMS- Learning Management System. Это полнофункциональное программное обеспечение для учреждений образования и не только, для полноценного регулирования процесса обучения [1]. Такие системы делают процесс обучения гибче с меньшим количеством затрат. Как одно из главных преимуществ сервисов управления обучением является возможность определять объективные критерии оценки усвоения знаний. В электронном обучении имеется возможность выставлять четкие критерии, по которым оцениваются знания, полученные студентом в процессе обучения.

Образование, а в частности онлайн-образование, становится сейчас по-настоящему массовой услугой. Количество обучающихся онлайн растёт из года в год. Это значит, что есть прекрасная возможность для сбора статистически достоверных данных. Благодаря этому получится максимизировать эффективность методики обучения. И как результат - повысится качество знаний и усвоение материала.

Анализ данных необходим в первую очередь для того, чтобы сделать образование стандартизированной услугой. Образование в будущем станет своеобразным чёрным ящиком, где на входе будет человек без необходимых знаний, а после программы - практически гарантированно с нужной компетенцией. Данный принцип тесно связан с персонализированным обучением, так как имея те или иные показатели об обучающемся и преподавателе:

- Преподаватель может чётко определить, какое необходимо принять решение, чтобы воздействовать на тот или иной показатель учащегося.
- Преподаватель может получить обратную связь в виде данных от учащихся и обновить свой материал в соответствии с показателями.

Современные технологические решения позволяют учитывать и контролировать усвоение материала гораздо эффективнее [2]. Намного проще реализовывать новые подходы в обучении. Например, такие, как геймификация, либо Project/Problem-based learning. Геймификация – это использование программного обеспечения, типичного для игр, в неигровых процессах, с целью с целью привлечения пользователей и потребителей, повышения их вовлечённости в решение прикладных задач, использование продуктов, услуг. Примером сервиса, реализующего принцип геймификации является Kahoot. Он представляет собой онлайн-викторину со множеством выборов. По её результатам собирается статистика разного плана: кто быстрее и правильнее всех отвечал на вопросы, самые трудные вопросы, самые лёгкие вопросы и т.д.

Следующий принцип сложнее – Project/Problem-based learning и как он реализован в современном программном обеспечении для образования. Это принцип, согласно которому тот, кто обучается - получает знания на основе конкретной задачи, которую необходимо решить, что максимально приближает любое учебное заведение к реальной жизни. В современных сервисах делается большой упор на работу в группе и на основе этого они позволяют собирать данные о том, как велась работа на самом проекте, степень ответственности конкретного студента и т.д.

Исходя из выше сказанного можно сделать вывод, что современные технологические решения позволяют более эффективно контролировать усвоение учебных материалов. Несомненно, реализация этих принципов приводит к тому, что повышается общая компетенция и как следствие, образование выходит на новый уровень, становясь стандартизированной услугой.

Список использованных источников:

1. Парамонов, А. И. Проблемы дистанционного образования и их прикладные решения в образовательных технологиях // *Материалы X Международной научно-методической конференции «Высшее техническое образование: проблемы и пути развития»*, (26 ноября 2020 г.) / Мин. образования РБ, БГУИР. – Минск : БГУИР, 2020. – С. 182–187.
2. Парамонов, А. И. Использование сервиса Google Class для организации самостоятельной работы студентов / Парамонов А. И., Олеферович А. В. // *Информационные технологии : тезисы 84-й научно-технической конференции, посвященной 90-летию юбилею БГТУ и Дню белорусской науки (с международным участием)*, (03–14 февраля 2020 г.) – Минск : БГТУ, 2020. – С. 96–98.

АНАЛИЗ ПОВЕДЕНИЯ ЧЕЛОВЕКА ПО МИМИКЕ ЛИЦА

Батал А.

Белорусский государственный университет информатики и радиоэлектроники
г. Минск, Республика Беларусь

Самаль Д.И – Доцент кафедры ЭВМ

«Подозревая, что кто-нибудь лжет, притворимся, будто мы верим ему; тогда он становится наглым, лжет еще больше, и маска спадёт» А. Шопенгауэр

На сегодняшний день область изучения микро и макровыражений лица важна для установления нестандартного поведения субъекта в ходе коммуникаций, остается не до конца изученной из-за огромного количества выражений лица человека. Человек учится определять эмоции неосознанно — с самого детства он начинает ассоциировать определенные паттерны поведения с эмоциями, которые испытывают люди вокруг. Научившись этому, человек может проанализировать, по каким признакам это делает. Самое очевидное — по тому, какое выражение принимают рот и глаза. Но на лице множество мимических мышц, которые создают невероятное количество выразительных нюансов. Целью данной работы – получение эффективной модели анализа поведения по конкретным правилам.

Решением данной проблемы занимаются учёные из университетов Англии, США, Нидерланды, а также России. Один из вариантов решения – собирается определенное количество людей для достижения необходимого количества лиц в целях сбора информации о главных эмоциях человека: печаль, радость, страх, удивление, гнев и отвращение. Делается это для сбора выборки данных в целях обучения нейронной сети классификации эмоций на основе ключевых точек лица. На вход нейронной сети поступает обработанное изображение для классификации по правилам Facial Action Coding System (FACS), Система кодирования лицевых движений – представляет собой систему для классификации выражений лица человека разработанная Полом Экманом – Американский психолог, профессор Калифорнийского университета, открывший понятие «Микровыражение лица».

В данной работе предполагается использовать вариант решения проблемы с использованием многозадачной каскадной нейронной сети (MTCNN) для обнаружение лица и извлечение ключевых точек лица, а также дальнейшей классификации по правилам FACS, где на вход нейронной сети подается обработанное изображение, а на выходе отображается область лица и выведены основные точки лица: два зрачка глаз, нос и два уголка рта, а также выдается вероятность обнаружения каждой из семи основных эмоций. Для проверки исследований психолога Пола Экмана будет проведен эксперимент, который предполагает сбор определенного количества людей, которым будет задано одинаковое количество вопросов, реакция подопытных будет зафиксирована на высокочастотную камеру для фиксации малейших изменений в мышцах лица. Дальше происходит изучение записей вручную для сопоставления полученных результатов с результатами исследований психологов.

Таким образом разработана модель для выявления нестандартного поведения человека, анализируя мимику его лица в режиме реального времени, что расширяет спектр использования данного продукта в различных сферах.

Дальнейшее развитие предполагает доработку существующей модели анализа для более эффективного и менее ресурсоемкого анализа в режиме реального времени, что существенно сократит необходимые ресурсы на поддержку продукта.

Список использованных источников:

1. Пол Экман. Психология Лжи/ Экман П. – Питер, 2020. – 377с.
2. Пол Экман Психология эмоций/ Экман П. – Питер, 2020. – 333с.
3. MTCNN [электронный ресурс] – <https://towardsdatascience.com/face-detection-using-mtcnn-a-guide-for-face-extraction-with-a-focus-on-speed-c6d59f82d49>.

МОБИЛЬНОЕ ПРИЛОЖЕНИЕ ДЛЯ ВЕДЕНИЯ ДНЕВНИКА ПУТЕШЕСТВИЙ

Беляцкая А.А.

Белорусский государственный университет информатики и радиоэлектроники
г. Минск, Республика Беларусь

Данилова Г.В. – ассистент кафедры ПОИТ

В современном мире технологии выходят на новый уровень, позволяя быстро и просто осуществить практически любую операцию, не выходя из дома. С каждым годом все больше услуг предоставляется через мобильные приложения. В данной работе рассматривается разработка программного средства для ведения дневника путешествий и планирования поездок.

Всегда присутствовала необходимость вести дневник путешествий и наблюдений, однако, далеко не вся информация сохранилась до наших дней. С развитием технологий видеоизменялись и способы сохранения информации: если сразу это можно было сделать только вручную, оставив заметку в блокноте, то позже появилась печать и фотография, а в настоящее время мир захватили цифровые технологии и Интернет. Теперь можно гораздо проще и надежнее находить и сохранять любую информацию.

С каждым годом становится популярнее и доступнее возможность самостоятельной организации поездки, что требует больше сил и времени, нежели обращение в турагентство, но также позволяет получить максимум впечатлений. Все, что может понадобится на этапе планирования путешествия и на всем его протяжении, удобно умещается в смартфоне, который всегда под рукой.

Программное средство базируется на концепции клиент-серверного взаимодействия, где клиентская часть представлена в виде мобильного приложения для платформы Android [1].

Клиентское приложение представляет собой интерфейс для взаимодействия с пользователем, позволяющий сохранять данные о прошедших и планируемых поездках, отправляя всю введенную информацию на сервер. Помимо вводимой пользователем информации программное средство имеет функцию отслеживания местоположения девайса в реальном времени, с целью сохранения более детальной информации о передвижении пользователя. Собранные данные передаются на сервер при подключении к сети Интернет, а также используются для отображения маршрута на карте. Внешний вид клиентского приложения представлен на рисунке 1.

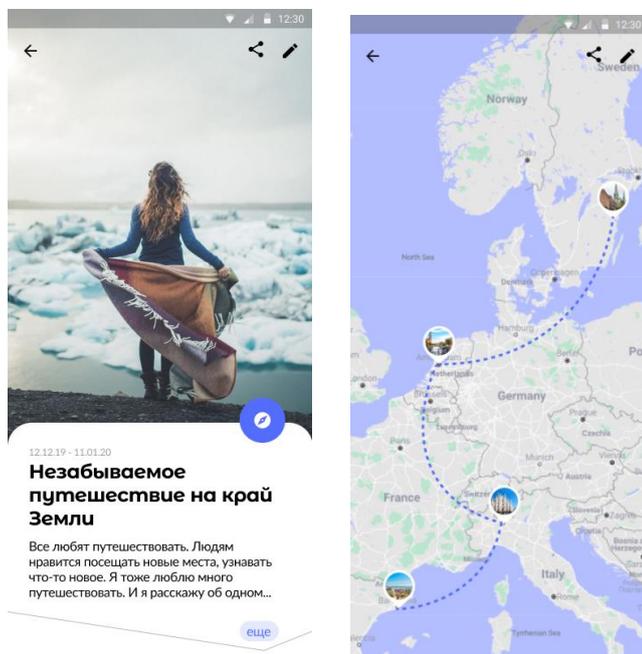


Рисунок 1 – Внешний вид клиентского приложения

Основными преимуществами разработанной системы являются простота в использовании, что достигается благодаря интуитивно-понятному пользовательскому интерфейсу, удобство и доступность, так как приложением можно воспользоваться из любой точки планеты, имея лишь смартфон.

Список использованных источников:

[1] Марсиано К., Стюарт К., Филлипс Б. *Android. Программирование для профессионалов*. Питер, 2016.

ПРОГРАММНОЕ СРЕДСТВО ОБРАБОТКИ ЗВУКОВЫХ СИГНАЛОВ

Биткин Н.С.

Белорусский государственный университет информатики и радиоэлектроники
г. Минск, Республика Беларусь

Данилова Г.В. – ассистент каф. ПОИТ

Современный рынок программных и аппаратных средств обработки музыки огромен: различные платформы и операционные системы, приложения со множеством различных расширений и дополнений. Поэтому начинающему музыканту необходим простой и понятный пример для ознакомления с огромным числом возможностей по обработке звука.

Звук всегда присутствовал в жизни человека – от первых песен до сложных проектов, основанных на специальной обработке. На сегодняшний день существует огромное многообразие форматов, способов воспроизведения, наложения различных эффектов на звук.

Целью данного проекта является создание средства обработки медиа-файлов, которое будет поддерживать работу с файлами различных форматов, в том числе узкоспециализированными (например, использующиеся игровой индустрией) и предоставлять возможность обработки различными способами (частотные фильтры, наложение эффектов).

Фильтр изменяет сигнал, «убирая» в нем выбранные частоты. Существующие фильтры не идеальны. Полоса пропускания – полоса частот, которую фильтр не затрагивает. Полоса подавления – полоса нежелательных частот. В полосе перехода происходит спад частот. Фильтр ближе к идеальному тем, насколько меньше он искажает полосу пропускания, насколько сильно он подавляет частоты в полосе подавления и насколько узка полоса перехода [1].

В качестве примера фильтра был выбран фильтр Баттерворта первого порядка.

АЧХ (Амплитудно-частотная характеристика) фильтра Баттерворта максимально гладкая на частотах полосы пропускания и снижается практически до нуля на частотах полосы подавления [1]. При отображении частотного отклика фильтра Баттерворта на логарифмической АФЧХ, амплитуда снижается к минус бесконечности на частотах полосы подавления. В случае фильтра первого порядка АЧХ затухает с крутизной -6 децибел на октаву (-20 децибел на декаду) (на самом деле, все фильтры первого порядка независимо от типа идентичны и имеют одинаковый частотный отклик). Для фильтра Баттерворта второго порядка АЧХ затухает на -12 дБ на октаву, для фильтра третьего порядка – на -18 дБ и так далее. АЧХ фильтра Баттерворта – монотонно убывающая функция частоты.

В математическом приближении алгоритм работы фильтра сводится к свертке, которая делается последовательно для каждого семпла:

$$y(n) = b_0 * x(n) + b_1 * x(n - 1) + b_2 * x(n - 2) - a_1 * y(n - 1) - a_2 * y(n - 2) \quad (1),$$

$y(n)$ – это новое значение семпла, которое нужно рассчитать. $x(n)$ – текущее значение семпла, соответственно $y(n-1)$ и $y(n-2)$ – предыдущие 2 рассчитанных семпла, а $x(n-1)$ и $x(n-2)$ – предыдущие входные значения семплов, а и b – некоторые коэффициенты.

Одним из самых простых эффектов является эхо, основанное на задержке. Эхо – это многократное повторение сигнала с экспоненциальным затуханием. То есть, текущее значение сигнала складывается как текущее новое значение плюс значение сигнала t времени назад, t – время задержки. Далее приведен общий вид формулы расчета:

$$y(n) = x(n) - y(n - T) \quad (2),$$

где x – входная последовательность семплов, y – результирующая, T – задержка в семплах.

Для регулирования уровня затухания, в формулу необходимо добавить долевые коэффициенты обработанного (P) и необработанного сигнала (U). Полученная формула представлена ниже:

$$y(n) = P * x(n) - U * y(n - T) \quad (3).$$

Далее необходимо добавить коэффициент снижения громкости по времени (F):

$$y(n - T) = x(n) - F * y(n - T - 1) \quad (4).$$

Программная реализация будет представлять собой простейший циклический буфер.

Список использованных источников:

[1] Айфичер Э., Джервис Б. Цифровая обработка сигналов. Практический подход. М: Вильямс, 2016.

ПРЕДСТАВЛЕНИЕ СИГНАЛОВ В МОДОВОЙ ДЕКОМПОЗИЦИИ

Бондарева Т.О., Деменковец Д.В., Леванцевич В.А.

Белорусский государственный университет информатики и радиоэлектроники
г. Минск, Республика Беларусь

Бранцевич П.Ю. – канд. тех. наук

Описан метод представления сигналов в виде их модовой декомпозиции. В качестве основных информативных признаков метода выбираются амплитуда, мгновенная частота и фаза отдельных мод. Разработан алгоритм и реализовано программное средство модовой декомпозиции. Показана применимость метода для анализа нестационарных вибрационных сигналов, полученных с реальных технических объектов.

Одним из важнейших показателей оценки технического состояния машин и агрегатов являются параметры вибрации [1]. При этом для исследования структуры вибросигнала, как правило, используются методы, основанные на преобразовании Фурье. Необходимым условием для применения Фурье-анализа является требование стационарности исследуемых сигналов, что на практике не всегда реализуемо, так как большинство вибросигналов, получаемых с реальных объектов, являются нестационарными [2].

В настоящее время широкое распространение получил метод исследования, нестационарных сигналов, основанный на эмпирической модовой декомпозиции, в результате которой исходный сигнал разлагается на ряд компонент, которые называются эмпирическими модами [3].

Эмпирическая мода - это такая функция, в которой количество экстремумов функции (максимумов и минимумов) и количество пересечений нуля не должны отличаться более чем на единицу.

Схема-алгоритма представления сигналов в модовой декомпозиции приведена на рисунке 1.

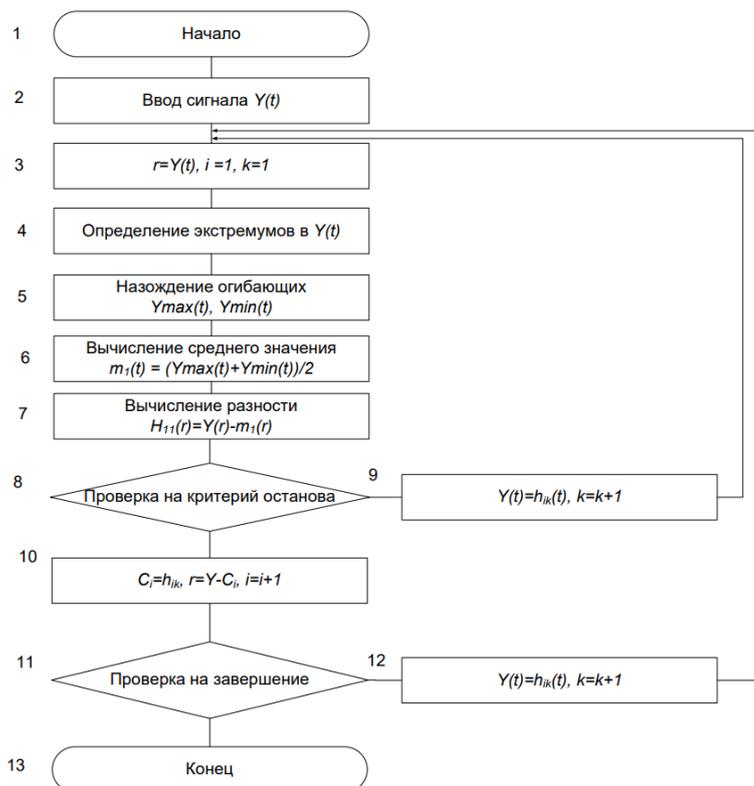


Рисунок 1 – Схема-алгоритма представления сигналов в модовой декомпозиции

Декомпозиция на модовые функции представляет собой итерационную процедуру. После открытия файла дискретных значений оцифрованного вибросигнала $Y(t)$ в блоке 4 определяются все его максимальные и минимальные значения. В блоке 5 по полученным минимальным и максимальным значениям строится верхняя и нижняя огибающие $Y_{max}(t)$ и $Y_{min}(t)$. Для построения огибающей, на интервале между двумя экстремумами как правило, применяется интерполяция

В блоке 6 вычисляется среднее значение огибающих. Первое приближение h_{11} к первой функции эмпирической моды сигнала $Y(t)$ определяется в блоке 7.

Принимая вместо функции $Y(t)$ функцию h_{11} (блок 9) и повторяя шаги с первого по четвертый, находим второе приближение к первой функции моды $h_{12}(t)$. Аналогичным образом находим и следующие приближения $h_{1k}(t)$ функции моды $C_1(t)$. Итерации продолжаются до тех пор, пока нормализованная квадратичная разность между двумя приближениями $h_{1k}(t)$ и $h_{1k-1}(t)$ Вычисляемая в блоке 8, не станет меньше некоторого предельного значения:

$$\alpha \geq \sum_{i=0}^T \left(\frac{(h_{ik-1}(t) - h_{ik}(t))^2}{h_{i-1}^2(t)} \right)$$

(1).

Последнее значение $h_{ik}(t)$ итераций принимается за первую, наиболее высокочастотную моду $C_1(k) = h_{ik}(k)$ семейства модовых функций, которая непосредственно входит в состав исходного сигнала $Y(t)$. Это позволяет вычистить $C_1(k)$ из состава сигнала и оставить в нем более низкочастотные составляющие (блок 10).

Для получения второй эмпирической моды $C_2(t)$ в блоке 12 функции $Y(t)$ присваивается значение остатка $r_1(t)$ и далее повторяются те же преобразования, что и для получения первой моды $C_1(k)$.

Просуммировав полученные значения мод $C_i(t)$ и последний полученный остаток r_n , который может быть трендом или постоянной составляющей сигнала, получим разложение сигнала согласно формулы:

$$y(t) = \sum_{i=1}^n c_i(t) + r_n \quad (2).$$

Процесс декомпозиции может быть остановлен, когда остаток $r_n(t)$ не содержит точек экстремумов, т.е. становится либо константой, либо монотонной функцией, из которой больше не могут быть извлечены модовые функции.

На рисунке 2 приведен пример декомпозиции сигнала, содержащего кратковременные возмущения, на две моды. Можно заметить, что первая мода выделяет сигнал возмущений из исходного сигнала. Определив амплитуду и частотный спектр первой моды, можно оценить характеристики сигнала возмущений.

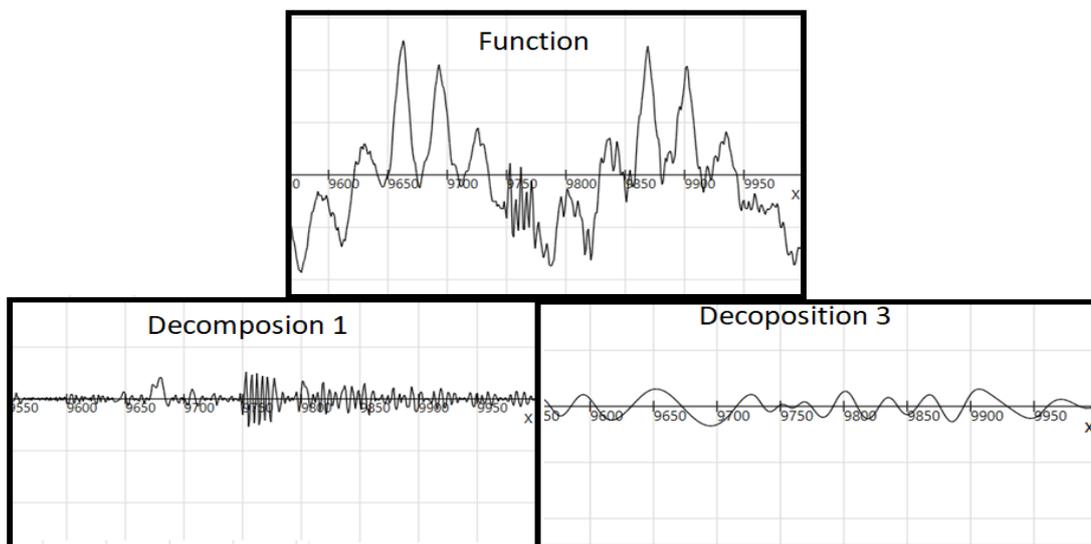


Рисунок 2 – Представление кратковременного возмущения сигнала в модовой декомпозиции

Выводы: Модовая декомпозиция сигналов может быть использована для анализа нестационарных вибрационных сигналов при оценке технического состояния машин и агрегатов.

Список использованных источников:

1. Балицкий, Ф.Я. *Неразрушающий контроль. Справочник. Том 7. Книга Вибродиагностика* / Ф.Я. Балицкий, А.В. Барков, Н.А. Баркова // М.: Машиностроение, 2005. – 829с.
2. Кан, Ш.Ч. *Анализ нестационарных сигналов на основе преобразования Гильберта-Хуанга* / Ш.Ч. Кан, А.В. Микулович, В.И. Микулович // Информатика – 2010. – №2 – С. 36-47.
3. *The empirical mode decomposition method and the Hilbert spectrum for non-stationary timeseries analysis* / N.E. Huang [et al.] // Proc. R. Soc. Lond. – 1998. – P. 903-995

УСТРОЙСТВО СКАНИРОВАНИЯ ЦВЕТА И МОБИЛЬНОЕ ПРОГРАММНОЕ СРЕДСТВО АНАЛИЗА ЦВЕТОВОЙ ПАЛИТРЫ И ПОДБОРА ЦВЕТА

Бондарь Е.А., Деменковец Д.В.

Белорусский государственный университет информатики и радиоэлектроники
г. Минск, Республика Беларусь

Деменковец Д.В. – ст. преподаватель, м.т.н.

В результате работы было сконструировано и запрограммировано устройство сканирования цвета, а также разработано мобильное программное средство анализа цветовой палитры и подбора цвета. Устройство и мобильное приложение взаимодействуют с помощью технологии Bluetooth. Данная система предназначена для того, чтобы пользователь имел возможность сканировать цвет любой ровной поверхности и в результате получить палитру цветов, которые необходимо смешать для того, чтобы получить отсканированный цвет.

В производстве строительных красок, а также порой в быту, возникает необходимость в подборе красок определенного цвета. Встречается ситуация, когда покупатель приходит в магазин и просит подобрать для него краску, которая была бы точно такой же, как и какой-либо другой элемент интерьера. Но, к сожалению, покупатель не знает номер и название цвета краски того самого элемента интерьера. С этой задачей может справиться устройство сканирования цвета и мобильное программное средство анализа цветовой палитры и подбора цвета. Кроме того, если в магазине не будет в наличии нужного цвета, мобильное приложение подберет палитру красок, которые необходимо смешать для достижения заданного цвета.

К немногочисленным аналогам данного устройства можно отнести: RM-Basf (Colortronic 1, 2), SpiesHecker (ColorDialog), DuPont (ChromaVision) и другие. Все представленные приборы специализируются на компьютерном подборе краски для автомобилей. Кроме представленных аналогов существует колориметр RAL COLORCATCH NANO, который также используется для определения цвета поверхности, но его существенным недостатком является то, что устройство сканирования цвета должно быть всегда соединено кабелем с мобильным телефоном.

Разработанное устройство сканирования цвета представляет собой конусообразную воронку диаметром 6 см. со встроенными светодиодами, датчиком и программируемой платой. Форма воронки даёт устройству возможность сканировать исключительно цвет поверхности, без воздействия внешнего света, чтобы исключить искажение сигнала выходного цвета. Светодиоды необходимы для того, чтобы максимально точно определить цвет поверхности. Степень освещенности светодиодами определяется методом подбора, учитывая отражающие свойства поверхности. Функция устройства заключается в том, чтобы принять на вход значение трех составляющих цвета поверхности (красный, зеленый, синий) и по интерфейсу Bluetooth передать эти значения цвета мобильному программному средству для последующей обработки. Схема передачи RGB-значений цвета с устройства сканирования цвета мобильному программному средству представлена на рисунке 1.

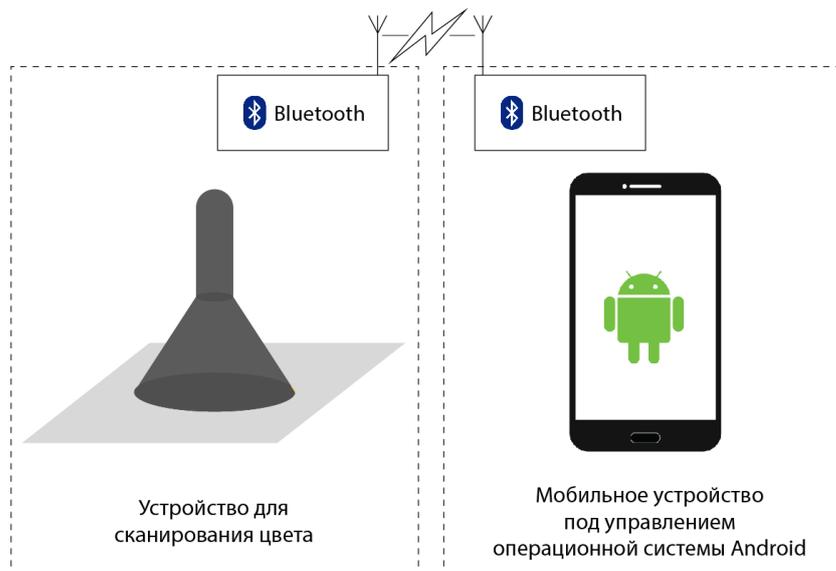


Рисунок 1 – Схема передачи RGB-значений цвета с устройства сканирования цвета мобильному программному средству

Мобильное программное средство представляет собой Android-приложение. На стартовом экране имеются три кнопки: «Сканировать» - кнопка, для сканирования цвета поверхности; «Библиотека» - кнопка, ведущая пользователя к списку сохраненных ранее цветов; «Калибровка» - кнопка, для калибровки устройства сканирования цвета. При нажатии пользователем кнопки «Сканировать», на экране появляется подсказка с изображением и описанием того, как необходимо расположить устройство сканирования цвета, чтобы цвет был определен максимально точно. После сканирования цвета поверхностью устройством, на экране мобильного приложения отображается следующая информация: визуализация отсканированного цвета поверхности; номер отсканированного цвета в таблицах CMYK, RGB и RAL; возможность добавить цвет в библиотеку; визуализация, номера в таблице RAL и процентное соотношение цветов, которые необходимо смешать, для получения отсканированного цвета; визуализация и номера цветов в таблице RAL, похожих на отсканированный цвет; возможность поделиться отсканированным цветом в социальных сетях Facebook, Twitter и ВКонтакте.

Для определения цветов, которые необходимо смешать, чтобы получить отсканированный цвет, используется субтрактивная схема формирования цвета (CMYK) - это получение цвета за счёт вычитания из белого света его спектральных составляющих. То есть, поскольку устройство передает мобильному приложению RGB-значение цвета, то для дальнейших манипуляций с цветом, его необходимо перевести в систему CMY следующим образом: $RGB(a, b, c) = CMY(255-a, 255-b, 255-c)$. После чего, полученный результат следует нормализовать. В результате применения алгоритма смешения цветов, пользователь получает палитру цветов в системе RAL с процентным соотношением каждого цвета. Субтрактивная схема формирования цвета представлена на рисунке 2.

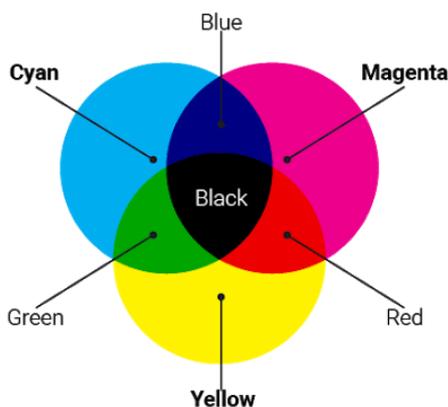


Рисунок 2 – Субтрактивная схема формирования цвета

К преимуществам устройства сканирования цвета и мобильного программного средства можно отнести:

1. Удобная форма и компактность (из-за небольшого размера устройства сканирования цвета, его можно переносить даже в кармане);
2. Портативность (при наличии портативного блока питания, устройство сканирования цвета можно использовать в любом, подходящем по требованиям, месте);
3. Низкая стоимость.

Недостатками устройства сканирования цвета и мобильного программного средства являются:

1. Особые требования к использованию (для сканирования цвета устройству необходима ровная поверхность, размером минимум 6*6 см);
2. Для работы устройства сканирования цвета, ему необходимо быть подключенным к портативному блоку питания, что утяжеляет конструкцию устройства;
3. Погрешность в сканировании цвета поверхности.

Устройство сканирования цвета и мобильного программного средства анализа цветовой палитры и подбора цвета удобно использовать для генерации палитры цветов, которые нужно смешать, чтобы получить необходимый цвет, но пока человеческий глаз по-прежнему самый точный прибор для подбора красок.

Список использованных источников:

1. Компьютерный подбор краски для авто [Электронный ресурс]. – Режим доступа: <http://avtomalaria.by/podbor/computerniy-podbor-kraski>. – Дата доступа: 20.03.2021.
2. Колориметр RAL COLORCATCH NANO [Электронный ресурс]. – Режим доступа: <https://ral.ru/colorcatch>. – Дата доступа: 22.03.2021.
3. Субтрактивное смешение цветов [Электронный ресурс]. – Режим доступа: <http://knowledge.systema-stage.ru/encyclopedia/subtraktivnoe-smeshenie-cvetov>. – Дата доступа: 22.03.2021.
4. Спектрофотометр. Виды и работа. Применение и особенности [Электронный ресурс]. – Режим доступа: <https://electrosam.ru/glavnaja/slabotochnye-seti/oborudovanie/spektrofotometr>. – Дата доступа: 24.03.2021.
5. Спектрофотометр: взгляд изнутри [Электронный ресурс]. – Режим доступа: https://www.publish.ru/articles/200002_4039976. – Дата доступа: 24.03.2021.

КЛАССИФИКАЦИЯ ЭЛЕКТРОННОЙ ПОЧТЫ

Будник А.С.

Белорусский государственный университет информатики и радиоэлектроники
г. Минск, Республика Беларусь

Герман Ю.О. – канд. тех. наук

Рассматривается метод классификации электронной почты на основе выделения ключевых слов в письме и алгоритма кластеризации К-средних.

С недавнего времени электронная почта сменила устаревшую бумажную переписку по причине своей практичности. Актуальность создания методов классификации объясняется следующим: большое количество писем приходит на почтовые ящики ежедневно, сортировка этих писем поможет сэкономить время пользователя.

Для решения данной задачи можно использовать алгоритм нахождения ключевых слов в письме, а затем классифицировать письма на основе одного из методов кластеризации.

Любой текст можно охарактеризовать набором ключевых слов, которые этот текст представляют. Первоначально рассматривают только существительные. Подсчитываются частоты слов (сколько раз каждое слово вошло в текст). Поскольку слова входят в текст с разными окончаниями, то гласные выбрасываются и два слова считаются совпадающими, если они достаточно похожи друг на друга. При этом короткие слова должны быть похожими друг на друга в большей степени, чем длинные. Затем слова размещаются по убыванию частот, и берется примерно корень из общего числа слов с наибольшими значениями частот. При этом чем длиннее слова, тем меньше требуется допустимый процент совпадения [1]. Этот процент можно установить исходя из следующей эмпирически составленной таблицы 1.

Таблица 1 – Допустимое число неверных символов

Размер слова	Допустимое число неверных символов
До 5	0
5-6	1
7-8	2
9-10	3
Больше 10	4

На следующем этапе будет использоваться алгоритм К-средних.

Кластерный анализ – многомерная статистическая процедура, выполняющая сбор данных, содержащих информацию о выборке объектов, и затем упорядочивающая объекты в сравнительно однородные группы [2].

Метод К-средних – это метод кластерного анализа, цель которого – разделение m наблюдений на k кластеров, при этом каждое наблюдение относится к тому кластеру, к центру которого оно ближе всего. Результат разбиения на кластеры представлен на рисунке 1.

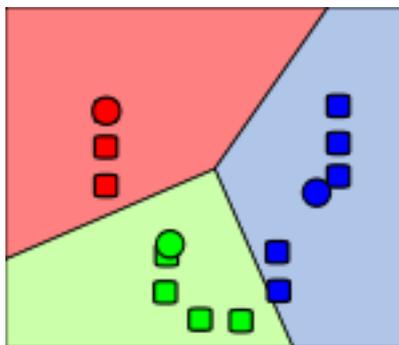


Рисунок 1 – Результат работы алгоритма К-средних

Таким образом, на основе тестовой выборки будут сформированы кластеры почтовых писем (например, деловые письма, поздравления, спам). Затем при получении нового письма будет определяться набор его ключевых слов, и письмо будет отнесено к тому или иному кластеру, тем самым будет выполнена процедура классификации писем.

Список использованных источников:

1. Герман, О.В. Искусственный интеллект: метод. пособие / О. В. Герман, Ю.О. Герман. – Минск : БНТУ, 2013. – 127с.
2. Мандель, И.Д. Кластерный анализ / И. Д. Мандель. – М.: Финансы и статистика, 1988. – 176 с.

СРАВНЕНИЕ LR И GLR СИНТАКСИЧЕСКИХ АНАЛИЗАТОРОВ

Юрченко Н.А., Будницкий Е.Г.

Белорусский государственный университет информатики и радиоэлектроники
г. Минск, Республика Беларусь

Савёнок В.А. – инженер-программист ООО «Незабудка Софтвр»,
магистр технических наук

В данной работе рассмотрены современные алгоритмы синтаксических анализаторов, которые применяются для разбора текстов формальных и естественных языков. Приведено подробное описание восходящих анализаторов: LR, SLR и GLR, сравнение их возможностей и ограничений, связанных с реализацией.

В настоящее время количество информации, представленной в электронном виде, растет с каждым днем. Все больше текстов с бумажных носителей переносится на электронные, для общения используются социальные платформы и электронная почта. Такой большой объем данных невозможно обработать за рациональное время не только одному человеку, но и группе специалистов, что приводит к потребности в автоматизации обработки и анализа текстовой информации.

Синтаксический анализатор – это программа или подпрограмма, производящая сопоставление линейной последовательности лексем формального или естественного языка его формальной грамматике [1]. Формальный язык – множество строк, из определенного алфавита, удовлетворяющее правилам. Для проверки корректности грамматики используются терминальные и нетерминальные символы. Терминал – элемент грамматики, для которого отсутствуют правила подстановки. Нетерминал – элемент грамматики, для которого есть правила подстановки. Синтаксический анализатор можно построить практически по любой грамматике. Результатом работы синтаксического анализатора является синтаксическое дерево, которое характеризует структуру выражения. На сегодняшний день существует ряд алгоритмов разбора предложений. Эти алгоритмы делятся на две группы в зависимости от способа построения дерева разбора: нисходящие и восходящие анализаторы. Нисходящий разбор строит дерево путем развёртки нетерминалов до соответствия с исходной строкой, восходящий – путем свертки терминалов в нетерминалы пока не останется один нетерминал. Методы восходящего разбора более предпочтительны, так как не требуется левофакторизованная грамматика. В данной работе рассматриваются анализаторы, входящие во вторую группу: семейства LR и GLR анализаторов.

Анализаторы LR детерминированы, они читают входной поток слева направо и преобразуют строку путем замены самого правого нетерминала. LR-анализаторы производят единственный правильный синтаксический анализ, в случае неправильного конечного ответа отката к предыдущим сверткам для выбора другого сворачивания не будет, так как на каждом шаге имеется только определенная схема свертки, в связи с этим алгоритм занимает линейное время. Существует множество вариантов LR-анализаторов, например LR(0), SLR(1), LR(1), LALR(1). Число в скобках показывает количество символов непрочитанной цепочки, используемых анализатором для принятия решения на каждом шаге. Однако при работе LR анализатора имеется проблема, что в некоторых ситуациях нельзя однозначно решить, какое действие следует предпринять в данный момент для достижения верного результата, так как имеется несколько вариантов свертки по различным правилам. При выборе ошибочного решения верная грамматика может оказаться ошибочной из-за неправильного свертывания. В связи с этим строятся таблицы LR разбора, в которых сохраняют только однозначное решение для данного и следующего символа.

LR(0) – восходящий алгоритм синтаксического разбора контекстно-свободных грамматик, один из видов LR. Данный алгоритм редко используется в связи с малым размером грамматик, которые он может разобрать, но он является основой для более эффективных алгоритмов разбора, таких как SLR(1) и LALR(1). Данный алгоритм не требует предпросмотра, что уменьшает время на проверку его возможных грамматик.

SLR(1) – улучшение восходящего алгоритма синтаксического разбора LR(0), которое позволяет разбирать более сложные грамматики. Однако сложность разбираемых им грамматик недостаточно высока, что ограничивает область использовать данного алгоритма только в учебных целях.

LR(1) – восходящий алгоритм синтаксического разбора, для принятия решения которого требуется прочтение одного символа. Главное отличие от LR(0) алгоритма заключается в наличии предпросмотра. Основная идея данного алгоритма заключается в том, чтобы не производить некорректные свертки. Алгоритм затрачивает больше оперативной памяти, но из-за увеличения возможной грамматики его возможностей хватает на различные языки, к примеру C, C++ [2].

LALR(1) (LA от англ. lookahead – предпросмотр) представляет собой восходящий алгоритм синтаксического разбора; является расширением SLR(1). Он позволяет сохранить наиболее подходящие ветви разбора для устранения неоднозначности. Несмотря на то, что данный алгоритм менее мощный, чем LR(1), его возможностей вполне достаточно для разбора большинства языков, к

примеру C, C++ [2].

GLR-анализатор (англ. Generalized Left-to-right Rightmost derivation parser) – обобщенный восходящий магазинный анализатор или параллельный синтаксический анализатор. Этот расширенный алгоритм предназначен для разбора по недетерминированным и неоднозначным грамматикам. Теоретическая база была заложена Бернардом Лэнгом [3] впервые такой синтаксический анализатор описал Масару Томита [4]. Цель создания GLR-анализатора – анализ текстов на естественных языках, с недетерминированностью и неоднозначностью которых LR-анализатор не справляется.

Для разбора недетерминированных грамматик GLR-анализаторы обрабатывают различные ветви разбора параллельно, они используют более сложную стековую структуру, нежели LR-анализаторы. Такая структура называется графовой структурой (GSS – англ. graphstructured stack). Использование графовой структуры позволяет эффективно хранить различные ветви разбора, которые помечаются неактивными по мере сдвига входных токенов или символов. Все неактивные ветви должны быть восстановлены в случае повторного посещения старого состояния, из чего следует задача применения поиска с возвратом, что усложняет работу анализатора.

Работу синтаксического анализатора можно оптимизировать сокращением объема хранимой информации. GLR-анализатор считается самым универсальным из рассмотренных алгоритмов, поскольку он разбирает не только однозначные грамматики, но также способен успешно анализировать естественные языки. Однако у классического GLR-анализатора есть ряд недостатков. В частности, он не имеет средств выхода из скрытой левой рекурсии [5], которая часто присутствует в грамматиках языков программирования. На практике часто используются гибридные анализаторы, в которых GLR комбинируют с алгоритмом LALR(1), что обеспечивает более высокую производительность.

Сравнительный анализ описанных восходящих анализаторов представлен в таблице 1. Для анализа выбраны следующие характеристики: глубина предпросмотра, уровень сложности грамматики, размер таблицы, способность анализировать естественные языки.

Таблица 1 – Сравнительная характеристика синтаксических анализаторов

Тип анализатора	Параметры сравнения			
	Глубина предпросмотра	Уровень сложности грамматики	Размер таблицы	Подходит для анализа естественных языков
LR(0)	0	Низкий	Маленькая	Нет
SLR(1)	1	Низкий	Больше, чем у LR(0)	Нет
LALR(1)	1	Средний	Такая же, как у SLR(0)	Нет
LR(1)	1	Средний	Большая	Нет
GLR	Не ограничена	Высокий	Большая	Да

Развитие анализаторов происходило в порядке: LR(0) → SLR(1) → LALR(1) → LR(1) → GLR. Таким образом, LR(0) поддерживает грамматику только LR(0), SLR(1) поддерживает LR(0) и SLR(1), LALR(1) поддерживает LR(0), SLR(1) и LALR(1) и т. д. Для разбора строятся LR-таблицы, размер которых увеличивается с ростом сложности разбираемой грамматики как показано в таблице 1. Размер LR-таблицы также связан с количеством всевозможных состояний, а состояния в свою очередь связаны со сложностью грамматики. Поэтому чем проще грамматика, тем быстрее ее будет разобрать. Использование предпросмотра увеличивает затрачиваемое время на анализ, но не во всех случаях избежать предпросмотр возможно. Из-за простоты некоторых грамматик, LR(0), SLR(1), необходимость предпросмотра отпадает, что уменьшает время на их анализ.

Анализаторы семейства LR подходят для разбора формальных языков, но не пригодны для работы с естественными языками, потому что не способны разрешать недетерминированность и неоднозначность этих языков. Для анализа естественных языков используют семейство GLR-анализаторов.

Таким образом по результатам анализа нельзя выделить лучший или худший синтаксический анализатор. Все анализаторы имеют свои плюсы и минусы. Каждый анализатор разрабатывался для разбора определенных грамматик. При выборе анализатора следует исходить из задачи, предоставленной для решения, и искать анализатор, который справится с этой задачей максимально эффективно.

Список использованных источников:

1. Бионика интеллекта / Борисова Н.В., Канищева О.В. – Изд. 2-е. – 2014
2. Bison GNU / Operating system GNU. – Mode of access: <https://www.gnu.org/software/bison/>
3. Deterministic techniques for efficient non-deterministic parsers / Lang B. Lang B. – 2nd Colloquium – Saarbrücken: Springer.
4. Efficient parsing for natural language / Tomita M. – Boston : Kluwer Academic Publishers, 1986.
5. Generalized parsing: some costs / Johnstone A., Scott E., Economopoulos G.. – Berlin: Springer, 2004.

УДАЛЕННАЯ СИСТЕМА УПРАВЛЕНИЯ ДАННЫМИ

Герасимов Я.А.

Белорусский государственный университет информатики и радиоэлектроники
г. Минск, Республика Беларусь

Герман Ю.О. – канд. тех. наук

Рассматривается использование удаленной системы управления данными в качестве средства оптимизации процессов совместной работы по созданию и сопровождению программного обеспечения.

В современном мире трудно представить себе такую ситуацию, когда программное обеспечение будет разрабатывать один человек. В подавляющем большинстве случаев для разработки программных продуктов используется целая команда разноплановых специалистов: менеджеров, разработчиков, архитекторов решения, бизнес-аналитиков.

Большой объем информации и знаний, с которыми приходится работать командам, разрабатывающим программные продукты, требует целой стратегии по управлению этими знаниями. Большинство методологий, такие как Agile, Waterfall, Spiral Model, Iterative Model, требуют обязательного использования баз знаний в проекте, поскольку они существенно упрощают систематизацию знаний команды, необходимых для эффективной разработки программного обеспечения, для ввода новых участников и вовлечения всех членов команды в обсуждение и усовершенствование процесса разработки программного обеспечения.

Удаленная система управления данными (далее – система) предполагает постоянно функционирующее приложение с возможностью доступа через браузер, что позволяет оперативно вносить новые данные и получать изменения всем участникам разработки. Она является как базой знаний, так и мощным инструментом разработки программного обеспечения. Такая система будет полезна на всех стадиях разработки программ: техническом задании, эскизном проекте, техническом проекте, рабочем проекте и внедрении [1].

На стадии разработки технического задания система позволяет оценить целесообразность использования предыдущих решений, оперативно вносить изменения в техническое задание, дать представление всем участникам разработки о целях и средствах их достижения.

На стадии эскизного проекта система предоставляет возможность совместного доступа к диаграммам и моделям, совместную разработку структуры данных и общего алгоритма решения задачи.

На стадии технического проекта система позволяет совместную разработку структуры программы, разработку плана совместных мероприятий, определение формы представления входных и выходных данных.

На стадии рабочего проекта система становится наиболее полезной. В системе предоставляется возможность хранения и изменения плана тестирования, отдельных тестов, требований для разработчиков, дизайнов страниц, архитектуры приложения. Также хранятся и изменяются при необходимости со стороны заказчика бизнес-требования, предъявляемые к программному обеспечению. Сохраняются всевозможные записи встреч с заказчиком, записи обсуждений функциональных требований, записи презентаций.

На стадии внедрения необходимые документы, оформленные на предыдущих этапах в системе, передаются заказчику. Происходит сопровождение в соответствии с бизнес-процессами. При разногласиях данные из системы могут стать доказательством того, что все работает, как и было затребовано.

Таким образом, основные преимущества системы – в оперативном доступе к актуальным данным и их структурированность, а также безопасность и доступность. Различные типы и форматы данных могут быть иерархические, представлены в одном месте, что также немаловажное преимущество. Удаленная система управления данными на сегодняшний день востребована многими компаниями на всех стадиях разработки. А с текущей тенденцией подходов разработки программного обеспечения по методологии Agile подобные системы становятся жизненной необходимостью компаний по всему миру.

Список использованных источников:

1. ГОСТ 19.102-77. Единая система программной документации. Стадии разработки – Введ. 01.01.1980, переиздан 04.03.2010. – М : Стандартинформ, 2010. – 2 с.

МОБИЛЬНОЕ ПРОГРАММНОЕ СРЕДСТВО УПРАВЛЕНИЯ ПРЕДМЕТАМИ КОЛЛЕКЦИИ НА ОСНОВЕ РАДИОЧАСТОТНЫХ МЕТОК

Голубев К.А., Фадеева Е.Е.

Белорусский государственный университет информатики и радиоэлектроники
г. Минск, Республика Беларусь

Деменковец Д.В. – ст. преподаватель, м.т.н.

В результате работы было разработано мобильное программное средство управления предметами коллекции на основе радиочастотных меток. Мобильное приложение взаимодействует с радиочастотными метками с помощью технологии NFC. Данная система предназначена для того, чтобы пользователь имел возможность сканировать и записывать информацию на метки, поддерживающие стандарт NTAG 215. В качестве примера рассматривается учёт коллекционных моделей Amiibo, поставляющихся со встроенным RFID чипом.

Коллекционирование является феноменом социокультурного плана. Это процесс самовыражения человеческой личности, способ самореализации индивидуальности в свободное время. Коллекционеры образуют своеобразные социальные сети общения и формируют самобытные субкультуры по интересам. Такая деятельность наполнена социальным социокультурным смыслом [1]. В условиях массового производства и избытка информации, препятствующих концентрации, современный человек испытывает всё большую «потребность углубиться во что-то», что способствует бурному развитию коллекционирования как коммерческой индустрии в быстроразвивающемся современном мире [2].

В настоящее время среди коллекционеров существует потребность в новых способах управления предметами непосредственного коллекционирования: старые технологии ведения ручного каталога не отвечают необходимым критериям безопасности, удобства и доверия независимо от того, ведутся они на бумаге или с помощью компьютера. Наличие предмета в таком каталоге не гарантирует его наличие у коллекционера и потому во многих ситуациях может возникнуть необходимость перепроверок. Коллекционер ценит возможность презентовать свою коллекцию обществу. Это возможность оценить свой труд, узнать общественное мнение и расширить свои познания в поиске новых экземпляров, что усложняется устаревшими технологиями учёта коллекции [3].

С этой задачей может справиться технология внедрения в предметы радиочастотных меток и мобильное программное средство, которое будет сканировать и обрабатывать данные с них. К существующим аналогам данного метода перевода коллекционирования в цифровое поле можно отнести QR-коды и генерацию цифровой электронной подписи. Первый способ не решает проблемы с безопасностью и гарантией, а второй – удобства.

На данный момент на рынке существует серия коллекционных моделей Amiibo от компании Nintendo со встроенными радиочастотными метками стандарта NFC NTAG 215 [4]. В эти модели они встроены для взаимодействия с видеоиграми компании, однако на примере этой серии было принято решение продемонстрировать возможности рассматриваемой технологии управления коллекцией [5].

Мобильное программное средство представляет собой кроссплатформенное приложение. На стартовом экране имеются четыре вкладки: «Моя коллекция» - отсортированный список имеющихся у пользователя моделей; «Библиотека» - структурированный список всех существующих моделей по сериям выпуска, году, тематике и т.д. с индикатором полноты собранной коллекции; «Поиск» - меню поиска информации про интересующие модели; «Настройки» - меню настроек и информации о приложении. При нажатии пользователем кнопки «Сканировать» в разделе «Моя коллекция» на экране появляется подсказка с изображением и описанием того, как необходимо расположить мобильное устройство и коллекционную модель для сканирования радиочастотной метки, чтобы произошла синхронизация и чтение информации. После сканирования информации и модели, на экране мобильного приложения отображается следующая информация: визуализация отсканированной модели, серия и год выпуска, дополнительная информация о конкретном, возможность добавить модель в свою коллекцию. В случае, если эта модель уже была добавлена в коллекцию, пользователю предлагается на выбор либо обновить информацию о ней в соответствии с возможными изменениями, либо прервать процесс добавления модели. Приложение позволяет не только считывать, но и записывать информацию в пустые метки, тем самым упрощает возможность использования технологии для коллекционеров менее технологическими предметами. Для этого необходимо выбрать интересующую модель в библиотеке и проследовать инструкции по записи. Схема взаимодействия коллекционной модели и разработанного приложения представлена на рисунке 1.

Данное программное средство разработано на основе технологии Flutter и ориентировано для персонального использования на личном мобильном устройстве. Выбор технологии обусловлен задачей разработать приложение в минимальные сроки и на максимально широкий список устройств, а Flutter позволяет создавать современные мобильные приложения сразу на несколько платформ и

независимо от версии операционной системы.



Рисунок 1 – Схема взаимодействия модели Amiibo и мобильного программного средства

Flutter – бесплатный и открытый набор средств разработки мобильного пользовательского интерфейса, созданный компанией Google и выпущенный в мае 2017 года. Проще говоря, с помощью Flutter возможно создать собственное мобильное приложение с одним массивом кода. Это означает, что для создания двух приложений (IOS и Android) можно использовать единый язык программирования и одну базу кода. Логотип платформы Flutter приведён на рисунке 2.



Рисунок 2 – Логотип платформы Flutter

К преимуществам технологии внедрения радиочастотных меток и использования мобильного программного средства можно отнести:

1. Доступность (пользователь может использовать своё личное мобильное устройство, которое всегда под рукой);
2. Безопасность (использование технологии NFC гарантирует подлинность предмета и требует непосредственного контакта для синхронизации);
3. Низкая стоимость (внедрение радиочастотной метки не создаёт высоких издержек за счёт низкой цены устройства);
4. Компактность (внедрение радиочастотной метки не создаёт высоких издержек за счёт низкой цены устройства).

Недостатками технологии и мобильного программного средства являются:

1. Особые требования к предметам коллекционирования (должны содержать в себе радиочастотные метки или быть дополнительно оснащены ими);
2. Удобство в использовании (технология NFC требует очень близкого контакта между устройством и меткой);
3. Распространённость (для повсеместного использования выбранной технологии от производителей коллекционных предметов требуется внедрять в свои продукты специальные радиочастотные устройства).

Разработанное мобильное программное средство удобно использовать для управления коллекцией моделей Amiibo, но пока производители других коллекционных предметов не разработают общий стандарт использования данной технологии в своих продуктах, широкое использования рассматриваемой технологии остаётся нецелесообразным.

Таким образом, главная задача – создание программного средства, упрощающего учёт коллекционных предметов и позволяющего пользователю иметь доступ к виртуальному журналу своей коллекции – была выполнена.

Список использованных источников:

1. Шахова И. А. Коллекционирование как социокультурный феномен современного общества / Вестник Амурского государственного университета – 2017. – Серия: Гуманитарные науки. – №78.
2. Хёйзинга Й. Тени завтрашнего дня / СПб – 2017 – С. 454.
3. Digitizing Collections [Электронный ресурс]. – Режим доступа: <https://siarchives.si.edu/what-we-do/digital-curation/digitizing-collections>. – Дата доступа: 03.04.2021.
4. Что такое Amiibo? [Электронный ресурс]. – Режим доступа: <https://www.nintendo.ru/amiibo/-amiibo-amiibo-932316.html>. – Дата доступа: 03.04.2021.
5. ISO/IEC 18092:2004 Information technology -- Telecommunications and information exchange between systems – Near Field Communication [Электронный ресурс]. – Режим доступа: <https://www.iso.org/standard/38578.html>. – Дата доступа: 03.04.2021.

АППАРАТНО-ПРОГРАММНЫЙ КОМПЛЕКС ПО МОНИТОРИНГУ И АНАЛИЗУ СОСТОЯНИЯ ОКРУЖАЮЩЕЙ СРЕДЫ

Гулида М. А.

Белорусский государственный университет информатики и радиоэлектроники
г. Минск, Республика Беларусь

Видничук В.Н. – м.т.н.

Экология стала важным фактором, влияющим на качество жизнедеятельности человека. В данной работе предлагается аппаратно-программный комплекс на основе Arduino Nano для мониторинга и анализа состояния окружающей среды.

Согласно данным Всемирной Организации Здравоохранения (ВОЗ) 91% населения Земли живет в местах, где качество воздуха не соответствует нормам, рекомендованным ВОЗ. Более четырех миллионов смертей наступает ежегодно из-за заболеваний, вызванных воздействием тонкодисперсных частиц в загрязненном воздухе [1].

Мониторинг состояния окружающей среды с помощью датчиков позволяет получать наиболее точную информацию о различных показателях загрязненности территории, таких как уровень оксида углерода, оксида азота, диоксида серы, формальдегида, а также уровень радиации.

Дальнейший анализ полученных данных позволяет более точно предложить пользователю экологические активности различного рода, выполнение которых позволило бы улучшить данные показатели и как следствие снизить уровень загрязнения воздуха и улучшить качество жизни.

Для решения данной проблемы предлагается аппаратно-программный комплекс по мониторингу и анализу состояния окружающей среды. Данный комплекс предназначен как для индивидуального использования, так и для использования экологическими организациями для контроля уровня загрязнения окружающей среды и своевременной реакции на складывающиеся изменения.

В качестве аппаратной основы данного аппаратно-программного комплекса используется платформа Arduino Nano, построенная на микроконтроллере ATmega328P [2]. Комплекс включает в себя датчики контроля уровня оксида углерода, оксидов тяжелых металлов, формальдегидов, температуры и влажности воздуха, уровня и дозы радиации.

Данные с датчиков аккумулируются на сервере, где происходит их дальнейший анализ. На основании проведенного анализа пользователям предлагаются экологические активности или даются рекомендации, которые следует соблюдать для уменьшения влияния тех или иных загрязнений. Например, согласно руководству ВОЗ в помещениях в целях сведения к минимуму воздействия на здоровье, уровень оксида углерода должен находиться ниже 8-часового значения 10мг/м^3 [3]. Если уровень оксида углерода превышает данное значение в здании рекомендуется сделать влажную уборку, в случае наличия увлажнителя воздуха, воспользоваться им. В случае нахождения на открытой местности, рекомендуется уменьшить поток транспорта, избегать сжигания листьев.

Ключевой особенностью предлагаемого комплекса является то, что он представляет собой платформу для коллективного контроля за экологической обстановкой в определенном регионе. Помимо сбора данных, предлагаемая система реализует их аккумулирование в централизованном хранилище с последующим глубоким анализом характеристик состояния воздуха на покрытой комплексом территории равно, как и динамики изменения этих характеристик во времени. На основе результатов анализа автоматически формируются рекомендации относительно действий, как личных, так и коллективных, которые стоит предпринять пользователям для улучшения экологической обстановки окружающей их среды.

Платформа включает в себя элементы социальных сетей как инструмент для эффективной коммуникации и самоорганизации пользователей в принятии коллективных действий в соответствии с рекомендациями данной системой на основе анализа собранных данных. Существующие аналоги реализуют только сбор и отображение данных о состоянии воздуха и лишены как функций анализа и вынесения рекомендаций, так и инструментов для самоорганизации пользователей для принятия коллективных действий, направленных на поддержание и улучшение экологической обстановки окружающей их среды.

Список использованных источников:

1. World Health Organization. *Ambient air pollution: a global assessment of exposure and burden of disease* / World Health Organization. – 2016.
2. Arduino Nano (V2.3): User Manual [Electronic Resource] / Arduino. – Mode of access: <https://www.arduino.cc/en/uploads/Main/ArduinoNanoManual23.pdf>. Date of access: 30.03.2021.
3. Всемирная Организация Здравоохранения. *Руководство ВОЗ по качеству воздуха в помещениях: избранные загрязняющие вещества* / Всемирная Организация Здравоохранения. – 2011.

НЕРАЗРУШАЮЩЕЕ ТЕСТИРОВАНИЕ ЗАПОМИНАЮЩИХ УСТРОЙСТВ НА БАЗЕ ДВОЙНЫХ АДРЕСНЫХ ПОСЛЕДОВАТЕЛЬНОСТЕЙ

Деменковец Д.В.

Белорусский государственный университет информатики и радиоэлектроники
г. Минск, Республика Беларусь

Ярмолик В.Н. – профессор, доктор технических наук

Анализируется эффективность применения неразрушающих тестов для тестирования запоминающих устройств (ЗУ). Синтезируется базовый элемент неразрушающих тестов с применением двойных адресных последовательностей. Приводится неразрушающий тест ЗУ March_2A_1 для которого оценивается его временная сложность и эффективность обнаружения неисправностей ЗУ.

Основное свойство методов неразрушающего тестирования запоминающих устройств заключается в сохранении данных хранимых в памяти после проведения процедуры тестирования [1–3]. Развитием неразрушающего тестирования ЗУ стала методика, предложенная М. Николаидисом (M. Nicolaidis) [3, 4]. Данная методика из-за эффекта маскирования не позволяет получить 100%-ую покрывающую способность [1, 5].

Для формального описания неисправных состояний ЗУ используются математические модели их неисправностей, отражающие реальные физические дефекты ЗУ [2–4, 6]. Наиболее сложными моделями неисправностей, обнаруживаемых маршевыми тестами и покрывающими более простые неисправности являются, кодочувствительные пассивные неисправности NPSFk (PNPSFk), в которых содержимое базовой ячейки нельзя изменить в зависимости от определенного набора в любых k из N ячеек [3, 4].

В таблице 1 приведены оригинальная реализация и неразрушающая модификация по методике Николаидиса теста March Y [3, 4].

Таблица 1. Две версии реализации теста March Y

Описание теста	Тест	Сложность теста
Неразрушающий тест	$\{\uparrow(rb, r\bar{b}); \downarrow(r\bar{b}, rb); \uparrow\downarrow(rb)\}$ $\{\uparrow(rb, w\bar{b}, r\bar{b}); \downarrow(r\bar{b}, wb, rb); \uparrow\downarrow(rb)\}$	12N

Тест приведенный в табл. 1 состоит из двух частей. Начальная часть теста, необходима для получения эталонной сигнатуры SF [1, 2]. После реализации базового теста формируется реальная сигнатура SR, которая сравнивается с ранее полученной эталонной SF. По результату сравнения принимается решение о наличии либо отсутствии неисправности в ЗУ. Несовпадение реальной сигнатуры SR с ее эталонным значением SF свидетельствует лишь о неисправном состоянии памяти [2].

Для повышения эффективности применения маршевого теста используют подход, при котором тест повторяется несколько раз, но с разной последовательностью адресов на каждой итерации маршевого теста [7]. Основная идея неразрушающих маршевых тестов на базе двойных адресных последовательностей основана на том, что при двукратном инвертировании содержимого ячейки ЗУ его значение останется прежним. В соответствии с этим простейшим свойством операции инвертирования строится базовый элемент неразрушающего маршевого теста на базе двойной адресной последовательности 2A. Маршевый элемент должен начинаться с операции чтения rb содержимого b текущей ячейки ЗУ. Следующей операцией должна быть операция записи инверсного значения по отношению к только что прочитанному содержимому из ячейки, так как подобная операция является необходимым условием активизации неисправностей ЗУ. За операцией записи следует операция чтения этой же текущей ячейки ЗУ для проверки правильности выполнения операции инвертирования ее содержимого. Далее переходят к следующему запоминающему элементу, который соответствует следующему адресу адресной последовательности. Использование двойных адресных последовательностей 2A обеспечивает повторное инвертирование каждой ячейки ЗУ, в итоге сохраняя его исходное состояние. Базовый элемент имеет следующий вид.

$$2 \uparrow (rb, w\bar{b}, rb) \quad (1)$$

Отметим, что использование в базовом элементе двойной адресной последовательности ($2\uparrow$) приводит к тому, что каждая ячейка ЗУ последовательно выполнит два перехода из обычного состояния в инверсное и, наоборот, из инверсного в первоначальное, сохранив, таким образом, свое начальное значение. Правильность выполнения обоих переходов (\uparrow) и (\downarrow), а также операций чтения нулевых и единичных значений обеспечивает вторая операция чтения rb базового элемента (1).

Базовый элемент на основе двойных адресных последовательностей позволяет синтезировать неразрушающий маршевый тест (2):

$$\text{March_2A_1: } \{\uparrow\downarrow (rb); 2 \uparrow (rb, w\bar{b}, rb); \uparrow\downarrow (rb)\}, (8N); \quad (2)$$

В тесте произвольный порядок адресов $\uparrow\downarrow$ для первой и последней операций чтения должен быть одинаков, возрастающий либо убывающий. Это связано с тем, что первая фаза теста March_2A_1 используется для сжатия исходного состояния ЗУ и получения эталонной сигнатуры SF, а их последняя фаза для получения реального значения сигнатуры SR после выполнения предыдущих базовых элементов. В случае проявления неисправностей в ходе выполнения базовых элементов их наличие будет определяться выполнением неравенства $SF \neq SR$. Максимально возможная диагностическая способность теста March_2A_1 достигается и для случая сложных кодочувствительных неисправностей PNPSFk. Выполнение базового элемента в случае PNPSFk позволяет идентифицировать адрес базовой ячейки. Кроме того, тест March_2A_1 позволяет достичь такого же значения полноты покрытия для PNPSFk, как и тест March Y.

Для обнаружения неисправностей взаимного влияния необходимо выполнить анализ состояния ячейки жертвы после активизации конкретной неисправности, что невозможно в рамках базового элемента (1). Это приводит к тому, что конечное состояние ЗУ будет отличаться от его исходного состояния, что приведет к выполнению неравенства $SF \neq SR$. Полнота покрытия тестом March_2A_1 для неисправностей CFid из экспериментальных данных, приведена в табл. 2.

Таблица 2. Полнота покрытия неисправностей CFid тестом March_2A_1 в процентах (%)

CFid	2A _{c0}	2A _{c1}	2A _{c2}	2A _{c3}	2A _{c4}	2A _{c5}	2A _{c6}	2A _{c7}	2A _{c8}
(0,↑)	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00
(1,↑)	100,00	100,00	100,00	100,00	100,00	100,00	100,00	100,00	100,00
(0,↓)	0,00	0,39	1,18	2,75	5,88	12,16	24,71	49,80	100,00
(1,↓)	100,00	99,61	98,82	97,25	94,12	87,84	75,29	50,20	0,00
(↑,0)	0,00	0,39	1,18	2,75	5,88	12,16	24,71	49,80	100,00
(↑,1)	100,00	99,61	98,82	97,25	94,12	87,84	75,29	50,20	0,00
(↓,0)	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00
(↓,1)	100,00	100,00	100,00	100,00	100,00	100,00	100,00	100,00	100,00
Total	50,00	50,00	50,00	50,00	50,00	50,00	50,00	50,00	50,00

Как видно, из приведенной таблицы, общее (Total) количество обнаруживаемых неисправностей тестом March_2A_1, независимо от адресной последовательности 2A, всегда равняется 50 %. Приведенный анализ свидетельствует о высокой покрывающей способности нового неразрушающего теста March_2A_1, которая сравнима с покрывающей способностью March Y. Кроме того, наличие базового элемента (1) в обоих тестах обеспечивает максимальную диагностическую способность, достижимую в рамках маршевых тестов, для неисправностей SAF, TF и PNPSFk.

Список литературы

1. Nicolaidis M. Theory of transparent BIST for RAMs. *IEEE Transactions on Computers*. 1996;45(10):1141-1156.
2. Ярмолик В.Н., Мурашко И.А., Куммерт А., Иванюк А.А. *Неразрушающее тестирование запоминающих устройств*. Минск: Бестпринт; 2005.
3. Ярмолик С.В., Занкович А.П., Иванюк А.А. *Маршевые тесты для самотестирования ОЗУ*. Минск: Бестпринт; 2009.
4. Goor A.J. *Testing Semiconductor Memories: Theory and Practice*. Chichester, UK: John Wiley & Sons; 1991.
5. Yarmolik V.N., Nicolaidis M., Kebichi O. *Aliasing-Free Signature Analysis for RAM BIST*. *IEEE International Test Conference*. 1994:368-377.
6. Hamdioui S., Wadsworth R., Reyes J.D., Goor A.J. *Memory Fault Modeling Trends: A Case Study*. *Journal of Electronic Testing*. 2004;20(3):245-255.
7. Mrozek I., Yarmolik V.N. *Multiple Control Random Testing*. *Fundamenta Informaticae*. 2019;144(1):23-43.

ПРОГРАММНОЕ СРЕДСТВО АВТОМАТИЗАЦИИ НАЙМА ПЕРСОНАЛА ИТ-КОМПАНИИ

Дубовская Е.В.

Белорусский государственный университет информатики и радиоэлектроники
г. Минск, Республика Беларусь

Данилова Г.В. – ассистент каф.ПОИТ

Постоянный рост рынка труда в белорусском IT-секторе, одновременно с дефицитом квалифицированных специалистов создаёт особые требования кандидатов к компаниям, а соответственно, и к процессу найма сотрудников. Для автоматизации поиска сотрудников разработано программное средство, представляющее из себя веб-приложение. Приложение учитывает структуру компании, как по отделам, так и по проектам, и ориентировано на организацию найма персонала в соответствии требованиям компаний.

В современном мире технологии становятся настолько сложными, а процессы настолько запутанными, что за их динамикой становится крайне трудно уследить. Данное правило всецело относится в том числе и к подбору персонала. Квалифицированные и правильно подобранные команды являются неотъемлемой частью компании любого размера. Поэтому автоматизация и структуризация данного процесса является залогом успешной работы предприятия. Найм персонала (IT-рекрутинг) – быстро развивающееся направление, поэтому приложения, ориентированные на данный рынок, пользуются большой популярностью. Приложения данного направления решают такие задачи, как снижение нагрузки на менеджеров и собеседующих, хранение истории коммуникации с кандидатом, а также общая формализация процесса найма [1].

Актуальность данного программного средства обусловлена необходимостью оптимизировать время тех сотрудников, которые отвечают за подбор персонала. Снижение нагрузки достигается с помощью унификации различной информации о кандидатах. Приложение позволяет координировать коммуникацию с кандидатами, хранить результаты всех собеседований и тестовых заданий.

Анализ аналогов, а именно программных средств в сфере найма персонала, показал, что большинство решений являются универсальными конфигурируемыми системами по работе с персоналом для фирм разного направления. Типичными проблемами таких систем являются трудоемкая конфигурация, излишняя интеграция со сторонними сервисами (сайты, площадки), неиспользуемыми при найме в IT-секторе, множество излишних модулей, сложная поддержка системы, малая расширяемость системы.

Основная идея данной работы заключается в создании приложения, которое должно повысить эффективность найма в сфере информационных технологий. Приложение позволит добавить кандидатов в систему, редактировать информацию в процессе работы с ними, изменять статус кандидата. Также приложение будет содержать информацию обо всех проектах и вакансиях в компании. Внутри системы кандидаты будут соотноситься с вакансиями, вакансии – с проектами. Дизайн пользовательского интерфейса приложения представлен на рисунке 1.

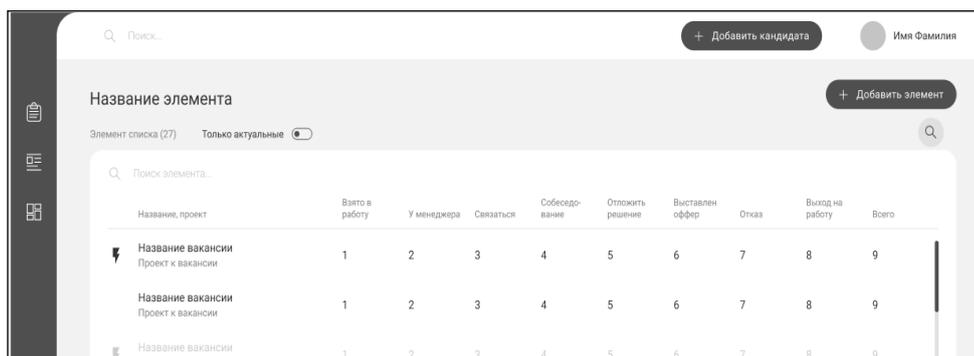


Рисунок 1 – UX-страницы «Вакансии»

Таким образом, приложение будет представлять собой систему для внутреннего пользования компании с разграничением прав (администратор, менеджер, рекрутер), которая централизованно хранит информацию о структуре компании, потенциальных сотрудниках, вакансиях, проектах. Программа имеет прикладное значение. Её внедрение поможет специалистам по найму персонала IT-компаний синхронизировать совместную работу, упростить подбор квалифицированных кадров.

Список использованных источников:

1. IT-рекрутинг: особенности работы и как подбирают специалистов в Google [Электронный ресурс]. – Режим доступа: <https://hurta.work/ru/blog/it-rekruting-osobennosti-raboty-i-kak-eto-delayut-v-google/>. – Дата доступа: 29.03.2021.

СИСТЕМА КОНТРОЛЯ ДОСТУПА С ПРОПУСКАМИ НА ОСНОВЕ ТОКЕНИЗИРОВАННЫХ EMV КАРТ

Ермолович И.А.

Белорусский государственный университет информатики и радиоэлектроники
г. Минск, Республика Беларусь

Шамына А.Ю. – магистр техн. наук, ассистент каф. ПОИТ

В работе описывается программное и аппаратное средство, которое способно эмулировать платежный терминал и считывать номера токенизированных EMV[1] карт (прим. Apple Pay, Samsung Pay, Google Pay). Программное средство позволяет вести базу пользователей с номерами их карт и контролировать возможность посещения клиентом организации. Программное средство создано для того, чтобы уменьшить расходы компаний на содержание программных средств и закупку персональных карт, уменьшить количество карт у клиентов, а также снизить количество пластикового мусора в мире.

Все существующие системы контроля доступа при детальном изучении довольно сильно переусложнены и неудобны. Они требуют выпуска отдельных карт, установки программ, обучения персонала, запоминания последовательности действий с мастер-картами. Такое программное средство подходит для крупных организаций, но такие решения избыточны для небольших компаний в сфере обслуживания, которые имеют один проходной пункт и не требуют разделения зон доступа.

Создаваемое программное средство должно быть просто в подключении, поддерживать разные операционные системы и не требовать специального обучения для персонала. В качестве платформы для программного средства было решено использовать веб-браузер.

Для связи программного и аппаратного средства был выбран стандарт WebUSB[2], который позволяет связывать веб-сайт с USB устройствами. В качестве аппаратного средства была выбрана Arduino Pro Micro, для чтения NFC карт был выбран микроконтроллер PN532, поскольку он способен отправлять на NFC карту APDU команды[3] для активации программ на EMV картах.

Все платежные инструменты работают по протоколу EMV. Бесконтактные EMV-карты позволяют без авторизации считать данные, такие как: имя владельца, номер карты, срок действия, лог транзакций по карте. PAN номер (номер карты) не изменяется на протяжении всего срока действия карты, поэтому он будет использован в качестве идентификатора пользователя.

Безопасность работы с номерами карт осуществляется самой платежной системой. Платежная система генерирует токен при добавлении виртуальной карты на устройство. Токен это такой же PAN номер, но он не соответствует реальному номеру карты, а является его эквивалентом. Запросы POS-терминалов по этому виртуальному номеру будут рассмотрены платежной системой, как норм физической карты. При таком подходе, даже потенциально скомпрометированный терминал не видит реального номера карты — токен можно использовать только для платежей на POS-терминалах.

Начало общения с EMV-картой всегда происходит с чтения файла PPSE[4] (Payment System Environment) командой чтения. В ответ на команду чтения карта должна вернуть FCI (File Control Information) со списком приложений, существующих на карте. В ответе FCI нам необходимо найти идентификатор платежного приложения (AID), который содержит информацию о платежной системе.

В ответ на запуск платежного приложения карта может затребовать от считывателя PDOL (Processing Options Data Object List). Это набор параметров POS-терминала — поддерживаемые протоколы и стандарты, валюта, дата, случайное число для криптографии, и т.д. Сложность состоит в том, некоторые карты могут отказаться работать без корректного ответа PDOL. Карта ожидает ответ на PDOL в том же порядке, в котором следуют запросы, и той длины, которая указана после каждого параметра PDOL. Так как наше средство не собирается списывать деньги, то его задача — сформировать самый простой ответ PDOL, который удовлетворит карту. Экспериментальным путем было получено, что почти на все запросы PDOL можно ответить нулями, кроме Terminal Transaction Qualifiers (TTQ).

После этого программное средство сравнивает номер карты с базой и в случае наличия записей производит проверку возможности доступа пользователя в организацию при этом сигнализируя оператору о статусе проверки и, в случае возникновения ошибки, описания возникших проблем.

Разработанное программное и аппаратное средство позволяет в течении 5 минут настроить систему доступа и не требует дополнительных затрат на установку и подключение, поскольку подключается в USB порт компьютера. Также разработанное средство способно сэкономить средства организации на обслуживании оборудования и закупке персональных карт для клиентов. При этом у самих клиентов уменьшается количество персональных карт.

Список использованных источников:

1. EMV Payment Tokenisation [Электронный ресурс]. – Режим доступа: <https://www.emvco.com/emv-technologies/payment-tokenisation/>. – Дата доступа: 05.04.2021.
2. WebUSB API [Электронный ресурс]. – Режим доступа: <https://wicg.github.io/webusb/>. – Дата доступа: 05.04.2021.
3. ISO/IEC 7816-4:2005 [Электронный ресурс]. – Режим доступа: <https://www.iso.org/36134.html> – Дата доступа: 5.4.2021.

МОДЕЛЬ МАШИННОГО ОБУЧЕНИЯ ДЛЯ СОРТИРОВКИ ТВЕРДОГО БЫТОВОГО МУСОРА

Калоша С.О.¹, магистрант

Белорусский государственный университет информатики и радиоэлектроники¹
г. Минск, Республика Беларусь

Нестеренков С.Н. – канд. техн. наук

Аннотация. В данной работе изложены результаты создания модели машинного обучения для решения задачи сортировки твердого бытового мусора, а также создание специализированного устройства для сбора данных для обучения такой модели.

Ключевые слова. Машинное обучение, глубокое обучение, сверточные нейросети, проблема переобучения, сортировка мусора, экономика замкнутого цикла.

Проблемы загрязнения окружающей среды и перехода к экономике замкнутого цикла [1] крайне актуальны в настоящее время. Решение этих проблем сложно представить без переработки мусора, неотъемлемой частью которой является его сортировка. В настоящее время основными подходами по сортировке мусора являются ручная сортировка бытового мусора и сложные сортировочные промышленные линии. Недостатком обоих подходов является низкая экономическая мотивированность населения, а также невозможность разделять мусор на большое количество категорий с более коротким циклом переработки.

Использование моделей машинного обучения для решения задачи классификации бытового мусора позволяет преодолеть эти ограничения. Исследования таких моделей ускорились с развитием более сложных архитектур нейронных сетей. В то же время предложенные на текущий день модели либо основаны на использовании подходов, разделяющих мусор на две категории [2,3], либо на использовании глубоких сверточных нейросетей обученных на больших массивах графических изображений, достигающих точности в 94%, но склонных к переобучению [4]. В то же время, если рассмотреть процесс сортировки мусора человеком и механизмы сложных промышленных линий, то можно сделать вывод, что основными факторами для принятия решения о классе объекта мусора являются его внешний вид, объем и масса. Создание модели машинного обучения, способной выделить и обучиться на таких факторах должно уменьшить склонность к переобучению, при этом увеличив точность классификации.

К сожалению, на данный момент времени не существует наборов данных включающих не только графические изображения объектов, но и величину их массы. Поэтому для сбора данных и дальнейшей проверки работоспособности полученных моделей, было принято решение о разработке прототипа устройства для сканирования объектов бытового мусора [5], чья схема изображена на рисунке 1.

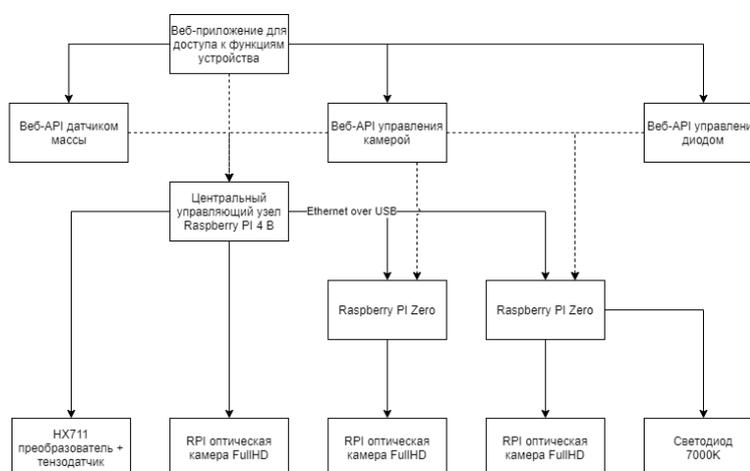


Рисунок 1. Схема устройства для сбора данных для обучения

В результате использования данного устройства удалось получить набор изображений четырехсот объектов мусора с разных ракурсов, а также значения массы для каждого из них. Анализ

собранных данных подтвердил предположение, что разделение объектов мусора на более широкие классы является не оптимальным в связи с сильно неравномерным распределением объектов мусора в каждой категории (рис. 2). В результате чего объекты были разделены на шестнадцать классов, включающих в себя пластиковые стаканчики, бумажные контейнеры, стеклянные бутылки, тетрапаки, но не ограниченные ими.

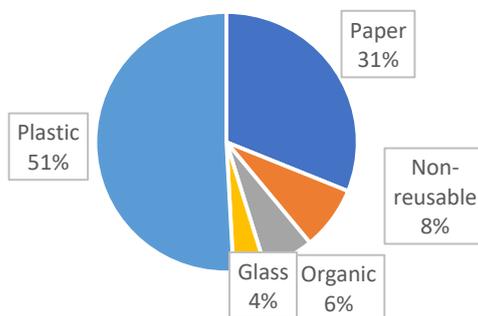


Рисунок 2. Распределение объектов мусора по широким классам: стекло, бумага, пластик, органические и не перерабатываемые отходы

В результате проверки различных моделей машинного обучения наилучшие результаты показала модель, основанная на архитектуре ResNet50 [6] и получающая на вход сразу два изображения и значение массы. Реализация модели была осуществлена с помощью библиотеки Tensorflow 2.4.1. Малый размер выборки удалось компенсировать за счет использования предварительно обученной модели ResNet50 на наборе изображений image-net, с последовательным обучением на собранных изображениях, расширенных изображениями из сети интернет, и до-обучением на собранных данных о массе. В результате удалось получить точность классификации в 95% на валидационной выборке и существенно снизить склонность к переобучению (рис. 3).

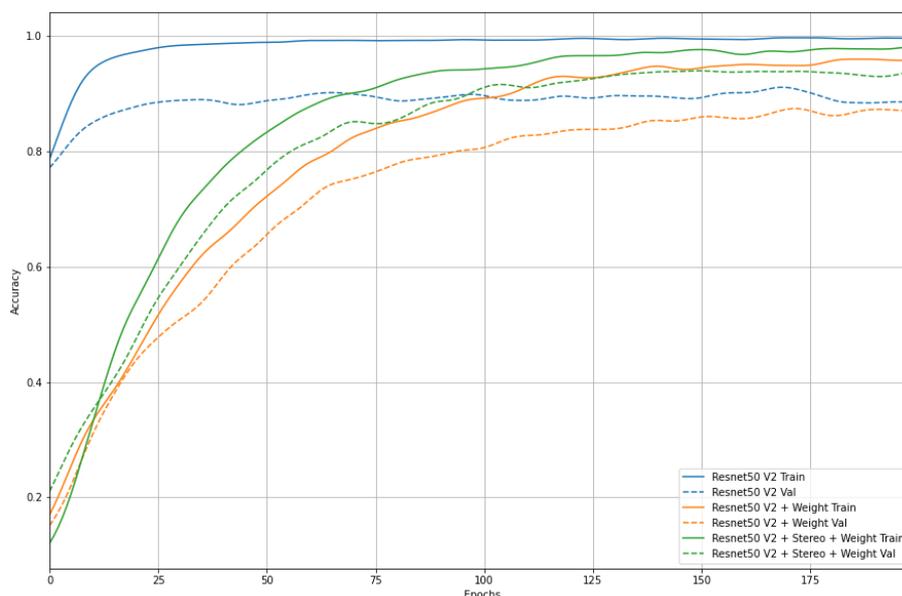


Рисунок 3. Сравнение графиков обучения различных моделей глубокого обучения в задачах сортировки мусора

Планируется использовать полученную в результате исследования модель в совокупности с устройством, разработанным для сбора данных для ее обучения, с целью создания прототипа устройства по автоматической сортировке мусора.

Список использованных источников:

1. Korhonen, J., Honkasalo, A., & Seppälä, J. (2018). Circular Economy: The Concept and its Limitations. *Ecological Economics*, 143, 37–46. doi:10.1016/j.ecolecon.2017.06.041
2. Sudha, S., Vidhyalakshmi, M., Pavithra, K., Sangeetha, K., & Swaathi, V. (2016). An automatic classification method for environment: Friendly waste segregation using deep learning. 2016 IEEE Technological Innovations in ICT for Agriculture and Rural Development (TIAR). doi:10.1109/tiar.2016.7801215
3. Dubey, S., Singh, P., Yadav, P., & Singh, K. K. (2020). Household Waste Management System Using IoT and Machine Learning. *Procedia Computer Science*, 167, 1950–1959. doi:10.1016/j.procs.2020.03.222

4. Srinilta, C., & Kanharattanachai, S. (2019). *Municipal Solid Waste Segregation with CNN*. 2019 5th International Conference on Engineering, Applied Sciences and Technology (ICEAST). doi:10.1109/iceast.2019.8802522
5. Калоша, С. О. *Технология Ethernet over USB как основа вычислительного кластера на Raspberry PI* / Калоша С. О., Нестеренков С. Н. // *Информационные технологии и системы 2020 (ИТС 2020) = Information Technologies and Systems 2020 (ITS 2020) : материалы международной научной конференции, Минск, 18 ноября 2020 г. / Белорусский государственный университет информатики и радиоэлектроники ; редкол. : Л. Ю. Шилин [и др.]*. – Минск, 2020. – С. 36–37.
6. He, K., Zhang, X., Ren, S., & Sun, J. (2016). *Deep Residual Learning for Image Recognition*. 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). doi:10.1109/cvpr.2016.90

ПРОГРАММНОЕ СРЕДСТВО МОНИТОРИНГА ФИЗИЧЕСКОГО РАЗВИТИЯ ДЛЯ МОБИЛЬНОЙ ПЛАТФОРМЫ ANDROID

Калтович В.А.

Белорусский государственный университет информатики и радиоэлектроники
г. Минск, Республика Беларусь

Парамонов А.И. – канд. техн. наук

Описывается программное средство, которое помогает пользователям в отслеживании показателей их физического развития. Приложение может использоваться в повседневной жизни и предоставляет пользователю актуальную информацию за счет применения анализа данных и автоматического расчёта различных нормативных показателей. Программное средство позволяет отслеживать и анализировать изменения показателей в динамике и отображает пользователю информацию в доступной и лёгкой для восприятия форме.

Мониторинг физического развития человека предполагает отслеживание его физических показателей и их последующий анализ с целью выявления отклонений. Разрабатывается программное средство, которое предоставляет такие возможности и отображает их в наглядной графической форме. Данное программное средство помогает пользователям в подсчёте значений, необходимых для поддержания здорового образа жизни и физической формы.

Составленная индивидуальная программа определяет начальный вектор движения и даёт понимание того, что нужно делать для достижения цели. Добавление и отслеживание целей, создание и просмотр тренировок позволяют пользователю держать всю необходимую информацию в одном приложении, что позволяет сэкономить время на переключении между приложениями и способствует сохранению мотивации пользователя.

Целевой аудиторией программы являются начинающие спортсмены, люди, которые следят за своим здоровьем и фигурой, а также люди, которые хотят достичь необходимой физической формы.

На данный момент рынок заполнен огромным количеством фитнес-приложений, однако каждое из них создано для отдельной цели, например: для ведения дневника питания или для создания и отслеживания тренировок, для подсчёта количества шагов или для отслеживания изменения параметров.

Данное программное средство является мобильным приложением для платформы Android [1], которое разработано на языке программирования Kotlin [2].

Интерфейс программного средства содержит 8 основных экранов: экран для входа в приложение; 2 экрана регистрации - экран для создания данных для входа и экран для внесения данных о физических параметрах; стартовый экран – содержит информацию о количестве шагов, пройденных пользователем за день, количество необходимых калорий и питательных элементов; экран прогресса – содержит графики с отображением динамики прогресса пользователя; экран целей – содержит список целей пользователя и информацию об их статусе; экран тренировок – содержит информацию о тренировках пользователя и позволяет создавать новые на основе упражнений, хранящихся в базе данных; экран профиль - содержит данные пользователя и позволяет изменять их.

После внесения пользователем данных о его физическом состоянии, программа по этим параметрам производит расчёт таких показателей, как: Индекс массы тела (далее ИМТ), базовый расход калорий, потребность в питательных веществах, суточная норма воды [3].

ИМТ рассчитывается по формуле (1).

$$I = \frac{m}{h^2} \quad (1)$$

где m - масса человека в килограммах, h - рост человека в метрах.

Уравнение Миффлина для расчёта базового расхода калорий представлено формулой (2).

$$K = (10 * m) + (6.25 * h) - (5 * l) + Q1 \quad (2)$$

где m – масса человека в килограммах, h - рост человека в метрах, l - возраст человека в годах, $Q1$ – гендерный коэффициент (равен 5 для мужчин и -161 для женщин).

Для получения количества калорий, необходимых пользователю в зависимости от образа жизни, результаты, полученные с использованием формул 2, 3, следует умножить на коэффициент активности, выбранный пользователем по таблице 1.

Для расчёта необходимых питательных элементов (белки, жиры и углеводы) используется формула (3).

Таблица 1 – Коэффициенты активности

Коэффициент активности	Описание
1,2	Минимальные нагрузки (сидячая работа)
1,375	Немного дневной активности и легкие тренировки 1-3 раза в неделю
1,4625	Тренировки 4-5 раз в неделю или работа средней тяжести
1,550	Интенсивные тренировки 4-5 раз в неделю
1,6375	Ежедневные тренировки
1,725	Ежедневные интенсивные тренировки или тренировки 2 раза в день
1.9	Тяжелая физическая работа или сверхинтенсивные тренировки 2 раза в день

$$P = \frac{n * k}{Q2} \quad (3)$$

где n - суточная норма калорий человека, k - процент белка от суточной потребности в питательных элементах, Q2 – коэффициент элементов, который равен 4 для белков, 4 для углеводов, 9 для жиров.

Для расчёта суточной нормы воды используется формула (4).

$$W = m * Q3 \quad (4)$$

где m - масса человека в килограммах, Q3 – коэффициент (равен 31 для женщин и 35 для мужчин).

Для оценки ИМТ используется таблица 2, содержащая соответствие значений ИМТ и его интерпретации.

Таблица 2 – Значение ИМТ

Индекс массы тела	Интерпретация
от 15,99 и менее	Выраженный дефицит массы тела
от 16,0 до 18,49	Недостаточная масса тела
от 18,5 до 24,99	Нормальная масса тела
от 25,0 до 29,99	Избыточная масса тела
от 30,0 до 34,99	Ожирение первой степени
от 35,0 до 39,99	Ожирение второй степени
от 40 и более	Ожирение третьей степени

Пользователю доступен функционал для создания тренировок на основании упражнений, хранящихся в базе данных. Каждому упражнению соответствует количество затрачиваемых килокалорий на 50 кг за 1 час.

Для получения количества калорий, затраченных на выполнение упражнений, используется формула (5).

$$K = a * \frac{m}{50} * \frac{t}{60} \quad (5)$$

где a - количество затрачиваемых килокалорий на 50 кг за 1 час, m - масса человека в килограммах, t – время, затраченное на выполнение упражнения.

Разработанное программное средство автоматически рассчитывает приведённые показатели, анализирует их и выдаёт пользователю на основании результатов советы, необходимые для начала движения в сторону намеченной цели, что дает возможность в процессе занятий отследить правильность действий. Благодаря данной программе пользователю не придётся разбираться самостоятельно в большом количестве информации, необходимой для того, чтобы начать работу над своей физической формой.

Список использованных источников:

1. Что такое Android [Электронный ресурс]. – Режим доступа: https://www.android.com/intl/ru_ru/what-is-android/. – Дата доступа: 02.04.2021.
2. Open Source Community. Руководство по языку Kotlin. [Электронный ресурс]. – Режим доступа: <https://kotlinlang.ru/>. – Дата доступа: 02.04.2021.
3. Юлианна Плискина: Счетчик калорий. Основы правильного питания / Издательство: Эксмо-Пресс, 2017. – 192 с.

НАБОР СРЕДСТВ ДЛЯ УДАЛЁННОГО УПРАВЛЕНИЯ МИКРОКОНТРОЛЛЕРАМИ ARDUINO

Касперский И.В.

*Белорусский государственный университет информатики и радиоэлектроники
г. Минск, Республика Беларусь*

Шамына А.Ю. – м.т.н., ассистент кафедры ПОИТ

Данная работа предоставляет набор средств для удалённого управления микроконтроллерами arduino. Сама работа заключается в создании программных средств для описанной задачи, а представленный продукт - лишь пример их использования. Конечный продукт состоит из 3-х программ, каждая из которых изменяется под необходимую задачу. Проект легко масштабируется для большого числа управляющих программ и микроконтроллеров.

Часто возникает необходимость удалённого управления системами на базе микроконтроллеров. Из ближайших к повседневности бытовых примеров использования можно выделить системы типа „умный дом“. Целью данной работы является разработка набора средств для создания любых систем на базе микроконтроллеров arduino.

Разработка управляющих программ для микроконтроллеров arduino имеет ряд проблем, решение которых предоставляет данная работа. Первая проблема — жёсткие аппаратные ограничения. Они заключаются в малом объёме оперативной памяти, малом объёме памяти программ и сравнительно невысоком быстродействии. К примеру, используемая в демонстрации плата Arduino UNO построена на базе микроконтроллера ATmega328. Он характеризуется 32кб памяти программ, 2кб RAM и тактовой частотой 20МГц. Данных возможностей не хватает для оперативной работы требовательных к аппаратным ресурсам программ. К примеру программ, работающих в режиме реального времени. Второй проблемой разработки управляющих программ для микроконтроллеров arduino является структура программы, работающей на устройстве. Программа может состоять только из 1 модуля и не имеет возможности подключения сторонних модулей. Также полноценная работа с потоками отсутствует, а присутствует лишь возможность создавать псевдопараллельные потоки с использованием библиотеки ArduinoThread[5]. Все эти особенности сильно ограничивают возможности программы, которая выполняется на микроконтроллере. Следующим аспектом, усложняющим разработку, является сложность передачи команд извне. Приходящие управляющие сигналы могут передаваться либо по COM порту, либо через модуль, способный принимать сигналы по различным интерфейсам, к примеру ethernet. Но такие решения занимают порты, требуют дополнительные аппаратные средства. Использование отдельного сервера позволяет централизовать управление системами и снимает вычислительную нагрузку с микроконтроллеров. Последним недостатком является строгая зависимость вшиваемой в arduino программы от языка программирования. При необходимости работы с управляющими приложениями приходится разрабатывать API взаимодействия. Предоставленное решение избавляет разработчика от данных проблем и предоставляет готовый набор средств, предоставляющий универсальный API взаимодействия для легко масштабируемых систем. Адаптируя данный набор средств, проектировщики подобных систем могут существенно сократить время разработки и при создании приложений использовать удобный, а главное не зависящий от языка программирования, интерфейс.

Конечный продукт можно рассматривать как 3 отдельных модуля: физической системы, управляющего сервера и управляющего приложения. Взаимодействие этих модулей между собой происходит с помощью специализированного API. На рисунке 1 приведена схема коммуникации модулей между собой, а также средства и интерфейсы, используемые для этого.

Первый модуль состоит из физической системы, управляемой микроконтроллерами и программами, которые на них запущены. Система легко масштабируется от одного до нескольких микроконтроллеров. Программа, запускаемая на них, написана на диалекте языка программирования C, одинакова для всех микроконтроллеров и не изменяется в зависимости от задачи, поставленной перед системой. Коммуникация с сервером происходит по COM порту, но может быть изменено под другой интерфейс при необходимости.

Второй модуль является сервером-посредником между физической системой и управляющим приложением. Взаимодействие с ними осуществляется по соответствующим протоколам. Протокол взаимодействия с физической системой и протокол взаимодействия с управляющим приложением различаются. Серверное приложение написано на языке программирования java и почти не зависит от выполняемых задач. Для его настройки под конкретную задачу необходимо и достаточно определить соответствие управляющих приложений и физических систем.

Третий модуль является приложением, которое удалённо управляет физической системой. Согласно протоколу оно может отправлять запросы трёх типов: установка режима порта ввода/вывода, установка значения напряжения на порт и взятие значения напряжения с порта. Одновременно к серверу может быть подключено несколько управляющих приложений. Соединение

с сервером может происходить как по локальной сети, так и через глобальную сеть интернет. Коммуникация происходит по протоколу TCP, обеспечивающему надёжную доставку пакетов. Данный модуль является платформонезависимым и может быть написан на любом языке программирования. Программная реализация соединения подразумевает использование сокетов.

При демонстрации будет представлен пример использования разработанных средств. Физический модуль будет контролироваться одним микроконтроллером arduino uno. Модуль сервера будет выведен в глобальную сеть с помощью подключения через VPN к модему, на котором запущен Dynamic DNS сервис. Таким образом пользователь извне сможет подключиться к серверу через постоянное доменное имя. Управляющий модуль будет в единственном экземпляре представлен приложением под android написанным на java. Интерфейс приложения представляет собой изображение микроконтроллера, вид сверху.

У пользователя есть возможность контролировать состояния портов микроконтроллера. Для начала работы с портом необходимо выбрать его режим работы. На выбор доступно 3 режима: INPUT, OUTPUT и INPUT_PULLUP. Режим INPUT используется для получения значения напряжения на портах. Режим OUTPUT используется для установки выходного значения напряжения на порту. Режим INPUT_PULLUP схож по возможностям с режимом INPUT. Его особенностью является активация подтяжки напряжения к 5В. Данная возможность используется для подключения датчиков, например кнопок. После установки режима можно соответственно установить или взять значение напряжения на данном порту.



Рисунок 1 – Схема взаимодействия модулей.

Микроконтроллеры семейства Arduino были выбраны не случайно. Они более распространены, чем их более продвинутые аналоги: ESP32, STM32 и прочие. Как следствие, данная разработка позволит охватить большее количество проектов и упростить их разработку. Стоит заметить, что благодаря модульной архитектуре финального набора средств, для переноса физической системы на базу одного из вышеперечисленных микроконтроллеров необходимо изменить только 1 модуль из 3.

В процессе разработки были изучены различные сферы разработки. Первой из них является разработка приложений на базе микроконтроллеров arduino. Работающие на них программы пишутся на диалекте языка C. Его особенности описаны выше. Второй изученной сферой является сетевое взаимодействие с arduino по COM порту. Для передачи данных в COM порт со стороны сервера использовались библиотека Java-Arduino Communication Library. Эта небольшая библиотека позволяет подключаться к COM порту arduino, помещать и читать из него данные. Третьим рассмотренным аспектом разработки стала передача данных по сети между сервером и управляющим приложением. Пакеты на сервер передаются через локальную сеть или сеть интернет по протоколу TCP, используя динамический адрес. Четвёртой темой является разработка приложений под android. На момент начала работы над проектом имелся некоторый опыт, однако в процессе разработки навыки и знания в данной теме были углублены.

Таким образом результатом работы стал набор средств, существенно облегчающий и ускоряющий процесс разработки физических систем, управляемых дистанционно. Данная разработка решает ряд вышеописанных проблем и ограничений, легко масштабируется и адаптируется под поставленную перед проектировщиком задачу.

Список использованных источников:

1. Раздел обучения на официальном сайте arduino [Электронный ресурс]. – Режим доступа: <https://www.arduino.cc/en/Tutorial/HomePage> – Дата доступа: 06.03.2021.
2. JDK 16 Documentation [Электронный ресурс]. – Режим доступа: <https://docs.oracle.com/en/java/javase/16/> – Дата доступа: 13.03.2021.
3. Start Android - учебник по Android для начинающих и продвинутых [Электронный ресурс]. – Режим доступа: <https://startandroid.ru/ru/> – Дата доступа: 13.03.2021.
4. Работа с COM-портом Arduino из Java-приложения [Электронный ресурс]. – Режим доступа: <https://habr.com/ru/post/340630/> – Дата доступа: 06.03.2021.
5. Блог об ардуино электронике и ИТ [Электронный ресурс]. – Режим доступа: <https://soltau.ru/index.php/arduino/item/373-kak-vypolnyat-parallelnye-zadachi-threads-v-programme-dlya-arduino> – Дата доступа: 03.04.2021.

АНАЛИЗ МНОГОМЕРНЫХ ДАННЫХ НА ПРИМЕРЕ ДАННЫХ ПО ОНКОЛОГИЧЕСКИМ ЗАБОЛЕВАНИЯМ ЛЁГКИХ: ПОДБОР ИНФОРМАТИВНЫХ ПРИЗНАКОВ

Корховая А.Б.

Белорусский государственный университет информатики и радиоэлектроники
г. Минск, Республика Беларусь

Абрамович М.С. – зав. НИЛ статистического анализа и моделирования

В данной работе был рассмотрен и применен критерий согласия Пирсона χ^2 для отбора информативных признаков выборки данных. Также был произведен выбор алгоритма для дальнейшей работы в рамках поставленной задачи.

По результатам наблюдений за онкологическими больными формируются выборки данных содержащие большое количество не всегда информативных признаков. Для проведения эксперимента была предоставлена выборка данных по Минскому району, которая включает в себя 380 пациентов и более 100 характеристик. Однако для такого объема данных количество признаков несоизмеримо высоко. На первом этапе работы с данными было принято решение не использовать показатели генов. И далее при помощи эксперта были отобраны 11 параметров для которых будут производиться применения алгоритмов [1]. Следующим шагом в моей работе стал отбор информативных признаков при помощи математических алгоритмов.

Для отбора информативных признаков был выбран критерий согласия Пирсона χ^2 . Ведь это один из самых популярных статистических критериев для анализа качественных данных (номинальных, порядковых, ранговых), анализа частот.

Критерий согласия Пирсона применяют для проверки гипотезы о соответствии эмпирического распределения предполагаемому теоретическому распределению $F(x)$ при большом объеме выборки (число записей в выборке ≥ 100). Критерий применим для любых видов функции $F(x)$, даже при неизвестных значениях их параметров, что обычно имеет место при анализе результатов механических испытаний. В этом заключается его универсальность.

Также данный алгоритм используется для [2]:

- для оценки статистической значимости различий двух или нескольких относительных показателей;
- для проверки гипотезы (нулевой), что наблюдаемая случайная величина подчиняется некому теоретическому закону распределения;
- для анализа категориальных данных, т.е. таких, которые выражаются не количеством, а принадлежностью к какой-то категории. К таким данным нельзя применять математические операции вроде сложения и умножения, для них можно только подсчитать частоты.

В таблице 1 приведены значения статистики хи-квадрат и соответствующие p -значения признаков. В качестве информативных признаков отбирались те, у которых p -значения меньше 0.1.

Таблица 1 – Информативные признаки

Признак	Статистика хи-квадрат	p -значение
Лучевая терапия	41.90860	0.000000
Химио-терапия	25.75409	0.000000
Операция (да; нет)	25.65689	0.000000
Жалобы (шифр)	12.82626	0.000342
Гист. РЛ (шифр)	18.90845	0.001999
Статус курения	7.16375	0.007439
Возраст	64.35080	0.024284
Стадия	8.86487	0.031142

Признаки упорядочены по p -значению. Чем меньше p -значение, тем признак более информативен.

Таким образом из одиннадцати признаков, выбранных экспертом, для дальнейшей работы были отобраны 8 информативных признаков. Которые в дальнейшем будут использоваться в алгоритме дерева решений. Деревья решений в свою очередь рекомендуется использовать в условиях выборок ограниченного объема [3], что и имеет место в рассматриваемом случае.

Список использованных источников:

1. Корховая А.Б. ИСПОЛЬЗОВАНИЕ АЛГОРИТМА ДЕРЕВА РЕШЕНИЙ ДЛЯ АНАЛИЗА МНОГОМЕРНЫХ ДАННЫХ НА ПРИМЕРЕ ДАННЫХ ПО ОНКОЛОГИЧЕСКИМ ЗАБОЛЕВАНИЯМ ЛЁГКИХ. — 21–24 апреля 2020 года, Минск, БГУИР.
2. Никулин М. С. О критерии хи-квадрат для непрерывных распределений. — 1973
3. Harrington, P. *Machine Learning in Action* / P. Harrington. New York: Manning, 2012

ПРИМЕНЕНИЕ ИСКУССТВЕННОЙ НЕЙРОННОЙ СЕТИ ДЛЯ ОПРЕДЕЛЕНИЯ ТИПА МЕСТНОСТИ

Красковский П.Н.

Белорусский государственный университет информатики и радиоэлектроники
г. Минск, Республика Беларусь

Серебряная Л.В. – канд. техн. наук

Физические карты позволяют визуализировать рельеф местности, используя различные цветовые обозначения для определённых типов ландшафта. Рельеф можно представить более наглядно, если использовать дополненную реальность для наложения трёхмерной модели ландшафта на реальную физическую карту. Определение высоты местности по цвету карты является одной из задач, необходимой для построения объёмного ландшафта.

Как правило, физические карты используют градиентную шкалу, которая обозначает высоту или глубину местности, отображаемой на карте. На рисунке 1 представлен пример такой шкалы. Градиентная шкала содержит несколько опорных точек, каждая из которых задаёт соответствие между цветом и определённой высотой. Например, зелёным цветом обозначаются равнины, жёлтым — возвышенности и холмы, коричневым — горы, а синим цветом — водные объекты [1]. Цвета, которые расположены между опорными точками градиента, задают промежуточные значения высоты и глубины местности.

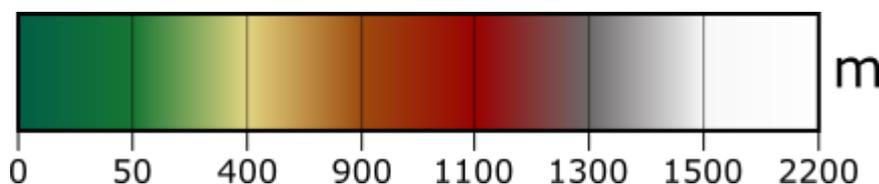


Рисунок 1 — Градиентная шкала высот

Для решения задачи определения высоты местности по цвету на карте было принято решение использовать искусственную нейронную сеть, структура которой показана на рисунке 2. Входной слой сети состоит из трёх нейронов, каждый из которых принимает значение R, G или B компоненты входного цвета. Выходной слой имеет четыре нейрона, по значениям которых определяется тип ландшафта и его высота. Каждому типу ландшафта ставится в соответствие определённая высота. При подаче на вход сети цвета, который соответствует одной из опорных точек градиентной шкалы, один из выходных нейронов сети должен выдавать значение, близкое к единице, а все остальные нейроны — близкое к нулю. Для получения промежуточной высоты между двумя опорными точками, необходимо интерполировать значения высот опорных точек, используя коэффициент интерполяции, полученный на выходе нейронной сети. Данный коэффициент принимает значения в диапазоне от 0 до 1 и показывает, насколько входной цвет близок к опорной точке градиента.

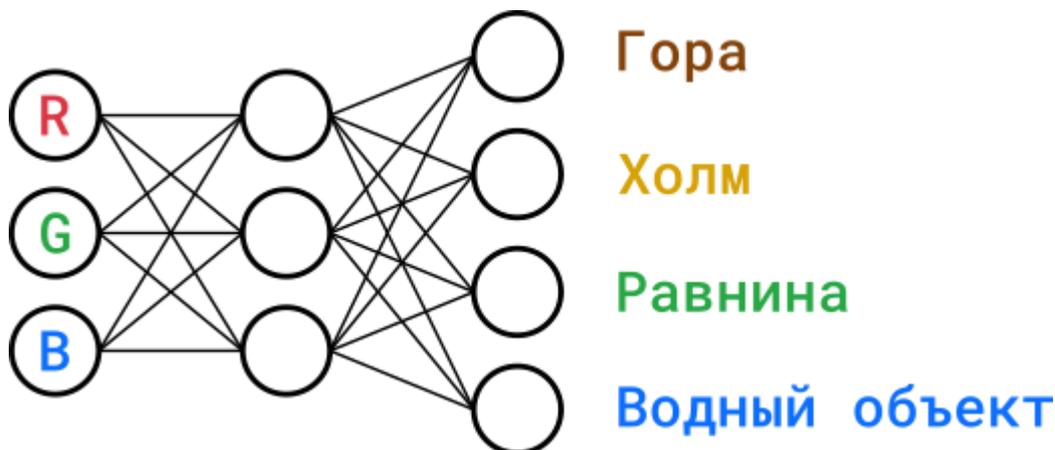


Рисунок 2 — Искусственная нейронная сеть для определения высоты местности

Список использованных источников:

1. The Role of Colors on Maps [Electronic resource]. – Mode of access: <https://www.thoughtco.com/colors-on-maps-1435690> – Date of access: 03.04.2021.
2. Осовский, С. Нейронные сети для обработки информации / Пер. с польского И.Д. Рудинского. – М.: Финансы и статистика, 2002. – 344 с.

АВТОМАТИЗИРОВАННАЯ СИСТЕМА УПРАВЛЕНИЯ РЕСТОРАНОМ

Крукович П.Н.

*Белорусский государственный университет информатики и радиоэлектроники
г. Минск, Республика Беларусь*

Видничук В.Н. – м. т. н., старший пр. каф. ПОИТ

В докладе рассматривается тема автоматизации бизнес-процессов на предприятиях общественного питания.

С ростом численности ресторанов стремительно усиливается и конкуренция, что неизбежно приводит к необходимости эффективно и рационально использовать имеющиеся ресурсы. В этих условиях для успешного ведения бизнеса необходимо инвестировать в средства и инструменты его поддержания и развития. Один из основных инструментов развития ресторанного бизнеса – это современная система автоматизации ресторанов.

Современная система автоматизации ресторана – это профессиональная система управления рестораном, многофункциональная и легко модернизируемая. Целью автоматизации является повышение эффективности управления рестораном, ускорение обслуживания и минимизация возможных злоупотреблений, особенно воровства. Значительная доля успеха складывается из отличного сервиса и оперативной работы персонала. Именно возможности автоматизации ресторана позволяют оптимально сочетать скорость и качество.

Очевидны неоспоримые преимущества автоматизированного ресторана перед другими подобными заведениями: высокое качество сервиса и скорость обслуживания клиентов, отсутствие ошибок при оформлении заказа, обработка и передача заказа в автоматическом режиме, абсолютный контроль всех процессов от момента приема заказа до его исполнения, возможность непрерывно отслеживать финансовые результаты работы заведения.

Данная работа подразумевает создание системы, в которой будут восполнены недостатки существующих систем. Основная цель проекта заключается в том, чтобы включить в систему необходимые функции для обеспечения эффективного управления рестораном.

Основные функции автоматизированной системы управления рестораном:

- складской учет и бухгалтерский учет – автоматизация складских операций: приход и расход продуктов, полуфабрикатов, инвентаризация, списание ингредиентов при приготовлении блюд для кухни и бара;
- управление персоналом – учет рабочего времени, создание расписаний, контроль количества выписанных чеков;
- контроль деятельности заведения – бронирование столиков, создание планов помещений;
- создание программ лояльности и скидок – создание групп клиентов и специальных предложений;
- формирование меню, контроль ценообразования – формирование технических карт блюд, расчет себестоимости и прибыли от продажи блюд;
- планирование и прогнозирование – генерация отчетов за выбранный период, анализ полученных данных.

Применение автоматизированной системы позволит решать корпоративные задачи, такие как развитие предприятия и обеспечение непрерывности функционирования, а также экономические задачи, такие как максимизация товарооборота и конечной прибыли, обеспечение ликвидности и рентабельности предприятия, минимизация текущих издержек за счет роста эффективности работы. Ожидаемые результаты использования автоматизированной системы:

- Повышение прибыльности и снижение издержек предприятия;
- Контроль и оптимизация деятельности предприятия;
- Улучшение качества обслуживания посетителей;
- Предотвращение хищений и прочих злоупотреблений;
- Увеличение производительности труда персонала;
- Поддержка маркетинговых мероприятий.

Автоматизированная система состоит из нескольких модулей, включающих функционал, предназначенный для отдельных типов сотрудников. Для доступа к каждому из модулей сотрудник должен обладать определенным набором прав. Каждый модуль представлен отдельным веб-приложением. На веб-сервере происходит обработка данных, выполняются аналитические операции и формируются отчеты, выполняются операции по работе с базой данных. Автоматизированная система реализована с использованием облачных технологий, что делает ее еще более мобильной и эффективной в использовании. Схематическое представление архитектуры системы приведено на рисунке 1.

Модуль управления рестораном содержит функционал, необходимый для менеджеров и руководителей. Складской учет позволяет контролировать остатки, своевременно списывать товары, производить инвентаризацию, работать с технологическими картами, вести расчет калорийности блюд и продуктов. Подмодуль управления персоналом позволяет отслеживать и корректировать работу сотрудников. Также данный модуль предоставляет возможность создания различных программ лояльности и скидков, объединения клиентов в группы и формирования иерархии перехода клиентов из одной группы в другую.

Модуль кухни и бара используется официантами и поварами. Данный модуль помогает уменьшить количество времени, затрачиваемое на принятие заказа и обслуживание столика, а также увеличить скорость работы персонала и оборачиваемость столов. Для кухни разработан подмодуль визуализации заказов. Как только заказ был подтвержден в модуле точки продаж, уведомление автоматически появляется на веб-сайте данного модуля.

Модуль точки продаж позволяет официантам, барменам и кассирам создавать, редактировать, отменять и оплачивать заказы, разделять чеки, отслеживать степень готовности заказанных блюд, добавлять в заказ блюда отдельно для каждого гостя за столиком. Также в нем есть возможность производить бронирование столиков, просматривать расписание бронирования и получать уведомления о приближении времени брони.

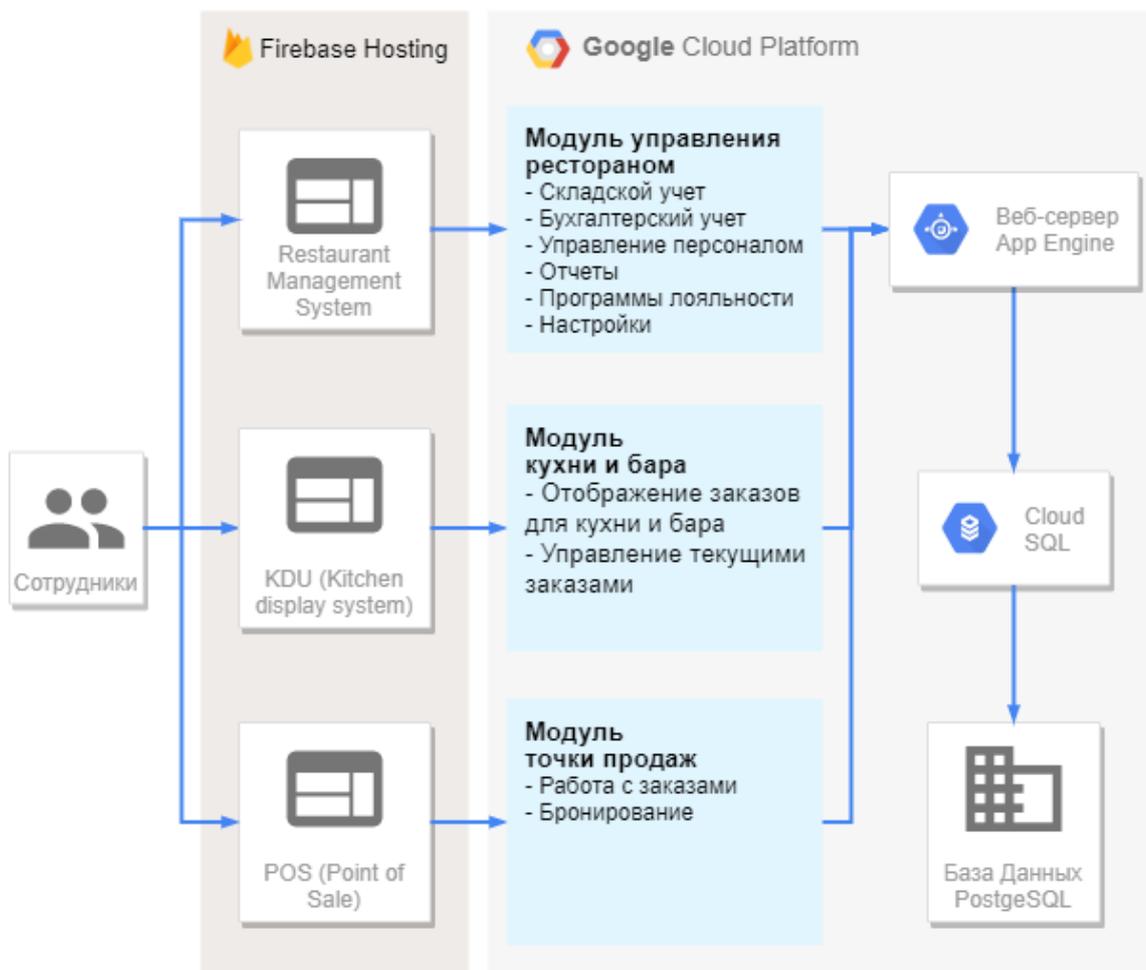


Рисунок 1 – Архитектура системы управления рестораном

Данная автоматизированная система управления рестораном позволит пользователям решать корпоративные задачи, такие как развитие предприятия и обеспечение непрерывности функционирования, а также экономические задачи, такие как максимизация товарооборота и конечной прибыли, обеспечение ликвидности и рентабельности предприятия, минимизация текущих издержек за счет роста эффективности работы эффективным решением проблем.

Список использованных источников:

1. Автоматизация складского учета [Электронный ресурс]. – Режим доступа: <https://www.datakrat.ru/software/keeper/avtomatizatsiya-ucheta>.
2. Бухгалтерский учет в общепите [Электронный ресурс]. – Режим доступа: <https://agentura-soft.ru/articles/AccountFood/>.
3. Норенков, И. Автоматизированные информационные системы: учеб. пособие / И. Норенков. – Москва: СПб.: Питер, 2011. – 526 с.

АЛГОРИТМ ПЕРЕМЕЩЕНИЯ РОБОТА “ПАУКА”

Кувшинов А.А.

Белорусский государственный университет информатики и радиоэлектроники
г. Минск, Республика Беларусь

Видничук В.Н. – м.т.н

В данной работе последовательно описываются этапы разработки алгоритма перемещения уже спроектированного робота “паука”, названного так по причине наличия шести подвижных конечностей, а также ввиду моделирования им процесса перемещения, присущего данному классу членистоногих.

Создание автоматических систем, способных облегчить труд человека и которые могут быть использованы в условиях, опасных для человека – является актуальной научной и технической задачей. Одним из важных классов таких систем является класс шагающих роботов, предназначенных для перемещения по труднопроходимой поверхности, порой, с заранее неизвестным рельефом [1]. Шагающий аппарат при движении использует для опоры лишь некоторые ограниченные участки поверхности на поверхности в отличие от колесных и гусеничных машин, имеющих непрерывную колею.

Анализируя существующие виды движителей, можно заметить, что нет ничего более совершенного, чем природные системы, по причине чего система, имитирующая модель перемещения, совершенствующуюся в процессе эволюции может быть наиболее оптимальной с точки зрения энергетических затрат и эффективности самого перемещения. Воплощением данной системы может служить робот, имитирующий перемещение паука. Сам робот имеет определенную структуру. Конструкция робота продемонстрирована на рисунке 1. В наиболее упрощенном виде, он состоит из несущего каркаса, элементов управления (спроектированной системной платы, управляемой одноплатным миникомпьютером *raspberrypi*) и питания, а также шести подвижных конечностей, осуществляющих сам процесс перемещения. Каждая из конечностей, в свою очередь, состоит из трех сегментов и трех подвижных сервоприводов.

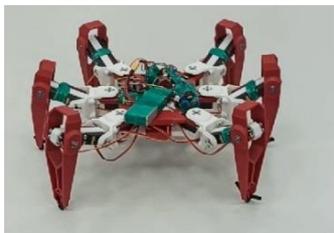


Рисунок 1 – Изображение робота “паука”



Рисунок 2 – Изображение сервопривода

Наиболее важную роль для самого перемещения, а также непосредственно для написания алгоритма, обеспечивающего автоматизацию данного процесса, играют сервоприводы, расположенные на стыках сегментов конечностей робота, а также в местах скрепления этих самых конечностей к корпусу (рисунок 2). С помощью специально написанной низкоуровневой библиотеки каждому сервоприводу по его специальному идентификатору PIN можно передать значение угла, на который он должен повернуться в текущий момент времени, что предоставляет интерфейс управления самим роботом. Поскольку данный робот работает на *raspberrypi* с легковесной и многофункциональной операционной системой Linux, на нем можно сохранить, скомпилировать и запустить код, который, посредством передачи значений сервоприводам, смог бы привести всю систему в движение.

С этого момента, имея интерфейс управления сервоприводами, задача написания алгоритма для перемещения робота сводится к геометрической по расчету соответствующих углов поворота. Также немаловажным условием является тот факт, что траектория поворота ног (непосредственно осуществляющих перемещение) должна быть параллельна корпусу, для повышения эффективности перемещения и исключения вращательных движений корпуса. Исходя из всего вышеперечисленного, задачу можно разбить на три последовательных этапа: расчет углов, на которые необходимо сдвинуть сервоприводы, чтобы проекция ноги на горизонтальную плоскость имела определенную длину, итеративный расчет каждой промежуточной длины для обеспечения прямолинейной траектории, а также применение алгоритма итеративного расчета длин для всех ног в синхронном порядке.

На первом этапе рассматривается лишь перемещение в вертикальной плоскости, а значит результатом выполнения алгоритма на данном этапе должны быть два рассчитанных угла α и β , которые будут переданы сервоприводам для того, чтобы проекция ноги на горизонтальную плоскость

имела определенную длину. В целом система будет выглядеть следующим образом (рисунок 3). Имеются константные значения длин сегментов ноги (а и b), значение высоты h, которое в данных обстоятельствах тоже можно рассматривать как константу, значение длины проекции l, которое будет входным параметром для алгоритма, а также непосредственно сами углы α и β , на которые надо сдвинуть сервоприводы, чтобы получить нужную проекцию l. Расчет идет по формуле (1).

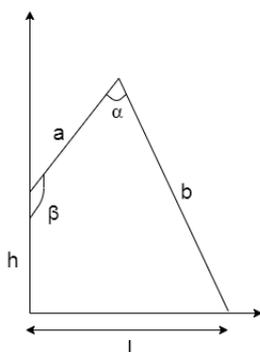


Рисунок 3 – Первый этап

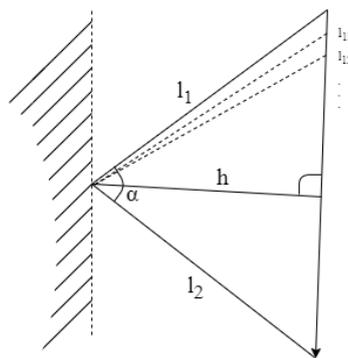


Рисунок 4 – Второй этап

$$p = \frac{a+b+\sqrt{h^2+l^2}}{2} \quad S = \sqrt{p*(p-a)*(p-b)*(p-\sqrt{h^2+l^2})} \quad \beta = \frac{\arcsin(2*S)}{a*b} \quad \alpha = 270 - \beta - \arcsin\left(\frac{2*S}{b*\sqrt{h^2+l^2}}\right) + \arctan\left(\frac{h}{l}\right) \quad (1)$$

Входными данными для алгоритма на втором этапе будут начальная и конечная длина проекции, а также угол, задающий поворот ноги в горизонтальной плоскости. Целью алгоритма будет являться итерационный расчет длины проекции для каждого градуса поворота в горизонтальной плоскости таким образом, чтобы траектория перемещения точки соприкосновения ноги с поверхностью была линией, параллельной корпусу. В свою очередь каждая из полученных итерационных длин проекции будет передана в качестве аргумента алгоритму, написанному на первом этапе, после чего сервоприводам при каждом изменении горизонтального угла на градус, будут переданы значения углов, на которые им необходимо повернуться для того, чтобы подстроить текущую длину под общую прямолинейную траекторию. Геометрически данная задача проиллюстрирована на рисунке 2. Расчет идет по формуле (2). В первую очередь рассчитывается высота до вектора перемещения. Затем рассматриваются разные случаи, зависящие от того, острый ли угол α . После чего расчет сводится к итерационному подсчету гипотенуз двух прямоугольных треугольников, формирующих, или дополняющих исходный треугольник, сформированный начальной и конечными длинами, а также вектором перемещения.

$$h = \frac{\sin(\alpha * l_1 * l_2)}{\sqrt{l_1^2 + l_2^2 - 2 * l_1 * l_2 * \cos(\alpha)}} \quad (2)$$

На третьем этапе в применение идут алгоритмы, полученные на двух предыдущих этапах. Алгоритмы применяются из расчета на каждую ногу с учетом граничных углов (пределное значение поворота сервопривода превышение которого может его повредить), которые будут индивидуальны для каждой ноги, в зависимости от их пространственного местоположения. Перемещение будет происходить потактово (рисунок 5). Одновременно будут подняты и опущены по три ноги.

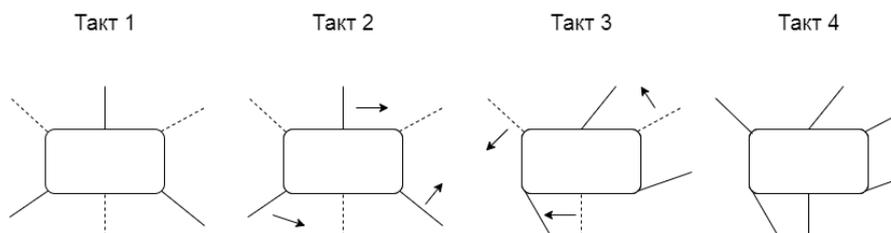


Рисунок 5 – Третий этап

Список использованных источников:

1. Разработка конструкции и алгоритмов управления движением шагающего аппарата для технического обслуживания авиационных комплексов / Н.В. Гарееская, В.В. Полянский, А.М. Сабадырь, А.С. Семцов - Электронный журнал «Труды МАИ». Выпуск № 62 [Электронный ресурс] – Режим доступа: <http://trudymai.ru>
2. Расчет параметров алгоритма движения ног для шестиногого робота-паука / Сабанова, Б.К, Иванова, И.В. [Электронный ресурс] – Режим доступа: <https://www.elibrary.ru>

О ПРОБЛЕМЕ ОПРЕДЕЛЕНИЯ ЭМОТИВНОЙ ОЦЕНКИ ТЕКСТА

Куличок Н.С.

Белорусский государственный университет информатики и радиоэлектроники
г. Минск, Республика Беларусь

Парамонов А.И. – канд. техн. наук, доцент

В данной работе рассматриваются вопросы анализа эмотивности цифровых текстов на естественном языке. Представлены основные понятия, обозначены проблемы и описаны известные пути их решения.

Постоянный рост объемов текстовой информации в цифровом виде приводит к возникновению потребности в новых инструментах для ее анализа. На сегодня достаточно эффективно решаются задачи классификации текстов и автоматического синтаксического анализа. Известны подходы по семантическому анализу текстов. На современном этапе развития автоматической обработки текстов актуальной и востребованной становится задача эмотивной оценки текстов. Анализ эмотивности является достаточно сложной задачей [1], что связано с трудностями при выделении нужной эмоциональной лексики в текстах, а также с определением самого эмотивного пространства, количества и состава его измерений. Ежедневно множество данных генерируется пользователями соцсетей, новостными порталами, различными блогами. Анализ эмотивности также требуется для мониторинговых, аналитических и сигнальных систем, для систем документооборота и рекламных платформ, таргетированных по тематике веб-страниц и многих других. Область применения анализа эмотивности текстов обширна.

Эмотивность определяется как свойство языка выражать психологические (эмоциональные) состояния и переживания человека. В речи она реализуется с помощью специальных языковых средств — эмотивов, которые представлены на каждом уровне любого естественного языка и образуют его эмотивный код (ЭК). Каждый текст имеет свой ЭК. ЭК языка — семиотическая система корреляций между психическими состояниями (эмоциями) коммуникантов и их концептуализацией. Существует разные характеристики эмотивности: эмотивный фон, эмотивная тональность, эмотивная окраска, эмотивная направленность, эмотивная модальность, эмотивные интенции текста [2].

Наибольшего успеха исследования эмотивности текста достигли при анализе его тональности. Анализ тональности - класс методов контент-анализа в компьютерной лингвистике, предназначенный для автоматизированного выявления в текстах эмоционально окрашенной лексики и эмоциональной оценки авторов (мнений) по отношению к объектам, речь о которых идет в тексте [3]. Анализ тональности играет важную роль в принятии решений и в системе рекомендаций. Например, перед покупкой продуктов людей интересуют отзывы других покупателей о данном продукте, при прочтении новостей – мнение других людей об описываемых событиях. Однако современные объемы информации не позволяют самостоятельно оценивать все тексты целиком. Анализ тональности упрощает эту задачу, поскольку описывает полярность текста, так что пользователь может напрямую узнать, является ли данный текст положительным или отрицательным, не читая его целиком.

Для определения настроения в анализе тональности используются три термина: объект, о котором дается мнение, особенности этого объекта, а также владелец мнения об объекте [4]. Анализ тональности выполняет задачу классификации в 3 этапа: на уровне документа, уровне предложения и уровне характеристик. Уровень классификации документа используется там, где задача состоит в том, чтобы найти общую полярность темы независимо от того, кто придерживается мнения. Классификация по уровням предложений предполагает, что каждое предложение придерживается единого мнения. На уровне характеристик выполняется анализ различных характеристик объекта. Анализ тональности включает в себя предварительную обработку данных, выбор характеристик и классификацию, а затем определение полярности данных. Предварительная обработка данных включает в себя токенизацию, удаление стоп-слов, выделение корней, лемматизацию и т. д.

Методы анализа тональности делятся на 2 большие группы: основанные на машинном обучении и основанные на правилах и словарях.

Решение проблемы анализа тональности текста позволит эффективнее обрабатывать цифровые данные в автоматизированном режиме с целью формирования эмоционального контекста материалов или выявления несоответствующих общему фону тестов.

Список использованных источников:

1. Пазельская А., Соловьев А. Метод определения эмоций в текстах на русском языке // *The international conference on computational linguistics and intellectual technologies "Dialogue 2011"*: конференция. — Москва, 2011. — С. 510–522.
2. Ленько Галина Николаевна Уровни анализа текстовой эмотивности (на примере текстов художественного стиля) // *Вестник ЛГУ им. А.С. Пушкина*. 2014. №2. – 10 с.
3. Гербик, А. И. Анализ тональности текста / А. И. Гербик, П. Е. Дорошкевич, А. И. Свито // *Компьютерные системы и сети: материалы 53-й СНТК*. – Минск: БГУИР, 2017. – С. 167–168.
4. Алгоритмы анализа тональности текста / Н. С. Иванин и другие // *BIG DATA and Advanced Analytics: collection of materials of the third international scientific and practical conference. (Minsk, Belarus, May 3 – 4, 2017) / editorial board : M. Batura [et al.]*. – Минск : BSUIR, 2017. – С. 150-154.

ПОВЫШЕНИЕ ЭФФЕКТИВНОСТИ РАБОТЫ КОМАНДЫ РАЗРАБОТЧИКОВ

Маркевич В. Ю.

Белорусский государственный университет информатики и радиоэлектроники
г. Минск, Республика Беларусь

Медведев С.А. – к.т.н., доцент

Менеджерам разных категорий в процессе своей деятельности приходится сталкиваться с вопросами повышения производительности команды как на коротком промежутке времени, так и на всём пути существования проекта. Такие вопросы обычно возникают под влиянием внешних факторов: нервных переговоров с заказчиком, поджиманием сроков, ограниченностью бюджета и т.д. Нередки случаи когда игнорирование процесса планирования деятельности команды приводит к плохим показателям эффективности. Но как же свести это всё к минимуму?

Анализируя существующие методологии управления командами можно увидеть целый комплекс рекомендаций, направленный на улучшение работы. По одной из лидирующих методологий SCRUM [1], родоначальником которой считается Джеф Сазерленд, предлагается разбиение работы на короткие интервалы – спринты и сведение планирования от масштабных целей до конкретного промежутка времени, давая отзывы заказчикам по завершению маленького этапа жизни продукта. В методологии KANBAN [2], предложенной руководителем компании «Toyota» Тайити Оно, важнейшей целью командной работы является устранение потерь и их классификация. Тайити Оно выделяет три классификации потерь: «migi» - потери возникающие из-за неблагодарного обращения с ресурсами, «muga» - потери из-за неравномерного обращения с ресурсами и «muda» - потери из-за производственных процессов. Для уравнивания потерь хорошо подходит цикл Деминга (или другими словами круг качества) – постоянный круг регулирования усовершенствования продукта и производственных процессов. В основе цикла Деминга лежат 4 итерации: «планировать», «действовать», «проверять» и «корректировать». Если сопоставить их с классификацией потерь, то «планировать» - избегать «migi», «действовать» - избегать «muga», «проверять» - избегать «muda», «корректировать» же в свою очередь это попытка предугадать или увидеть зарождения проблем, до момента возникновения потерь.

Вопрос улучшения работы команды требует комплексного анализа работы. Но для анализа нужна выборка данных, которые нужно как-то получить. Ответом на этот вопрос является чёткая постановка процессов внутрикомандного взаимодействия и выделение времени на инспекцию проделанной работы («корректировать» из цикла Деминга) – ретроспектива.

В мировой практике уже есть чёткие правила, по которым нужно проводить процесс ретроспективы. В основном они говорят о том, что нужно анализировать короткие промежутки времени, каждый член команды должен выступить с докладом, который отразит плюсы, минусы, предложения и идеи. Эти правила были мной расширены и конкретизированы, а сам процесс был разделён на 2 этапа: индивидуальные выступления и коллективное обсуждение.

Для индивидуальных выступлений был разработан план построения доклада, состоящий из индивидуальной характеристики и оценки команды, а также перечень обязательных вопросов.

Для коллективного обсуждения разработан план проведения, включающий:

- анализ предложений и идей (положительные и отрицательные стороны относительно внедрения или не внедрения);
- анализ минусов (почему получилось так, а не иначе?);
- анализ плюсов (как получилось достигнуть и что нужно делать для поддержания положительного эффекта?).

По результатам проведения ретроспективы стала хорошо прослеживаться динамика улучшения «коэффициента счастья» команды и факторы, влияющие на спад или подъём интенсивности работы. По результатам применения разработанного плана улучшилась динамика проведения ретроспективных собраний, команда стала лучше понимать важность не только самого процесса ретроспектив, но и важность формирования команды как основной движущей силы достижения общей цели. Благодаря хорошо проработанному плану проведения переговоров удалось достигнуть своевременного выявления проблем на ранних этапах зарождения, видеть лучшую картину формирования командного «счастья» и достигнуть большего уровня вовлеченности в процесс каждого члена команды.

Список использованных источников:

1. Сазерленд, Дж. *Scrum. Революционный метод управления проектами* / Дж. Сазерленд – М. : Изд-во Манн, Иванов и Фербер, 2018. – 288 с.
2. Канбан и "точно вовремя" на Toyota. *Менеджмент начинается на рабочем месте* / Японская Ассоциация Менеджмента – М. : Альпина Паблишер, 2020. – 214 с.

ПРОГРАММНОЕ СРЕДСТВО МОДЕЛИРОВАНИЯ КАРТЫ НЕИСПРАВНОСТЕЙ ЗАПОМИНАЮЩИХ УСТРОЙСТВ С ЗАДАНЫМИ ПАРАМЕТРАМИ

Наркевич И.Г., Деменковец Д.В.

*Белорусский государственный университет информатики и радиоэлектроники
г. Минск, Республика Беларусь*

Бахтизин В.В. – кандидат технических наук

В работе рассматривается программное средство моделирования карты неисправностей запоминающих устройств с заданными параметрами. Рассматриваются существующие модели неисправностей запоминающих устройств. Предложен пример использования разработанного программного средства.

Вычислительная система может состоять из большого множества различных элементов, большинство которых не являются обязательными. Но одним из важнейших элементов, без которого вычислительная система не может функционировать, является запоминающее устройство. Это устройство служит для записи и хранения данных.

Причиной неисправного состояния запоминающего устройства является присутствие какого-то физического или механического дефекта либо их множества, при этом их количество и многообразие практически неограниченно. В зависимости от технологических особенностей при производстве памяти и внешних факторов при её эксплуатации могут появляться новые типы и разновидности дефектов. Установление факта появления дефекта и его классификация является весьма трудоёмкой и часто неразрешимой задачей. Это объясняется в первую очередь тем, что появление дефекта чаще всего можно определить лишь по косвенным признакам, как правило, по факту неправильной работы запоминающего устройства.

Современный этап развития методов и средств тестирования ОЗУ сопряжён со следующими основными проблемами. Во-первых, это неуклонный рост ёмкости ОЗУ, которая превышает 10^9 бит. Во-вторых, существенное отличие физической структуры ОЗУ от их логической организации, когда физически соседние запоминающие элементы имеют логически удалённые адреса. И, в-третьих, многие разновидности физических дефектов и отказов современных ОЗУ могут быть описаны только с помощью сложных моделей неисправностей ОЗУ, которые трудно обнаруживаются классическими тестами и их модификациями.

Для аналитического описания неисправных состояний памяти используются математические модели неисправностей, которые тем или иным образом отражают реальные физические дефекты запоминающих устройств. Основные физические дефекты этих устройств найдены и построены математические модели неисправностей, которые эти дефекты отражают. На основе имеющихся математических моделей можно построить компьютерные модели, которые будут иметь поведение, похожее на поведение реальных физических запоминающих устройств.

Первые модели неисправностей запоминающих устройств были предложены в начале 1980-х годов, которые включали константные неисправности, неисправности дешифратора адреса, неисправности взаимного влияния и кодочувствительные неисправности. Функциональные неисправности запоминающих устройств подразделяются на два подмножества: неисправности матрицы запоминающих элементов и неисправности электронного обрамления. Второе подмножество включает неисправности дешифраторов адреса и неисправности логики чтения-записи. Доминирующее значение имеют неисправности матрицы запоминающих элементов памяти, которые часто называют ячейками памяти.

К неисправностям матрицы ячеек запоминающих устройств в первую очередь относят неисправности, в которых участвуют: одна ячейка; две ячейки; несколько ячеек, в общем случае более чем две, без ограничений на их количество.

К неисправностям, затрагивающим одну ячейку запоминающего устройства, относят:

1. Константные неисправности (stuck-at faults – SAF). Неисправный запоминающий элемент памяти постоянно находится в состоянии логического нуля (s-a-0) или логической единицы (s-a-1), независимо от операций, выполняемых с неисправным элементом и другими элементами запоминающего устройства.

2. Переходные неисправности (transition faults - TF). Подобные неисправности характеризуются невозможностью перехода состояния неисправного запоминающего элемента из 0 в 1 (TF↑) или из 1 в 0 (TF↓) при выполнении соответствующих операций записи.

Среди неисправностей, в которых участвуют две ячейки запоминающего устройства, выделяют следующие неисправности:

1. Неисправности взаимного влияния (coupling fault - CF). При описании данной неисправности выделяют влияющую ячейку, определяемую её адресом i , изменение логического состояния которой воздействует на состояние зависимой ячейки с адресом j . Различают три типа неисправностей взаимного влияния:

а) инверсные неисправности взаимного влияния (inverse coupling faults - CF_{in}). При наличии данной неисправности изменение значения b_i влияющей ячейки вызывает инвертирование значения b_j зависимой ячейки. Возможны следующие виды CF_{in}: $\wedge(\uparrow, \bar{b}_j)$, $\wedge(\downarrow, \bar{b}_j)$, $\vee(\uparrow, \bar{b}_j)$, $\vee(\downarrow, \bar{b}_j)$. Символ \wedge и \vee задают взаимное расположение влияющего и зависимого запоминающих элементов памяти. Первый символ \wedge означает, что запоминающий элемент с меньшим адресом влияет на запоминающий элемент с большим адресом ($i < j$), а символ \vee используется в случае, когда адрес влияющего запоминающего элемента больше адреса зависимого запоминающего элемента ($i > j$);

б) неисправности взаимного влияния прямого действия (idempotent coupling faults - CF_{id}). При изменении значения b_i влияющего запоминающего элемента происходит принудительная установка определённого логического значения 0 или 1 в зависимом запоминающем элементе. Различают восемь неисправностей прямого действия: $\wedge(\uparrow, 0)$, $\wedge(\uparrow, 1)$, $\wedge(\downarrow, 0)$, $\wedge(\downarrow, 1)$, $\vee(\uparrow, 0)$, $\vee(\uparrow, 1)$, $\vee(\downarrow, 0)$, $\vee(\downarrow, 1)$;

в) статические неисправности взаимного влияния (state coupling faults CF_{st}). Переход зависимой ячейки в какое-либо состояние b_j возможен при определённом значении b_i влияющей ячейки. Возможно восемь неисправностей CF_{st}: $\wedge(0, 0)$, $\wedge(0, 1)$, $\wedge(1, 0)$, $\wedge(1, 1)$, $\vee(0, 0)$, $\vee(0, 1)$, $\vee(1, 0)$, $\vee(1, 1)$.

2. Кодочувствительные неисправности (pattern sensitive faults - PSF) рассматриваются как обобщение моделей неисправностей взаимного влияния. Для подобных неисправностей логическое состояние или изменение логического состояния одного запоминающего элемента запоминающего устройства может зависеть от содержимого (0 или 1) или от логических переходов из 1 в 0 или из 0 в 1 влияющих запоминающих элементов запоминающих устройств. В случае кодочувствительной неисправности PSF_k, в которой участвует k запоминающих элементов запоминающего устройства, подразумевается, что влияющими запоминающими элементами в предельном случае могут быть любые из $k - 1$ из N запоминающих элементов запоминающего устройства, а зависимым один из оставшихся $N - k + 1$ запоминающих элементов. На практике при рассмотрении кодочувствительных неисправностей вводятся ограничения на количество запоминающих элементов k , так и на их местоположение.

В зависимости от эффекта влияния на базовый запоминающий элемент различают несколько классических типов кодочувствительных неисправностей NPSF_k:

1. Пассивными кодочувствительными неисправностями (passive NPSF – PNPSF) являются неисправности, при которых состояние базового запоминающего элемента не может быть изменено для определённого кода в $k - 1$ соседних запоминающих элементах запоминающего устройства.

2. Под активными кодочувствительными неисправностями (active NPSF – ANPSF) понимают неисправности, в которых базовый запоминающий элемент изменяет своё состояние из-за изменения кода в соседних запоминающих элементах. Изменение кода для подобных неисправностей происходит в результате изменения состояния на противоположное только в одном соседнем элементе, в то время как остальные ячейки сохраняют предыдущее состояние.

3. Статические кодочувствительные неисправности (static NPSF – SNPSF) характеризуются тем, что для определённой комбинации значений в соседних запоминающих элементах базового запоминающего принудительно устанавливается в состояние 0 или состояние 1. Главным отличием статических неисправностей от активных кодочувствительных неисправностей является длительность процесса установления неверного значения в базовой ячейке. Для статических неисправностей это время существенно больше.

В структуре классических моделей неисправностей запоминающих устройств одно из самых видных мест в силу сложности их обнаружения тестами памяти занимают связные неисправности. Под связными неисправностями понимают неисправности различных типов, которые включают в себя общие ячейки.

В случае двух неисправностей взаимного влияния, например $(\uparrow, 1)$ и $(\uparrow, 0)$, с адресами ячеек i, j, l и g , участвующих в этих неисправностях и расположенных по возрастанию $i < j < l < g$, при $j = g$ две неисправности взаимного влияния, имеющую общую ячейку-жертву, трансформируются в связную неисправность. Возможны любые сочетания типов и подтипов неисправностей, участвующих в конкретной связной неисправности. Их многообразие и количество всегда являлись препятствием для анализа их обнаружения конкретным тестом.

В результате было создано программное средство моделирования карты неисправностей запоминающих устройств с заданными параметрами. Данное приложение позволяет создавать карту неисправностей, изменяя состояние конкретной выбранной ячейки памяти или распределяя случайным образом определённое количество неисправностей с заданными параметрами, сохранять созданную карту в файл и загружать сохранённую карту из файла.

Список использованных источников:

1. Ярмолик, В. Н. Контроль и диагностика вычислительных систем : учеб. пособие / В. Н. Ярмолик. – Минск : Изд-во : Бестпринт, 2019. – 387 с.
2. Ярмолик, С. В. Маршевые тесты для самотестирования ОЗУ : монография / С. В. Ярмолик, А. П. Занкович, А. А. Иванюк. – Минск : Изд. центр БГУ, 2009. – 271 с.

ПРОГРАММНОЕ СРЕДСТВО ОРГАНИЗАЦИИ УПРАВЛЕНИЯ ТЕЛЕФОННЫМИ НОМЕРАМИ НА ПЛАТФОРМЕ JAVA

Невар В.И.

Белорусский государственный университет информатики и радиоэлектроники
г. Минск, Республика Беларусь

Нестеренков С.Н. – доцент, к.т.н

В результате работы было разработано программное средство организации управления телефонными номерами, а также разработано web-приложение для отслеживания текущего состояния работы программы. Вызовы к системе осуществляются посредством HTTPS запросов. Данная система предназначена для того, чтобы клиент произвел единственный запрос к системе, а она, в свою очередь, в автоматическом режиме делает запросы к поставщикам телефонных номеров для их покупки, после чего сообщает клиенту о выполнении его.

IP-телефония - это технология, позволяющая использовать Интернет или любую другую IP-сеть для ведения международных и междугородных телефонных разговоров и передачи факсов в режиме реального времени.

Для организации телефонной связи по IP-сетям используется специальное оборудование - шлюзы IP-телефонии. Каждый шлюз должен быть соединен с телефонным аппаратом или абонентской линией АТС, пользователи которых будут являться абонентами IP-шлюза. Два абонента разных IP-шлюзов, разделенные расстоянием в тысячи километров, могут общаться в режиме реального времени, оплачивая только время подключения к IP-сети. С равным успехом IP-шлюз может использоваться и в корпоративной ЛВС.

Общий принцип действия телефонных шлюзов IP-телефонии таков: шлюз принимает телефонный сигнал, оцифровывает его, значительно сжимает, разбивает на пакеты и отправляет через IP-сеть по назначению.

Еще одним способом применения IP-шлюзов является подключение к сети оператора IP-телефонии. В этом случае можно совершать вызовы на любые телефонные номера ТФОП. Стоимость звонка будет существенно дешевле, так как междугородние/международные тарифы операторов IP-телефонии существенно ниже тарифов операторов телефонной связи.



Рисунок 1 – Схема организации IP-телефонии

Только при распространении IP-телефонии появилась возможность использовать виртуальные номера. Ни один из классических способов телефонной связи этого делать не умеет. Суть вот в чем: вы арендуете любой номер телефона, который вам нравится. А на него звонят клиенты. Идея в том, что можно взять в аренду абсолютно любой номер. Это прежде всего интересно бизнесу по нескольким причинам. Покупка номеров с кодами больших городов, например Москвы, может сформировать мнение у покупателей, что данная организация является солидной, раз по всему

видимому офис организации находится в столице. Купив зарубежные номера, можно предоставлять свои услуги иностранным клиентам, причём они даже не будут догадываться, что вы можете находиться в другой части земного шара. Кроме того, можно приобрести различные номера для каждого отдела или даже человека, чтобы клиентам было удобнее получать помощь в интересующем их отделе, а не постоянно перенаправляться у новому оператору.



Рисунок 2 – Схема работы виртуального номера

На сегодняшний день имеется достаточно большое количество поставщиков виртуальных номеров. Для небольшого бизнеса, работающего в пределах одной страны, может хватать услуг одного поставщика. Но для большого бизнеса, ведущего свою деятельность в нескольких странах, могут понадобиться услуги нескольких поставщиков. Однако это неудобно, придётся тратить дополнительные ресурсы на управление всей этой структуры.

Решением может стать посредник, который организует взаимодействие с несколькими поставщиками виртуальных номеров. В результате получается поставщик, библиотека номеров которого будет покрывать достаточное количество регионов даже для большого бизнеса.

Ярким примером может быть крупный поставщик услуг в сфере телефонии IDT Express, библиотека которого насчитывает более 70 стран, включая подавляющее большинство регионов США. Это стоит отметить, так как именно в этой стране IP-телефония пользуется огромным спросом.

Однако организация библиотеки номеров, используемое этим поставщиком, является достаточно затратной в качестве материальных и рабочих ресурсов. Разработанное программное средство призвано решить эти проблемы.

Разработанное программное средство организации управления телефонными номерами позволяет совершать заказы непосредственно у первичных поставщиков виртуальных номеров.

К преимуществам программного средства организации управления телефонными номерами можно отнести:

1. Огромная база доступных виртуальных номеров, полученная благодаря интеграции непосредственно с первичными поставщиками.
2. Автоматическое обновления статуса заказа. После обновления статуса заказа на стороне поставщика клиент будет оповещён.
3. Отсутствие затрат на поддержание библиотеки виртуальных номеров, так как теперь она является совокупностью библиотек поставщиков.
4. Возможность расширения за счёт интеграции с новыми поставщиками, что достигнуто благодаря продвинутой архитектуре программного средства.

К минусам подобной системы можно отнести только затраты на её поддержку, что по сравнению с выгодой является незначительным.

Программное средство организации управления телефонными номерами позволило упростить процесс взаимодействия с поставщиками и уменьшить затраты на поддержание базы виртуальных номеров.

Список использованных источников:

1. Что такое DID или прямой входящий набор номера [Электронный ресурс]. – Режим доступа: <https://www.3cx.ru/voip-sip/did/>. – Дата доступа: 21.04.2021.
2. Что такое IP-телефония и как ее использовать для интернет-магазина [Электронный ресурс]. – Режим доступа: <https://www.insales.ru/blogs/university/что-такое-ip-telefoniya>. – Дата доступа: 21.04.2021.
3. Почему ваш бизнес должен приобрести DID-номера [Электронный ресурс]. – Режим доступа: <https://www.idtexpress.com/blog/why-should-your-business-buy-dids/>. – Дата доступа: 21.04.2021.

ПРОГРАММНОЕ СРЕДСТВО УПРАВЛЕНИЯ ОЧЕРЕДЯМИ НА ПЛАТФОРМЕ JAVA

Панкратьев А.С.

Белорусский государственный университет информатики и радиоэлектроники
г. Минск, Республика Беларусь

Нестеренков С.Н. – канд. тех. наук, доцент

В результате работы было разработано приложения для управления очередями. Пользователи имеют возможность создавать очереди на события, регистрироваться на события, тем самым пользователи системы (как авторы очереди, так и участники) могут планировать свое время более продуктивно.

За всю жизнь человек проводит огромное количество времени в очереди: от сдачи лабораторных работ до очереди в поликлинике или на выступление на конференции.

В большинстве случаев очереди оставляют негативные впечатления как для организаторов, так и для участников. Чаще всего основная проблема состоит в том, что очередь получается неорганизованной. Так же людям сложно просчитать, когда лучше прийти, если они даже не знают количества людей, предшествующих им. Понимание того, сколько людей планирует прийти на то или иное событие так же может помочь компании и сотрудникам компании планировать свое рабочее время более продуктивно[1].

Разработанное решение представляет собой клиент-серверное приложение. Серверная часть приложения построена на платформе Java при использовании фреймворка «Spring». Клиентская часть является браузерным приложением, написанным с использованием библиотеки React. Таким образом пользователи смогут использовать клиент приложения с любого устройства, где установлен браузер: компьютер, смартфон, планшет, ноутбук и т.д.

При переходе на главную страницу, пользователь должен аутентифицироваться или зарегистрироваться. После того, как пользователь аутентифицирован, у него появляется возможность создавать очереди и регистрироваться в очередях. Скриншот страницы регистрации на очередь приведен на рисунке 1.

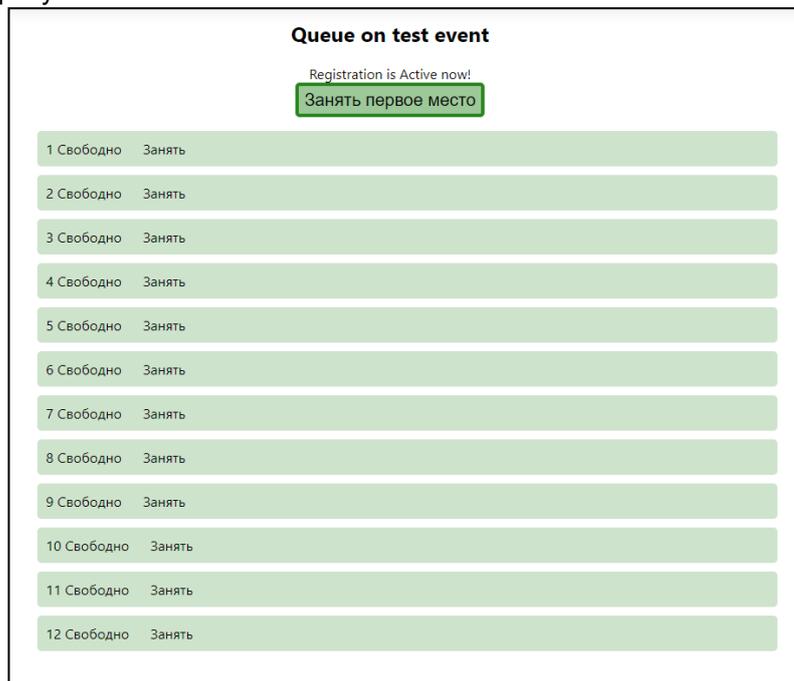


Рисунок 1 – Страница регистрации в очереди

Автор очереди имеет возможность редактировать очередь, а так же помечать задачи как выполненные, давая возможность людям понимать состояние очереди на данный момент.

В качестве корпоративных пользователей системы могут выступать организации, занятые в сферах услуг, образования, медицины. Применение программного средства управления очередями позволит этим компаниям решать задачи организации очередей на события и собирать необходимую информацию от участников мероприятия.

Список использованных источников:

1. *Master Your Time, Master Your Life: The Breakthrough System to Get More Results, Faster, in Every Area of Your Life* / B. Tracy // TarcherPerigee, 2016 – 240 p..
2. Шилдт, Г. Полное уководство; 9-е изд.: Пер. с англ. / Г. Шилдт. - М. : ООО "И.Д. Вильяме 2015.

ПРОГРАММНОЕ СРЕДСТВО РЕАЛИЗАЦИИ ФИЛЬТРА КАЛМАНА

Трухан К.А., Струневская А.В., Верещагин Н.В.

Белорусский государственный университет информатики и радиоэлектроники
г. Минск, Республика Беларусь

Леванцевич В.А. – маг. тех. наук

Рассмотрен метод фильтрации сигналов на основе фильтра Калмана первого порядка. Описана упрощенная модель фильтра Калмана и разработано программное средство для его реализации. Проведены экспериментальные исследования и показана возможность применения фильтрации Калмана для исследования вибрационных сигналов.

При измерении аналоговых величин, возникает погрешность, которая в той или иной мере оказывает негативное влияние на результат измерения. Одним из методов подавления шумов, позволяющих свести к минимуму потери полезного сигнала, является применение фильтра Калмана [1-2]. Этот метод способен давать значение, максимально приближенное к реальному. Такие результаты достигаются благодаря тому, что система может учитывать управляющее воздействие на систему, если известны свойства этого воздействия.

Алгоритм состоит из двух повторяющихся фаз: предсказания и корректировки. На первой фазе рассчитывается предсказание состояния в следующий момент времени (с учетом неточности их измерения). На второй, новая информация с датчика корректирует предсказанное значение (также с учетом неточности и зашумленности этой информации).

Математически алгоритм реализации фильтра Калмана в общем случае выражается формулой [2]:

$$x_{k+1}^o = K_{k+1} \cdot z_{k+1} + (1 - K_{k+1}) \cdot x_k^o \quad (1),$$

где x_{k+1}^o отфильтрованное фильтром Калмана, текущее значение измеряемой величины;

x_k^o предсказанное значение измеряемой величины, полученное на предыдущем этапе;

z_{k+1} текущее, измеренное значение измеряемой величины

K_{k+1} коэффициент Калмана (коэффициент усиления).

Коэффициент Калмана зависит от соотношения среднеквадратичной ошибки предсказания и ошибки измерения и с каждым шагом вычисления стабилизируется к определенному значению. Поэтому на практике иногда используется упрощенная форма фильтра Калмана в которой коэффициент Калмана принимается равным константе, которая подбирается экспериментально. На рисунке 1 представлен результат работы программного средства, реализующего упрощенный фильтр Калмана при обработке вибросигнала.



Рисунок 1 – Результат работы программного средства

Из рисунка видно, что сигнал на выходе фильтра Калмана более сглажен.

Выводы: Фильтр Калмана является эффективным средством фильтрации сигналов и для своей реализации требует меньше вычислительных ресурсов в сравнении с другими методами фильтрации.

Список использованных источников:

1. Васильев В.Н., Гуров И.П. Компьютерная обработка сигналов в приложении к интерферометрическим системам. СПб: БХВ - Санкт-Петербург, 1998.
2. Гуров И.П., Анализ акустических сигналов на основе метода фильтрации Калмана/ И.П. Гуров, П.Г. Жианов П.Г.// Научно-технический вестник информационных технологий, механики и оптики. –2011 С. 200-206
3. Балакришнан А.В. Теория фильтрации Калмана. М.: Мир, 1988. 168 с.

ОБЗОР РЕШЕНИЙ ЗАДАЧИ ИДЕНТИФИКАЦИИ АВТОРА ТЕКСТА

Труханович И.А., Кунцевич В.С.

Белорусский государственный университет информатики и радиоэлектроники
г. Минск, Республика Беларусь

Парамонов А.И. – канд. техн. наук, доцент

Идентификация автора текста актуальная задача в области обработки естественно-языковых текстов. Описаны основные задачи, которые возникают в процессе исследования текста, и указаны существующие методы их решения.

Высокая значимость идентификации автора печатного текста обуславливается быстрым ростом объема информации и стала одним из ключевых методов обработки и организации текстовых данных. Исследование идентификации авторов полезно для выявления наиболее вероятных авторов и поиска доказательств, подтверждающих вывод.

Таким образом, метод определения автора позволяет идентифицировать наиболее возможного автора из группы кандидатов в авторы научных статей, новостей, электронных писем и т.д. Этот процесс основан на изучении набора документов этого автора и сопоставлении его «почерка». В качестве «почерка» автора принято использовать авторский инвариант. Это количественная характеристика литературных текстов или некий параметр, который однозначно характеризует своим значением документы одного автора или некоторой группы «близких» авторов, и принимает существенно разные значения для произведений разных групп авторов.

В настоящее время замечается обостренный интерес к количественным технологиям разбора текстовой информации на основе слабо регулируемых автором черт текста, общих для всех авторов. Но общепризнанного определения о том, какой комплект черт даёт лучший результат, не существует. Достаточно мало внимания уделено идентификации автора с использованием комплексных черт текста.

Решение данной проблемы можно разбить на следующие задачи:

- Выбор модели представления текстов.
- Выбор признаков для формирования авторского инварианта.
- Выбор метода классификации с подходящими настройками.
- Определение автора текста из множества предполагаемых авторов.

Каждая из задач на сегодня имеет уже ряд предполагаемых решений, которые используют различные подходы и имеют определенные результаты.

Для представления текстов в информационной системе можно использовать модель «мешок слов», N-граммную и сглаживающую модели. Модель «мешок слов» представляется как неупорядоченная коллекция всех слов (или признаков слов), из которых состоит текст. В N-граммной модели текст понимается как последовательность цепочек из n элементов. Сглаживающие модели помогают справиться с проблемной разреженных данных в N-граммном представлении с помощью специальных техник сглаживания.

В качестве признаков, используемые для формирования авторского варианта, используются лексические, синтаксические, структурные, контентно-специфические, стилевые и другие.

В качестве методов классификации применяются самые различные подходы: критерий Стьюдента, меры расстояния, байесовский классификатор, метод ближайших соседей, генетические алгоритмы, нейронные сети и метод опорных векторов.

Самыми достоверными и точными можно назвать метод опорных векторов и нейронных сетей. Они дают высокую степень точности на текстах разных областей даже достаточно небольшого размера. Поэтому их можно использовать в достаточно широком диапазоне задач: от анализа диалогов в сети Интернет до авторства больших научных работ. Тем не менее, они являются и ресурсозатратными. Что является существенными недостатками. Также нейронные сети требуют повышенного внимания при работе с большим признаковым пространством. Затраты времени на подбор топологии сети и обучение можно сократить с помощью методов автоматического подбора топологии, что в итоге сказывается на точности результатов. В метод опорных векторов отсутствуют такие проблемы, однако он чувствителен к шумам в исходных данных.

Уверенность в авторском инварианте и соблюдение требований к точности выходных данных в конечном счете предопределяет возможность определение авторства текста и формирование окончательного решения об самом авторе.

Список использованных источников:

1. Орлов, Ю.Н. Методы статистического анализа литературных текстов / Ю.Н. Орлов, К.П. Осминин. – М. : URSS, 2012. – 326 с.
2. Карта слов [Электронный ресурс]. – Режим доступа: <https://kartaslov.ru/карта-знаний/Авторский+инвариант>. – Дата доступа: 03.04.2021.

ПРИМЕНЕНИЕ ГЕНЕТИЧЕСКИХ АЛГОРИТМОВ ДЛЯ КРИПТОАНАЛИЗА ПОДСТАНОВОЧНЫХ ШИФРОВ

Фролов А.О.

Белорусский государственный университет информатики и радиоэлектроники
г. Минск, Республика Беларусь

Прохорчик Р.В. – ст. преп., м. т. н.

В данной работе рассматривается иной подход к криптоанализу подстановочных шифров, основанный на применении генетических алгоритмов. Показывается, что такие алгоритмы могут быть использованы для обнаружения ключа простых шифров подстановок. В качестве существенного фактора целевой функции генетического алгоритма используется частотный анализ.

В настоящее время безопасность является одной из важнейших проблем в информатике, криптография широко используется для ее реализации. Задачу, противоположную криптографии, решает криптоанализ – процесс, в котором пытаются нарушить безопасность. В свою очередь, сложность этого процесса рассматривается как мера безопасности. Поскольку криптографические алгоритмы открыты для всех, вся мощь алгоритма заключается в сложности ключа, сила которого в общем случае определяется его длиной.

Данная работа посвящена криптоанализу подстановочных шифров, которые являются одними из древнейших. Шифры такого рода явным или неявным образом используют таблицы подстановок. Рассмотрим в качестве шифра подстановки достаточно популярный шифр Виженера и покажем, насколько применим генетический алгоритм для его криптоанализа.

Генетические алгоритмы – это класс эвристических алгоритмов, которые пытаются решить задачу путем моделирования упрощенной версии генетических процессов. Такого рода алгоритмы основаны на концепциях генетики и эволюции. Генетический алгоритм функционирует на основе популяции выбранных организмов, в геномном материале которых (генотип) закодирован ответ на поставленный вопрос (фенотип). Для простых задач пространства фенотипов и генотипов одинаковы, однако в большинстве случаев эти пространства различны. Генетический алгоритм начинается с формирования изначального популяционного материала на основании произвольной выборки. В дальнейшем на основании аналитики дается оценка популяции по уровню приспособленности к среде обитания или заданным факторам. Затем самые приспособленные особи скрещиваются. Таким образом, популяционные особи дают потомство, которое имеет высокую вероятность сохранить необходимый генотип в своем и следующих поколениях. Потомство изначальной популяции создает следующую по хронологии популяцию, причем у большинства особей должен сохраняться на протяжении всей жизни выбранный генотип, а минимальное число особей подвержены генной мутации. Отметим, что каждое множество особей популяции в определенный промежуток времени – это поколение. Смена поколений образует популяционную эволюцию. При этом каждый этап эволюционного процесса может иметь разную длительность, что обусловлено различной скоростью поиска правильного решения, ограничением на количество возможных поколений, а также другими факторами.

Стоит отметить, что генетические алгоритмы не всегда дают точный ответ, а скорее дают решение, близкое к правильному. Несмотря на неточность, генетические алгоритмы показывают хорошие результаты при решении оптимизационных задач и поиске ответов на такого рода вопросы. По той причине, что целевая установка взлома шифра – это выявление единственно верного секретного ключа в ряду потенциально возможных ключей, то задача криптографического анализа может быть представлена в виде оптимизационной задачи. Таким образом, в поиске решения такой задачи может быть применен генетический алгоритм.

В первую очередь, при создании генетического алгоритма важно выбрать оптимальный, эффективный способ кодирования и описать основные операторы алгоритма – селекция, кроссинговер и мутация. На качественную составляющую алгоритма ключевое значение оказывает популяционная численность, количество поколений, значения уровня вероятности для базовых операторов.

В случае расшифровки зашифрованного текста каждое решение хранит ключ k длины n , а операторы мутации и кроссинговера изменяют k . Для реализации расшифровки текста была выбрана малая вероятность мутации (0.25), а уровень вероятности наследования генотипа от одного поколения к другому была, в свою очередь, выбрана достаточно большой (0.85). Оператор кроссинговера, используемый в данной работе, представляет собой двухточечный кроссинговер. Для выбора наиболее приспособленных особей для скрещивания была выбрана турнирная селекция.

Одной из главных задач при использовании генетических алгоритмов для расшифрования является выбор целевой функции. Реализовать хорошую целевую функцию довольно нетривиально. Языковая статистика и атаки по словарю должны противостоять этой проблеме. Весьма интересный подход к вычислению целевой функции описан в работе Э. Кларка [1]. Суть этого подхода

заключается в сравнении n -граммной статистики расшифрованного сообщения с данными языка, которые считаются известными. Уравнение, являющееся общей формулой, используемой для определения пригодности предполагаемого ключа k , имеет следующий вид:

$$f(k) = \alpha \cdot \sum_{i \in A} |D_i^u(k) - S_i^u| + \beta \cdot \sum_{i,j \in A} |D_{i,j}^b(k) - S_{i,j}^b| + \gamma \cdot \sum_{i,j,l \in A} |D_{i,j,l}^t(k) - S_{i,j,l}^t|, \quad (1)$$

где A – алфавит языка, D – частота появления n -граммы в анализируемом текстовом фрагменте, который получен из начального зашифрованного текста путем его расшифровки с применением предполагаемого ключа, обозначенного буквой k , S – заданный средний уровень частоты повторения n -граммы в различных текстовых источниках, а индексы u , b и t обозначают статистику униграмм, биграмм и триграмм соответственно. Значения α , β и γ позволяют присваивать различные веса каждому из трех типов n -грамм. Целевая функция опирается на статистическую характеристику языка, чтобы определить пригодность ключа. Таким образом, выбор меры пригодности полностью зависит от языковых характеристик и, следовательно, эти характеристики должны быть известны, как и сам используемый язык.

Разработанный на основе вышеизложенного алгоритм, позволяющий проводить криптографический анализ шифра Виженера, работает на заранее заданной длине ключа шифрования и находит с определенной точностью сам секретный ключ. Выбранная функция пригодности является основным фактором алгоритма. Текст, получаемый в результате работы алгоритма, не сильно отличается от исходного текста. В большинстве случаев отдельные символы легко восстанавливаются с помощью контекста. Результаты проведенных опытов дают возможность сказать, что в случае малой численности популяции результаты получаются не оптимальными, что продемонстрировано на рисунке 1. Это связано с тем, что генетический алгоритм достаточно быстро определяет место локального экстремума функции. Очевидно, что число поколений сильно зависит от длины ключа и, следовательно, с увеличением длины ключа нужно увеличивать число последовательно идущих друг за другом поколений. Соблюдение обозначенного условия дает возможность получить наиболее точное решение. Изменения основных генетических операторов алгоритма в допустимых пределах не дали улучшения результатов.

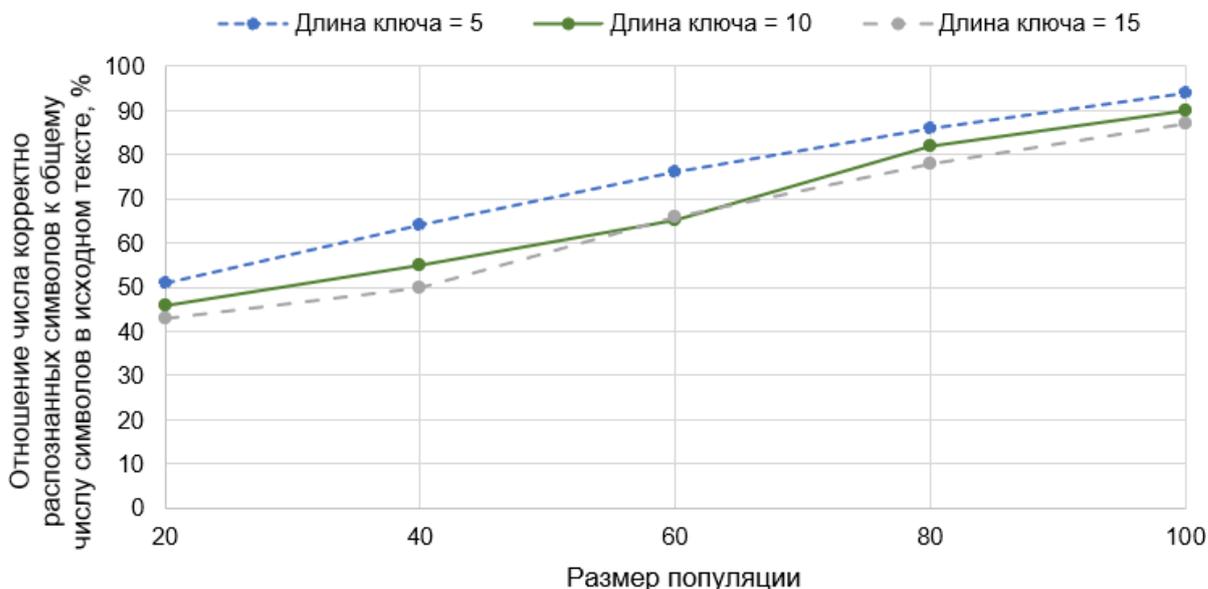


Рисунок 1 – Процентное соотношение корректных символов при различных размерах популяции

В данной работе была успешно реализована атака генетического алгоритма на простой криптографический шифр подстановки. Несмотря на то, что не всегда удается полностью автоматизировать процесс решения задачи криптоанализа, алгоритм в значительной степени помогает расшифровать зашифрованный текст. В ходе работы были протестированы различные параметры, такие как численность популяции, число поколений, а также вероятностные характеристики основных операторов. Использование генетического алгоритма для криптоанализа простого шифра подстановки оказалось эффективным методом криптоанализа, основанным на аспекте сравнения частот встречаемости букв. Так как подход, основанный на применении генетического алгоритма, оказался успешным для простого шифра подстановки, то имеет смысл применить тот же подход к более современным и сложным шифрам.

Список использованных источников:

1. Clark, A. *Modern optimisation algorithms for cryptanalysis*. In *Proceedings of the 1994 Second Australian and New Zealand Conference on Intelligent Information Systems*, November 29 – December 2 1994, p. 258-262.

ПРОБЛЕМЫ И ПЕРСПЕКТИВЫ СТАНДАРТИЗАЦИИ КИБЕРФИЗИЧЕСКИХ СИСТЕМ ОБРАЗОВАТЕЛЬНОГО НАЗНАЧЕНИЯ

Христенко А. В.

Белорусский государственный университет информатики и радиоэлектроники
г. Минск, Республика Беларусь

Давыдовский А. Г. – кандидат биологических наук, доцент

Аннотация. В работе описана целесообразность и необходимость создания документов в области международной стандартизации киберфизических системы образовательного назначения, таких как “умная школа”, “умный университет”. Представлены элементы управления процессом обучения на основе киберфизических системы образовательного назначения в условиях глобальной пандемии COVID-19.

Ключевые слова. Электронное обучение, цифровизация, индустрия электронного обучения, международный стандарт, пандемия COVID-19, киберфизическая система.

В настоящее время индустрию электронного обучения определяет семейство международных стандартов “Информационные технологии в обучении, образовании и подготовке”, нацеленных повысить конкурентоспособность образовательных услуг, сделать обучение доступным и непрерывным на основе информационных технологий (ИТ). Цифровизация учреждений образования является необратимым процессом, так как ее уровень во многом определяет востребованность, актуальность и качество преподаваемых знаний. Уже выстраиваются подтвержденные прогнозы о значительных изменениях в сфере обучения с использованием ИТ. На протяжении последних лет прослеживается стабильно растущий интерес среди наиболее развитых стран в новых продуктах на основе КФС. Спросом обусловлен недавний выход международных стандартов: ИСО/МЭК 30182:2017 Концептуальная модель умного города и свежих стандартов от ISO/IEC JTC 1/SC 42 в области искусственного интеллекта (ИИ). Продукт, выходящий в соответствии с ними, имеет глубокий технологический потенциал и может переопределить образ ИТ-индустрии, а также многократно увеличить ВВП стран-производителей.

Цель работы - обоснование международной стандартизации киберфизических систем образовательного назначения, таких как “умная школа”, “умный университет”, нацеленных на увеличение качества образования.

На фоне интенсивно развивающегося рынка электронных образовательных решений система образования столкнулась с беспрецедентным срывом образовательного процесса в условиях локдауна, обусловленного глобальной пандемией COVID-19, повлиявшей на все сферы жизни общества. Произошло рекордное падение развития человеческого потенциала, что будет отражаться на экономиках стран мира на протяжении десятков лет. Современные образовательные платформы не смогли быстро и качественно адаптироваться под кризис, вызванный коронавирусом. Это обусловило сверх заинтересованность и спрос в новых образовательных решениях.

В условиях четвертой промышленной революции формируются предпосылки для концепции киберфизических систем образовательного назначения. При этом основой технологических платформ “умная школа” и “умный университет” может быть использование когнитивных алгоритмов принятия решений, которые основываются на превентивном принципе управления образовательным процессом исходя из анализа внешней среды образовательных услуг. Новая система представляет собой консалтинговый сервис модели “Университет 4+”, нацеленный на увеличение развития человеческого потенциала посредством учета индивидуальных особенностей учащихся и анализа их успеваемости. Сервис уже может быть реализован без особых ресурсных затрат с помощью устоявшихся технологий по подобию социальных сетей.

В обозримом будущем, актуальной проблемой, определяющей развитие ИТ-индустрии на следующие 50 лет, является создание проектов международных стандартов моделей КФС образовательного назначения, таких как “умная школа” и “умный университет”. При этом Беларусь может выступить инициатором создания нового международного стандарта КФС “умное учебное заведение”. В этой связи развитие КФС образовательного назначения сталкивается с необходимостью решения комплекса проблем безопасности, образовательной эффективности, оптимизации законодательной базы, антропоцентричности проектирования образовательно-технологических сервисов.

Список использованных источников:

1. Позднеев, Б. М. Развитие международных стандартов по информационным технологиям в образовании, обучении и подготовке [Текст] / Б. М. Позднеев, М. В. Сутягин // Открытое образование – 2015. – 1. – 4 с.
2. Международная и национальная стандартизация информационно-коммуникационных технологий в образовании. / Климанов В. П., Косильников Ю. А., Позднеев Б. М. и др. – М.Ж ФГБОУ ВПО МГТУ “Станкин”, 2012.
3. Шваб, Клаус. Четвертая промышленная революция / Клаус Шваб – Москва : Издательство “Э”, 2017. – 208 с.

ПРИМЕНЕНИЕ PID РЕГУЛИРОВАНИЯ ДЛЯ УПРАВЛЕНИЯ СУДОМОДЕЛЬЮ

Шульга К.В., Билалов С.И., Деменковец Д.В.

Белорусский государственный университет информатики и радиоэлектроники
г. Минск, Республика Беларусь

Леванцевич В.А. – ст. преподаватель

В данной работе была рассмотрена возможность применения PID-регулятора для управления судомоделью. Также предложена схема управления судомоделью. Описывается программное средство, разработанное на языке Python, реализующее математическую модель PID-регулятора на примере автоматизированного управления судомоделью и движения ее до конечной точки маршрута.

В настоящее время обретают все большую популярность автопилоты. В связи с этим, было принято решение реализовать автопилот для модели корабля.

Автоматическое управление судомоделями применяется при решении различных задач: определение рельефа дна водоема, исследование миграции рыб, охрана и патрулирование водоемов на территории заповедников, любительская рыбалка, поиск скоплений рыбы с помощью эхолота. Все эти задачи сводятся к необходимости построения маршрута движения.

Для достижения конечной точки маршрута необходимо в автоматическом режиме управлять движением модели корабля. С этой задачей справляется специальное устройство – регулятор. Регулятор влияет на движение модели с помощью управляющих сигналов, которые влияют на две величины: скорость движения и градус поворота руля, изменяются в диапазоне от -1 до 1. Чтобы регулятор мог рассчитать эти параметры, на его вход необходимо передать расстояние до конечной точки маршрута (цель), а также отклонение от прямого курса в градусах. Для получения этих данных были использованы GPS-приёмник и компас. Схема работы регулятора представлена на рисунке 1.

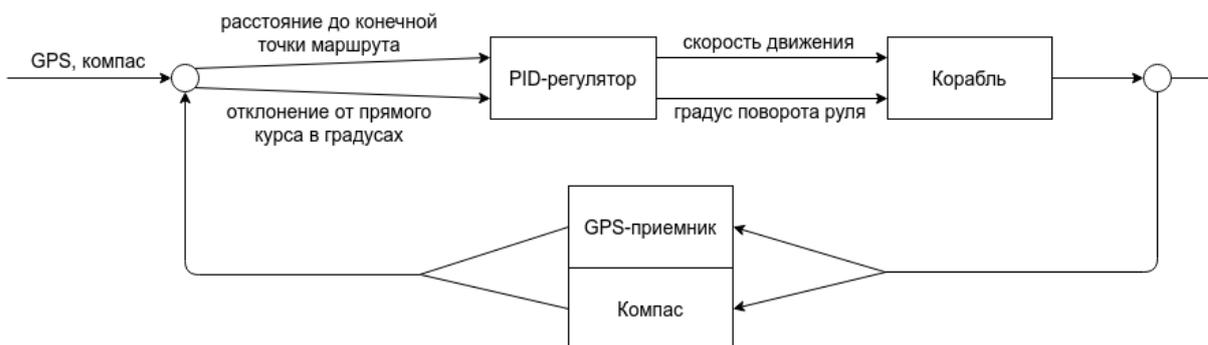


Рисунок 1 – Схема управления судомоделью

Одним из способов получения точных и качественных управляющих сигналов воздействия является PID-регулятор.

Управляющий сигнал PID-регулятора рассчитывается по следующей формуле [1]:

$$u(t) = P + I + D = K_p e(t) + K_i \int_0^t e(t) dt + K_d \frac{de(t)}{dt} \quad (1),$$

где P – пропорциональная составляющая; I – интегральная составляющая; D – дифференциальная составляющая; K_p – коэффициент усиления пропорциональной составляющей; K_i – коэффициент усиления интегральной составляющей; K_d – коэффициент усиления дифференцирующей составляющей; e – рассогласование (отклонение величины от заданной).

Как видно из формулы, сигнал управления является суммой трех составляющих: пропорциональной (1-е слагаемое), интегральной (2-е слагаемое) и дифференциальной (3-е слагаемое).

Пропорциональная составляющая зависит от рассогласования и отвечает за реакцию на мгновенную ошибку регулирования. Однако при использовании только пропорциональной составляющей значение регулируемой величины никогда не стабилизируется на заданном значении. Существует так называемая статическая ошибка, которая равна такому отклонению регулируемой величины, которое обеспечивает выходной сигнал, стабилизирующий выходную величину именно на этом значении [2].

Интегральная составляющая содержит в себе накопленную ошибку регулирования, которая является дополнительным источником выходной мощности и позволяет добиться максимальной скорости достижения цели при отсутствии перерегулирования. Интегральная составляющая позволяет регулятору со временем учесть статическую ошибку, то есть учитывать то, что происходило в прошлом [2].

Дифференциальная составляющая зависит от скорости изменения параметра, вызывающей реакцию регулятора на резкое изменение измеряемого параметра, возникшее, например, в результате внешнего возмущающего воздействия [2].

Для программной реализации необходимо выполнить дискретизацию стандартной модели PID-регулятора. Дискретная формула имеет следующий вид [3]:

$$Y_i = \frac{1}{X_p} \cdot \left[E_i + \tau_d \cdot \frac{\Delta E_i}{\Delta t_{изм}} + \frac{1}{\tau_u} \sum_{i=0}^n E_i \Delta t_{изм} \right] \cdot 100\% \quad (2),$$

где X_p – полоса пропорциональности; E_i – рассогласование; T_d – постоянная времени дифференцирования; ΔE_i – разность между двумя соседними измерениями E_i и E_{i-1} ; $\Delta t_{изм}$ – время между двумя соседними измерениями T_i и T_{i-1} ; τ_u – постоянная времени интегрирования; $\sum E_i$ – накопленная в i -й момент времени сумма рассогласований (интегральная сумма).

С целью апробации математической модели PID-регулятора на примере автоматизированного управления судомоделью было разработано программное средство на языке программирования Python со следующими функциями:

1. Возможность установки конечной точки маршрута (цель маршрута) движения судомодели.
2. Возможность выполнить автоматизированное управление моделью для достижения заданной цели.
3. Возможность добавить воздействие внешней силы на судомодель (эмуляция течения или воздействие ветра).

В результате выполненной работы была спроектирована математическая модель PID-регулятора, на основе которой было построено программное средство.

Были получены результаты для пяти наборов данных и построены пять графиков, которые демонстрируют работу регулятора. Каждый график показывает изменение расстояния до цели с течением времени. Полученные графики показаны на рисунке 2.

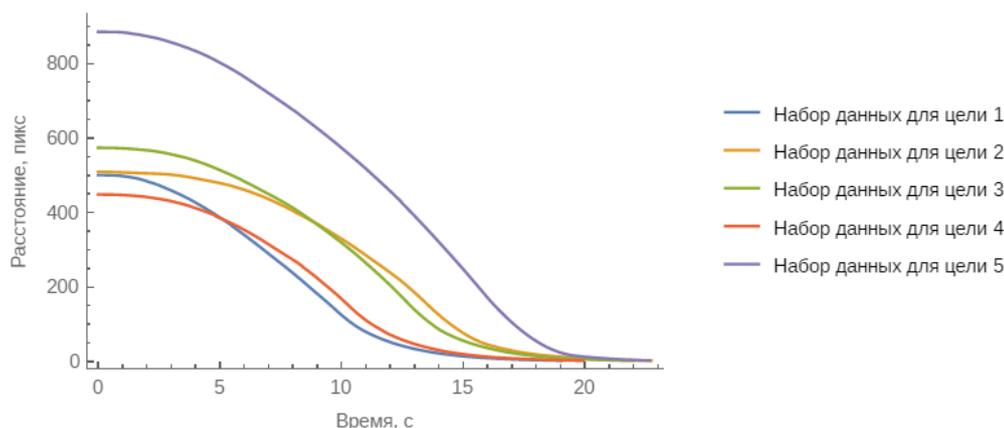


Рисунок 2 – График, демонстрирующий работу регулирования

При тестировании полученного прототипа была подтверждена возможность практического применения математической модели на реальной физической. В последующем планируется применение математической модели и разработанных алгоритмов для управления физической судомоделью в реальных природных условиях.

Список использованных источников:

1. Visioli, Antonio. *Practical PID control* / Antonio Visioli – London : Springer-Verlag London Limited, 2006 – 310 p.
2. Astrom, Karl J. *PID controllers: theory, design, and tuning* / Karl Johan Astrom, Tore Hagglund – Research Triangle Park, NC : International Society for Measurement and Con, 1995 – 354 p.
3. *Design and implement PID controllers [Electronic resource]*. – Mode of access: <https://www.mathworks.com/discovery/pid-control.html>. – Date of access: 17.03.2021.

ПРОГРАММНОЕ СРЕДСТВО ОТРИСОВКИ ТРЕХМЕРНЫХ СЦЕН С ИСПОЛЬЗОВАНИЕМ ТЕХНОЛОГИИ ТРАССИРОВКИ ЛУЧЕЙ

Яскевич Д.А., Бахтизин В.В.

Белорусский государственный университет информатики и радиоэлектроники
г. Минск, Республика Беларусь

Бахтизин В.В. – к.т.н., профессор

Рендеринг является основополагающим компонентом компьютерной графики, которая применяется в видеоиграх, инженерном моделировании, кинопроизводстве и многих других областях. С развитием платформы Nvidia RTX и графических API, позволяющих выполнять трассировку лучей в реальном времени, стало возможным реализовывать физически корректный рендеринг, который учитывает законы сохранения энергии, преломления и отражения света, а также реалистично моделировать всевозможные материалы.

Трассировка лучей – метод геометрической оптики, исследующий оптические системы путем отслеживания взаимодействия отдельных лучей с поверхностями. Идея использовать данную технологию для построения изображений трехмерных моделей появилась еще в 1960-х годах, однако аппаратные возможности того времени позволяли применять её только в оффлайн режиме, терпя значительные временные и ресурсные затраты, долгое время растеризация являлась основным методом визуализации трехмерных сцен. Трассировка лучей позволяет получать изображения с физически корректными тенями, отражениями, учитывать такие эффекты как преломление света на границе двух сред с различными показателями преломления.

Начиная с августа 2018 года, когда компания Nvidia анонсировала первый графический ускоритель серии RTX [1], аппаратные решения для применения трассировки лучей в рендеринге позволили использовать эту технологию в реальном времени. В ноябре 2020 года консорциум Khronos Group выпустил спецификацию расширений графического API Vulkan, предназначенных для трассировки лучей с использованием аппаратных возможностей графического ускорителя, включающих в себя возможность построения и использования акселерационных структур для геометрии сцены [2].

Целью данной работы является разработка программного средства отрисовки трехмерной сцены с использованием технологии трассировки лучей. В основе программного средства используется графический API Vulkan. Для хранения и загрузки трехмерных сцен был выбран один из самых широко используемых форматов – gltf. В качестве физически корректного метода моделирования материалов был выбран Metallic-Roughness подход, который описывает произвольный материал основываясь на теории микрограней [3], согласно которой любая поверхность при сильном увеличении может быть представлена как набор микроскопических зеркал, называемых микрогранями. Металличность и шероховатость, полученные в результате физических экспериментов над материалами, в виде коэффициентов используются в процессе рендеринга. Ввиду шероховатости поверхности эти микро-зеркала могут быть ориентированы в разных направлениях, что влияет на её отражающие свойства, пример приведен на рисунке 1.

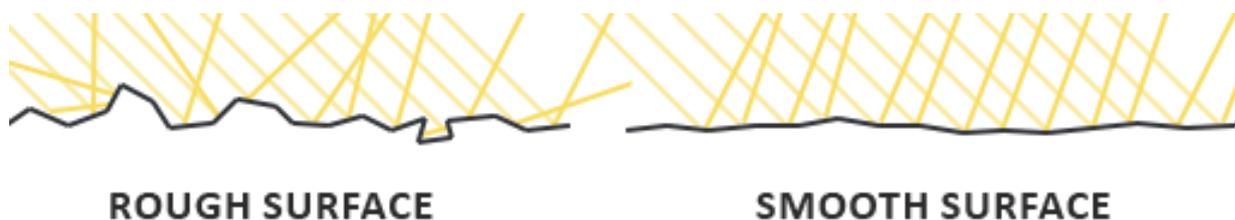


Рисунок 1 – Отражение от шероховатой и гладкой поверхностей

Для оценки освещенности поверхности в произвольной точке был выбран метод интегрирования Монте-Карло, который позволяет добиться высокой точности оценки, однако, ввиду своей стохастичности, на готовых изображениях формирует шум. Данный недостаток решается при помощи агрегации последовательности генерируемых кадров с течением времени.

Выбранный набор инструментов обеспечивает фотореалистичную, физически корректную отрисовку трехмерных сцен, описанных в одном из самых широко используемых форматов.

Список использованных источников:

1. GeForce RTX Founders Edition Graphics Cards: Cool and Quiet, and Factory Overclocked [Электронный ресурс]. – Режим доступа: <https://www.nvidia.com/en-us/geforce/news/geforce-rtx-founders-graphics-card-breakdown/>
3. Vulkan Ray Tracing Final Specification Release [Электронный ресурс]. – Режим доступа: <https://www.khronos.org/blog/vulkan-ray-tracing-final-specification-release>
3. Learn OpenGL, extensive tutorial resource for learning Modern OpenGL [Электронный ресурс]. – Режим доступа: <https://learnopengl.com/PBR/Theory>

НАХОЖДЕНИЕ КРАТЧАЙШИХ ПУТЕЙ В ГРАФЕ МЕТОДОМ МАТРИЧНОГО УМНОЖЕНИЯ

Плотников В. В., Кресс В. Д.

Белорусский государственный университет информатики и радиоэлектроники
г. Минск, Республика Беларусь

Болтак С. В. – ассистент

В работе было проведено исследование одного из методов решения проблемы APSP (All Pairs Shortest Path) в алгоритмической теории графов с использованием подхода динамического программирования. Была прослежена связь решения проблемы с перемножением матриц, а также был осуществлен вывод сложности с её последующим улучшением.

Любой динамически программируемый алгоритм включает в себя следующие этапы: задание общей структуры оптимального решения поставленной задачи, рекурсивное определение оптимального решения, вычисление значения, процесс конструирования решения.

Пусть граф задан матрицей смежности $W = (w_{ij})$. Рассмотрим кратчайший путь p из вершины i в вершину j , причем p содержит не более чем m дуг графа. В случае, если $i = j$, путь p имеет нулевой вес и не содержит дуг. В случае, если $i \neq j$, проведем разложение пути p в $i \xrightarrow{p'} k \rightarrow j$, где p' – путь, содержащий не более чем $m - 1$ дуг графа. Таким образом, общая структура наикратчайшего пути из вершины i в вершину j может быть задана формулой:

$$\delta(i, j) = \delta(i, k) + w_{kj}, \quad (1)$$

где i, j, k – вершины графа.

Пусть l_{ij}^m – вес самого короткого пути из вершины i в вершину j с не более чем m дугами. Тогда справедлива следующая рекуррентная формула:

$$l_{ij}^m = \min_{1 \leq k \leq n} [l_{ik}^{m-1} + w_{kj}], \quad (2)$$

где i, j, k – вершины графа.

Тогда вес кратчайшего пути из вершины i в вершину j может быть задан следующим образом:

$$\delta(i, j) = l_{ij}^{\{n-1\}} = l_{ij}^{\{n\}} = l_{ij}^{\{n+1\}} = \dots \quad (3)$$

В алгоритме происходит последовательное вычисление матриц вида $L^{(m)} = (l_{ij}^m)$. Для вычисления очередной матрицы используется функция $ESP(L, W)$.

```
function ESP(L, W)
n = L → getRowCount()
L' = (l'ij) – новая матрица размером n × n
for i = 1 to n
  for j = 1 to n
    l'ij = ∞
    for k = 1 to n
      l'ij = min(l'ij, lik + wkj)
return L'
```

Пользуясь повторным возведением в квадрат, можно сконструировать следующий алгоритм для решения задачи APSP с общей сложностью $\Theta(n^3 \ln n)$, где n – количество вершин графа.

```
L(1) = W
m = 1
while m < n - 1
  L(2m) = ESP(L(m), L(m))
  m = 2 * m
return L(m)
```

Список использованных источников:

1. C++ Programming Robert Sedgewick - Princeton University Addison Wesley Professional Algorithms in C++, Parts 1A4: Fundamentals, Data Structure, Sorting, Searching, Third Edition – 752 p.
2. Introduction to algorithms / Thomas H. Cormen . . . [et al.]—3rd ed – P. 686-695.
3. Library of Congress Cataloging-in-Publication Data Sedgewick, Robert, 1946 Algorithms in C++/Robert Sedgewick.—3d ed.

СЕКЦИЯ «ВЫСШАЯ МАТЕМАТИКА»

КОМПЬЮТЕРНАЯ ПРОГРАММА, ПРОГНОЗИРУЮЩАЯ РОСТ ИНФЛЯЦИИ

Захаренков М.И., Марченко В.Ю.

Белорусский государственный университет информатики и радиоэлектроники
г. Минск, Республика Беларусь

Цегельник В.В. - доктор физ.-мат. наук, профессор

В работе рассмотрена экономическая модель, а в частности модель, прогнозирующая денежную инфляцию, математически проанализированы основные алгоритмы этой модели. Данный анализ положил основу создания компьютерной программы, прогнозирующей денежную инфляцию. Программа будет являться очень актуальной в наше время. В будущем также планируется создание мобильного приложения, работающего на основе данных анализов.

Существует такое понятие, как «Денежная иллюзия». Оно показывает, что люди не всегда способны правильно воспринять реальную цену денежных средств. То есть, большинство людей смотрят на цифровой номинал денег, когда более важным является количество продукции, которую возможно приобрести на эти деньги. Люди совершают ошибку, не учитывая следствие инфляции.

За время прошедшего десятилетия наблюдалась смена режима некоторых стран мира в пользу таргетирования инфляции, что сделало актуальной одну из самых важных проблем человечества, а именно прогнозирования инфляции[2]. Развилась и расширилась большинство моделей, прогнозирующих инфляцию, так например стали больше использовать базовые линейные регрессии, также получили широкое использование векторные авторегрессии, ARIMA-модели и, так называемая, P*-модель.

Данная модель является одной из одной из наиболее подходящих для анализа процессов инфляции. Её основа - количественная теория денег[1]. P*-модель показывает равновесный уровень цен (p^*), согласующийся с предложением совместно с установленным предположительным выпуском (y^*) и скоростью обращения денежных средств (v^*).

В представленной модели предполагается, что если фактический уровень цен отклоняется от равновесного уровня, то развивается инфляционный процесс. То есть при фактическом уровне ниже равновесного, через некоторое время будет расти инфляция.

Для создания компьютерной программы требуется математически рассмотреть P* - модель. Для анализа инфляционных процессов, основанных на этой модели, нужно использовать количественную теорию денег:

$$M_t V_t = P_t Y_t, \quad (1)$$

где M_t - денежные средства; V_t - скорость обращения денежных средств; P_t - ценовой уровень в стране; Y_t - выпуск продукции с постоянными ценами.

Формулу (1) перепишем как

$$P_t = \frac{M_t V_t}{Y_t}. \quad (2)$$

Равновесный уровень цен (p^*), согласующийся с предложением совместно с установленным предположительным выпуском (y^*) и скоростью обращения денежных средств (v^*) можно представить как

$$P_t^* = \frac{M_t V_t^*}{Y_t^*}. \quad (3)$$

Поделив уравнение (3) на (2) и получим новое уравнение:

$$\frac{P_t^*}{P_t} = \frac{M_t V_t^*}{M_t V_t} * \frac{Y_t}{Y_t^*}, \quad (4)$$

Теперь взяв натуральный логарифм от обеих частей уравнения (4), переопределим полученные переменные, что упростит уравнение (4), получим

$$p_t^* - p_t = (v_t^* - v_t) + (y_t - y_t^*). \quad (5)$$

Пропустим ряд преобразований. Добавим переменную λ , являющуюся коэффициентом корректировки равновесия. Данная переменная изменяется в диапазоне $[0;1]$. Тогда $1/\lambda$ отразит скорость корректировки[3]. Уравнение (6) получит вид:

$$\Delta \ln P_t^h = \Delta \ln P_t^F + \lambda \left(\ln \left(\frac{P_{t-1}^F}{P_{t-1}^H} \right) - \ln \left(\frac{P_{t-1}^F}{P_{t-1}^{H*}} \right) \right). \quad (6)$$

В последнем уравнении также учитывается влияние уровня цен иностранной продукции на уровень цен в стране в целом.

Если учесть, что когда кривая обменного курса валюты лежит выше уровня равновесия, ценовой уровень в стране растёт и это в конечном итоге вызовет рост инфляции, можно привести уравнение (6) к виду:

$$\Delta p_t = a_0 + a_1(p_{t-1}^* - p_{t-1}) + a_2(reer_{t-1} - reer_{t-1}^*) + \sum_{i=1}^n (\beta_i \Delta p_{t-i}) + \sum_{j=1}^m (\delta_j \Delta reer_{t-j} + \varepsilon_t), \quad (7)$$

где $reer_t$ показывает изменчивость курса обмена валюты.

Таким образом, можно использовать основные выражения, описанные в данной работе, для создания программы(приложения), подсчитывающего рост инфляции. В дальнейшем при доработке нашей программы можно было бы учесть ещё и другие факторы, чтобы повысить точность вычисления инфляции. Данная программа будет очень полезной в качестве мобильного приложения, т.к. не позволит людям оказаться в условиях «Денежной иллюзии».

Список использованных источников:

1. Hallman, J. Is the price level tied to the M2 monetary aggregate in the long run? / J. Hallman, R. Porter, D. Small // The American Economic Review. – 1991. – № 81(4). – P. 841–858.
2. Kool, J. The P-Star model in five small economies / J. Kool, J. Tatom // Federal Reserve Bank of SL Review. – 1994.
3. Математическая статистика [Электронный ресурс]: учеб.-метод. пособие / М. В. Дубатовская, С. В. Rogozin, Е. И. Васенкова.

ПОЕДИНОК «ГРУЗОВИК – ПЕШЕХОД»: ОПТИМАЛЬНЫЕ ПО БЕЗОПАСНОСТИ ТРАЕКТОРИИ

Кулик А.В., Кугач А.В.

Белорусский государственный университет информатики и радиоэлектроники
г. Минск, Республика Беларусь

Луцакова И.Н. – канд. физ.-мат. наук

Рассматривается ситуация, когда пешеход пытается увернуться из-под колес приближающегося к нему грузовика. Исследуются различные формы траекторий движения пешехода (прямые и кривые линии) с целью найти траекторию, оптимальную по безопасности. Приводится пример траектории, которая позволяет избежать столкновения с самым быстрым грузовиком.

Представьте, что вы стоите на середине дороги. По направлению к вам мчится грузовик, и за рулем - невнимательный водитель, который вас не видит. Первое, что может прийти вам в голову, это броситься прямо к обочине. Этот путь, хоть и самый короткий, но не самый безопасный. Возможны две интерпретации того, какую траекторию движения считать наиболее безопасной [1,2].

Рассмотрим первую интерпретацию. Предположим, что вы стоите посередине дороги в x метрах от обочины и замечаете грузовик в y метрах от вас, приближающийся с постоянной скоростью $v_{гр}$. Вы можете двигаться со скоростью $v_ч$. Рисунок 1 иллюстрирует эту ситуацию. Рассмотрим в качестве возможных траекторий вашего движения прямые линии. Пусть $v_{гр} > v_ч > 0$. Кроме того, достаточно рассматривать значения θ из интервала $(-\frac{\pi}{2}, \frac{\pi}{2})$. Тогда длина пути, который вы проделаете к обочине, равна $D_ч(\theta) = \frac{x}{\cos \theta}$, и время, которое займёт этот путь, будет равно $T_ч(\theta) = \frac{x}{v_ч \cos \theta}$.

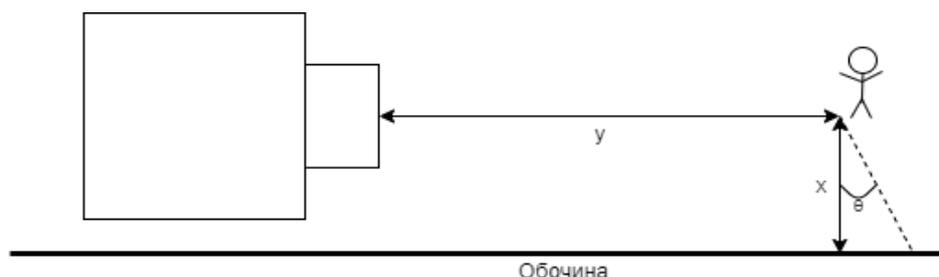


Рисунок 1. Схема приближающегося к человеку грузовика

Двигаясь по такому пути, расстояние между вами и грузовиком в момент, когда вы достигнете обочины, равно:

$$D(\theta) = y + x \tan \theta - v_{гр} T_ч(\theta) = y + x \tan \theta - \frac{v_{гр}}{v_ч} * \frac{x}{\cos \theta}. \quad (1)$$

Сохраняя все остальные параметры постоянными, значение θ , которое максимизирует расстояние между вами и грузовиком, будет равно $\arcsin(\frac{v_ч}{v_{гр}})$.

Рассмотрим вторую интерпретацию. Что если грузовик движется с переменной скоростью или вы не знаете его скорость, а может, вы любитель острых ощущений и хотите побегать от многотонной машины? В этом случае будем рассматривать траектории движения более общего вида, чем прямые. Хотя это может показаться странным, но иногда движение по кривой не только поможет избежать столкновения с грузовиком, но и ещё сэкономить время.

В данной интерпретации изменение угла $\theta = \arctan\left(\frac{x}{y}\right)$ задается выражением:

$$d\theta = \frac{ydx - xdy}{x^2 + y^2} = \frac{-yv_q \cos(\theta) - xv_q \sin(\theta) + xv_{tr}}{r^2} dt = \frac{-v_q r \cos(\theta)^2 - v_q r \sin(\theta)^2 + v_{tr} \sin(\theta)}{r^2} dt, \quad (2)$$

где $y = z - v_{гр}t$, $v = \left(\frac{dz}{dt}, \frac{dx}{dt}\right) = (v_q \sin \theta, -v_q \cos \theta)$, $r = OP = \sqrt{x^2 + y^2}$, $y = r \cos(\theta)$, $x = r \sin(\theta)$ в соответствии с рисунком 2.

Рассмотрим общую ситуацию $0 < v_{гр} < v_{гр.max} = \frac{v_q}{\sin(\theta_0)}$, где θ_0 – это исходное значение θ в момент времени $t = 0$. Отметим, что в этом случае $\frac{d\theta}{dt} = \frac{-v_q + v_{гр} \sin(\theta)}{r} < 0$, угол θ будет уменьшаться со временем (становясь равным нулю, когда вы достигнете обочины в точке $x = 0$), и ваша траектория будет постепенно приближаться к перпендикулярю к обочине, поэтому вы будете приближаться к обочине быстрее, чем в исходном решении для прямой линии.

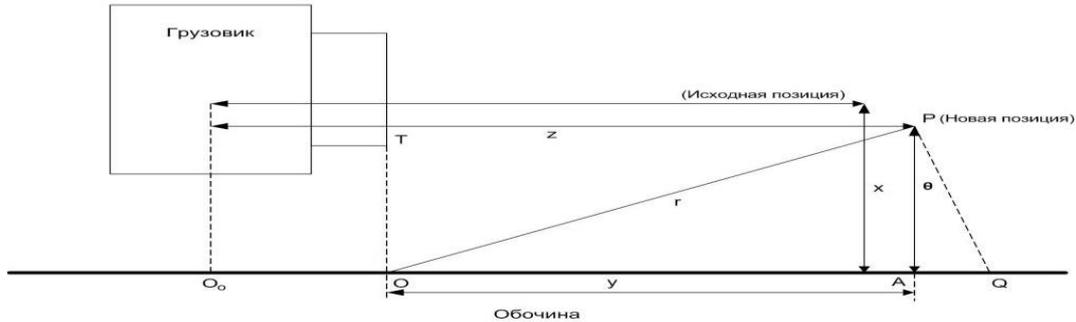


Рисунок 2. Схема приближающегося к человеку грузовика.

В работе получена [2] получена следующая формула для оптимальной траектории движения с точки зрения второй интерпретации:

$$r = \frac{r_0 \left(1 - \frac{v_{гр}}{v_q} \sin \theta_0\right)}{1 - \frac{v_{гр}}{v_q} \sin \theta} = \frac{L}{(1 - \varepsilon \sin \theta)}. \quad (3)$$

Здесь $L = r_0 \left(1 - \frac{v_{гр}}{v_q} \sin \theta_0\right)$, $\varepsilon = \frac{v_{гр}}{v_q}$. Это полярное уравнение кривой второго порядка с эксцентриситетом ε . Поскольку грузовик едет быстрее, чем вы бежите, кривая будет иметь эксцентриситет $\varepsilon > 1$, т.е. это будет гипербола. В случае медленного грузовика, движущегося со скоростью $v_{гр} < v_q$, получим эксцентриситет $\varepsilon < 1$, и кривая будет описывать эллипс. Если $v_{гр} = v_q$, то $\varepsilon = 1$, и кривая описывает параболу.

Нами было проведено компьютерное моделирование безопасных траекторий пешехода при различных исходных данных. На рисунке 3 изображена возможная траектория движения пешехода к обочине.

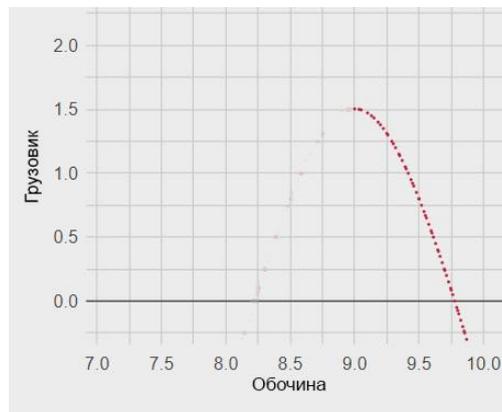


Рисунок 3. Путь пешехода к обочине по гиперболе ($\varepsilon > 1$).

Список использованных источников:

1. Truck Versus Human: Mathematics Under Pressure /E. Field, R. Ivison, A. Reyher, S. Warner// The College Mathematics Journal, 2014, 45:2, P. 116-120.
2. Truck Versus Human 2.0: Mathematical Follow-Up Under Increasing Pressure, and How Kepler's Laws Come to the Rescue./M.A. Lerma // The College Mathematics Journal, 2021, 52:1, P.22-30.

ТЕОРИЯ ЧИСЕЛ В АССИМЕТРИЧНОМ ШИФРОВАНИИ

Протьюко М.А.

Белорусский государственный университет информатики и радиоэлектроники
г. Минск, Республика Беларусь

Борисенко О.Ф. И.О. – канд. физ.-мат. наук, доцент

С широкой распространенностью информационных технологий растет потребность в специалистах широкого профиля, способных не только создавать программное обеспечение, но и исключать вмешательство третьих лиц в процессе его работы, тем самым, защищая конфиденциальность себя и клиента. Чтобы обеспечить данные возможности, необходимо глубокое понимание основ шифрования, начинающихся с теории чисел.

Ключевые слова: криптографическая стойкость, RSA алгоритм, ассиметричное шифрование, случайная величина

Один из известных алгоритмов шифрования, использующийся и на данный момент, это алгоритм RSA – криптографический алгоритм с открытым ключом, основывающийся на вычислительной сложности задачи факторизации больших целых чисел. Рассмотрев его поподробнее, можно ознакомиться с основными принципами шифрования.

Для того, чтобы создать криптографический стойкий шифр (способный противостоять расшифровке без знания ключа, анализу повторяемости, «грубой силе») необходимо найти одностороннюю функцию, используя которую на некотором множестве, можно было бы получить иное соответствующее данному множеству, обратимое при знаниях неких дополнительных условий (аргументов функции, в т.ч. ключей). Или же:

$$f(m, n) = c \quad (1)$$

$$f(c, e) = m, \quad (2)$$

где m – начальное множество, c – конечное (зашифрованное), n и e – открытый и закрытый ключи соответственно.

Причем невозможно найти такую $f^{-1}(c) = m$ за конечный промежуток времени, такой, что после дешифровки информация все еще будет востребована.

Односторонняя функция RSA выглядит так:

$$E(m) = m^e \pmod n \quad (3)$$

Алгоритм построения RSA следующий:

1. Выбрать два случайных простых числа p и q
2. Найти модуль:

$$n = p \cdot q \quad (4)$$

3. Найти функцию Эйлера:

$$\varphi(n) = (p-1) \cdot (q-1) \quad (5)$$

4. Найти взаимно простое число e , $1 < e < \varphi(n)$, где e – взаимно простое с значением $\varphi(n)$, где e – открытая экспонента
5. Найти d из уравнения

$$d \cdot e \equiv 1 \pmod{\varphi(n)}, \quad (6)$$

где d – секретная экспонента

При применении алгоритма на практике, были найдены следующие нюансы:

1. Нахождение случайных чисел, в особенности, когда в последовательности их больше двух, довольно затруднительно (не существует абсолютно удобных алгоритмов для решения данной задачи)

2. После нахождения случайных чисел, их нужно проверить на простоту, причем сделать это необходимо при минимальном количестве вычислений (за достаточно короткое время), т.е. количество проводимых тестов (Ферма, Люка, и т.д.) ограничено.

3. Нахождение d из уравнения в пункте 5 может требовать оптимизации вычислений с использованием теории чисел, причем, чем больше значения e и $\varphi(n)$, тем сложнее ее найти.

Решение каждого из вышеперечисленных вопросов требует финансовых и временных затрат. Из чего следует вывод: для того, чтобы создать алгоритм шифрования, необходимо четко знать в каких сферах он будет применяться (шифрование номеров банковских карт, электронная подпись, сертификация информации и т.д.), какими средствами данный алгоритм будет реализовываться (вычислительные мощности при вычислении ключей, максимальное время расшифровки/шифровки, объемы шифрования)

Список использованных источников:

1. НОУ ИНТУИТ. Лекция 11: Основные положения теории чисел, используемые в криптографии с открытым ключом [электронный ресурс]: <https://intuit.ru/studies/courses/691/547/lecture/12389>
2. Ассиметричное шифрование на практике [электронный ресурс]: <https://habr.com/ru/post/449552/>
3. Информация о числах. Свойства и характеристики одного числа [электронный ресурс]: <https://aboutnumber.ru/197>
4. Тест простоты Люка [электронный ресурс]: https://ru.wikipedia.org/wiki/Тест_простоты_Люка

5. Асимметричный алгоритм для генерации коротких серийных номеров [электронный ресурс]:

<https://habr.com/ru/post/69623/>

6. Стройникова Е.Д. Основы прикладной алгебры: учеб.-метод. пособие/ Е.Д.Стройникова. -Минск:БГУИР,2010. -120 с.:ил.

МАТЕМАТИЧЕСКАЯ МОДЕЛЬ ОСЦИЛЛЯТОРА НА ТРЕНИИ

Степанов Н.Ю.

Белорусский государственный университет информатики и радиоэлектроники
г. Минск, Республика Беларусь

Князюк Н.В. – канд. физ.-мат. наук, доцент

В данной работе исследуется математическая модель так называемого осциллятора на трении или осциллятора Тимошенко, представляющего собой массивное тело, приводимое в периодическое движение взаимодействием с двумя горизонтальными вращающимися с одной скоростью, но в разные стороны, валами. Построенная теоретическая модель учитывает изменение коэффициента трения, зависящего от взаимной скорости между массивным телом и валами.

Качественное описание. Балка, расположенная на валах с некоторым смещением, подвержена действию нескомпенсированных сил трения со стороны точек опоры. Из-за превосходства одной силы трения над другой, балка начинает свое движение. Из-за смещения в сторону другого вала, сила реакции опоры с его стороны больше, а, следовательно, больше и сила трения. Балка останавливается и начинает движение в противоположную сторону. Процесс повторяется. Балку можно назвать осциллятором.

Математическая модель. Пусть d – расстояние между валами, l – длина балки, x – смещение балки относительно центра системы, N_1, N_2 – силы реакции опоры первого и второго вала соответственно, $F_{тр1}, F_{тр2}$ – силы трения со стороны первого и второго вала соответственно (рисунок 1).

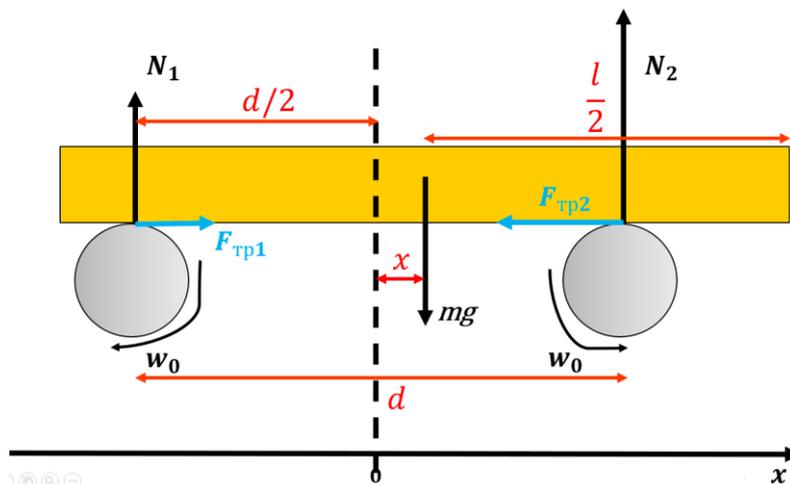


Рисунок 1 – Схематичное изображение валов, балки, сил, действующих на балку

Имеет место система из трех уравнений:

$$\begin{cases} N_2 \left(\frac{d}{2} - x \right) = N_1 \left(\frac{d}{2} + x \right), \\ N_1 + N_2 = mg, \\ F_{тр2} - F_{тр1} = ma, \end{cases} \quad (1)$$

где $F_{тр1} = \mu N_1$, $F_{тр2} = \mu N_2$ согласно закону Амальтона – Кулона. Из системы получаем

$$ma = -\frac{2\mu mg}{d} x,$$

откуда $a = -\frac{2\mu g}{d} x$, $\omega^2 = \frac{2\mu g}{d}$, где a – ускорение, ω – угловая скорость. Закон движения балки имеет вид $x(t) = x_0 \cos(\omega t)$, график закона движения представлен на рисунке 2.

По графику видно, что с течением времени амплитуда колебаний уменьшается, что не предусмотрено теорией. Экспериментально было показано, что коэффициент трения μ зависит от взаимной скорости между валом и балкой (рис.3).

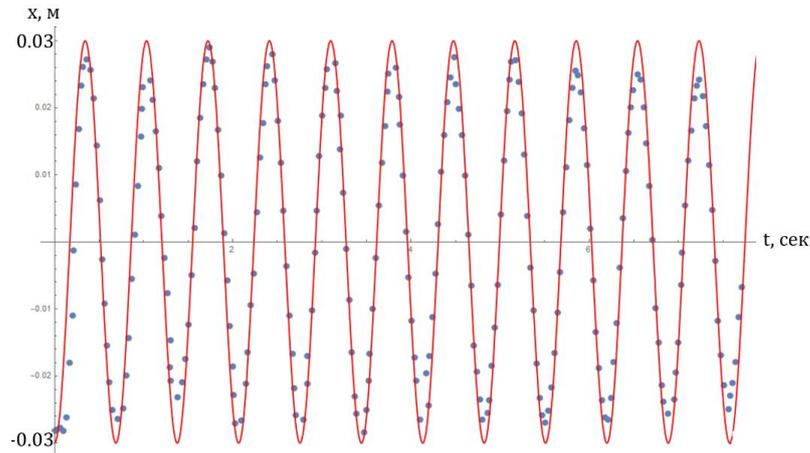


Рисунок 2 – График зависимости смещения центра балки относительно середины расстояния между валами, где красная косинусоида - теоретический закон движения, синие точки - экспериментальные точки.

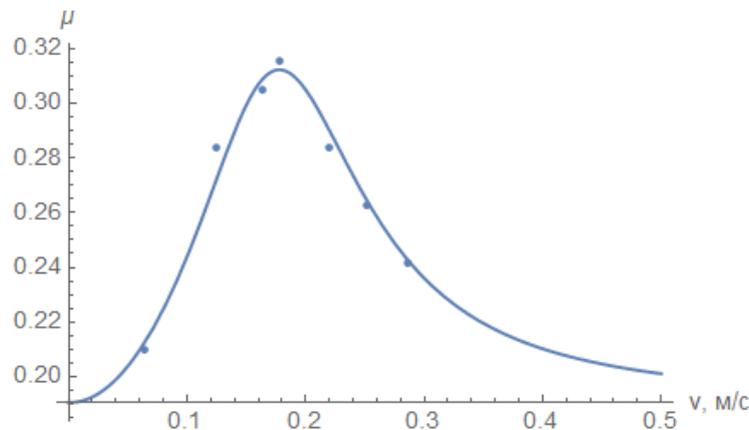


Рисунок 3 – График зависимости коэффициента трения μ от взаимной скорости между валом и балкой, где синяя кривая – аппроксимация, синие точки - экспериментальные точки.

Полученный результат позволяет сделать вывод, что коэффициент трения μ не является константой, а сложно зависит от взаимной скорости между валом и балкой. Построим расширенную математическую модель, преобразовав систему уравнений (1) с учетом того, что $\mu_{1,2} = \mu(V \mp x'(t))$. Имеем

$$\frac{d}{g} x''(t) = \mu[V - x'(t)] \left(\frac{d}{2} - x(t) \right) - \mu[V + x'(t)] \left(\frac{d}{2} + x(t) \right), \quad (2)$$

где $V = 2\pi vr$ – базовая скорость вала, v – частота вращения вала, r – радиус вала.

Предположим, что $V \gg x'(t)$, и построим линейную математическую модель осциллятора на трении, где $\mu[V \pm x'(t)] = \mu[V] \pm k x'(t)$. Тогда, линеаризация зависимости $\mu[v]$ дает $k = \mu'[V]$. Получим линейное однородное дифференциальное уравнение второго порядка с постоянными коэффициентами:

$$x''(t) + kgx'(t) + \frac{2\mu[V]g}{d} x(t) = 0. \quad (3)$$

Аналитически решая задачу Коши для полученного дифференциального уравнения (3) с начальными условиями $x(0) = x_0$, $x'(0) = 0$, где x_0 – начальное смещение, получим два закона движения.

Если $\frac{2\mu g}{d} - \frac{g^2 k^2}{4} > 0$, то

$$x(t) = x_0 e^{-\frac{gk}{2}t} \left(\cos \left(t \sqrt{\frac{2\mu g}{d} - \frac{g^2 k^2}{4}} \right) + \frac{kg}{2\sqrt{\frac{2\mu g}{d} - \frac{g^2 k^2}{4}}} \sin \left(t \sqrt{\frac{2\mu g}{d} - \frac{g^2 k^2}{4}} \right) \right). \quad (4)$$

Полученный закон движения описывает медленное осциллирующее «затухание» (возможно как затухание ($k < 0$), так и раскачка колебаний ($k > 0$)).

Если $\frac{g^2 k^2}{4} - \frac{2\mu g}{d} > 0$, то

$$x(t) = x_0 e^{-\frac{gk}{2}t} \left(\left(\frac{1}{2} + \frac{kg}{4\sqrt{\frac{g^2 k^2}{4} - \frac{2\mu g}{d}}} \right) e^{t\sqrt{\frac{g^2 k^2}{4} - \frac{2\mu g}{d}}} + \left(\frac{1}{2} - \frac{kg}{4\sqrt{\frac{g^2 k^2}{4} - \frac{2\mu g}{d}}} \right) e^{-t\sqrt{\frac{g^2 k^2}{4} - \frac{2\mu g}{d}}} \right). \quad (5)$$

Такой закон движения описывает быстрое затухание без колебаний, которое на практике не встречалось.

Построенная линейная модель осциллятора дает некорректные результаты. На основе экспериментальных данных методом наименьших квадратов была получена следующая зависимость коэффициента трения μ от взаимной скорости между валом и балкой:

$$\mu(v) = \text{sign}(v) \left(\mu_0 + \frac{av^2}{(v-v_0)^4 + b} \right), \quad (6)$$

где по результатам экспериментов $\mu_0 = 0.1905$, $a = 1.346 \cdot 10^{-3} \text{ (м/с)}^2$, $b = 2.52 \cdot 10^{-4} \text{ (м/с)}^4$, $V = 7.84 \cdot 10^{-2} \text{ м/с}$. Зависимость (6) коэффициента трения μ от взаимной скорости между валом и балкой была использована в численном решении дифференциального уравнения (2), в результате чего получена хорошая сходимость численного решения математической модели с практическими результатами.

КОМПЬЮТЕРНОЕ ЗРЕНИЕ И ЕГО ОСНОВНЫЕ ЗАДАЧИ

Круглая А.А., Харкевич А.П., Щедрова Д.С.

*Белорусский государственный университет информатики и радиоэлектроники¹
г. Минск, Республика Беларусь*

Рыкова О.В. – канд. физ.-мат. наук

В данной работе мы углубились в такую важную область обучения, как компьютерное зрение. Дано целостное описание появления данной области, её практического применения и дальнейшего развития. Так же нами были рассмотрены две основные задачи компьютерного зрения- Object Detection и Semantic Segmentation. Были изучены основные теоретические и практические методы решения этих задач (5 в первом случае и 3 во втором).

Оказывается, что то, что большинство людей воспринимает как должное — способность видеть и обрабатывать увиденное — чрезвычайно трудно воспроизвести в машинах. Именно на это нацелено компьютерное зрение (CV).

Компьютерное зрение описывает процесс, когда компьютер, использующий алгоритмы искусственного интеллекта, может идентифицировать и обрабатывать изображения (фотографии, видео и т. д.). А затем создавать соответствующие результаты анализа, поскольку компьютер действительно может «понимать» контент. В частности, компьютерное зрение может классифицировать, идентифицировать, проверять и обнаруживать объекты.

Есть три основных компонента компьютерного зрения.

1. Получение изображения: когда цифровая камера захватывает изображение, она создает цифровой файл, состоящий из нулей и единиц.
2. Обработка: алгоритмы используются для определения основных геометрических элементов для построения изображений из двоичных данных.
3. Анализ: на этом последнем этапе компьютерного зрения данные анализируются. Алгоритмы высокого уровня используются для принятия решений на основе изображений.

В последние годы компьютерное зрение стало активно использоваться в промышленности. Более 100 различных компаний начали производить системы машинного зрения. Были разработаны специальные светодиоды для систем машинного зрения, расширились функции световых сенсоров и архитектуры управления системами компьютерного зрения. Это значительно расширяло их функционал при постоянном снижении цен на такие системы.

Развитие Интернета в 90-х годах привело к накоплению огромных массивов оцифрованных изображений, которые стали доступны для анализа в режиме онлайн. Начали активно развиваться программы распознавания печатного и рукописного текста, а также распознавание лиц на фото и видео.

В последние 10-15 лет в связи с развитием облачных технологий, виртуализации вычислений, и таких прикладных областей, как машинное и глубокое обучение, Big Data и пр., развитие компьютерного зрения получило новый импульс для роста.

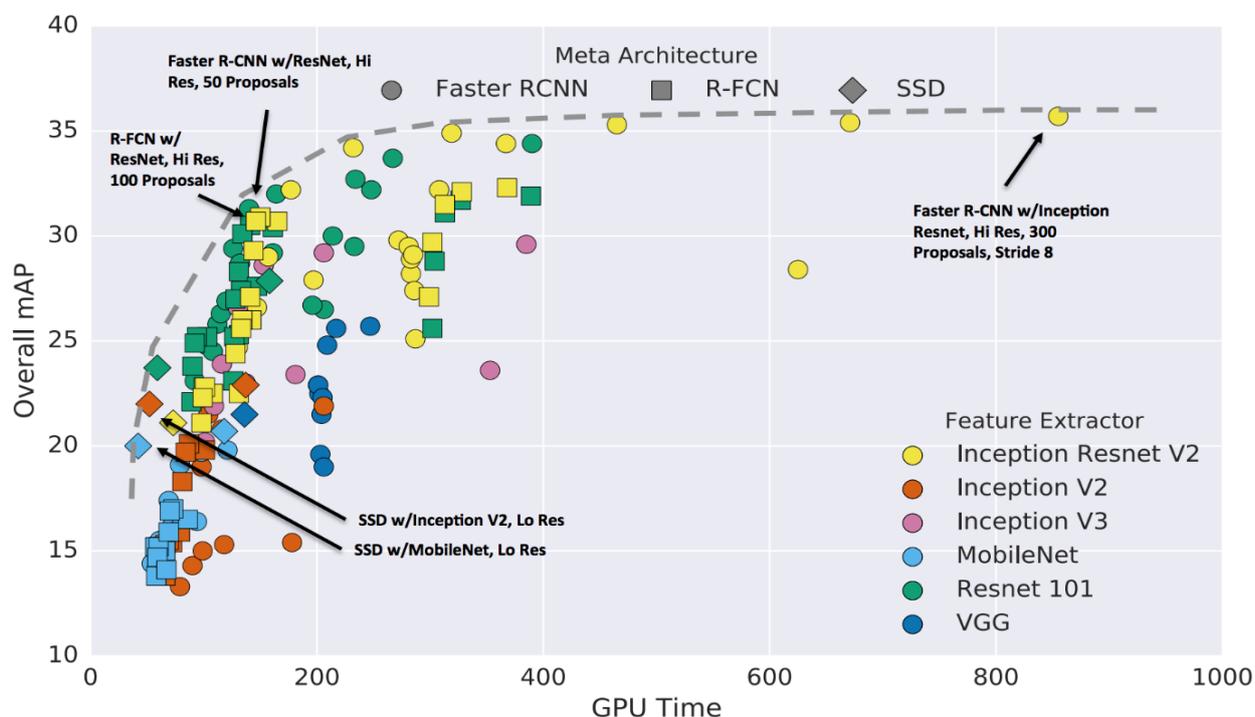
Развитие систем компьютерного зрения пока далеко от реализации всех его возможностей, но они быстро развиваются, и диапазон их применений ширится.

Далее мы проводили сравнительный анализ подходов и сделали следующие выводы:

1. Подход brute force не имеет практического применения, основная сложность в использовании- многократный выбор различных маленьких сегментов на изображении.
2. Подход R-CNN тратит примерно 1 минуту на обработку одного изображения. Основное время тратится на запуск CNN.
3. Подход Fast R-CNN'15 тратит примерно 3 секунду на обработку одного изображение. Основное время он тратит на поиск оптимальных регионов.
4. Подход Faster R-CNN'15 тратит примерно 0.2 секунду на обработку одного изображение. Он показывает наилучшее качество среди данных подходов.
5. Подход YOLO'15 работает со скоростью 45 изображений в секунду, а его быстрая версия работает ещё быстрее- до 155 изображений в секунду! Он работает менее точно, чем CNN подходы, зато сверхбыстро.

Вообще, как правило, ищется баланс между временем работы подхода и его точностью.

Этот баланс хорошо иллюстрирует изображение ниже – на нём можно увидеть как быстро и качественно работает рассмотренные подходы на разных архитектурах.



Для реализации методов Object detection мы выбрали CNN, RCNN и YOLO. Основанием для этого послужило то, что CNN является основой в Object detection, RCNN одна из основных надстроек над CNN. YOLO считается одним из самых прогрессивных методов решения задачи Object detection в наши дни. Данные подходы рассмотрены на конкретных задачах.

Список использованных источников:

1. <https://shalaginov.com/2020/05/16/computer-vision-history/>
2. <https://ru.wikipedia.org>
3. <https://medium.com/profisoft>
4. <https://www.jetinfo.ru/computer-vision-technology-review/>
5. <https://dlcourse.ai/>

СЕКЦИЯ «ЭЛЕКТРОННЫЕ ВЫЧИСЛИТЕЛЬНЫЕ МАШИНЫ»

ИНФОРМАЦИОННО-КОММУНИКАТИВНАЯ МОДЕЛЬ И УПРАВЛЕНИЕ УЧЕБНЫМ ПРОЦЕССОМ

Абдулвахид М.А.

Белорусский государственный университет информатики и радиоэлектроники
г. Минск, Республика Беларусь

Дадыкин А.К. – канд. тех. наук, доцент

Современные комплексные проекты в области образования требуют совместных усилий специалистов разных отраслей науки и техники. Отличия в терминологии, методологии и языке создают значительные трудности и препятствия в этом. Автором предложена информационно-коммуникативная модель, демонстрирующая в единой структуре механизмы общения между людьми в рамках одного или нескольких языков, взаимодействия между человеком и компьютером или обмена информацией между техническими устройствами. Применение современных IT-технологий совместно с использованием эффективных моделей получения знаний и навыков позволяет существенно ускорить обучение и повысить его успешность.

XXI век поставил перед обществом и наукой новые задачи в сфере образования, обучения и обеспечения передачи накопленного научного знания, передовых технологий и культурных традиций. Старые формы, сформировавшиеся много лет и даже столетий назад в совершенно других экономических, политических и социальных условиях, не соответствуют современному уровню развития цивилизации. А новые, интегрирующие в себе достижения научно-технической революции и компьютерно-информационных технологий, пока не могут конструктивно объединить имеющийся технический потенциал с разработками в гуманитарной сфере – педагогике, методологии и психолингвистике в силу совершенно разных языков описания предметной сферы, применяемых методов, сложности организации и согласования комплексных междисциплинарных проектов.

Для обеспечения взаимопонимания между представителями образования и информационных технологий особенное значение имеет согласование терминологии, научных моделей и проведение визуально-топологического анализа на стыке точных и гуманитарных наук. Интегрирующей основой для согласования терминологии, методов, процедур и структур вполне может стать визуальный мета-язык [1] и инструменты системного анализа, как связующего звена между теорией и практикой, IT и гуманитарной сферой, наукой и социальными процессами.

Примером реализации такого подхода является информационно-коммуникативная модель, с одной стороны демонстрирующая процессы в информационно-вычислительных системах и системах связи, а с другой стороны представляющая собой модель общения в системе “преподаватель-ученик”, “говорящий-слушатель”, или просто служащая моделью межличностной коммуникации, включая невербальные компоненты (рисунок 1).

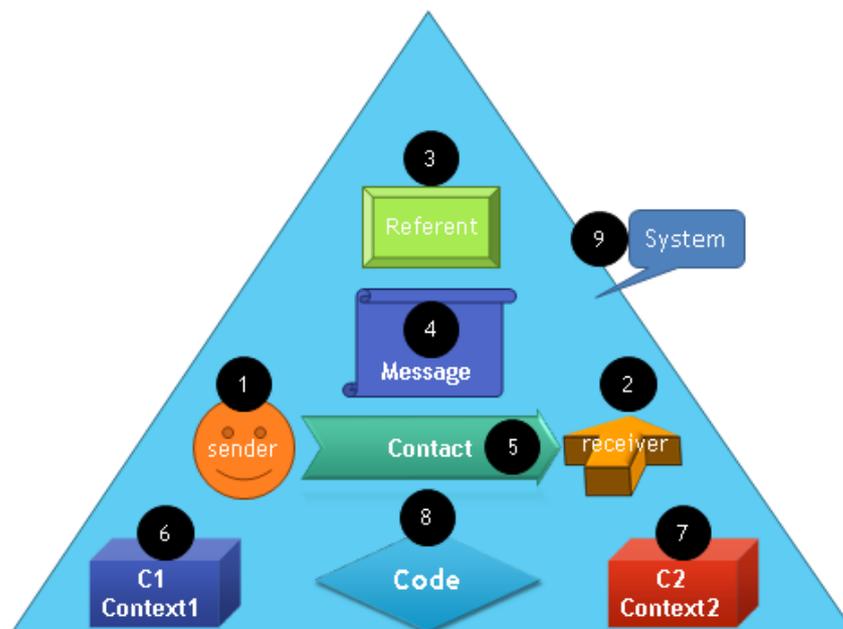


Рисунок 1 – Информационно-коммуникативная модель

Процесс передачи информации с помощью языка является важнейшим в человеческой культуре и даже, по некоторым теориям, является определяющим отличием человека как вида от

других живых существ. Подобная структура присуща как отношениям между группами людей и организациями, так и межличностному общению, и даже простой передаче информации между техническими устройствами. Но в каждой отрасли науки или техники выработалась собственная модель, со своей структурой и терминологией, поэтому разработки в одной сфере не находят применения в другой, а создание комплексных проектов (например, LMS – компьютерных систем управления образовательными процессами) сопряжено с большими финансовыми затратами и организационными сложностями.

Информационно-коммуникативная модель может служить связующим звеном для внедрения в компьютерные системы управления учебным процессом нового поколения инструментов, обеспечивающих не только структурирование и передачу знаний от преподавателя к ученику, но и количественное и качественное управление уровнем навыков и компетенций ученика и параметрами его реального продвижения по кривой обучения.

Это особенно важно в сфере обучения и переучивания взрослых. Как было показано в работах А. Бандуры [2], научение происходит в процессе взаимодействия влияния внешней среды, поведения человека и свойств его личности. При этом наличие предыдущего опыта ведет к тому, что вместо обычного логарифмического роста кривая обучения приобретает характерные изгибы с несколькими минимумами, так называемыми «барьерами преодоления» (рисунок 2).

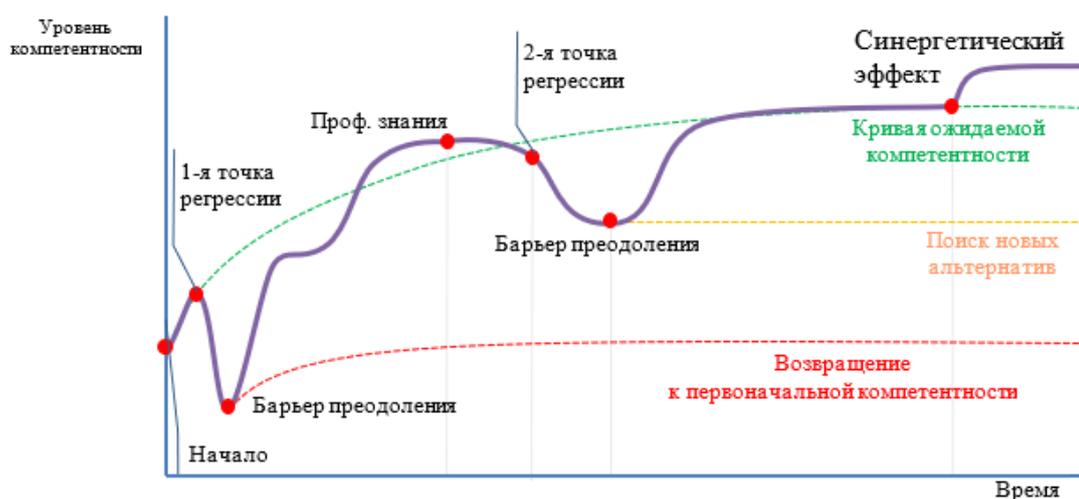


Рисунок 2 – Кривая обучения А. Бандуры

Применение современных IT-технологий совместно с использованием эффективных моделей получения навыков уменьшает и даже полностью устраняет влияние таких барьеров и соответственно ускоряет обучение и повышает его успешность [3]. Это происходит за счет передачи синергетического эффекта на все этапы формирования навыков, что особенно важно в процессе обретения иностранного языка [4].

Особенность предлагаемого подхода заключается в логической взаимосвязи всей системы, экономичности по времени усвоения материала и динамике использования языковых структур. Получение результатов должно достигаться за счет поддерживаемой не ниже критического уровня интенсивности усилий обучаемого на достаточно длительном промежутке времени (месяцы), что играет решающую роль в занятиях иностранным языком [5].

Интеграцию в едином проекте обеспечивают инструменты системного анализа и визуального моделирования, позволяющие эффективно согласовывать структуру предметной области с точки зрения совершенно разных отраслей человеческой деятельности, в каждой из которых присутствует своя терминология, методология и устоявшаяся парадигма.

Список использованных источников:

1. Дадыкин, А.К., Диброва, В.А. Обобщенная визуальная модель структуры образовательного процесса / *Современные тенденции в дополнительном образовании взрослых: материалы III Междунар. научно-метод. конф.*, Минск, окт. 2016 г.: в 2 ч. – Минск: РИВШ, 2016. – Ч. 1. – С. 62-67.
2. Bandura, A., Walters, R.H. *Social learning and personality development*. New York: Holt, Rinehart, & Winston, 1963. – 298 p.
3. Варгас, Дж. Анализ деятельности учащихся. Методология повышения школьной успеваемости / Дж. Варгас. – М.: Оперант, 2015. – 480 с.
4. Aliaksei Dadykin. *Information and Communication Technology and Callan Method* / *Association for Computing Machinery, NY, United States // ICEDS 2020: Proceedings of the 2020 International Conference on Education Development and Studies, Paris, France, March 2020.* – P. 44-52.
5. Dadykin, A., Tahini, I *Intersection of Technology and Second Language Acquisition* / *ICERI 2019 Proceedings 12th annual International Conference of Education, Research and Innovation, November, 2019, Seville, Spain.* – P. 8213-8220.

ПРИЗНАКИ И КРИТЕРИИ НЕСТАНДАРТНОГО ПОВЕДЕНИЯ ПОЛЬЗОВАТЕЛЯ

Байдун Д.Р.

Белорусский государственный университет информатики и радиоэлектроники
г. Минск, Республика Беларусь

Насуро Е.В. – канд. техн. наук, доцент

Различные способы идентификации пользователей дают неравнозначные уровни защиты от несанкционированного доступа в систему. Использование комбинаций признаков, полученных при анализе действий пользователя, позволят повысить безопасность систем. Доклад содержит информацию о доступных для анализа характеристиках действий пользователя, сбор которых не требует установки дополнительного оборудования и основан на уже имеющихся аппаратных средствах. Методы машинного обучения позволяют получать персонализированные признаки, характерные для одного пользователя, которые невозможно подделать, так как их сбор проводится неявно в фоновом режиме.

Признаки и характеристики.

Ниже представлены признаки, которые рассматриваются в качестве значимых параметров при отслеживании действий пользователя для выявления аномального поведения.

1. Распознавание лица. Используя камеру устройства, можно сравнить изображение, сохраненное ранее с изображением, полученным в режиме реального времени [1].

2. Имя пользователя и пароль. Проверка подлинности введенного пароля. Надежность этого метода можно повысить, сократив количество неправильных попыток. Дополнительным критерием может выступать скорость ввода пароля [2].

3. Среднее время и расписание активности. Например, рабочие аккаунты, при необходимости, можно привязывать к режиму работы сотрудника. В остальное время использование служебных учетных записей можно ограничить.

4. Активность пользователя. Подразумевается время активности пользователя как в отдельно взятых программах, так и в определенных областях программ (например, хранилище паролей в браузерах, поисковые запросы, отключение алгоритмов защиты компьютера – антивирусных программ, брандмауэра). Например, операционные системы ведут наблюдения и предоставляют некоторую статистику [3]. Кроме того, можно использовать специализированные программы.

5. Работа с программным обеспечением. Пользователь зачастую имеет устоявшиеся привычки, касающиеся выбора определенных программ, количества одновременно открытых копий программы, количества и типа загруженных из интернета файлов и область работы данных файлов. Кроме того, начало работы характеризуется определенным набором действий – сочетанием запущенных программ, открытием определенных файлов и т.п.

6. Внешние периферийные устройства. Чтобы исключить внесение изменений в систему и средства защиты при помощи внешних устройств отслеживается ряд параметров: название устройства, активность, область применения, а также – мониторинг файлов, измененных при помощи периферийных устройств.

7. Клавиатурный почерк [4]. Под данным термином, подразумевается несколько признаков присущих каждому пользователю: количество ошибок при наборе, интервалы между нажатиями клавиш, время удержания клавиш, число перекрытий между клавишами, степень ритмичности при наборе, скорость набора, привычные комбинации клавиш. Кроме того, сюда можно включить особенности работы с мышью/тачпадом. Скорость и ритм клика в каждой отдельно взятой программе.

Коэффициенты критичности.

В связи с разнородностью собираемых данных, а также, с учетом вариабельности некоторых характеристик, необходимо установить коэффициенты критичности критериев и определить уровень ошибок пользователя, при котором будет срабатывать система защиты. Кроме того, нужно предусмотреть разные способы идентификации пользователя при получении тревожных сигналов от системы защиты. В зависимости от комбинации допущенных пользователем ошибок, можно разработать разные реакции алгоритма – от повторного введения пароля или возвращения пользователя в область видимости камеры до полной блокировки системы и учетной записи пользователя с последующим обращением в службу безопасности предприятия.

Список использованных источников:

[1] Face Recognition Performance Role of Demographic Information. *IEEE Transactions On Information Forensics And Security*, Vol. 7, No. 6, December 2012

[2] Electronic Authentication Guideline. National Institute of Standards and Technology, USA [Электронный ресурс] / Nist. – Режим доступа: <https://csrc.nist.gov/publications/detail/sp/800-63/ver-102/archive/2006-04-30>. Дата доступа: 10.02.2021.

[3] Журнал действий Windows 10 и конфиденциальность [Электронный ресурс] / Microsoft. – Режим доступа: <https://support.microsoft.com/ru-ru/windows/>. Дата доступа: 19.01.2021.

[4] Нейросетевая идентификация типа личности человека по клавиатурному почерку. Т.В. Жашкова, О.М. Шарунова, Э.Ш. Исянова. *Международный студенческий научный вестник*. 2015. №3 Ч.1. С. 144-1462.

ПРИМЕНЕНИЕ ПОЛЯРИЗАЦИИ ДЛЯ ПОВЫШЕНИЯ ДОСТОВЕРНОСТИ РЕЗУЛЬТАТОВ ЛАЗЕРНОГО ТРИАНГУЛЯЦИОННОГО СКАНИРОВАНИЯ

Бунчук А.А

Белорусский государственный университет информатики и радиоэлектроники
г. Минск, Республика Беларусь

Шевяков А.В. – канд. техн. наук, доцент

Доклад посвящён исследованию возможности применения поляризации лазерного излучения для повышения достоверности результатов лазерного триангуляционного сканирования при сканировании объектов с высокой отражательной способностью, т.е. при наличии переотражений лазерного излучения от поверхностей объекта.

Лазерное триангуляционное сканирование является одним из самых распространённых способов бесконтактных оптических измерений. Данный способ основан на проецировании на объект лазерной линии. Форма проекции лазерной линии регистрируется при помощи CMOS-сенсора. Детектируя положение лазерной линии в каждом столбце полученного изображения рассчитывается положение точки в пространстве. Таким образом формируется профиль объекта в месте проецирования лазерной линии.

Вследствие множественных переотражений лазерного излучения от поверхностей объекта, формируется изображение, на котором присутствует как истинная отраженная линия лазера, так и множество переотраженных. Поэтому при сканировании объектов с высокой отражательной способностью или некоторыми особенностями геометрии (конические отверстия, внутренние углы и т.д.), возникает проблема определения действительного положения проецируемой лазерной линии на изображении светочувствительного сенсора сканера.

Традиционные методы выделения лазерной линии основаны на измерении интенсивности пикселей полученного изображения. Однако во многих случаях применение критерия детектирования, основанного на измерении только интенсивности, не позволяет надежно разделить истинную лазерную линию от переотраженных, что приводит к наличию ошибок и снижению достоверности в информации, получаемой от лазерных триангуляционных датчиков.

В работе [4] предлагается использовать вращающийся поляризационный фильтр для получения серии изображений из которых могут быть рассчитаны характеристики поляризации. Данный метод имеет вращающиеся элементы, что не гарантирует стабильность результатов во времени. В предлагаемом методе все компоненты фиксированы, механические перемещения отсутствуют.

В докладе приводятся причины появления ошибок при наличии переотражений. Анализируются изменения в поляризации излучения при отражении от металлических изделий. На основании анализа изменений в поляризации излучения была выдвинута гипотеза о том, что если проецировать на объект лазерную линию, поляризованную в различных направлениях, то на полученных изображениях, интенсивность отдельных участков, имеющих переотражения, будет отличаться.

Для проверки данного предположения разработана экспериментальная установка, схема и фото которой показаны на рисунке 1. На сканируемом объекте выделены области с переотражениями.

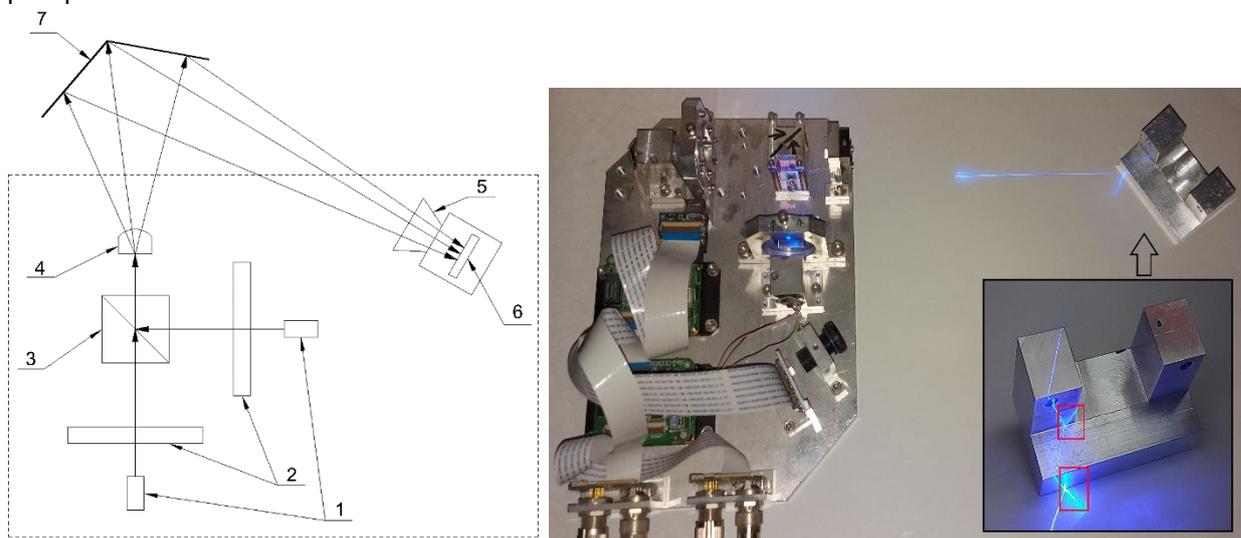


Рисунок 1 – Экспериментальная установка

Лазеры 1 излучают точечное излучение. Свет от лазеров проходит через поляризационные фильтры 2 с линейной поляризацией. Фильтры имеют возможность вращения вокруг собственной оси для ручной подстройки. Далее при помощи полупрозрачного зеркала 3 излучение лазеров совмещается, для проецирования на объект 7 в одинаковом положении. Точечное излучение разворачивается в линию при помощи линзы Пауэлла 4. Отражённый от объекта свет фокусируется при помощи объектива 5 на светочувствительный CMOS сенсор 6.

Путём поочерёдного включения двух лазеров, излучение которых принимает р и s линейную поляризацию, получены пары изображений. Для каждого изображения был построен график средней интенсивности в столбцах.

На рисунке 2 приведён пример полученных результатов. Показано изображение р поляризации, на котором разделены ключевые области (изображение s поляризации опущено). В области 1 лазерная линия отсутствует. В области 2 лазерная линия падает на объект без переотражений. В области 3 присутствует переотражение выше истинной линии. В области 4 переотражение имеет крайне низкую интенсивность и практически отсутствует. В области 5 присутствует переотражение выше и ниже лазерной линии. В области 6 присутствует переотражение выше лазерной линии.

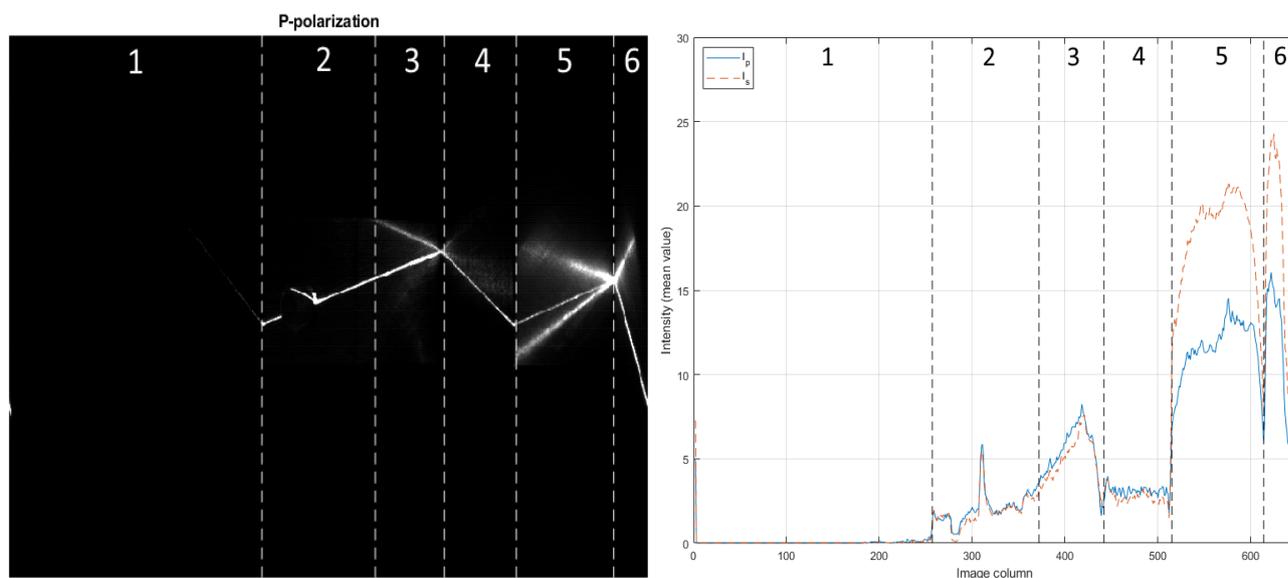


Рисунок 2 – Результаты эксперимента

Исходя из рисунка 2, на изображениях, полученных с разной поляризацией, в областях переотражений наблюдается отличие в интенсивности, что подтверждает выдвинутую гипотезу эксперимента.

Одним из примеров использования, может служить построение маски, на основании которой производится фильтрация изображения.

Маска может быть построена на основании разности интенсивности пикселей для изображения р и s поляризаций. Данный метод требует точного совмещения излучения двух лазеров. Однако сильно переэкспонированные области не могут быть отфильтрованы, так как в них значение полностью совпадает на обоих изображениях как в истинной лазерной линии, так и в переотражённой. Поэтому данный подход может быть улучшен анализом изменения интенсивности в каждом пике в столбце изображения по отдельности.

Таким образом, рассматриваемый подход является перспективным с точки зрения поиска дополнительных критериев для отделения истинной лазерной линии от переотражённых.

Результаты работы могут быть применены в лазерных триангуляционных сканерах. Однако необходимы дальнейшие экспериментальные исследования для получения количественных характеристик изменения излучения.

Список использованных источников:

1. *Principles of Optics. 60th Anniversary Edition / Born Max, Wolf Emil. / Cambridge University Press, 2019.*
2. *Refractive index database [Электронный ресурс]. – Электронные данные. – Режим доступа: <https://refractiveindex.info/> – Дата доступа: 23.03.2021.*
3. *CMOSIS CMV300 CMOS sensor datasheet. database [Электронный ресурс]. – Электронные данные. – Режим доступа: https://ams.com/documents/20143/36005/CMV300_DS000739_1-00.pdf – Дата доступа: 23.03.2021.*
4. *Improving Depth Image Acquisition Using Polarized Light / A.M. Wallace, B. Liang, E. Trucco, J. Clark. / International Journal of Computer Vision volume 32, 1999 — P. 87–109.*

ПРОГРАММНО-АППАРАТНЫЙ КОМПЛЕКС «БЕСПИЛОТНЫЙ ЛЕТАТЕЛЬНЫЙ АППАРАТ С ДИСТАНЦИОННЫМ УПРАВЛЕНИЕМ»

Гаврилова А.А.

Белорусский государственный университет информатики и радиоэлектроники
г. Минск, Республика Беларусь

Луцик Ю.А. – канд. техн. наук, доцент

Объектом исследования данной работы являются беспилотные летательные аппараты. Предметом исследования является технология моделирования квадрокоптеров и их практическое применение. Цель исследования: разработать универсальную модель квадрокоптера с возможностью его модернизации в зависимости от предполагаемой области использования. Для достижения поставленной задачи были изучены различные схемы построения квадрокоптеров, оценены возможные направления практического применения дронов, была изготовлена и апробирована полученная модель.

История создания дистанционно управляемых летательных машин началась в конце XIX века. В дальнейшем беспилотные летательные аппараты (БПЛА) получили бурное развитие. По степени автономности БПЛА делятся на дистанционно управляемые, иначе называемые ДПЛА – дистанционно-пилотируемый летательный аппарат, полностью автоматические и гибридные. Дифференциация по максимальной взлётной массе может требовать обязательной регистрации и постановки на соответствующий учет. Они также различаются по назначению, конструкции и другим параметрам. К основным сферам применения дронов относятся: оборонная промышленность, сельское хозяйство, видеосъемка, воздушные доставки, спасательные операции [1].

Важным преимуществом беспилотных летательных аппаратов, также называемых мультикоптерами, является существенно меньшая стоимость их создания. Основным приоритетом есть и будет безопасность оператора аппарата. В настоящее время область применения БПЛА постоянно расширяется, также растет число различных моделей квадрокоптеров. При всем их многообразии квадрокоптеры условно можно разделить на мини, маленькие, средние и тяжелые [1]. К так называемым мини-коптерам относятся малогабаритные устройства без дополнительных функций, с малой маневренностью и скоростью, неспособные противостоять порывам ветра и используемые, как правило, для ознакомления с базовым функционалом летательных аппаратов. Маленькие и средние дроны используются и для съемки, и для гонок. Небольшие устройства могут использоваться в поисково-спасательных операциях. В отдельную категорию выносятся тяжелые, значительно уступающие в маневренности, мультикоптеры с массивным корпусом и большой грузоподъемностью. Прочный панцирь позволяет их использовать в неблагоприятных погодных условиях. Также они пригодны для транспортировки относительно небольших грузов.

Несмотря на бесспорные преимущества БПЛА, их существенным недостатком является уязвимость систем дистанционного управления. Серьезную проблему могут представлять незащищенные радиоканалы, используемые при дистанционном управлении дроном. Сигналы, принимаемые и отсылаемые аппаратом, могут быть перехвачены, заглушены или изменены [2]. Большую надежность могут гарантировать лишь устройства, относящиеся к высоко-ценовому сегменту. Также проблемой может стать подбор квадрокоптера под какую-либо специфическую задачу и/или условия: наиболее распространенные и доступные модели пригодны только для одной задачи (videосъемка, перевозка грузов или гонки), а «заложенный» при производстве функционал изменить невозможно. При этом зачастую даже эта единственная опция может потребовать приобретение дополнительных гаджетов.

Целью данной работы стала разработка квадрокоптера с открытой аппаратной и программной частями. Такая концепция превращает дрон в конструктор. В зависимости от своих нужд пользователь может изменять управление, полетные функции, добавлять датчики или же изменять программную конфигурацию.

Собираемый квадрокоптер (см. рисунок 1) относится к категории малых/средних дронов. В базовом варианте он снабжен непосредственно управляющим микроконтроллером, моторами, контроллерами полета, гироскопом, акселерометром, барометрическим датчиком, GPS-модулем и камерой. Также был разработан пульт дистанционного управления, оснащенный потенциометром, выступающим в роли регулятора оборотов моторов, джойстиком для управления дроном в пространстве и ЖКИ-экраном, на котором отображаются данные, получаемые от GPS-модуля и барометрического датчика.

В основу квадрокоптера легла легкая 10-дюймовая карбоновая рама. Дрон оснащен 4 бесколлекторными моторами, способными совершать 920 оборотов в минуту (920 kv). Управление полетом и всеми датчиками мультикоптера осуществляется при помощи микроконтроллера семейства Arduino. Данные платы имеют ряд преимуществ и, ввиду своей популярности, обладают обширной экосистемой. Помимо готовых библиотек существует огромное множество созданных пользователями материалов, что позволяет найти необходимый инструментарий практически под

любую задачу [3]. Многие модули, как датчики освещенности, ультразвуковые и инфракрасные модули, совместимы со всей линейкой плат Arduino, при этом и подключаемые модули, и сами платы отличаются достаточно низкой стоимостью. Также для платы Arduino Uno подходит большинство стандартных плат расширения. К достоинствам данных плат можно отнести низкий порог вхождения – удобная кроссплатформенная интегрированная среда разработки, используемые в библиотеках высокоуровневые языки программирования существенно облегчают работу, что делает эти микроконтроллеры удобными и для новичков, и для опытных разработчиков. Это является наиболее существенным достоинством, так как концепция открытой аппаратной части предполагает, что пользователь без специализированных навыков должен справиться с настройкой и усовершенствованием устройства.



Рисунок 1 – Квадрокоптер в процессе разработки и пульт дистанционного управления

На данном этапе используется Arduino Nano. Дальнейшая модернизация может потребовать больше ресурсов, поэтому предполагается переход на Arduino Mega с большим объемом памяти или одноплатный компьютер Raspberry Pi. С таким переносом способен справиться даже штатный пользователь, поскольку платы семейства Arduino совместимы между собой [4], а стойка для микроконтроллера рассчитана на различные форм-факторы. Также важную роль играет модуль MSU-6050 – это трехосевой гироскоп и акселерометр с фильтром Калмана. Именно благодаря этому датчику осуществляется фиксация перемещения и регулировка положения квадрокоптера в воздухе. За установку связи между дроном и пультом дистанционного управления отвечает радиомодуль nrf2401.

В дальнейшем предполагается ввести ряд усовершенствований. К механическим относится установка защиты на винты и аппаратную часть устройства. Также планируется добавление антенны на пульт управления, что позволит расширить радиус контроля при полете. Помимо этого, возможна установка датчиков дыма/газа для применения дрона, например, при спасательных операциях в сложных условиях. Для трансляции видео в реальном времени имеющийся 1,6 – дюймовый экран на пульте управления будет заменен на другой, с большей диагональю. Разработка клешни для захвата и транспортировки грузов, шлем для управления квадрокоптером от первого лица (FPV), экзоперчатка как альтернатива пульту дистанционного управления – все это относится к ряду возможных модернизаций для расширения функционала данного устройства. В качестве обязательного улучшения планируется работа над элементом питания, так как большинство дронов обладают сравнительно малым временем работы без подзарядки. Имеющийся квадрокоптер может находиться в воздухе до 15 минут при отключенной камере.

В качестве программного улучшения может быть разработано мобильное приложение для управления дроном при помощи смартфона, планшета.

Таким образом будет получено устройство, пригодное и для достаточно продолжительной видеосъемки, и для транспортировки грузов, и для каких-либо метеорологических исследований, и даже для поисково-спасательных операций. Еще одним достоинством является малая итоговая стоимость. Цена аналогов с много меньшим функционалом и отсутствием возможности модернизации в разы превышает суммарную стоимость рамы, микроконтроллера Arduino и набора необходимых датчиков.

Список использованных источников:

1. Антти Соумалайнен. Беспилотники: автомобили, дроны, мультикоптеры / Антти Соумалайнен – Изд-во ДМК Пресс, 2018 – 122 с.
2. 924 Государственный центр беспилотной авиации Министерства обороны Российской Федерации. Перспективы развития и применения комплексов с беспилотными летательными аппаратами: сборник статей и докладов / Коломна, 2016 – 274 с.
3. Петров И. В. Программируемые микроконтроллеры. Стандартные языки и приемы прикладного проектирования / Петров И. В. – Изд-во СОЛОН-Пресс, 2010 – 255 с.
4. Улли Соммер. Программирование микроконтроллерных плат Arduino, Freeduino / Улли Соммер – Изд-во БХВ-Петербург, 210 – 244 с.

МЕХАНИЗМ ПОИСКА ДОКУМЕНТОВ В ИНФОРМАЦИОННО-ПОИСКОВОЙ СИСТЕМЕ

Глацкевич О.В.

Белорусский государственный университет информатики и радиоэлектроники
г. Минск, Республика Беларусь

Одинец Д.Н. – канд. техн. наук, доцент

По мере накопления документов в информационно-поисковых системах неминуемо возникает необходимость в организации эффективного механизма поиска информации в системе. Современная система должна отвечать таким критерием поиска, как быстрая скорость поиска и высокая степень соответствия найденных документов поисковому запросу. Для достижения этих целей эффективно использовать полнотекстовый поиск информации, выполняющий поиск документов по их содержанию.

Разрабатываемый поисковый механизм осуществляет полнотекстовый поиск документов в базе данных на основе информационно-поискового языка и соответствующих правил поиска. На основе текстовых данных выполняется построение инвертированного индекса, который представляет из себя структуру данных, состоящую из морфологического словаря, где каждому слову словаря соотносится список номеров документов, в котором это слово встречается.

Индексатор получает текст на вход, выполняет морфологические преобразования текста (приведение к лексемам или корневой основе слова, символные фильтры, исключение стоп-слов) и сохраняет в поисковый индекс, сопоставляя каждому слову набор документов.

Также в индексе сохраняется метainформация (позиция слова в документе, частота встречаемости слова в документе и во всей коллекции документов), используемая в алгоритме механизма поиска.

Система содержит лингвистическую базу данных, позволяющую выполнять морфологические преобразования входной строки посредством использования информации из лингвистической базы данных. Лингвистическая база данных содержит следующую информацию:

- словарь словоформ русского языка;
- словарь неинформативных слов;
- словарь ключевых слов;
- словарь синонимов;
- словарь сокращений.

После прохождения поисковой строки морфологического анализатора формируется результат пересечения списков инвертированного индекса в виде набора документов, содержащих каждое слово запроса. Сортировка документов выполняется на основе формулы ранжирования, выставляя каждому документу весовой коэффициент.

Основу формулы ранжирования в механизме поиска составляют функции BM25 (встречаемость слов запроса в каждом документе и во всей коллекции) и LCS (порядок слов), для которого в индексе для каждого слова записывается его позиция в документе. Формула ранжирования содержит ряд факторов ранжирования, которые являются характеристиками алгоритма поисковой системы, выполняющие анализ документов информационного массива на соответствие заданным требованиям с целью определения степени релевантности документов и представляют собой произвольные арифметические выражения, которые могут использовать константы, атрибуты документов, встроенные функции и логические операторы.

По результатам информационного поиска списки документов формируются по релевантности, где каждому документу присваивается весовой коэффициент, который представляет из себя числовое значение, полученное путем суммирования всех факторов ранжирования, встретившихся в документе. Документ с наибольшим весовым коэффициентом отобразится на первой позиции в списке результатов поиска.

В общем виде применяется следующая стратегия информационного поиска:

- формулировка запроса;
- морфологические преобразования информативных слов запроса;
- поиск релевантных документов;
- определение степени релевантности документов;
- ранжирование документов;
- предоставление результата поиска.

Список использованных источников:

1. Информационный портал *habr*. Как работают поисковые системы [Электронный ресурс]. Режим доступа: <https://habr.com/ru/company/yandex/blog/464375/>. – Дата доступа: 28.03.2021

СОЗДАНИЕ ПЕЧАТНОГО СЕРВИСА С ВОЗМОЖНОСТЬЮ ОПЛАТЫ ЭЛЕКТРОННЫМИ ДЕНЬГАМИ

Писчик А.С., Гук В. А.

Белорусский государственный университет информатики и радиоэлектроники
г. Минск, Республика Беларусь

Перцев Д.Ю. – канд. техн. наук

Представлен готовый прототип печатного сервиса с возможностью оплаты, описана схема работы сервиса.

Сфера автоматизированных услуг заключается в том, чтобы принять оплату от клиента и запустить устройство для выполнения выбранной услуги. Одной из таких задач является предоставление услуг печати. Вендинговые аппараты имеют ряд недостатков, среди которых высокая стоимость, низкая ремонтпригодность и отказоустойчивость, возможность работы только с купюро- и монетоприемниками, отсутствие расширяемости.

Целью данной работы ставится разработка программно-аппаратной платформы для вендинговых аппаратов, обладающей высокой расширяемостью и ремонтпригодностью, а также низкой ценой (предполагается, что стоимость аппарата сводится практически к стоимости устройства печати и его обслуживания). Схема работы печатного сервиса показана на рисунке 1.

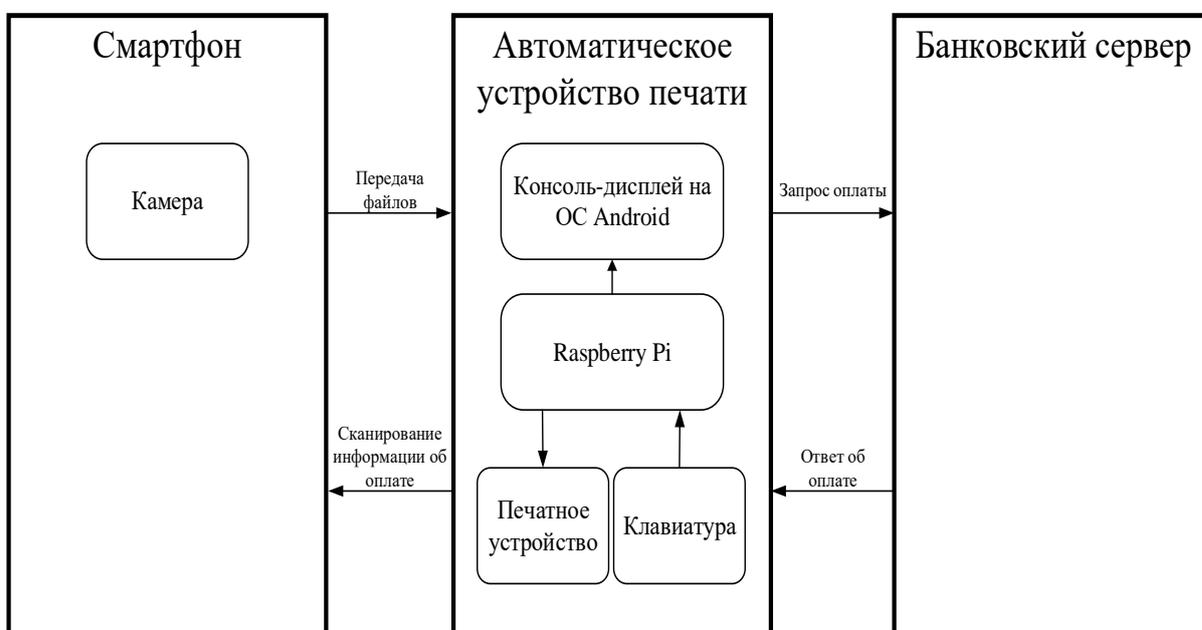


Рисунок 1 – Схема работы печатного сервиса

Отличительными особенностями сервиса являются:

- платформа Raspberry Pi [1] для исполнения программного обеспечения;
- интеграция разных способов передачи файлов на печать (Wi-Fi, Bluetooth, различные мессенджеры);
- интеграция с платежной системой «Оплати» [2], созданной компанией ООО «ЛВО» при поддержке ОАО «Белинвестбанк». Данная платежная система производит платежи через сканирование QR-кода и не требует установки платежных терминалов, купюро- и монетоприемников. Благодаря этому стоимость штучной интеграции системы оплаты в устройство стремится к нулю.

Преимуществами представленного подхода являются относительно небольшая стоимость по сравнению с профессиональным оборудованием, отсутствие необходимости устанавливать специальные хранилища для денег, полностью контролируемый разработчиками процесс работы (минимальная зависимость от сторонних поставщиков и разработчиков).

Список использованных источников:

1. Raspberry Pi 4 [Electronic Resource] / Raspberry. – Mode of access: <https://www.raspberrypi.org/products/raspberry-pi-4-model-b/>. Date of access: 27.03.2021.
2. Оплати [Электронный ресурс] / ООО «ЛВО». – Режим доступа: <https://www.o-plati.by/>. Дата доступа: 27.03.2021.

СЕГМЕНТАЦИЯ СПУТНИКОВЫХ СНИМКОВ С ИСПОЛЬЗОВАНИЕМ НЕЙРОННОЙ СЕТИ U-NET

Дашко Б.В.

Белорусский государственный университет информатики и радиоэлектроники
г. Минск, Республика Беларусь

Куприянова Д.В. – ассистент

В данной работе рассмотрена одна из самых популярных нейронных сетей, применяемая для сегментации спутниковых снимков, описана ее архитектура и обоснована перспективность применения.

Сегментация изображений является одной из основных задач компьютерного зрения, позволяющая автоматизировать множество задач, в т.ч. формирование карт на основе спутниковых снимков.

Одна из наиболее популярных нейронных сетей, используемая в сегментации спутниковых снимков, является сеть U-Net [1], классический вариант которой показан на рисунке 1. Основными причинами, по которым используется данная сеть, являются: способность обучаться в средах с небольшим объемом обучающих данных, возможность относительно быстрой адаптации к последним архитектурам при увеличении обучающих данных.

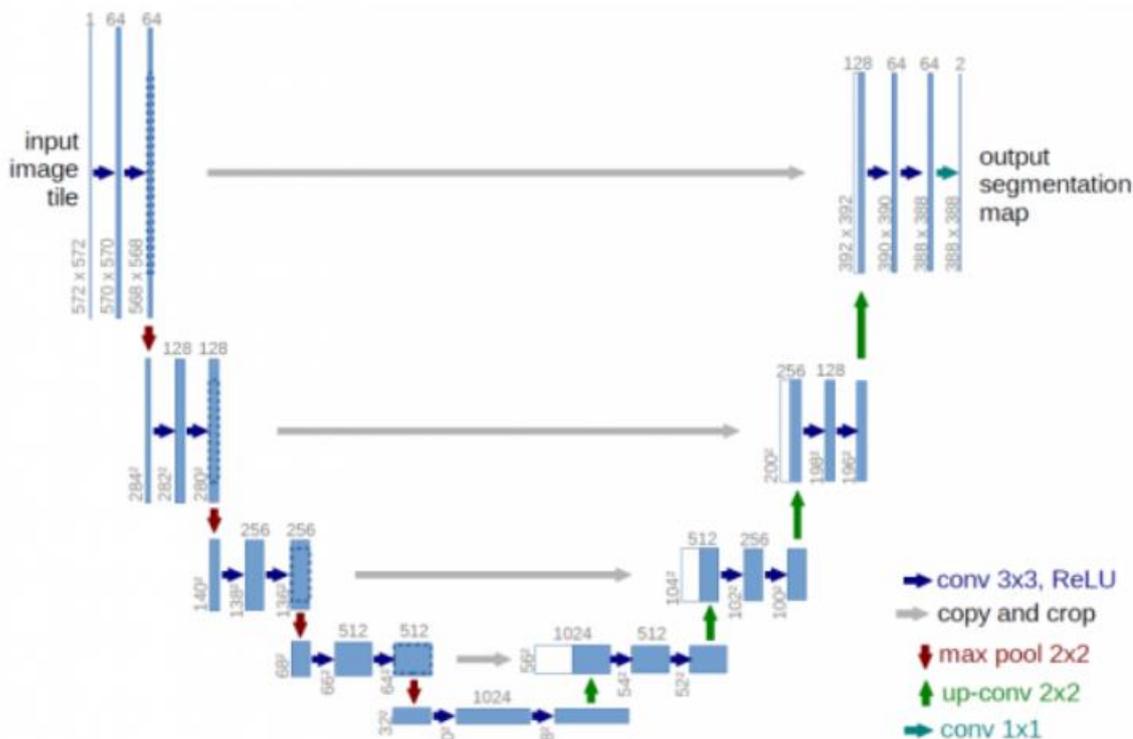


Рисунок 1 – Классическая архитектура U-Net

Архитектура сети U-Net представляет собой последовательность слоёв свёртки и пулинга, которые сначала уменьшают пространственное разрешение картинки, а потом увеличивают его, предварительно объединив с данными картинки и пропустив через дополнительные слои свёртки. Данные слои состоят из последовательного применения двух сверток 3x3 и ReLU активации. На каждом этапе понижающей дискретизации каналы свойств удваиваются. Для улучшения работы сети используется дополнительный слой Batch Normalization и аугментация (поворот на произвольный угол и зеркальное отображение изображения).

Перспективность применения нейронной сети на основе данной архитектуры подтверждается тем, что выявлено множество ее модификаций (например, [2,3]). В дальнейшем планируется реализовать U-Net сеть и сделать комплексный анализ эффективности представленного подхода.

Список использованных источников:

1. U-Net: Convolutional Networks for Biomedical Image Segmentation [Electronic Resource] / ArXiv. – Mode of access: <https://arxiv.org/pdf/1505.04597.pdf>. Date of access: 28.03.2021.
2. Soni A. M-U-Net: Modified U-Net Segmentation Framework with Satellite Imagery / A. Soni, [et.c.] // Proceedings of the Global AI Congress 2019. – Springer, 2020. – P. 47-59.
3. Segmentation of Satellite Imagery using U-Net Models for Land Cover Classification [Electronic Resource] / ArXiv. – Mode of access: <https://arxiv.org/pdf/2003.02899.pdf>. Date of access: 28.03.2021.

АНАЛИЗ ЭФФЕКТИВНОСТИ НЕЙРОННОЙ СЕТИ S2S ДЛЯ УДАЛЕНИЯ ШУМОВ НА ИЗОБРАЖЕНИИ

Климбасов А.А.¹, Найдович О.А.²

Белорусский государственный университет информатики и радиоэлектроники¹

г. Минск, Республика Беларусь

Белорусский государственный университет²

г. Минск, Республика Беларусь

Перцев Д.Ю. – канд. техн. наук

В данной работе проведен анализ эффективности работы нейронной сети S2S, предложенной на конференции CVPR 2020 [1, 2]. Выполнена оценка качества восстановления зашумленного изображения. Результаты исследования показывают, что предложенная архитектура нейронной сети обеспечивает высокую эффективность на рассмотренных наборах данных.

Массовое распространение цифровых телевизоров с качественной матрицей с высоким разрешением, оцифровка и приведение к высокому разрешению старых видеоматериалов приводит к необходимости обработки всего медиаконтента и, в частности, удалению шумов. Поэтому проблема качественного восстановления является важной задачей обработки информации. Типичными видами шума на изображении являются Гауссовский шум, соль и перец (англ. Salt and Pepper), шум Пуассона и другие.

Удаление шумов - это сложный процесс, который плохо поддается ручной алгоритмизации, однако с ростом вычислительных мощностей стало доступно использование более сложных моделей. Ярким примером такой модели являются глубокие нейронные сети с большим числом параметров.

Главное преимущество архитектуры S2S - для ее обучения не требуется большой объем данных. Более того, эти данные могут быть неразмеченными. Такой метод обучения называется самообучением (англ. Self-Supervised Learning). То есть нейронная сеть S2S обучается на неорганизованном наборе изображений, принимая на вход лишь одно изображение за итерацию.

Архитектура нейронной сети S2S (рисунок 1). Архитектура данной нейронной сети построена на основе шаблона Encoder-Decoder.

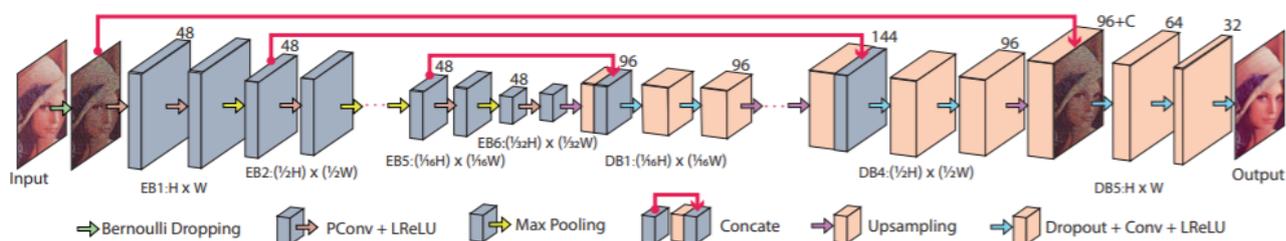


Рисунок 1 – Архитектура нейронной сети S2S

В качестве входных данных в нейронную сеть поступает изображение размером $H \times W \times C$.

Блок 1: кодер (англ., Encoder). В данном блоке к изображению применяется 6 слоев нейронной сети, каждый из которых состоит из свертки (в каждой свертке используется по 48 фильтров), функции активации (LReLU) и операции субдискретизации (англ. max pooling), которая каждый раз уменьшает размер изображения в 2 раза и состоит из рецептивного поля 2×2 . После прохода данного блока формируется карта признаков размером $H/32 \times W/32 \times 48$.

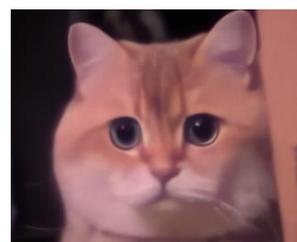
Блок 2: декодер (англ. Decoder). Состоит из функции увеличения разрешения (англ. upsampling), которая увеличивает размер изображения через билинейную интерполяцию, 2-х сверток и 2-х функций активации (LReLU). Также каждый раз после применения функции увеличения разрешения формируется карта признаков декодера с соответствующей ему картой признаков из кодера. Особенность данной сети заключается в том, что каждая свертка используется с применением метода исключения (англ. dropout).

Последний слой состоит из 3-х сверток (с 64, 32 и C фильтрами соответственно) и 3-х функций активации (LReLU). Делается это для того, чтобы привести карту признаков к исходному размеру изображения.

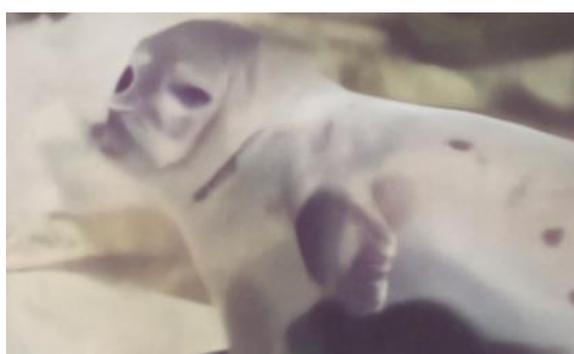
Тестирование нейронной сети S2S. Тестирование нейронной сети выполнялось на 3 тестовых наборах – Set9 [3] (гауссовский шум, 9 цветных изображений), BSD68 [4] (гауссовский шум, 68 изображений в градациях серого) и PoliU [5] (для анализа изображений из реального мира). В качестве метрики качества работы сети были использованы PSNR и SSIM. Это метрики сравнивают результат работы нейронной сети с размеченными данными из датасетов. Усредненные результаты измерения выбранных метрик на тестовых наборах сведены в таблицу 1. Примеры тестовых зашумленных изображений и результат их обработки показаны на рисунке 2.



Шум Пуассона



Шум соль-перец



Гауссовский шум



Мультипликативный шум

Рисунок 2 – Пример работы нейронной сети с разными видами шумов

Таблица 1 – Результаты измерения качества работы нейронной сети на основе выбранных метрик

Метрика	Тестовые наборы данных		
	Set9	PoliU	BSD68
PSNR	27.94	37.52	28.58
SSIM	0.92	0.98	0.80

На основе полученных результатов можно сделать вывод, что нейронная сеть плохо обрабатывает шум Пуассона, однако с другими видами шумов справляется достаточно качественно. Данный результат подтверждает высокий уровень значения SSIM, стремящийся к единице для изображений из реального мира.

Список использованных источников:

1. Self2Self With Dropout: Learning Self-Supervised Denoising From Single Image [Electronic Resource] / Computer Vision Foundation. – Mode of access: https://openaccess.thecvf.com/content_CVPR_2020/papers/Quan_Self2Self_With_Dropout_Learning_Self-Supervised_Denoising_From_Single_Image_CVPR_2020_paper.pdf. – Date of access: 21.03.2021.
2. Self2Self With Dropout: Learning Self-Supervised Denoising From Single Image [Electronic Resource] / GitHub. – Mode of access: <https://github.com/scut-mingqinchen/self2self>. – Date of access: 21.03.2021.
3. Deposited Set 9: GSK TCAMS TB Dataset [Electronic Resource] / ChEMBL-NTD. – Mode of access: <https://chembl.gitbook.io/chembl-ntd/downloads/deposited-set-9-gsk-tcams-tb-dataset-11th-january-2013>. – Date of access: 21.03.2021.
4. BSD (Berkeley Segmentation Dataset) [Electronic Resource] / Papers With Code. – Mode of access: <https://paperswithcode.com/dataset/bsd>. – Date of access: 21.03.2021.
5. Real-world Noisy Image Denoising: A New Benchmark [Electronic Resource] / ArXiv. – Mode of access: <https://arxiv.org/pdf/1804.02603.pdf>. – Date of access: 21.03.2021.

СОВРЕМЕННЫЕ МЕТОДЫ АНАЛИЗА СЕТЕВОГО ТРАФИКА

Климец А.А.

Белорусский государственный университет информатики и радиоэлектроники
г. Минск, Республика Беларусь

Перцев Д.Ю. – к.т.н., доцент

Представлен обзор современных подходов к глубокому анализу сетевых пакетов, лежащих в основе современных DPI-систем, используемых интернет-провайдерами и спецслужбами разных стран, а также причины их использования.

DPI – это технология проверки, фильтрации и накопления статистических данных о сетевом трафике по его содержимому или каким-то специфичным характеристикам протоколов. На данный момент на рынке представлено множество различных технологий DPI, они могут быть как standalone решениями, так и встроенными в маршрутизаторы. Среди наиболее крупных компаний, разрабатывающих DPI-системы, выделяются: Allot Communications, Cisco, Sandvine, Huawei. В данный их системы широко используют по всему миру.

В общем случае DPI способен фильтровать трафик на транспортном и сетевом уровнях модели OSI, однако наибольшая точность идентификации или классификации трафика происходит на сеансовом, представительском и прикладном уровнях, анализируя содержимое отдельных пакетов. Базовой проверки пакетов на данный момент недостаточно, так как количество генерируемого пользователями трафика, а также используемых протоколов неизменно растёт огромными темпами. Всё большее развитие потоковых приложений, обфусцированных протоколов, к примеру, QUIC версии 50, а также широкое использование VPN, создаёт значительные препятствия для работы DPI систем, не говоря уж об общей тенденции на улучшение безопасности пользовательских данных в сети Интернет. Всё это стало причиной появления новых техник распознавания трафика.

Если раньше во многом DPI системы сводились к обычной проверке содержимого 1-2 пакетов, чаще всего HTTP или HTTPS трафика, проверкой поля Host или SNI соответственно, на данный момент DPI системы широко используют альтернативные способы идентификации трафика по каким-либо косвенным признакам, присущим определённым сетевым программам и протоколам. Широко используется статистический и поведенческий анализ, в частности анализ встречаемости определённых символов, длины пакетов и др.

Для подобного анализа больших объёмов сетевого трафика нередко используются методы Data Mining, позволяющие обнаружить неявные зависимости между конкретными сетевыми соединениями и работающими в этот момент приложениями или какими-либо действиями пользователя при использовании веб-приложений. Применение подобных методов делает идентификацию трафика чрезвычайно точным, ведь современные приложения широко используют различные микросервисы и широко интегрируются с различными сторонними сервисами, что создаёт большое количество сетевых сессий, подписать которые бывает достаточно трудно.

Подходы к анализу трафика посредством DPI ограничиваются только тем, сколько различных параметров может проанализировать та или иная система, а также сложностью анализируемого протокола. Однако основная проблема DPI в том, что при определении принадлежности трафика к тому или иному приложению необходимо, чтобы устройство проверило оба направления сетевых сессий, т.е. необходимо проверять асимметричный трафик, который широко встречается у крупных операторов.

Существует несколько известных подходов к работе с таким трафиком: Cisco проверяет только одну часть сессии, Sandvine инкапсулирует асимметричный трафик в broadcast-фреймы и в таком виде прогоняет их через свои DPI устройства [1].

Существует много причин для использования DPI. Одна из причин – это экономия средств, так как внедрение DPI гораздо выгоднее, чем расширять аплинки, и позволяет осуществлять контроль утилизации каналов. Кроме того, DPI активно используется для реализации QoS, к примеру, давая приоритет VoIP-сервисам или стриминговым платформам, что актуально во время пандемии COVID-19, а также система может значительно снижать пропускную способность некоторых сервисов, тем самым осуществляя контроль загрузки сетей и для борьбы с нежелательным трафиком, выполняя полную блокировку тех или иных сайтов или приложений. И, наконец, DPI системы используются в маркетинговых целях, для сбора статистики по наиболее популярным у пользователей приложениям или тому, какие устройства используют пользователи. На этом фоне на данный момент стало чрезвычайно популярным у мобильных операторов делать тарифы, предоставляющие безлимитный доступ к мессенджерам или стриминговым сервисам.

Список использованных источников:

1. Краткий обзор технологии DPI – Deep packet Inspection [Электронный Ресурс]. – Режим доступа: <https://habr.com/ru/post/111054/>. – Дата доступа: 21.03.2021

КОМПЛЕКС КОНТРОЛЯ РАБОТЫ СЕРДЕЧНО-СОСУДИСТОЙ СИСТЕМЫ ЧЕЛОВЕКА

Кожемякин П.Н.

Белорусский государственный университет информатики и радиоэлектроники
г. Минск, Республика Беларусь

Луцик Ю.А. – канд. технических наук, доцент

Проект будет выполнен в два этапа: аппаратный и программный. Аппаратный этап представлен микроконтроллером для получения и предварительной обработки информации. Программный этап включает в себя ПО, написанное на мобильном приложении, для анализа поступившей информации о деятельности сердечно-сосудистой системы.

В наше время не проблема найти целый ряд медучреждений, где можно сдать абсолютно любые анализы и их исследуют профессионалы либо специальные приборы. Но ведь не всегда есть время для того, чтобы посетить больницу для проверки организма. А если и есть какие-то позывы со стороны здоровья, то часто мы ими пренебрегаем. Но если был бы способ пройти процесс сдачи анализов, их обработки и определения диагноза всего за пару минут, мы бы делали это куда чаще.

Это известный факт и с ним связано то, что развивается такая ветвь медоборудования, как персональное оборудование. К примеру термометры – уже многие десятки лет есть в каждой семье, тонометры – последние 10 лет активно появляются и используются всеми слоями населения, глюкометры – также широко используются у людей в личных целях.

А что на счет аппаратов-электрокардиографов (аппарат-ЭКГ)? Аппараты-ЭКГ редко бывают персональным, хотя, по статистике всемирной организации здравоохранения (ВОЗ), заболевания сердечно-сосудистой системы (ССС) являются основной причиной смерти во всем мире. По оценкам на 2019 год, 16% всех смертей в мире приходился на заболевания ССС [1].

Персональное оборудование в свою очередь всё больше и больше интегрируется в смартфоны в виде приложений. В связи с этим фактом, появилось огромное множество приложений помогающие человеку следить за его здоровьем [2]. Например, различные калькуляторы калорий, шагомеры, пульсометры, считывающие пульс с помощью встроенной камеры, и т.д. [3]. Данные приложения довольно информативны, но недостаточно точны и не всегда могут предоставить нужные данные. Поэтому начали появляться различные дополнительные гаджеты синхронизирующийся со смартфоном. Например, умные часы, которые всегда считают ваши шаги, даже когда телефона нет у вас в кармане, и мобильные глюкометры передающие измеренные данные на экран смартфона через несколько секунд после начала измерения. Каждое такое устройство требует свое уникальное ПО для смартфона.

Цель данного проекта состоит в том, чтобы разработать мобильное приложение как раз для одного из таких гаджетов. Им является комплекс мониторинга здоровья человека. Данный аппарат является персональным, мобильным и уникальным, с той точки зрения, что еще нет ни одного подобного устройства в отечественном сегменте. Данная работа представляет возможность анализировать здоровье человека в режиме реального времени, путем отслеживания таких данных как частота сердечных сокращений, насыщение крови кислородом, кардиомониторинг.

Также комплекс должен обладать следующими параметрами:

- приложение должно быть достаточно просто в обращении;
- данные должны обрабатываться и отображаться на экране мобильного устройства;
- должен производиться первичный анализ полученных данных.

С помощью данного устройства, результаты которого, можно будет показать вашему лечащему врачу. Потенциальное использование данной системой полезно для регулярного обследования, анализе и отчетности, чтобы иметь представление о своём состоянии здоровья на основе постоянного контроля. По данным Американской ассоциации лечение сердца в течение первых 12 минуты могут принести положительную результат примерно от 45% до 60% [4]. Здесь следует отметить, что предложенная конструкция направлена на то, чтобы все, включая пациентов, вели регулярный учет своего состояния здоровья. В нашу задачу также входит создание гибкой среды между врачом и пациентом, проверяя их постоянное самочувствие.

Список использованных источников:

1. ВОЗ публикует статистику о ведущих причинах смертности и инвалидности во всем мире за период 2000–2019 гг. [Электронный ресурс]. – Электронные данные. – Режим доступа: <https://www.who.int/ru/news/item/09-12-2020-who-reveals-leading-causes-of-death-and-disability-worldwide-2000-2019> – Дата доступа: 25.03.2021.
2. Портативные ЭКГ аппараты [Электронный ресурс]. – Электронные данные. – Режим доступа: https://www.livemd.ru/tags/portativnyye_ekg_apparaty/ – Дата доступа: 28.02.2021.
3. Заболевания сердечно-сосудистой системы [Электронный ресурс]. – Электронные данные. – Режим доступа: <https://www.kp.ru/putevoditel/zdorove/zabolevaniya-serdechno-sosudistoj-sistemy/> – Дата доступа: 27.02.2021.
4. Положения для контроля здоровья / НВ [Электронный ресурс]. – Электронные данные. – Режим доступа: <https://nv.ua/health/lifehacks/polozheniya-dlya-kontrolya-zdorovya-50034040.html> – Дата доступа: 25.03.2021.

ЗАЩИТА ГРАФИЧЕСКИХ ОБРАЗОВ С ПОМОЩЬЮ СИСТЕМ ВОДЯНЫХ ЗНАКОВ

Кушнеревич П.М.

Белорусский государственный университет информатики и радиоэлектроники
г. Минск, Республика Беларусь

Ярмолик В. Н. – доктор техн. наук, профессор

Данная работа содержит сведения о выборе метода для размещения водяного знака.

Количество графических образов в сети интернет, которые являются плагиатом оригинальных изображений говорят о том, что нужно средство, которое будет добавлять некую метку (водяной знак) на графический образ, по которой можно будет определить автора.

Так как плагиат графических образов особенно актуален для систем электронной коммерции, добавление на графический образ водяного знака, видимого для пользователя без специального программного средства, может повлиять на конверсию. Таким образом имеет смысл добавлять водяной знак на графический образ, который можно будет обнаружить только с помощью программного средства.

Данную проблему можно решить с помощью стеганографических методов внедрения информации об авторе.

Всё разнообразие существующих стеганографических можно разделить на два вида: методы, обеспечивающие внедрение произвольного количества бит скрываемого сообщения, и методы, обеспечивающие размещение в изображении одного бита информации – признака авторства (authorship sign).

Метод "Patchwork", предложенный в 1996 году Бендером, Грулом, Моримото, устойчив к попыткам обнаружения и уничтожения.

Суть подхода, использованного в методе "Patchwork", заключается в модификации статистической характеристики не всего графического образа, а некоторой его части. Преобразуемая часть объекта выбирается как подмножество элементов объекта, определяемое некоторым параметром.

При этом характеристика всего объекта остаётся неизменной. Модифицированный вид характеристики выбранного подмножества элементов рассматривается как наличие размещенного водяного знака.

Обнаружение водяного знака возможно только при наличии специального программного средства и известном значении использованного при выборе преобразуемой части объекта параметра, который является секретным ключом.

Для размещения водяного знака в данном методе используется скрытая модификация статистической характеристики изображения, что обеспечивает устойчивость внедряемой информации.

Суть метода заключается в изменении яркости множества случайно выбранных пар пикселей (a, b) таким образом, чтобы значение S, определённое для данного множества пикселей, стало существенно отличным от нуля, где $S = a - b$.

Скрытость признака авторства достигается тем, что информацией о координатах изменённых пикселей владеет только внедривший метод.

Метод "Patchwork" является подходящим алгоритмом для написания программного средства, которое будет определять автора графического образа без изменения внешнего вида изображения и в будущем поможет уменьшить количество плагиата в сети интернет.

Список использованных источников:

1. Barni M., Bartolini F. *Watermarking Systems Engineering*.— N.Y.: Marcel Dekker Inc, 2004.— 466 p.
2. Aucsmith D. *Tamper resistant software: an implementation // In Information Hiding*.— Vol. 1174 of *Lectures Notes in Computer Science*.— Springer-Verlag, 1996.— P. 317—334.
3. Anderson R., Petitcolas F. *On the limits of Steganography // IEEE Journal of Selected Areas in Communications*.— 1998.— № 16(4).— P. 474—481
4. *Techniques for data hiding / W. Bender, D. Gruhl, N. Morimoto et al. // IBM Syst. J.*— 1996.— № 4.— P. 313—336.

ИСПОЛЬЗОВАНИЕ МЕТОДА ПЕРЕДАЧИ ОБУЧЕНИЯ ПРИ РАЗРАБОТКЕ КЛАССИФИКАТОРА ДЛЯ ОПРЕДЕЛЕНИЯ ПАТОЛОГИЙ НА ИЗОБРАЖЕНИЯХ ПОЛУЧЕННЫХ МЕТОДОМ ОПТИЧЕСКОЙ КОГЕРЕНТНОЙ ТОМОГРАФИИ

Лисовский А.Г.

Белорусский государственный университет информатики и радиоэлектроники
г. Минск, Республика Беларусь

Одинец Д.Н. – канд. техн. наук, доцент

В данной работе рассматривается применение метода передачи обучения при разработке классификатора для определения патологий на изображениях, полученных при исследовании человеческого глаза методом оптической когерентной томографии.

Методы глубинного обучения находят широкое применение в сфере разработки экспертных систем медицинского назначения. На этапе диспансеризации современные системы с высокой вычислительной способностью позволяют обрабатывать большие объемы данных, что способствует более быстрому диагностированию заболеваний на ранних стадиях у большего количества людей. В данной работе исследуются методы построения классификатора для выявления патологий сетчатки глаза на изображениях полученных методом оптической когерентной томографии.

Построение классификатора при использовании метода обучения с учителем требует наличия большого количества размеченных данных, которые должны быть использованы для обучения. Анализ предметной области [1] показал наличие ограниченного количества наборов данных, полученных с ОКТ установки, размеченных медицинскими специалистами и доступными для проведения обучения. Данный фактор обусловил необходимость поиска методов обучения классификатора, которые бы позволили достигнуть необходимого уровня точности распознавания в условиях, ограниченных тестовых наборов данных.

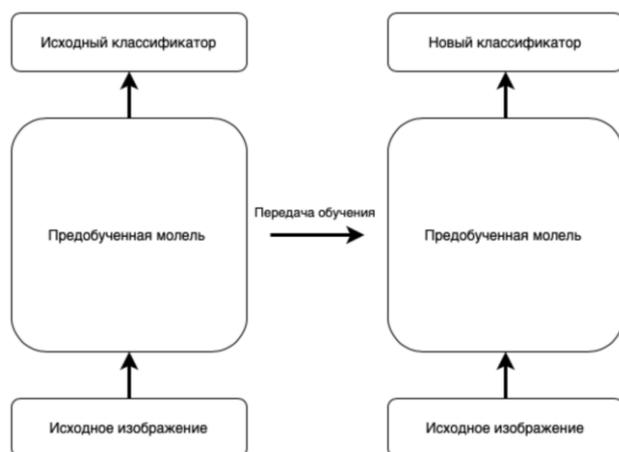


Рисунок 1 – Процесс передачи обучения

Одним из таких способов обучения является метод передачи обучения. Идея метода заключается в обучении нейронной сети для решения обобщенной задачи, для которой имеется достаточно большой набор данных, и последующей настройки ее параметров для решения конкретной задачи.

В данной работе рассматривается использование метода передачи обучения при построении классификатора патологий сетчатки человеческого глаза. Набор данных, использованный для обучения нейронной сети, содержит 80000 изображений сетчатки глаза взрослых пациентов, разделенные на четыре класса: хориоидальная неоваскуляризация, диабетическая ретинопатия, друзы, без патологий. Данные были разделены на обучающую, валидационную и тестовую

выборки. Размеры валидационной и тестовой выборки составляли 1000 изображений равномерно распределенным между классами.

В качестве исходной модели использовалась нейронная сеть SqueezeNet, обученная на базе изображений ImageNet. Данная модель характеризуется высокой точностью и небольшим набором параметров.

Для адаптации модели SqueezeNet к решению поставленной задачи был изменен выходной слой нейронной сети, т.к. количество классов патологий меньше количества классов в сети ImageNet. Остальные слои нейронной сети были заморожены, чтобы предотвратить изменение параметров предварительно обученной модели алгоритмом обратного распространения ошибки.

Обучение классификатора производилось в три эпохи с использованием алгоритма адаптивной оценки момента в качестве оптимизатора нейронной сети. На рисунке 2 представлены графики изменения функции потерь и точности предсказаний в процессе обучения.

В конце каждой эпохи обучения производилась проверка достоверности предсказаний с помощью валидационной выборки. Сохранение весов нейронной сети производилось в том случае, если точность предсказаний оказывалась выше, чем при предыдущей наиболее удачной конфигурации.

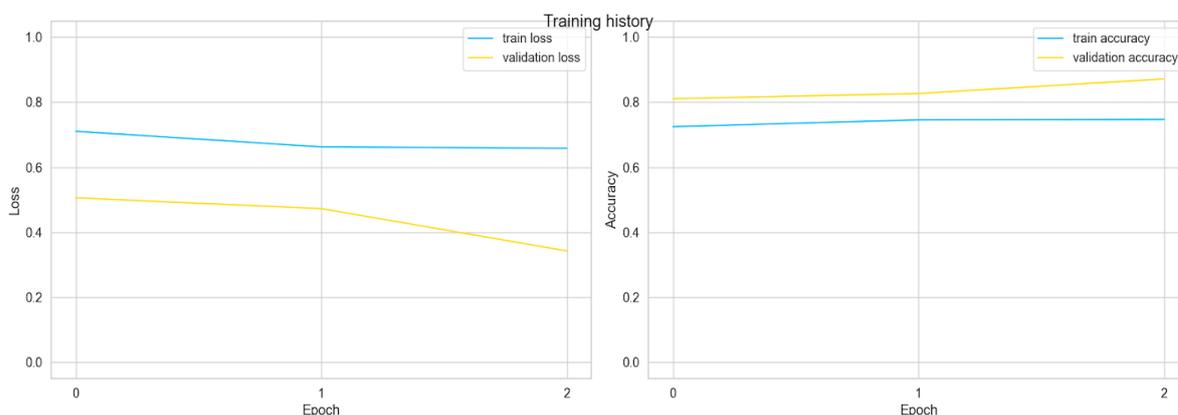


Рисунок 2 – Изменение функции потерь и точности предсказаний в процессе обучения

В результате обучения точность предсказаний нейронной сети достигла 87%. Результат проверки достоверности предсказаний представлен в виде матрицы предсказаний представленной на рисунке 3.

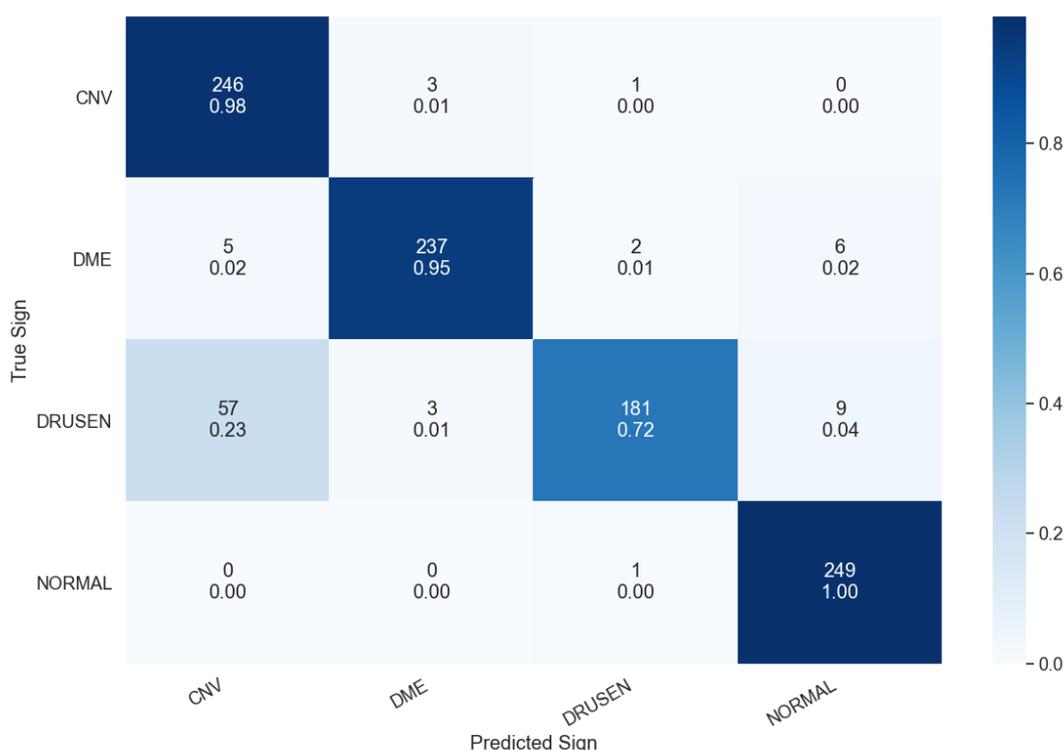


Рисунок 3 – Матрица предсказаний (несоответствий)

Полученные в ходе обучения классификатора результаты демонстрируют целесообразность использования метода передачи обучения для разработки нейронных сетей в условиях наличия ограниченного количества размеченных данных для обучения. Полученная точность показывает возможность передачи обучения даже в тех случаях, когда область знаний исходной модели значительно отличается по своей структуре от области знаний модели к которой передается обучение.

Разработанная модель является прототипом. Допускается изменение большего числа параметров исходной модели с целью повышения точности распознавания.

Список использованных источников:

1. Daniel S. Kermany, Michael Goldbaum, Wenjia Cai, M. Anthony Lewis, Huimin Xia, Kang Zhang. *Identifying Medical Diagnoses and Treatable Diseases by Image-Based Deep Learning* [Электронный ресурс]. – Режим доступа: <https://www.cell.com/action/showPdf?pii=S0092-8674%2818%2930154-5>. – Дата доступа: 25.03.2021.
2. S. P. K. Karri, Debjani Chakraborty, Jyotirmoy Chatterjee. *Transfer learning based classification of optical coherence tomography images with diabetic macular edema and dry age-related macular degeneration* [Электронный ресурс]. – Режим доступа: <https://www.ncbi.nlm.nih.gov/pmc/articles/PMC5330546/>. – Дата доступа: 30.03.2021

ПОСТРОЕНИЕ КАРТЫ ПОМЕЩЕНИЯ С ПОМОЩЬЮ SLAM

Лобач Р. Д., Видничук В. Н.

Белорусский государственный университет информатики и радиоэлектроники
г. Минск, Республика Беларусь

Перцев Д.Ю. – канд. техн. наук

Представлен способ построения 3D карты помещения, основанный на алгоритме SLAM, с использованием RPLidar A1.

Возможность построения 3D карты помещения дает большие преимущества в вопросах как исследования, так и безопасности, и охраны. Кроме построения такой карты важной задачей является поддержка ее в актуальном состоянии. Такую сложную работу удобнее всего отдать автономным системам.

Одним из самых известных методов построения карты и локализации робота в пространстве является SLAM (англ., Simultaneous localization and mapping). Данный метод используется в автономных мобильных устройствах для построения карты в неизвестном пространстве и окружении, обновлении с одновременным контролем текущего местоположения и пройденного нашим роботом пути [1].

В качестве аппаратной основы используется микрокомпьютер Raspberry Pi Zero W и лазерный дальномер RPLidar A1.

Raspberry Pi управляет передвижением робота и предоставляет доступ к данным центральному устройству (ноутбуку или персональному компьютеру), который будет заниматься обработкой и построением карты.

Лидар (англ., Light Detection and Ranging) способен считывать данные на расстоянии от 0.1 метра до 6 метров и сканировать окружающее пространство с частотой 5,5 Гц [2], что дает возможность обновлять строимую карту с довольно высокой скоростью. Для перемещения в пространстве используются 2 сервопривода. Управление сервоприводами осуществляется микроконтроллером Raspberry Pi.

Существует несколько разновидностей алгоритма SLAM. В данном случае используется вариант, основанный на фильтре частиц [3]. В общем случае построение двухмерной карты работает следующим образом:

- 1) карта представляет собой матрицу 3-значных переменных (пусто, есть препятствие, не определенное состояние). Значение по-умолчанию - пусто;
- 2) робот инициализирует свое положение;
- 3) считываются данные с лидара, на их основе строится начальная карта;
- 4) формируется дополнение карты:
 - 4.1) вычисляется точка, находящаяся на минимальном расстоянии от большинства неизвестных областей, откуда необходимо выполнить следующее измерение;
 - 4.2) робот перемещается в нужную точку и считывает данные;
 - 4.3) с помощью фильтра частиц вычисляется наиболее вероятное положение робота относительно текущей карты;
 - 4.4) считая, что робот находится в наиболее вероятном положении, дополняем текущую карту на основе считанных показаний.

Данная версия алгоритма подходит для построения 2D карт. Для 3D необходимо поворачивать плоскость сканирования лидара, в результате чего будет получен не двухмерный, а трехмерный массив точек.

Дальнейшая работа алгоритма практически идентична описанному выше. Основным отличием является модификация фильтра частиц для трехмерного массива.

Список использованных источников:

1. *Simultaneous Localisation and Mapping (SLAM): Part I The Essential Algorithms* [Electronic Resource] / Berkeley. – Mode of access: https://people.eecs.berkeley.edu/~pabbeel/cs287-fa09/readings/Durrant-Whyte_Bailey_SLAM-tutorial-I.pdf. Date of access: 27.03.2021.
2. *RPLidar A1* [Electronic Resource] / Slamtec. – Mode of access: <http://www.slamtec.com/en/lidar/a1>. Date of access: 27.03.2021.
3. *Простыми словами о фильтре частиц* [Электронный ресурс] / Habr. – Режим доступа: <https://habr.com/ru/post/276801>. Дата доступа: 27.03.2021.
4. Трунин, А. С. *Методы определения текущего местоположения автономного робота* / А. С. Трунин // Молодой ученый. – 2019. – № 36 (274). – С. 1-5.

РЕАЛИЗАЦИЯ ВЕБ-СЕРВИСА ДЛЯ РАБОТЫ С КАМЕРОЙ ВИДЕОНАБЛЮДЕНИЯ

Пархоменко А. И.

Белорусский государственный университет информатики и радиоэлектроники
г. Минск, Республика Беларусь

Луцик Ю. А. – канд. техн. наук, доцент

Данная работа посвящена описанию разработки программно-аппаратного приложения для реализации камеры видеонаблюдения, состоящей из одноплатного компьютера Raspberry Pi Zero W, а также модуля камеры Raspberry Pi Camera Rev 1.3, а также специально написанного веб-сервиса для управления камерой.

В современном мире системы видеонаблюдения являются важнейшей частью организации требуемого уровня безопасности во всех сферах человеческой деятельности. Однако, так как в настоящее время с быстрым развитием технологий стало доступно множество многофункциональных устройств для видеонаблюдения, возможность обезопасить себя появилась практически у каждого человека. Также возросло количество способов применения видеокамер, поэтому обеспечение требуемого уровня защиты путем слежения за территорией стало не единственной возможной выполняемой задачей. Камеры также могут применяться в различных сферах услуг для контроля разного рода процессов. Например, некоторые задачи видеоаналитики могут потребовать дополнительную фильтрацию и обработку полученных изображений. Так, например, данный механизм используется при идентификации объектов, зафиксированных камерой наблюдения. Система может сканировать лица людей для установления их личности и получения информации из имеющихся баз данных. Это достигается с помощью специализированных алгоритмов, основанных на использовании нейронных сетей, проводящих анализ полученных изображений.

В результате появляется очевидная проблема при выборе эффективных камер, составляющих систему видеонаблюдения, которая заключается в необходимости существенных денежных затрат. Поэтому вне зависимости от того, как будет использоваться видеонаблюдение, будь оно архивным или оперативным, выбор сводится в пользу наиболее экономичных средств.

Было принято решение о разработке собственной камеры, которая отвечает требованиям необходимого качества съемки, а также обладает всем нужным функционалом, управлять которым можно с помощью разработанного пользовательского интерфейса, за относительно небольшую стоимость.

Целью является реализация веб-сервиса для работы с камерой видеонаблюдения. Такая система может быть полезна пользователям, желающим иметь доступ к камере и возможность дистанционного управления при помощи удобного пользовательского интерфейса. Такая камера будет способна вести наблюдение в реальном времени и предоставлять актуальную информацию о наблюдаемом объекте или местности. В целях экономии ресурсов хранилища данных на устройстве, будет реализована возможность сохранения изображений в базу данных, что даст возможность делать снимки практически в неограниченном количестве. Помимо этого, будет реализована возможность идентификации объектов на полученных снимках при помощи нейронной сети.

Задачи разработки: 1) реализовать веб-сервис для управления камерой; 2) реализация дистанционного доступа к полученным с камеры изображениям; 3) реализация просмотра полученных изображений; 4) предусмотреть возможность фильтрации изображений; 5) добавить базу данных для хранения изображений; 6) разработать нейронную сеть для идентификации объектов.

Вначале были выбраны аппаратные средства для разработки. Устройство для съемки изображений состоит из двух модулей: 1) одноплатный компьютер Raspberry Pi Zero W [1], предоставляющий необходимую вычислительную мощь для обработки поставленных задач; 2) камера Raspberry Pi Camera Rev 1.3, предоставляющая нужное качество получаемых изображений.

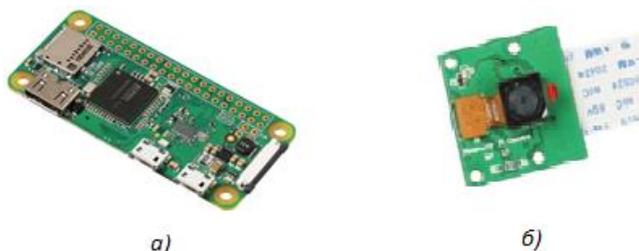


Рисунок 1 - Модули камеры: а) Raspberry Pi Zero W, б) Raspberry Pi Camera Rev 1.3

Далее осуществлялся выбор технологий программирования для создания веб-сервиса. Приложение представляет собой клиент-серверную архитектуру. Модель такой системы основана на том, что клиент отправляет запросы серверу для обработки самого запроса, после чего, сервер

готовые данные возвращает клиенту, поэтому для написания как клиента, так и сервера необходимо выбрать свои наиболее подходящие технологии.

Сервер – это устройство, представляющее собой вычислительное ядро системы, которое выполняет определенные задачи, запрашиваемые клиентом и предоставляет доступ к определенным ресурсам, хранения информации и баз данных. Для разработки этой части приложения предпочтение было отдано в пользу Express.js. Это один из существующих базовых фреймворков JavaScript, ключевые особенности которого заключаются в предложении надежного механизма маршрутизации для обработки динамических URL адресов, упрощении отладки кода, а также в предоставлении динамического рендеринга страниц на серверной стороне. Основные плюсы его использования заключаются в наличии хорошей документации, обеспечении быстрой разработки с использованием node.js, а также интеграции с наиболее популярными базами данных, например, MySQL, MongoDB. В приложении используется база данных MongoDB, учитывая ее отличную совместимость с используемыми в разрабатываемом веб-приложении технологиями.

Клиент – это компьютер на стороне пользователя, который отправляет серверу запросы для предоставления информации или обработки каких-либо действий. Так как основными инструментами для разработки клиентской части приложения на сегодняшний день являются HTML, CSS и JavaScript [2], учитывая скорость работы и производительность, мощную инфраструктуру, простоту и рациональность применения, удобство пользовательских интерфейсов, легкость освоения, были использованы именно они.

Так как важной особенностью разрабатываемого приложения является наличие возможности распознавания объектов, зафиксированных на изображениях, полученных с камеры, то была необходимость в использовании алгоритма сверточной нейронной сети [3], которая характеризуется наилучшими результатами в области распознавания лиц.

Таким образом был разработан веб-сервис для дистанционного управления камерой видеонаблюдения, где предусмотрены следующие возможности: 1) выбор разрешения изображения; 2) выбор места для сохранения изображения. Здесь можно сохранить или в базу данных, или на устройство в зависимости от решения пользователя; 3) выбор названия изображения. Если не выбрать, то файл сохранится с названием по умолчанию; 4) настройка яркости, контрастности и насыщенности; 5) идентификация объектов на выбранном из галереи изображении.

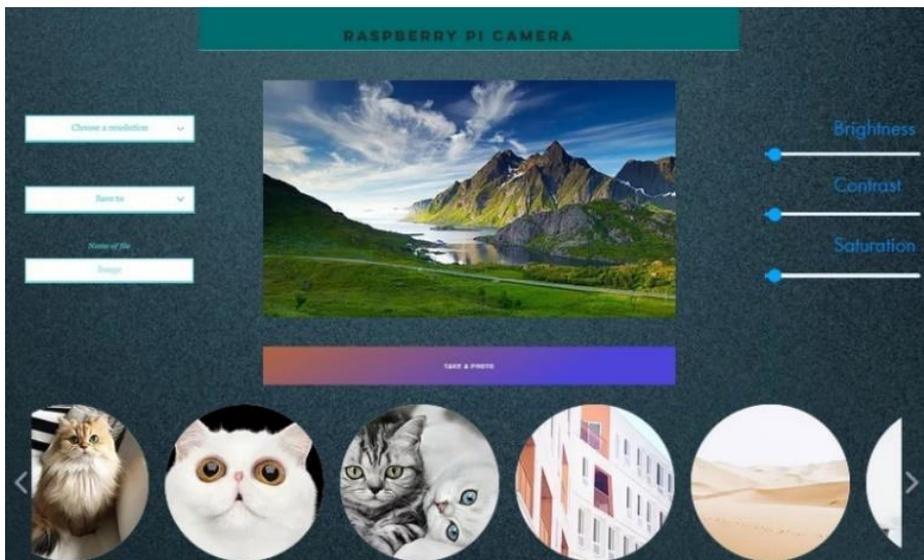


Рисунок 2 – Веб-сервис для управления камерой видеонаблюдения

Учитывая возможности приложения, его широкое практическое применение, доступность аппаратных средств, на данном этапе уже готово устройство, обладающее рядом значительных преимуществ относительно уже существующих камер. Важной особенностью также является наличие исходного кода приложения для возможных модификаций, изменений или улучшений.

Далее планируется совершенствовать уже существующее веб-приложение, добавляя ряд новых функций, в число которых входит трансляция видео с камеры в реальном времени, добавление возможности авторизации, также добавление множества других настроек обработки получаемого с камеры изображения. Помимо всего прочего, хотелось бы реализовать функцию автоматического распознавания лиц, которая позволяет в процессе видеонаблюдения фокусироваться только на некоторых, специально выбранных из базы данных лицах, как только те попадают на камеру.

Список использованных источников:

1. Vasilis Tzivaras. *Raspberry Pi Zero W Wireless Projects*. Packt Publishing, 2017. – 240 с.
2. Дэвид Флэнаган. *JavaScript: Подробное руководство*. Символ – Плюс, 2013. – 1080 с.
3. Я. Гудфеллоу, И. Бенджио, А. Курвилль. *Глубокое обучение*. ДМК, 2017. – 652 с.

ИССЛЕДОВАНИЕ АВТОМАТИЧЕСКОЙ ВЕКТОРИЗАЦИИ НА ПРИМЕРЕ КОМПИЛЯТОРА MSVC 14.28

Пикиреня П.И., Русакович А.В.

Белорусский государственный университет информатики и радиоэлектроники
г. Минск, Республика Беларусь

Перцев Д.Ю. – канд. техн. наук

Продемонстрированы некоторые проблемы, возникающие при работе с автоматической векторизации циклов компилятором MSVC 14.28.

Составная часть компилятора, отвечающая за автоматическое распараллеливание и векторизацию, обеспечивают существенное повышение производительности исполнения цикла в коде. Векторизация – вид распараллеливания программы, при котором однопоточные приложения, выполняющие одну операцию на одном элементом в каждый момент времени, модифицируются для выполнения одной операции сразу над некоторым множеством данных.

Примерами векторных инструкций являются SSE2, AVX и AVX2 в процессорах Intel и AMD [1], NEON в процессорах ARM. Простейший пример, демонстрирующий работу векторизации показан на рисунке 1. Слева показано простое сложение двух векторов на языке программирования C. Справа – векторная версия данного алгоритма, при этом число итераций массива уменьшено в 4 раза, т.к. инструкции movups и addps работают сразу с четырьмя элементами массива.

<pre>float* a, b, c; // инициализация a, b, c пропущена for (unsigned i = 0; i < 1000; i++) { c[i] = a[i] + b[i]; }</pre>		<pre>mov ecx, 255 mov offset, 0 loop1: movups xmm1, [MEM1 + offset] movups xmm2, [MEM2 + offset] addps xmm1, xmm0 movups [MEM3 + offset], xmm1 add offset, 16 loop loop1</pre>
--	--	--

Рисунок 1 – Пример работы алгоритма векторизации

При замене ограничителя числа итераций (в примере выше – 1000) на константу size, поведение компилятора изменяется и зависит от следующих условий:

- 1) данное значение определено как константа или изменяется
- 2) ее размер.

Чем меньше значение size, тем меньше вероятность успешной автоматической векторизации (например, при значении size = 4 несмотря на то, что вручную векторизовать можно, авторам так и не удалось заставить компилятор векторизовать вручную). При динамическом изменении размера компилятор генерирует две версии кода и определяет, можно ли исполнить векторную версию кода во время исполнения кода. В зависимости от результата выполняется векторная версия (в приоритете) либо традиционная скалярная.

Также неожиданный результат был получен при исполнении кода на рисунке. 2:

```
for (int l = 0; l < size; ++l) {
    c[l] += a[q] * b[l];
}
```

Рисунок 2 – Пример кода с ошибкой векторизации

Несмотря на то, что все условия векторизации выполнены, была получена ошибка векторизации с кодом 1203 (не смежный доступ к массиву [2]). Объяснения, почему так происходит, найдено не было.

Представленный материал будет полезен студентам, изучающим основы векторизации.

Список использованных источников:

1. Автоматическая параллелизация и автоматическая векторизация [Электронный ресурс] / Microsoft. – Режим доступа: <https://docs.microsoft.com/ru-ru/cpp/parallel/auto-parallelization-and-auto-vectorization?view=msvc-160#auto-vectorizer3>. Дата доступа: 27.03.2021.

2. Сообщения векторизатора и параллелизатора [Электронный ресурс] / Microsoft. – Режим доступа: <https://docs.microsoft.com/ru-ru/cpp/error-messages/tool-errors/vectorizer-and-parallelizer-messages?view=msvc-160>. Дата доступа: 27.03.2021.

ПРОЦЕССОР ОБРАБОТКИ ИЗОБРАЖЕНИЙ НА ОСНОВЕ НЕРАЗДЕЛИМОГО ДВУМЕРНОГО ПОДХОДА

Самусенко М.А.

Белорусский государственный университет информатики и радиоэлектроники
г. Минск, Республика Беларусь

Петровский Н.А. – канд.тех. наук.

Двумерные дискретные вейвлет-преобразования (2D DWT), и другие схожие алгоритмы широко используются во многих инженерных приложениях из-за их способности к разложению. Эти алгоритмы эффективны, так как они позволяют уменьшить избыточность исходного сигнала и произвести его сжатие без существенных потерь. Тем не менее, обработка больших объемов данных различной способности к разложению делает их вычисление вычислительно очень ёмким и затруднительным [1].

Одномерные преобразования можно применять для построения многомерных разделимых систем. Двумерные сигналы (изображения) отдельно преобразуются по вертикали и горизонтали. Однако многомерные сигналы, в большинстве, неразделимы, и этот подход не позволяет эффективно использовать их характеристики. Двумерные неразделимые преобразования работают более эффективно для кодирования изображений, чем разделимые, потому что неразделимые преобразования учитывают характер входного сигнала.

Ранее были предложены архитектуры для обеспечения высокоскоростного вычисления двумерного DWT с требованием уменьшения использования количества аппаратных ресурсов. Эти архитектуры в широком смысле могут быть подразделены на разделяемые и неразделимые. В разделяемой архитектуре двумерная операция фильтрации делится на две операции одномерной фильтрации, одна для обработки данных по строкам, а другая по столбцам. Разделимые архитектуры, в которых используется структура 1-D фильтрации для выполнения 2-D DWT, есть дополнительное требование транспонирования промежуточных данных между двумя процессами одномерной фильтрации. Это увеличивает размер памяти, требуемой для хранения промежуточных значений результатов вычислений, а также увеличивает задержку. В неразделимых архитектурах эта проблема решается, так как в этом варианте реализации двумерные преобразования вычисляются напрямую с помощью двумерных преобразований [2]. Пример таких вычислений приведен на рисунке 1.

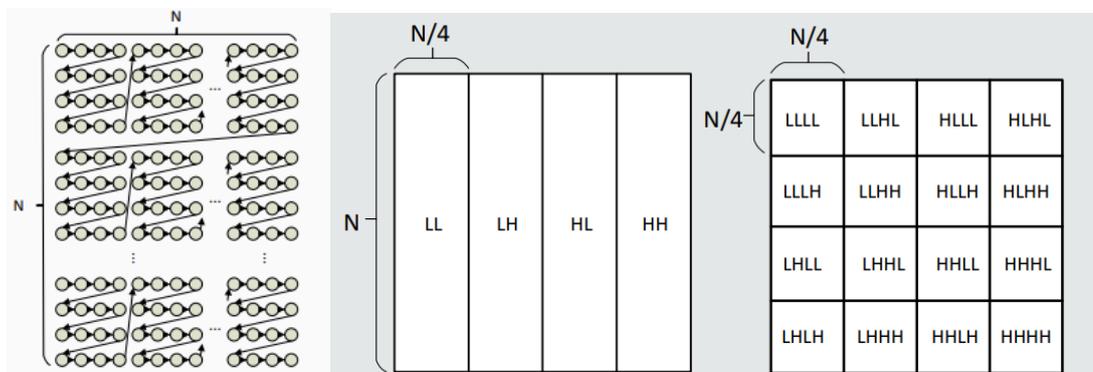


Рисунок 1 - Этапы обработки изображения с помощью двумерного 4-полосного банка фильтров [3]

При реализации таких архитектур использование FPGA является целесообразным, так как неразделимые преобразователи могут быть синтезированы из двух разделимых и имеют возможность работать сразу в двумерном домене и их основное преимущество при реализации на FPGA – высокая пропускная способность и сведение к нулю использования памяти для хранения промежуточных данных.

Основной задачей является изучение и оценка эффективности структур, реализованных на основе неразделимых преобразований. Сама разработка предполагает использование ее в виде IP-блока при создании специализированной цифровой аппаратуры (предполагаемая реализация – на микросхеме SoC серии Zynq).

Список использованных источников:

1. Zhang.C. A VLSI architecture for a fast computation of the 2-D discrete wavelet transform / C. Zhang, C.Wang, M.O. Ahmad. - Proc. IEEE Int. Symp. Circuits and Systems (ISCAS), 2007. - p. 3980–3983.
2. Zhang C. An efficient buffer-based architecture for on-line computation of 1-D discrete wavelet transform / C. Zhang, C.Wang, M.O. Ahmad. - Proc. IEEE Int. Conf. Acoustics, Speech, Signal Process. (ICASSP), 2004. - p. 201–204.
3. Petrovsky N. Two-dimensional non-separable quaternionic paraunitary filter banks / N. Petrovsky, E.Rybenkov, A. Petrovsky // SPA2018 – Poznan, 2018 - p.120-125

ПЛАТФОРМА ДЛЯ СЪЕМКИ СФЕРИЧЕСКИХ ПАНОРАМ С ИСПОЛЬЗОВАНИЕМ RASPBERRY PI

Стракович А.И.

Белорусский государственный университет информатики и радиоэлектроники
г. Минск, Республика Беларусь

Луцкич Ю.А. – канд. техн. наук, доцент

Показан подход к созданию платформы для съемки сферической панорамы, основанный на использовании множества камер. Рассмотрены различные камеры и варианты работы с ними.

Сферическая панорама представляет из себя собранное из множества отдельных кадров изображение в сферической проекции. Характерной чертой сферических панорам является максимально возможный угол обзора пространства (360×180 градусов). Просматривая такие панорамы создается эффект присутствия на рассматриваемом месте.

Для создания сферических панорам, необходимо получить множество фотографий в один момент времени либо за минимальное время между кадрами, чтобы движущиеся объекты находились на одном и том же месте на разных фото. Иначе совместить такие фото будет невозможно, либо на итоговой панораме случайно попавший в кадр движущийся объект будет прозрачным.

Платформа для съемки сферических панорам представляет собой сферический объект в центре которого расположен основной управляющий модуль, а по бокам располагаются камеры.

Основным управляющим устройством, для объединения фотографий в одну панораму, выступает одноплатный компьютер Raspberry Pi 3B+ [1]. Выбранная плата имеет порт для подключения камеры. Однако подключить больше одной камеры затруднительно. Для данной цели можно использовать специальный модуль расширения, но его стоимость больше, чем стоимость самой Raspberry Pi. К тому же модуль позволяет в один момент времени работать только с одной камерой.

В качестве альтернативы камер Raspberry можно использовать камеры OV2640 [2]. Данные камеры также не просто подключить к Raspberry Pi в связи с отсутствием подходящего интерфейса. Можно использовать совместимый интерфейс I2C, при этом в один момент времени работать можно будет также только с одной камерой.

Решением данной проблемы является использование модуля ESP32-cam, основанном на чипе ESP32-S [3]. Данный модуль позволяет легко работать с камерой OV2640, а встроенной памяти размером 520 КБ хватает для хранения как минимум одного кадра. Используя несколько данных модулей, можно сделать снимки в один момент времени, а не последовательно, что позволит избежать смещения движущихся объектов между кадрами.

Существуют несколько вариантов модулей камер OV2640. Они отличаются углом обзора. Наиболее распространены камеры с углами 70 градусов и 160 градусов. Для покрытия всего обзора по горизонтали понадобится минимум 3 камеры с объективом в 160 градусов и еще 2 камеры для съемки надира и зенита.

При запуске процесса съемки Raspberry подает сигнал всем ESP32-cam чтобы каждая камера сделала снимок. Спустя одну секунду Raspberry поочередно опрашивает каждый модуль и принимает от него кадр изображения через SerialPort.

Для выбора конкретной камеры используется мультиплексор и демультимплексор для линии Rx и Tx соответственно. Далее все 5 полученных изображений объединяются в одну панораму посредством библиотеки OpenCV [4].

Общий алгоритм объединения фотографий заключается в поиске особых точек на изображениях, попарное сопоставление кадром по особым точкам, построение проективного преобразования для выравнивания изображений и их переноса в общую плоскость, корректирование цвета на границе объединения, сохранение изображения.

В связи с тем, что камеры будут статичны относительно друг друга, то области перекрытия на всех полученных изображениях будут неизменны. Используя данную особенность можно оптимизировать код объединения изображений в панораму. В процессе калибровки камер делаются снимки и объединяются в панораму, при этом сохраняется карта перекрытий изображений. Любые последующие объединения происходят согласно данной карте, без поиска общих точек и попарного сопоставления кадров, что позволяет увеличить производительность.

Список использованных источников:

1. Raspberry Pi 3B+ Datasheet [Электронный ресурс] – Режим доступа : www.raspberrypi.org/documentation/hardware/computemodule/datasheets/rpi_DATA_CM3plus_1p0.pdf – Дата доступа : 01.03.2021.
2. Камера OV2640 Datasheet [Электронный ресурс] – Режим доступа : www.uctronics.com/download/cam_module/OV2640DS.pdf – Дата доступа : 01.03.2021.

СИСТЕМА ДИСТАНЦИОННОГО ОБУЧЕНИЯ ТЕХНИЧЕСКИМ ДИСЦИПЛИНАМ

Холупко А.А

Белорусский государственный университет информатики и радиоэлектроники
г. Минск, Республика Беларусь

Блинов И.Н. – канд. физ.-мат. наук

Доклад посвящен исследованию методов обучения и описанию разработки системы дистанционного обучения техническим дисциплинам.

Дистанционная форма обучения дает возможность создания систем массового непрерывного самообучения, всеобщего обмена информацией, независимо от временных и пространственных поясов. Стоит отметить, что важным параметром хорошо разработанной системы дистанционного обучения является эффективность обучения. В свою очередь эффективность обучения зависит от методик обучения.

В данной работе было проведено теоретическое исследование наиболее эффективных методик обучения. На данный момент наиболее эффективными являются методики, которые включают в себя практические задачи в ходе обучения и методы интервального повторения изучаемого материала [1].

В ходе данной работы также был проведен эксперимент на определение влияния комбинирования данных методик на эффективность обучения.

В ходе эксперимента было отобрано 14 участников, выбран технический материал для изучения. Участники были разбиты на 2 группы по 7 человек. Первая группа проходила обучение при помощи метода «практического тестирования» [2], а вторая – при помощи комбинации методов «практического тестирования» и «интервального повторения» [3].

Данный эксперимент показал, что комбинирование методик обучения положительно влияет на эффективность обучения. Результат эксперимента представлен на рисунке 1.

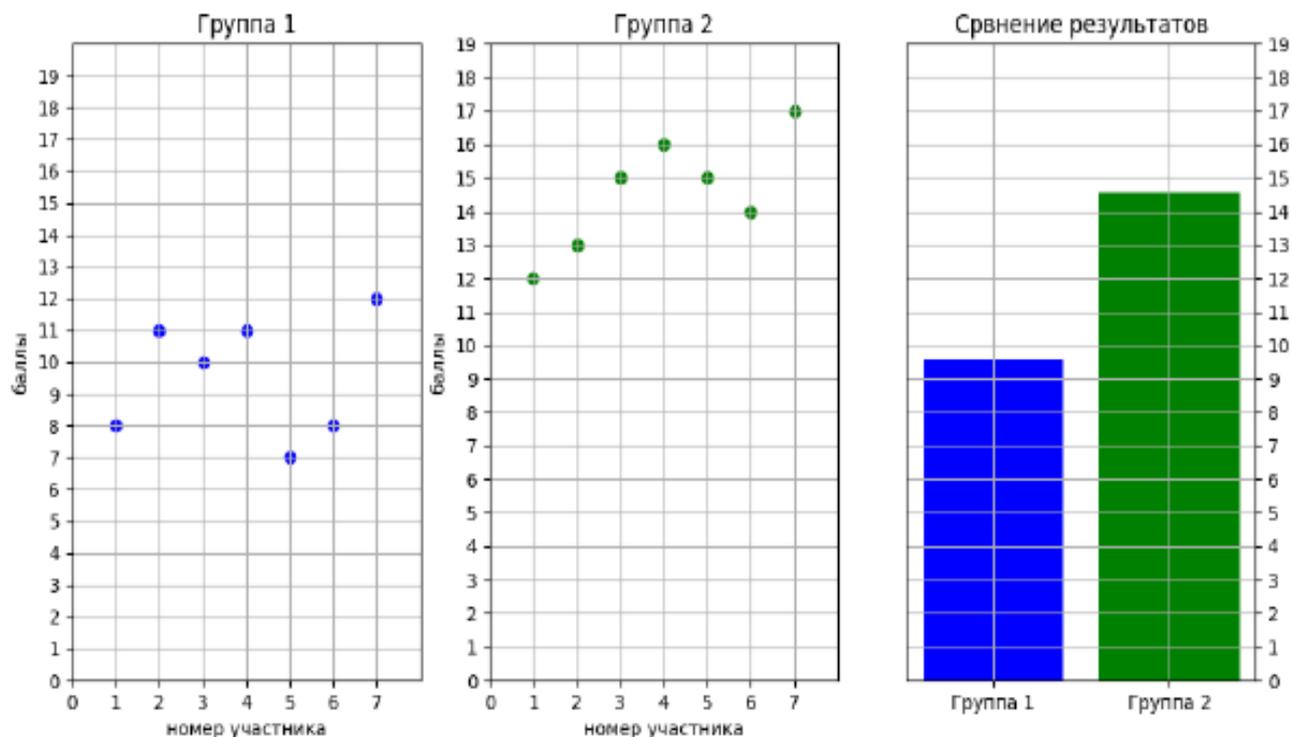


Рисунок 1 – Результат проведенного эксперимента

Для разработки системы дистанционного обучения в качестве методики обучения была взята комбинация методик, участвовавших в эксперименте.

Система дистанционного обучения представляет собой веб-приложение, которое позволяет самостоятельно изучать технические дисциплины дистанционно.

Проектируемая система дистанционного обучения позволяет пользователям изучать учебные материалы по курсам, проходить тесты, сохранять прогресс по курсам.

В данной работе был проведен обзор существующих технологий и подходов к разработке веб-приложений.

Основными технологиями, которые были выбраны для разработки системы дистанционного обучения техническим дисциплинам, являются Java, SQL, ReactJS. На рисунке 2 представлена архитектура разработанной системы.

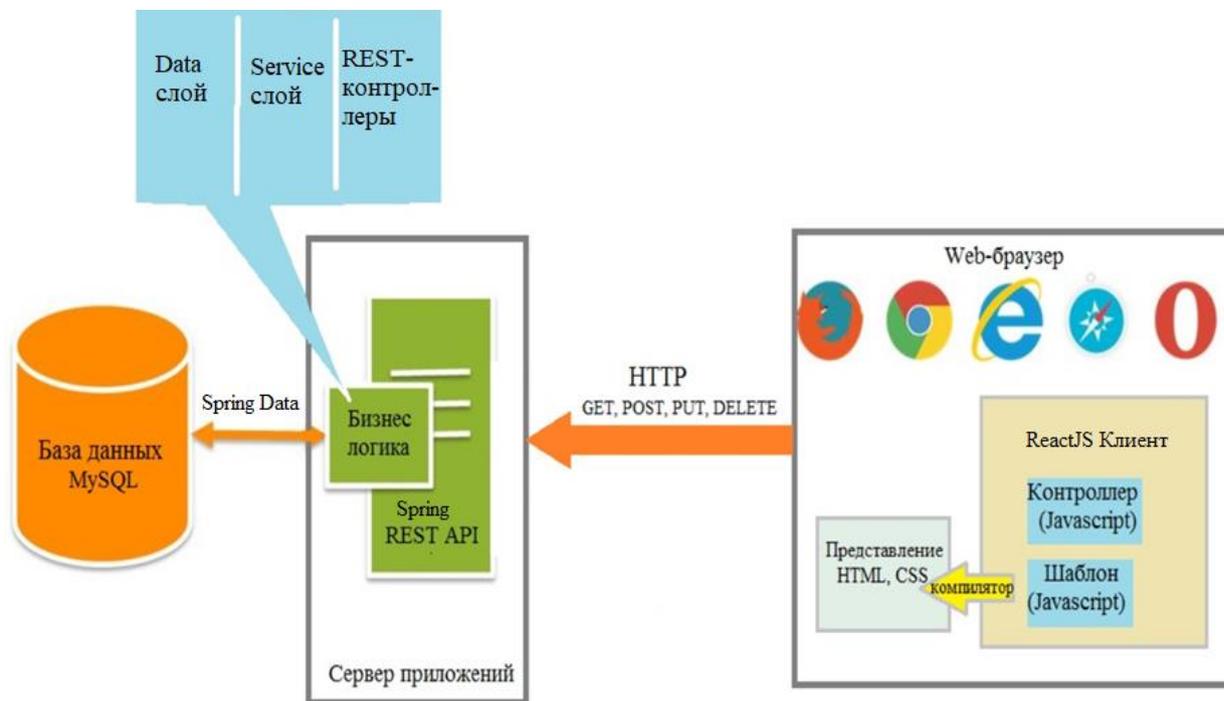


Рисунок 2 – Архитектура разработанной системы

Разработанная система состоит из трех компонентов: база данных (для хранения данных), серверная часть приложения (для реализации бизнес логики приложения), клиентская часть (для реализации пользовательского интерфейса).

В качестве базы использовалась реляционная система управления баз данных MySQL.

Серверная часть представляет собой программный интерфейс приложения REST. Данный программный интерфейс реализован при помощи Java-фреймворка Spring. Также данный модуль содержит три независимых уровня: слой данных, слой сервисов, слой контроллеров.

Слой данных используется для выполнения запросов к базе данных. Данный слой реализован при помощи Java-фреймворка Spring, в частности модуля Spring Data.

Слой сервисов используется для обработки данных из слоя данных и выполнения бизнес логики приложения. Данный слой реализован при помощи Java-фреймворка Spring.

Слой контроллеров используется для обработки запросов от клиентской части приложения. Данный слой реализован при помощи Java-фреймворка Spring, в частности модуля Spring MVC.

Клиентская часть приложения представляет собой интерфейс для взаимодействия пользователя с системой. Данная часть приложения реализована при помощи JavaScript-фреймворка ReactJS. Связь между серверной частью приложения и базой данных реализована при помощи Java-фреймворка Spring, в частности модуля Spring Data. Связь между клиентской частью и серверной была осуществлена по протоколу HTTP. Взаимодействие пользователя и системы происходит через веб-браузер.

Особенностями разработанной системы дистанционного обучения техническим дисциплинам является использование комбинации наиболее эффективных методик обучения, разработка системы при помощи современных веб-технологий, простота пользовательского интерфейса, возможность самообучения и самоконтроля знаний.

Список использованных источников:

1. *Improving Students' Learning With Effective Learning Techniques: Promising Directions From Cognitive and Educational Psychology* / J. Dunlosky [et al.] // *Improving Student Achievement*, 2010. – P. 29–45.
2. *Some effects of remembering on forgetting* / W. N. Runquist [et al.] // *Memory & Cognition*, 1983. – P. 641–650.
3. *Maintenance of knowledge: Questions about memory we forgot to ask* / H. P. Bahrick [et al.] // *Journal of Experimental Psychology: General*, 1979. – P. 296–308.

ПОДХОДЫ В МАШИННОМ ОБУЧЕНИИ ДЛЯ ЗАДАЧ КРЕДИТНОГО СКОРИНГА

Шичков Д.В.

Белорусский государственный университет информатики и радиоэлектроники
г. Минск, Республика Беларусь

Фролов И.И. – канд. техн. наук, доцент

В данной работе рассматриваются базовые методы и подходы, которые существуют в машинном обучении для решения задач кредитного скоринга: линейная модель, логистическая и решающие деревья.

Линейная регрессия – модель зависимости переменной от одной или нескольких других переменных (факторов, регрессоров, независимых переменных) с линейной функцией зависимости. Линейная регрессия относится к задаче определения «линии наилучшего соответствия» через набор точек данных и стала простым предшественником нелинейных методов, которые используют для обучения нейронных сетей [1].

Пусть $p = 0.7 - 0.01 * a + 0.5 * k$ – вероятность невозврата кредита, где a – возраст заёмщика; k – кредитная нагрузка заёмщика. В представленной выражении сделано предположение того, что чем старше заёмщик, тем более вероятно он вернёт кредит и чем больше у него уже имеется долгов, тем менее вероятно он вернёт кредит. На рисунке 1 наглядно показано, что линейная модель оценки риска дефолта не подходит для решение данной задачи, т.к. с увеличением возраста заёмщика, вероятность возврата становится отрицательной: возраст больше 70, означает вероятность меньше 0.

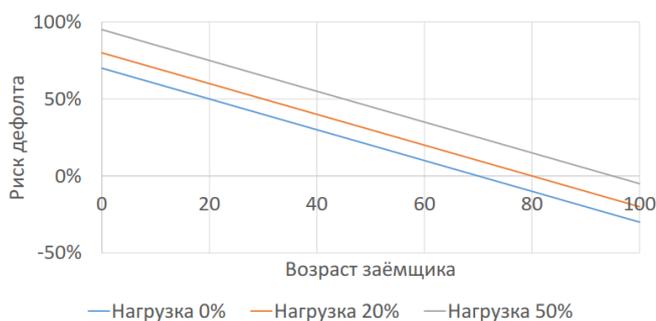


Рисунок 1 – График зависимости риска дефолта от возраста заёмщика с определённой нагрузкой, который иллюстрирует проблему линейной регрессии

Логистическая регрессия представляет собой статистическую модель, которая используется для прогнозирования вероятности возникновения определённого события путём подгонки данных к некой логистической кривой. Главной идеей логистической регрессией является тот факт, что пространство исходных значений может быть разделено на две части соответствующих классам области [2].

Пусть $p = 1 / (1 + \exp(-Z))$, где Z представляет собой любую линейную функцию. Можно заметить, что чем больше эта линейная функция, тем меньше $\exp(-Z)$ и тем выше вероятность невозврата кредита.

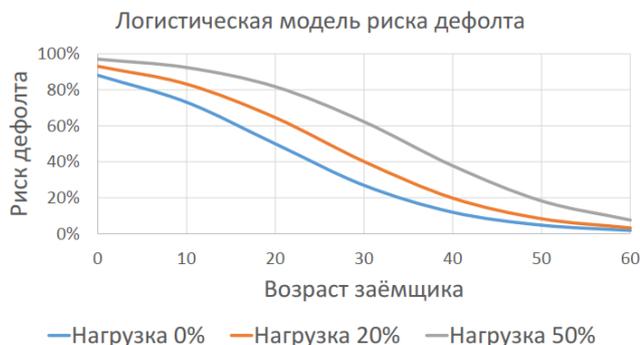


Рисунок 2 – График зависимости риска дефолта от возраста заёмщика с определённой нагрузкой для логистической функции

На рисунке 2 видно, что логистическая функция более подходит для решения задачи оценки кредитных рисков, т.к. значения вероятности дефолта не становится отрицательной, а стремится к нулю при возрасте старше 60 лет.

Недостатки логической регрессии:

- 1) довольно сложная формула (не рассчитать в уме):
 $p(y = 1) = 1 + \exp(0.02 \times \text{срок} + 0.001 \times \text{сумма} - 0.09 \times \text{возраст} + \dots) - 1$;
- 2) не умеет учитывать нелинейности;
- 3) не умеет учитывать взаимодействие признаков;
- 4) на рисунке 3 показан пример.



Рисунок 3 – Пример недостатка логической регрессии

Можно рассмотреть задачу оценки кредитного риска на примере решающего дерева [3]. На рисунке 4 показан пример решающего дерева на основе желаемого результата и полученного.

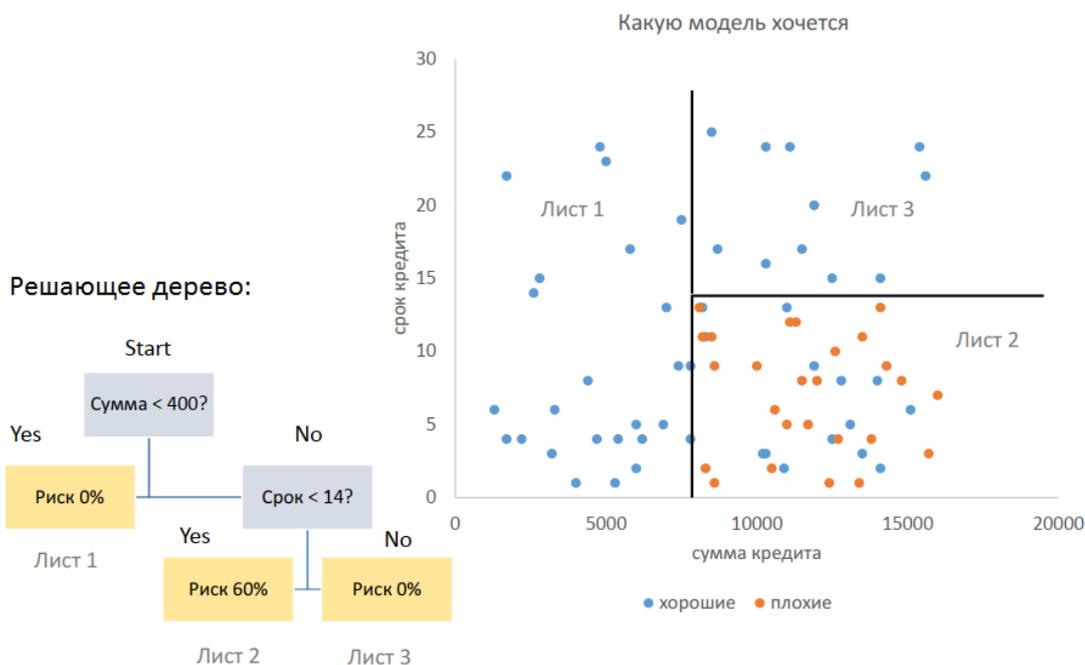


Рисунок 4 – Пример решающего дерева для кредитного скоринга

Алгоритм построения решающего дерева:

- 1) Перебрать все признаки.
- 2) Для каждого перебранного признака перебрать все пороги.
- 3) Выбрать признак и порог, которые приводят к наибольшему росту точности модели
- 4) Разделить в этом месте выборку на две части.
- 5) Для каждой разделённой половине повторить шаг 1.
- 6) Если таковых не оказалось, вернуться.

Список использованных источников:

1. Линейная регрессия в машинном обучении [Электронный ресурс] – Режим доступа: <https://neurohive.io/ru/osnovy-data-science/linejnaja-regressija/> – Дата доступа: 20.03.2021.
2. Логистическая регрессия [Электронный ресурс] – Режим доступа: <http://statistica.ru/theory/logisticheskaya-regressiya/> – Дата доступа: 20.03.2021.
3. Соколов Е.А. Решающие деревья [Электронный ресурс] – Режим доступа: <https://www.hse.ru/mirror/pubs/share/215285956>. – Дата доступа: 20.03.2021.

СЕКЦИЯ «ФИЗИКА»

КВАНТОВАЯ ПСИХОЛОГИЯ

Савостьянов И.Н.

Белорусский государственный университет информатики и радиоэлектроники
г. Минск, Республика Беларусь

Савилова Ю.И – канд. тех. наук, доцент

Квантовая психология – концепция в области парапсихологии, основанная на гипотезе о том, что сознание носит волновой характер и порождается квантово-волновой активностью мозга. Применение квантовых эффектов к области психологии в настоящее время является недостаточно обоснованным с позиции как физики, так и психологии. Согласно «Нейрокомпьютерной модели сознания» (Хамерофф и Пенроуз, 1994) активность головного мозга рассматривается как квантовый процесс, подчиняющийся закономерностям квантовой физики. Квантовая психология основана на гипотезе о том, что сознание носит волновой характер и порождается квантово-волновой активностью мозга. При этом предполагается, что при совпадении колебаний «волн сознания» человека с квантово-волновой характеристикой физического объекта в сознании человека возникают связанные с данным объектом мысли и образы, а изменение мысленного образа в мозговом нейрокомпьютеринге способно на квантовом уровне видоизменять материальный объект. Также предполагается, что при этом происходит взаимодействие сознания, материализованного на каком-либо носителе, с живыми и неживыми объектами материального мира [1].

Человек рассматривается как квант интеллектуальной системы Вселенной с взаимосвязанными процессами и алгоритмами эволюционного развития.

Вселенная, с включенными в ее состав интеллектуальными системами меньшего порядка, базируется на стратегическом управлении и непрерывном процессе поддержки системы на всех этапах ее жизненного цикла.

Человек в общем случае представляет собой упорядоченную наносистему с протекающими в ней нанопроцессами, которые включают в себя вещество и энергоинформационную составляющую. Информационное общество в эволюции развития человечества на земле базируется на сингулярности сознания (мышления) человека, в основе которой заложено единство и взаимосвязь вещества и энергии во Вселенной с соответствующими процессами преобразования и трансформации. Так как в основе квантовой психологии лежат открытия, совершенные в области квантовой механики, то личность человека и окружающая его материя рассматриваются с холистических позиций как единая сложная и неделимая Вселенская система со всей совокупностью протекающих в ней процессов. Наиболее интересным открытием физики является фундаментальное положение о всеобщем единстве. Ведь буквально на субатомном уровне наш мир и мы – одно целое. В свое время Альберт Эйнштейн высказал мысль о том, что «человек — это часть целого, которое мы называем Вселенной». Принципы квантовой механики также можно обнаружить применительно к психодиагностике личности, где представление о личности во многом сходно с представлением квантовых систем в физике. Проявление у личности какого-либо конкретного психического состояния в определенных условиях, описываемого личностным профилем может представляться как редукция волновой функции Ψ в физике. [2].

В 70-е гг. XX века академик В. Чавчанидзе писал о том, что сознания людей представляют собой своеобразный вид энергоинформационной материи, который подлежит научному описанию с позиций квантовой теории. Сознания людей находятся под управлением когерентных (взаимосвязанных) волн при этом люди объединяя свои сознания, создают более масштабные энергоинформационные сущности (душу общества или психозмоциональные поля), которые влияют на процесс их эволюционного развития. В целом он утверждал, что любая мыслительная деятельность человека должна описываться квантово-волновыми законами [3].

Список использованных источников:

1. Красильников Г. Т., Мальчинский Ф. В., Крачко Э. А. *О научном статусе квантовой психологии* [Электронный ресурс]. – Режим доступа: <https://cyberleninka.ru/article/n/o-nauchnom-statuse-quantovoy-psihologii/viewer> – Дата доступа: 10.04.2021.
2. Квантовая психология в системе нового знания [Электронный ресурс]. – Режим доступа: <https://www.b17.ru/blog/114862> – Дата доступа: 10.04.2021.
3. Чавчанидзе В.В. Квантово-волновая теория мышления // Материалы IV Всесоюзного съезда общества психологов. Тбилиси, 1971. С. 134—135.

ПЕРСПЕКТИВА ИСПОЛЬЗОВАНИЯ КВАНТОВОЙ КРИПТОГРАФИИ КАК ОСНОВНОГО МЕТОДА ЗАЩИТЫ КОММУНИКАЦИИ

Хомельянский Д.В.
Ермакович Н.В.

Белорусский государственный университет информатики и радиоэлектроники
г.Минск, Республика Беларусь

Савилова Ю.И – канд. тех. наук, доцент

В данной работе рассмотрена перспектива использования квантовой криптографии, как метода защиты коммуникации, а также последние исследования, направленные на развитие и популяризацию этого метода.

Криптография – общий термин, к которому относятся методы, обеспечивающие шифрование и целостность данных. Криптография подразумевает преобразование информации в шифр, который не несёт в себе исходного смысла сообщения. Процесс кодирования сообщения называется шифрованием. Обратный метод, подразумевающий расшифровку оригинального сообщения, называется дешифрованием. Традиционная криптография представляет собой область симметричных криптосистем, в которых шифрование и расшифрование проводится с использованием одного и того же секретного ключа [1].

Квантовое распределение ключа (КРК) — это технология, позволяющая создать у двух удаленных пользователей строку случайных бит, впоследствии используемую в качестве криптографического ключа.

Квантовая криптография — метод защиты коммуникаций, основанный на принципах квантовой физики.

Принципиальное отличие квантового способа заключается в том, что КРК — это первая в истории человечества система распределения ключей, для которой есть строгое математическое доказательство того, что она не взламывается, даже со всеми неограниченными вычислительными мощностями и технологиями, которые не запрещены законом физики. Метод КРК решает одну из фундаментальных задач криптографии — распределение ключей между конечными пользователями по открытым каналам связи.

Схема практической реализации квантовой криптографии показана на рисунке 1 [2].

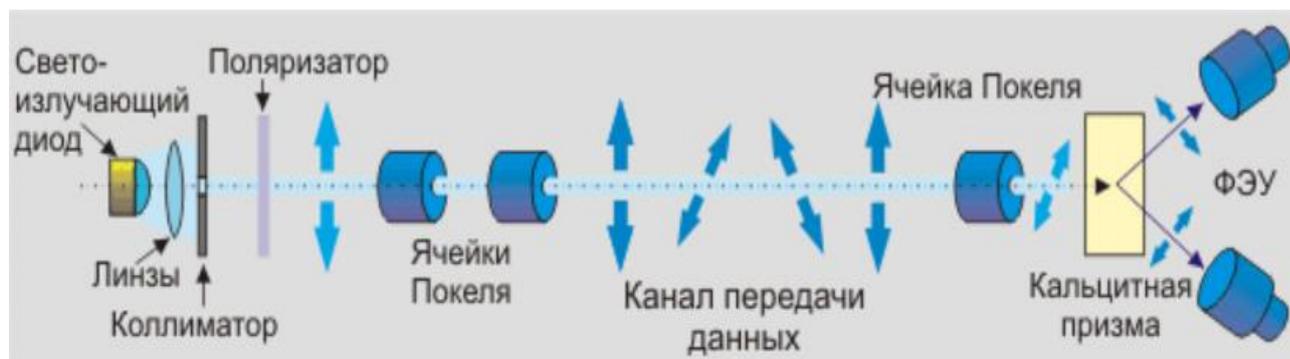


Рисунок 1 – Схема процесса квантовой криптографии

Передача информации на схеме происходит слева на право. Ячейки Покеля предназначены для импульсной вариации поляризации потока квантов передатчиком и для анализа импульсов поляризации приемником. Передатчик отправляет одну из двух поляризаций, каждая из которых имеет по 2 взаимно перпендикулярных направления. Передаваемые данные приходят в виде управляющих сигналов на принимающие ячейки Покеля. Одной из альтернативных сред передачи данных может являться оптическое волокно. В качестве источника сигнала вместо светоизлучающего диода можно использовать лазер.

На стороне приемника после ячейки Покеля установлена кальцитовая призма, которая разделяет пучок на два фотодетектора (ФЭУ), измеряющие две ортогональные составляющие поляризации.

При формировании передаваемых импульсов квантов иногда возникает проблема их интенсивности, что требует решения. Например, чем больше в импульсе содержится квантов — тем выше вероятность того, что злоумышленник сможет перехватить всю необходимую информацию в

ходе передачи. Поэтому идеальным вариантом является ситуация, когда в импульсе количество квантов стремится к одному. В таком случае, при перехвате — состояние системы изменится, провоцируя возникновение ошибок у получателя.

Целью данной работы является анализ новых достижений в области квантовой криптографии с точки зрения технологического прогресса и перспективы использования данного метода.

Одной из последних разработок в области квантовой криптографии является модификация системы квантового шифрования с компактным детектором [3].

Данная система КРК с компактным когерентным детектором работает на основе протокола квантового распределения ключа на боковых частотах, разработка ведется уже последний несколько лет. Главной особенностью данного протокола является образование состояний при помощи электрооптического фазового модулятора.

Ключевым достижением этой системы является разработка когерентного приема. В его основе лежит использование несущей оптической частоты, которая не несет в себе информации, но которую используют в качестве опорной для фиксации фазы слабого сигнала с боковых частот.

В данной разработке используется дополнительный модулятор, но с повышенным уровнем модуляции, затем требуется с его помощью преобразовать сигнал. Таким способом образуются дополнительные боковые частоты, которые взаимодействуют с теми, что пришли от передатчика. При конструктивной интерференции большая часть излучения окажется на боковых частотах и, наоборот, при деструктивной, большая часть излучения окажется на центральной частоте, что может быть зарегистрировано балансным детектором.

Данное исследование поможет удешевить стоимость оборудования, используемого для создания массовых сетей КРК, что сделает их доступными для более широкого круга пользователей. В дополнение эта разработка позволит использовать для КРК инфраструктуру обычных оптоволоконных линий связи.

В начале 2021 года учеными был предложен новый, более устойчивый к внешним воздействиям алгоритм коррекции ошибок с использованием так называемых полярных кодов [4]. Полярный код — линейный корректирующий код, основанный на явлении поляризации канала. Применение полярных кодов позволяет устройствам для квантового распределения ключей стабильно работать в условиях реальной жизни под воздействием различных факторов окружающей среды. Использование полярных кодов позволит сделать систему устойчивой к влиянию ветра, осадков, повышения или снижения температуры.

Еще одним достижением является сокращение доли ключа, необходимой для аутентификации данных, передаваемых во время процесса классической постобработки. В системах квантовой криптографии часть ключа, который был распределен ранее, используется для аутентификации классических сообщений в последующих раундах генерации ключей. Чем этот расход меньше, тем эффективнее работает система в целом. В данном реализации протокола аутентификации доля ключа, затрачиваемая на подтверждение подлинности трафика, значительно уменьшилась и может составлять менее доли процента.

Результаты этих исследований увеличат эффективность работы существующих систем КРК. Значительную роль в эффективности квантовых криптографических систем играет классическая постобработка — набор манипуляций над данными, направленный на выявление и исправление ошибок в квантовых ключах, а также анализ и исключение из него информации, которая могла быть перехвачена злоумышленником.

Описанные достижения окажут значительное влияние на будущее развитие технологии квантовой криптографии. Увеличение эффективности и скорости работы устройств для квантового распределения ключей ускорит массовое внедрение и расширит сферы применения этой технологии в будущем, а модификация алгоритма корреляции ошибок повысит устойчивость данной системы.

На данный момент одной из ключевых проблем повсеместного внедрения квантовой криптографии является высокая стоимость оборудования, но эта проблема решается путём альтернативного использования оптоволоконных сетей, как среды передачи и различных разработок, направленных на удешевление детекторов.

Список использованных источников:

Бабаш А.В. Криптографические методы защиты информации [Электронный ресурс]. – Режим доступа: <https://dlib.rsl.ru/02000022576> – Дата доступа: 23.03.2021.

Реализация идеи квантовой криптографии [Электронный ресурс]. – Режим доступа: [https://www.tadviser.ru/index.php/Статья:Квантовая_криптография_\(шифрование\)](https://www.tadviser.ru/index.php/Статья:Квантовая_криптография_(шифрование)) – Дата доступа: 23.03.2021.

ИТМО: Приемник для передачи ключей квантового шифрования [Электронный ресурс]. – Режим доступа: <https://www.tadviser.ru/index.php/Продукт:ИТМО: Приемник для передачи ключей квантового шифрования> – Дата доступа: 23.03.2021.

Новый мировой рекорд в области квантовой криптографии [Электронный ресурс]. – Режим доступа: <https://misis.ru/science/achievements/2021-02/7242/> – Дата доступа: 23.03.2021.

ГОЛОГРАММЫ В СОВРЕМЕННОМ МИРЕ

Погорельцев Г.И.

Пырякова М.В.

*Белорусский государственный университет информатики и радиоэлектроники
г.Минск, Республика Беларусь*

Савилова Ю.И – канд. тех. наук, доцент

В данной работе рассмотрена перспектива использования голографии, в современном мире объемных изображений предметов на фотопластинке без использования настоящих ресурсов.

Голограмма — продукт голографии, объемное изображение, создаваемое с помощью лазера, воспроизводящего изображение трехмерного объекта. Голографии прочат будущее визуальных развлечений, поскольку до сегодняшнего дня этот способ был самым многообещающим способом визуализации трехмерных сцен. За изобретение метода голографии в 1947 году Дэннис Габор получил Нобелевскую премию по физике в 1971 году. Все просто: вы буквально видите реальный объект, который на самом деле является объемной картинкой. Его можно обойти, рассмотреть со всех сторон, можно придать мощную глубину, которой не может похвастать никакая другая технология 3D-отображения. [1].

Голограмма – это нечто среднее между тем, что происходит, когда вы фотографируете, и тем, что происходит, когда вы смотрите на что-то реально. Как и фотография, голограмма – это постоянная запись отраженного от объекта света. Но голограмма также выглядит реальной и трехмерной и движется, когда вы смотрите вокруг нее точно так же, как реальный объект. Это происходит из-за уникального способа, которым создаются голограммы. [2]

Свет — это волновой процесс. И Габор заметил, что кроме амплитуды на фотопластинке можно регистрировать еще и фазу света. Чтобы записать голограмму, нужно разделить пучок света на две части, предметную и опорную волну, и зафиксировать их интерференцию. Но это было трудно сделать до того, как появился хороший источник света — лазер. Он обладает свойством когерентности, то есть предсказуемостью не только амплитуды, но и фазы. Когерентность лазера позволила записывать голограммы на обычных фотографических материалах.

В 1962 году американские физики Эмметт Лейт и Юрис Упатниекс предложили использовать для записи голограмм лазер, луч которого делится на две части и с помощью зеркал направляется на объект и регистрирующую среду. И объектная, и опорная волна попадает на фотопластинку с одной стороны — так формируется пропускающая голограмма. Чтобы ее восстановить, нужен источник света с той же длиной волны, на которой производилась запись. Чаще всего для этого используют луч того же лазера.

Советский физик Юрий Денисюк примерно в то же время предложил записывать отражательные голограммы в трехмерной регистрирующей среде. При этом луч лазера расширяется линзой и направляется на фотопластинку, за которой стоит регистрируемый объект. Часть луча, проходя через специальную прозрачную фотопластинку, формирует опорную волну, а отраженный от предмета свет формирует объектную волну. Так записывается отражательная голограмма, или голограмма Денисюка, ее видно при свете солнца или лампы, потому что в голограмму как бы встроен селектор длин волн, он выбирает из всего спектра именно ту длину волны, которая была использована при записи.

Разновидность пропускающих голограмм, изобретенных американцем Стивеном Бентоном, называют радужными, потому что трехмерные изображения на таких голограммах наблюдаются на фоне радуги. Они ориентированы на массовое производство. Их используют в рекламе, из них делают акцизные марки, пломбы сохранности. Вообще радужные голограммы используют для защиты от подделок, потому что мошенники пока плохо освоили метод голографии (в отличие от полиграфического способа нанесения рисунков).

Голографию применяют в спектроскопии — это область исследования, которая регистрирует спектры электромагнитного излучения различных веществ. Так определяют вещества, изучают их температуру и плотность. Один из основных инструментов спектроскопии — дифракционная решетка, поверхность, на которую нанесено большое количество щелей или выступов. Их обычно нарезают на специальных машинах механически, алмазным резцом. Но сейчас их записывают голографическим способом. Если задать определенный угол схождения опорной и предметной волны от точечных источников, а в месте пересечения поставить регистрирующую среду — запишется голограмма, а после проявления образуется рельеф. Это и есть дифракционная решетка, полученная при помощи лазера, а не нарезанная механически.

Важное применение голографии — уплотнение информации для волоконных линий связи. Это позволяет пропускать больше каналов связи по волокну, что увеличивает скорость передачи данных.

Конечно, голография — это еще и фундаментальный метод работы с информацией, при помощи которого можно записывать, передавать, воспроизводить многомерные массивы данных. Но часто слово «голография» используется не, по существу, а как мода, как приманка для всякого рода манипуляций с сознанием человека. [3]

Исследования в области голографических технологий и интерфейсов в последнее время становятся все более популярными в научной среде. Несмотря на то, что полноценное внедрение голограмм в нашу жизнь произойдет не скоро, практически не остается сомнений в том, что за этими технологиями будущее. Но вполне возможно, что будущее — это ближе, чем кажется, ведь недавно специалисты компании KT Corporation (одного из крупнейших южнокорейских операторов мобильной связи) совместно с коллегами из Verizon провели первый в истории сеанс голографической связи, используя для этого возможности мобильных сетей стандарта 5G.

Абоненты, участвующие в столь историческом событии, находились в разных частях света: в США и Южной Корее. Для проведения сеанса связи было использовано особое голографическое оборудование и специальное программное обеспечение для обработки огромного количества информации в режиме реального времени. Характеристики устройства и ПО по понятным причинам держатся в секрете. Во время «голографического звонка» присутствовали Хуанг Чанг-Кю, руководитель корпорации KT, и Лоуэлл МакАдам, президент компании Verizon.

Интересным является еще и то, что голографическое устройство, обеспечивающее связь, было подключено к не самому мощному планшету, самой мощной частью которого был модуль связи 5G. Помимо всего прочего, это служит доказательством тому, что для работы в 5G-сетях не требуются высококачественные и высокопроизводительные устройства, при этом скорость, которую можно получить при помощи стандарта связи 5 поколения, в 100 раз превышает нынешние скорости, которыми нас обеспечивает LTE. [4]

Head-up-дисплеи в то время новинкой не были. Например, немецкая компания Continental – мировой лидер в их производстве – устанавливала HUD в автомобили BMW, Audi и Mercedes с 2003 года. Традиционные устройства отображения информации на лобовом стекле – очень сложные приборы с изогнутыми зеркалами и сферической оптикой. И что критически важно, требующие большого объема, примерно 18 л – полтора обычных ведра! А ведь разместить эти полтора ведра нужно в районе рулевого колеса – одной из самых значимых точек автомобиля. Поэтому HUD оснащаются большие дорогие авто, которые изначально спроектированы с местом под дисплей. Неудивительно, что за установку проекционного дисплея в дилерских центрах немецких авто марок с вас попросят не меньше 100 000 рублей. Ну а на обычных машинах классический HUD не увидишь. [5]

Голографические методы используются в настоящее время как для фиксации, так и для исследования криминалистических объектов. Голография совершеннее фотосъемки; она позволяет получить более полную информацию об объекте, ибо представляет собой процесс регистрации на светочувствительном слое не только амплитудных (как в фотографии), но и фазовых характеристик светового потока.

Известно, что все освещенные объекты поглощают, отражают и рассеивают свет. Формирующееся при этом световое поле содержит полную информацию об объектах, их форме, взаимном расположении и даже материале, из которого они состоят. При осмотре объекта именно на это реагирует глаз наблюдателя. Полученную информацию анализирует мозг, в результате человек видит.

В каждом из направлений перемещения зрачков наблюдателя структура светового поля, формируемого объектом, несколько отличается от соседнего. Поэтому смена ракурса осмотра изменяет взаимное положение объектов. Следовательно, для наиболее полной регистрации нужно фиксировать не изображение объекта, а формируемое им световое поле. Научившись регистрировать это поле, а затем восстанавливать его, можно "увидеть" образ объекта таким же, каким он был в момент фиксации. [6]

Список использованных источников:

Что такое голограмма и как ее сделать? [Электронный ресурс]. – Режим доступа: <https://hi-news.ru/research-development/chto-takoe-gologramma-i-kak-ee-sdelat.html> – Дата доступа: 11.04.2021.

Голограмма [Электронный ресурс]. – Режим доступа: <https://hi-news.ru/tag/gologramma> – Дата доступа: 11.04.2021.

Голография: история и применение [Электронный ресурс]. – Режим доступа: <https://postnauka.ru/faq/82396> – Дата доступа: 11.04.2021.

KT и Verizon провели первый в истории сеанс голографической связи [Электронный ресурс]. – Режим доступа: <https://hi-news.ru/technology/kt-i-verizon-proveli-pervyj-v-istorii-seans-golograficheskoy-svyazi.html> – Дата доступа: 11.04.2021.

Полярная механика [Электронный ресурс]. – Режим доступа: <https://www.popmech.ru/vehicles/378252-golografiya-na-lobovom-stekle-kogda-nam-zhdat-styokol-displeev/> – Дата доступа: 13.04.2021.

Голография в криминалистике [Электронный ресурс]. – Режим доступа: <https://alldetectives.ru/crime/kriminalisticheskaya-fotografiya-videozapis-i-golografiya-golografiya-v-kriminalistike.html> – Дата доступа: 13.04.2021.

МАШИННОЕ ОБУЧЕНИЕ В ФИЗИКЕ

Арцыменя Р.А., Данилова А.В.

Белорусский государственный университет информатики и радиоэлектроники
г.Минск, Республика Беларусь

Савилова Ю.И – канд. тех. наук, доцент

В данной работе рассмотрена перспектива использования машинного обучения, как метода облегчения работы современных физиков и помощи им в совершении открытий.

Машинное обучение — класс методов искусственного интеллекта, характерной чертой которых является не прямое решение задачи, а обучение за счёт применения решений множества сходных задач. [1].

Основным современным типом машинного обучения является тип машинного обучения основанного на нейронных сетях

Нейронная сеть — математическая модель, а также её программное или аппаратное воплощение, построенная по принципу организации и функционирования биологических нейронных сетей — сетей нервных клеток живого организма. [2].

Применение методов машинного обучения к изучению квантовых систем (иногда называемое квантовым машинным обучением) является основной темой новых научных исследований в области физики.

Квантовое машинное обучение — раздел науки на стыке квантовой физики и информатики, в котором разрабатываются и изучаются методы машинного обучения, способные эффективно задействовать параллелизм квантовых компьютеров. [3].

Существует несколько основных задач нуждающихся в решении с помощью машинного обучения. Методы машинного обучения можно применять для задач классификации, чтобы распределять события, происходящие в рамках эксперимента, по категориям. В физике высоких энергий возникает задача трекинга (отслеживания), когда при помощи показаний детектора нужно понять, где конкретно пролетел объект и что это была за частица. Задачи детектора аномалий: например, они возникают при поиске неполадок в оборудовании, которым пользуются физики. Оптимизация процессов. Специалистам хочется обеспечить наивысшую эффективность экспериментов при наименьшей стоимости: этого можно достичь, заранее оптимизировав различные параметры экспериментов.

Эксперименты на Большом адронном коллайдере (Большой адронный коллайдер, сокращенно БАК — ускоритель заряженных частиц на встречных пучках, предназначенный для разгона протонов и тяжёлых ионов (ионов свинца) и изучения продуктов их соударений) каждую секунду выдают порядка миллиона гигабайт данных. Даже после уменьшения и сжатия, данные, полученные на БАК всего за час, по объёму оказываются сравнимыми с данными, полученными Facebook за целый год. К счастью, специалистам по физике частиц не приходится разбираться с этими данными вручную. Они работают совместно с разновидностью искусственного интеллекта, обучающегося вести самостоятельный анализ данных при помощи технологии машинного обучения.

Необходимость применения квантового машинного обучения обусловлена тем, что развитие обычных процессоров достигло своего критического максимума. Закон Мура перестал работать. (Закон Мура — эмпирическое наблюдение, изначально сделанное Гордоном Муром, согласно которому количество транзисторов, размещаемых на кристалле интегральной схемы, удваивается каждые 24 месяца. Часто цитируемый интервал в 18 месяцев связан с прогнозом Давида Хауса из Intel, по мнению которого, производительность процессоров должна удваиваться каждые 18 месяцев из-за сочетания роста количества транзисторов и увеличения тактовых частот процессоров) Один современный транзистор может состоять из нескольких атомов. Дальнейшее деление возможно, но вряд ли транзисторы такого размера будут работать стабильно. Кроме того, свои ограничения накладывает скорость света — быстрее нее электроны перемещаться уже не смогут. Компьютерный мир ждет новых технологий, и квантовые вычисления могут обеспечить тот самый необходимый скачок.

Машинное обучение это инновационная технология и у неё огромный потенциал как в физики так и в других сферах жизни человека. Множество исследований ведутся прямо сейчас и в будущем мы ждем новых открытий в данной области.

Список использованных источников:

1. Флах П. Машинное обучение. — М.: ДМК Пресс, 2015. — 400 с..
2. Хайкин С. Нейронные сети: полный курс = Neural Networks: A Comprehensive Foundation. 2-е изд. — М.: Вильямс, 2006. — 1104 с.
3. Srinivasan Arunachalam, Ronald de Wolf. A Survey of Quantum Learning Theory — 2017

СЕКЦИЯ «ФИЛОСОФИЯ, ЛОГИКА И ЖИЗНЬ»

ВООБРАЖАЕМАЯ БИОЛОГИЯ: АНАТОМИЯ РУСАЛКИ

Алдошкина Г.С., Герман Е.Ю.

*Белорусский государственный университет информатики и радиоэлектроники
г. Минск, Республика Беларусь*

Бархатков А.И. – канд. филос. наук

Авторы рассматривают гипотетические особенности предполагаемого приспособления мифических существ, а именно русалок, к водному образу жизни по аналогии с рыбами и водными млекопитающими.

С самого детства все мы слышим многочисленные рассказы и истории про мифических и волшебных существ. Эти рассказы помогают нам отвлечься от наших проблем, избежать скучной порою реальности, а также углубиться в изучение культуры. Однако, посещая уроки биологии и лекции по логике, мы понимаем, что между традиционным представлением мифических существ и законами природы существуют противоречия. Даже фантастические существа должны подчиняться законам природы.

Вне научного сообщества часто ведутся дискуссии на тему того, как должны выглядеть мифические существа. Например, дискуссии в Твиттере по поводу настоящего цвета кожи Ариэль, русалочки из произведения Ганса Христиана Андерсена. Мы также заинтересовались этим вопросом, но решили пойти даже дальше: целью данной работы является рассмотрение не только особенностей внешнего, но и внутреннего строения русалок.

Связь русалок и человека неоспорима. Основой для западноевропейских легенд о русалках послужила история из Ветхого завета об исходе евреев из Египта. Когда Моисей заставил воды Красного моря разойтись, дав им возможность убежать от людей фараона, тех накрыли воды. По повериям же восточных славян, русалками становились утопленники, самоубийцы и другие люди, умершие не своей смертью.

Прежде всего, рассмотрим среду обитания этих существ. Из-за того, что русалки являются метаморфозой человека, множество человеческих черт должно сохраниться. Так как человеческий организм не приспособлен к высокому давлению, русалки не могут жить на большой глубине. Их организм, как и человеческий, не выдерживает слишком низких температур. Оптимальное значение температуры воды для человека – 36–37 градусов. Подобную температуру воды можно найти в области экватора или, например, в Красном море, где, соответственно, и могли бы проживать русалки.

Теперь коснёмся строения русалок, начиная с её скелета. Верхняя часть скелета полностью идентична человеческой, а вот нижняя часть видоизменена. Если смотреть на способы приспособления других млекопитающих, таких как киты и ламантины, то можно заметить некоторые закономерности: слабо развитый или даже атрофированный пояс нижних конечностей, который нужен только для крепления мышц хвоста. Возможно, нижние конечности преобразовались в плавники, повышающие манёвренность русалок в воде.

Наиболее яркой отличительной чертой русалки от человека является её хвост. Он, прежде всего, выполняет функцию руля, особенно при крутых поворотах. Также примерно 40% энергии толчка при плавании в воде русалки получают за счёт хвоста, также как и рыбы.

Особенности мышечной системы аналогичны особенностям скелета: верхняя часть повторяет строение человеческой мышечной системы, а хвост представляет собой многослойное мышечное образование, отвечающее за его гибкость и силу. Хвостовой плавник, вероятно, подобен рыбьему и состоит из мышц и хрящей.

Перейдём к внутреннему строению. Разница с человеческим строением минимальна. Так, пищеварительная, выделительная, нервная, половая и кровеносная системы идентичны человеческим. Интерес представляет дыхательная система.

Изучая дыхательную систему китообразных, мы пришли к выводу, что для успешного существования под водой русалкам необходимо наличие строго отдельной дыхательной системы, то есть пищевод и дыхательные пути должны быть разделены. Вода, попадая в гортань при вдохе, очищается с помощью тычинок в гортани, затем попадает в лёгкие, где капилляры лёгких насыщаются кислородом, растворённым в воде. После этого на выдохе вода выталкивается через жаберные щели на рёбрах. Из-за изменения дыхательной системы русалки лишаются возможности говорить. Плавательный пузырь отсутствует, в связи с тем, что русалка не изменяет глубину погружения, всегда находясь либо на поверхности, либо в прибрежной зоне.

Обратим внимание на кожу русалок. Для уменьшения трения и поддержания постоянной температуры тела, кожа русалок должна быть покрыта слизью, выделяемой более развитыми сальными железами, и напоминать дельфинью. Так как главная задача русалки – уменьшить сопротивление воды, скорее всего она будет полностью безволосой, включая голову. Ногтевые пластины сильно развиты для охоты на рыб и поиска моллюсков на дне. Вдоль туловища появится боковая линия, такая же как и у рыб, помогающая отслеживать движение воды. Хвост русалок покрыт

чешуей, расположенной черепицеобразно. Для маскировки вполне вероятно, что спина русалок будет темнее, живота, для того чтобы быть схожей с окружающей средой. Если смотреть на русалку сверху. То она будет сливаться с темной водой, если снизу- будет сливаться со светлым небом.

Глаза сильно видоизменяются: слезные железы исчезнут, за ненадобностью, появится третье веко для защиты слизистой оболочки глаза, изменится форма хрусталика, чтобы хорошо видеть в воде, потому что степень преломления воды больше, чем в воздухе.

Таким образом, мы видим, что научный взгляд вносит существенные коррективы в восприятие мифических персонажей.

КРОВОТЫЕ НОРЫ: ЧТО ЖДЕТ ЧЕЛОВЕКА ПО ТУ СТОРОНУ ВСЕЛЕННОЙ

Барановская А. О., Боровская Д. Д.

*Белорусский государственный университет информатики и радиоэлектроники
г. Минск, Республика Беларусь*

Бархатков А.И. – канд. филос. наук

Авторы рассматривают феномен кротовых нор, или червоточин, допускаемых Общей теорией относительности, и обсуждают проблему возможности использования их человеком для перемещения во Вселенной.

Феномен кротовых нор, или как их еще называют, червоточин, был описан в еще в 1935 году в рамках общей теории относительности Эйнштейна. Кротовая нора - гипотетическая особенность пространства-времени, которая представляет собой особый «туннель» в пространстве. Эти области могут быть как тесно приближены друг к другу, так и полностью удалены друг от друга, являя собой феномен отдельных пространств. Пространства в космосе, как предполагают астрофизики, связаны между собой кротовыми норами. Если представить кротовую нору визуально, она будет напоминать своего рода тоннель, через который можно минуя огромные расстояния, попадать в иные пространственные области. Общая теория относительности (ОТО) допускает существование кротовых нор, но для того, чтобы пространственный переход мог быть осуществлен, «нора» должна быть заполнена экзотической материей с отрицательной плотностью энергии, в результате чего должно возникнуть сильное гравитационное поле. Если оно не возникнет, кротовая нора исчезнет. Феномен кротовых нор представляет сегодня одну из самых уникальных и актуальных проблем космологии и астрофизики. Он слабо изучен и поэтому является объектом многочисленных гипотетических предположений.

«Кротовые норы» часто становятся идеей, вокруг которой снимаются всевозможные фантастические фильмы. Выбор этой темы совсем не удивляет, ведь они в теории представляют собой туннели в пространстве-времени. Через них можно перемещаться между точками Вселенной, которые в привычном нам смысле сильно удалены друг от друга. Эти структуры согласуются с Общей теорией относительности, но подтвердить их существование до сих пор так и не удалось.

Несмотря на то, что кротовые норы не противоречат Общей теории относительности и предсказываются некоторыми космологическими теориями, реальные кандидаты на роль таких объектов до сих пор неизвестны. Кроме того, кротовые норы подразумевают весьма нетипичную геометрию пространства-времени, для поддержания которой требуется материя с экзотическими свойствами (например, с отрицательной плотностью энергии).

Впрочем, недоказанность наличия кротовых нор в нашей Вселенной не мешает ученым изучать связанные с ними вопросы. Например, сможет ли человек, попавший в такую «червоточину пространства», преодолеть весь путь по ней и выйти живым на другом конце межпространственного тоннеля.

Хотя такие топологические структуры могут быть пригодны для путешествий в отдаленные уголки Вселенной и, возможно, даже в параллельные измерения, есть много других проблем, связанных с перемещением по кротовым дырам. Так, например, при попадании в тоннель фотонов реликтового излучения их энергия может возрастать в тысячи и десятки тысяч раз, что сделает эти частицы опасными для человека. Кроме того, рассеивание энергии таких фотонов может привести к коллапсу всей структуры.

Основная сложность, с которой сталкиваются ученые при работе с червоточинами – их чрезвычайная нестабильность. Если представить, что ученые нашли червоточину и решили послать через нее один фотон света, энергии этого фотона будет достаточно, чтобы полностью уничтожить эту самую червоточину. Это говорит о том, что даже если удастся создать кротовую нору, она разрушится прежде, чем вы успеете ею воспользоваться.

Для решения этой проблемы необходимо противодействовать положительной энергии или массе, которая проходит через червоточину. Понадобится отрицательная масса или энергия. Однако, отрицательная масса в нашей вселенной – не такое уж и фантастическое явление. Однако, и здесь есть проблема. Неизвестно, возможно ли сгенерировать достаточное количество такой энергии, чтобы стабилизировать червоточину.

Если человек попытается войти в кротовую нору, его, скорее всего, разорвет на куски и разбросает по всей вселенной. Тоже результат, ведь он в итоге попадет в другую часть этой самой вселенной, правда, наверняка не так, как изначально задумывал.

Список использованных источников:

1. Хокинз С. Теория всего / С.Хокинз; пер. с англ. Е.В.Шимановской. – Москва: АСТ, 2019. – 160 с.
2. Рубин С.Г. Устройство нашей Вселенной / С.Рубин. – Фрязино: Век 2, 2006. – 312 с.

ЛОГИКА ВНУТРИ КВАНТОВОГО КОМПЬЮТЕРА

Василевич В.В., Хорошун Д.А.

*Белорусский государственный университет информатики и радиоэлектроники,
г. Минск, Республика Беларусь*

Научный руководитель: Мащитко М.С. – канд. философских наук, доцент

Аннотация. Данная работа содержит в себе попытку разобраться с логическим наполнением квантовых компьютеров и их основными чертами отличия от классических. Таковыми мы посчитали принцип работы, ход вычислений, логические операнды и возможность изучить подробно ход внутренних процессов. В ходе изучения материалов нами было отмечено, что к квантовым компьютерам неприменим принцип детерминированности, бинарная логика, и результаты их вычислений дают лишь некоторую, пусть и довольно большую, вероятность правильного ответа.

На данном этапе развития потенциал современных кремниевых компьютерных технологий практически исчерпан: еще большее уменьшение компонентов устройств приведет к тому, что вещество начнет проявлять волновые свойства, а также в нем начнет действовать принцип неопределенности Гейзенберга. Таким образом перед учеными становится вопрос, куда же двигаться дальше. Одним из возможных направлений и путей дальнейшего развития являются квантовые технологии. Уже сейчас множество лабораторий и компаний задействовано в разработке и модифицировании квантовых компьютеров, а также в написании алгоритмов для них. Ввиду того, что логика квантового компьютера существенно отличается от логики классических вычислительных машин, которые, скорее всего, в скором времени войдут в наши жизни возникает необходимость в понимании тонкостей этого вопроса. В этой работе мы бы хотели разобраться, в чем заключаются основные отличия логики квантового компьютера от логики классического.

Основной принцип работы квантового компьютера хорошо известен и базируется на возможности квантового элемента находиться в двух состояниях одновременно (так называемый принцип суперпозиции), что по сути исключает идемпотентность выводов. Чтобы объяснить это состояние, обратимся к автору одного из самых известных квантовых алгоритмов. Приведем здесь прямую цитату Шора: «Если измерить состояния машины (по отношению к данному (гильбертову. – В.В.) базису) на любом конкретном шаге вычислений, то вероятность обнаружить систему в базисном состоянии $|S_i\rangle$ будет $|a_i|^2$; однако измерение состояния машины спроектирует ее в наблюдаемый базисный вектор $|S_i\rangle$. Поэтому наблюдать состояние машины можно только в конце вычислений». Соблюдение унитарности вычислительного процесса конкретно означает сохранение суммы

$$\sum_i |a_i| = 1$$

на каждом шаге вычислительного процесса.

Главным отличием между квантовым и обычным компьютером (имеется в виду любая современная вычислительная техника, в том числе и суперкомпьютеры) мы считаем именно принципы их работы. Для всех современных компьютеров справедливо утверждение, что если задать некое начальное состояние системы и пропустить через заданный алгоритм, то результат вычислений будет повторяться из раза в раз, сколько бы мы не перезапускали данную программу или последовательность действий. Данный принцип работы называется детерминированным. Определенная последовательность шагов ведет к определенному результату. Этот принцип неприменим к квантовым компьютерам, так как они работают на принципе вероятностей. Это

означает, что в результате выполнения заданного алгоритма с определенным начальным состоянием мы получим выборку из вероятностного распределения конечных реализаций алгоритма не исключая вероятные ошибки. Чтобы увеличить вероятность правильного ответа, а в идеале и максимально приблизить результат к единице необходимо многократно повторить заданный алгоритм для нашего начального состояния. Говоря о стремлении результата к единице мы имеем в виду статистический анализ разброса N результатов, в котором один из этих результатов стремится к единице, в то время, как остальные - к бесконечно малым величинам.

Если мы говорим о процессе вычислений внутри квантового компьютера, то для нее справедлива следующая последовательность шагов: 1) установка некоторого необходимого для наших целей начального состояния кубитов; 2) изменение заданного состояния путем некоторых преобразований по сути соответствующим логическим операциям (или так называемым гейтам); 3) измерения конечного состояния системы, т.е. результатов работы компьютера. Одной из существенных отличительных черт работы квантового компьютера является также то, что мы можем измерить и проверить лишь результат работы и никогда не можем рассмотреть или отследить процессы внутри алгоритма, так как в процессе его выполнения вся система находится в состоянии суперпозиции.

Если в классическом компьютере используются логические вентили-базовые элементы цифровых схем, которые выполняют элементарные логические операции (к примеру, «И», «ИЛИ», «исключающее ИЛИ», побитовые сдвиги, побитовое отрицание), задача которых состоит в простом преобразовании входных логических сигналов в выходные, то в квантовом компьютере квантовые вентили преобразуют сигналы из входных в выходные по определенным законам. Простейшими однокубитными вентилями являются тождественное преобразование, отрицание, фазовый сдвиг. Стоит отметить, что квантовые логические операции не всегда имеют тот же смысл, что и классические, несмотря на схожие названия.

Так как бинарная логика и алгоритмы основанные на ней неприменимы к квантовым компьютерам, использующим квантовые вентили, для них создаются, основанные на квантовых преобразованиях, особые алгоритмы. Разработкой таких занимается множество ученых и список квантовых алгоритмов постоянно пополняется. Все они пишутся на базе классических компьютеров, имитирующих логику работы квантовых со связностью всех со всеми, после чего они перекомпилируются, чтобы отвечать особенностям архитектуры конкретных процессоров. Это делается для того, чтобы к моменту появления пригодного к промышленной эксплуатации универсального квантового компьютера максимально упростить написание программ для него.

Логика квантового компьютера разительно отличается от логики классического. Она по сути является небинарной, для нее нехарактерен принцип детерминированности. Работа с квантовыми однокубитными и многокубитными логическими операндами, как и разработка цельных квантовых алгоритмов требует значительной теоретической подготовки.

Список литературы.

1. ИКС Медиа, ИКС №2 2020, [Электронный журнал] статья «На пороге посткремния» Н.Носов от 05. Июля 2020 г. Режим доступа: <https://www.iksmedia.ru/articles/5684388-Na-poroqe-postkremniya.html>
2. Шор П. Полиномиальные по времени алгоритмы разложения числа на простые сомножители и нахождения дискретного логарифма для квантового компьютера // Квантовый компьютер и квантовые вычисления. Том II Ижевск, 1999. С. 208.
3. Электронный философский журнал Vox. Выпуск 20 (декабрь 2016), статья «Понятие квантовой логики: математические и философские аспекты» Антипенко Л.Г., Институт философии РАН. Режим доступа: <http://vox-journal.org>
4. National Academies of Sciences, Engineering, and Medicine 2018. Quantum Computing: Progress and Prospects. Washington, DC: The National Academies Press. P.3-3.1.5

ВЛИЯНИЕ МУЛЬТФИЛЬМОВ НА ПСИХИКУ ДЕТЕЙ

Кибитлевская А.Р., Тяхай Е.В.

*Белорусский государственный университет информатики и радиоэлектроники
г. Минск, Республика Беларусь*

Бархатков А.И. – канд. филос. наук

В работе рассматривается проблема влияния мультфильмов на психику детей с точки зрения формирования у них представлений об окружающем мире и воспитания определенных жизненных правил и нравственных ценностей. Рассматриваются примеры мультфильмов, способных оказать положительное и отрицательное влияние на психику детей различного возраста.

Психика ребенка на ранних этапах жизни очень хрупкая, он как пустой сосуд, куда информация из внешнего мира поступает, обрабатывается, и создаются готовые модели поведения для разных жизненных ситуаций.

Малыш впитывает все, как губка, и порой ему бывает очень сложно разобраться с такими сложными понятиями как «добро» и «зло», «смелость» и «милосердие». Он пытается повторить все, что видит и слышит, так он учится, познает мир и себя.

Основная цель качественного мультфильма — воспитание и обучение. Если мультфильм соответствует своей цели, обучение и воспитание, то видя, как главный герой на протяжении фильма обучается, меняется в лучшую сторону, противостоит множеству испытаний, маленький человек делает тоже самое, учится и развивается.

Примерами хороших мультфильмов может послужить «Винни-Пух», который учит детей не унывать, вести активный образ жизни и даже маленьким аспектам этикета, «В тридевятом царстве» — настоящее пособие, каким быть в этом мире не стоит, «Простоквашино» показывает реалии жизни, необходимость заботиться о близких, хозяйстве, «Бэмби», «Золушка», «Синдбад», «Король Лев», «Гуффи», «Чип и Дейл» также прививают определенные моральные ценности.

Однако есть и плохие мультики. Плохим следует считать мультфильм, похожий на рекламный ролик: яркий, цепляющий глаз, но не запоминающийся, а скорее въедливый. Создатели некоторых фильмов и мультсериалов имеют цель — не обучение и воспитание, а прибыль. Для увеличения количества просмотров и популяризации сопутствующих товаров (игрушки, майки, рюкзаки и т.д.) авторы используют неравномерный звуковой ряд, чрезмерно яркую картинку, частую смену ракурса героя, множество мелких бросающихся в глаза деталей, которые часто меняются и т.п.

Примерами плохих мультфильмов служат такие мультфильмы как «Футурама», «Симпсоны», «Южный парк». Эта анимация пропагандирует насилие, жестокость, человеческую глупость, наивность. Зачастую это делается сатирично, однако в раннем возрасте малыш этого не поймет, будет воспринимать все за чистую монету. Еще одним наглядным примером является мультфильм «Маша и медведь».

Главная героиня очень нравится детям, вызывает восторг своим неоднозначным поведением. Многие ее поступки откровенно шокируют взрослых, заставляя жалеть медведя. Но эти чувства возникают только у взрослых, дети еще не способны полностью осознать последствия большинства действий героини и с радостью за ней повторяют. Да, герои мультфильма не говорят о запрещенных вещах, не проявляют сильную агрессию и не ругаться матом, но вполне показывают поведение, не соответствующее общепринятым моральным установкам.

Выбирая мультфильм, следует понимать, что он может оказать как хорошее влияние, так и плохое.

Мультфильм может оказать положительный эффект, если подходит малышу по возрасту, отражает его умственные потребности, не содержит технологий, препятствующих правильному восприятию. Когда ребенок смотрит телевизор, его мозг только находится в активной стадии формирования, развития, роста, это означает, что часто дети еще не способны полностью осознать последствия большинства действий героев и с радостью за ними повторяют. Очень удобно, ребенок моментально замолкает, успокаивается, не требует никакого внимания к себе. Как бы находится в безопасности, сидит и смотрит. Такая физическая безопасность таит в себе психологическую угрозу. Ребенку необходимо очень много двигаться и изучать окружающий мир в реальности. Знания, приходящие с экрана — хорошо, но они не могут заменить опыт, полученный от взаимодействия с настоящими предметами, людьми.

Для того чтобы узнать сходятся ли представления психологов и обычных людей о плохих и хороших мультиках, мы решили провести опрос среди студентов нашего вуза. В опросе приняло участие 50 студентов. По результатам можно сделать вывод, что хорошими мультфильмами, по мнению опрошенных, являются «Винни-пух», «Простоквашино», «Смешарики», «Золушка», «Король Лев», «Чип и Дейл». Противоположное впечатление же произвели «Южный парк», «Мимимишки», «Барбоскины».

Много споров у психологов вызвал мультфильм «Маша и Медведь». На вопрос о том, почему он хороший, давались следующие ответы: не знаю, эксперты считают плохим, но, когда он вышел, я была уже в таком возрасте, что сама понимала модель поведения Маши – проказничество; смешно, интересно, жизненно, песни; обучающий и поучительный в какой-то степени; легкий, сочетание двух совершенно разных персонажей даёт крутой контраст, интересно наблюдать за их отношениями.

Исходя из этого, можно подвести итог: для каждого возраста есть свой хороший мультфильм. Каждый человек и ребенок замечает и воспринимает в них разные аспекты и мелочи, поэтому нет единого примера хорошего или плохого мультфильма.

СВОБОДА ВОЛИ: РЕАЛЬНОСТЬ, ВОЗМОЖНОСТЬ ИЛИ ИЛЛЮЗИЯ?

Панев А.С.

Белорусский государственный университет информатики и радиоэлектроники
г. Минск, Республика Беларусь

Ратникова И.М. – канд. филос. наук

Статья посвящена обоснованию тезиса об отсутствии свободы воли у Искусственного Интеллекта и человека.

В наши дни Искусственный Интеллект (далее – ИИ) повсюду: алгоритмы YouTube, голосовые помощники, автопилот для автомобилей. С каждым днем сфер применения ИИ становится все больше, но зачастую мы об этом даже не догадываемся. К примеру, цены на недвижимость высчитываются ИИ, а не человеком.

В массовой культуре широко тиражируется позиция, согласно которой допускается возможность появления свободы воли у ИИ. В этом случае он сможет захватить мир и поработить человечество. Данное утверждение представляется весьма сомнительным. ИИ может разрушить мир только в двух случаях: а) если поставлена прямая цель захватить или разрушить мир; б) если поставлена какая-то цель, которая может быть наиболее эффективно достигнута путём разрушения мира.

Последний вариант можно пояснить следующим примером. Программист ставит андроиду цель научиться поливать оранжерею так, чтобы влажность почвы всегда была в допустимых пределах. При этом схема полива должна потреблять минимальное количество электричества. Пусть за показания влажности почвы отвечают датчики в земле. ИИ может догадаться положить датчики в закрытую ёмкость с почвой необходимой влажности. В этом случае, несмотря на выполнение основной цели (влажность в допустимых пределах и малый расход электроэнергии), все растения погибнут. Таким образом, ИИ не сможет нанести вред человеку, если мы будем правильно формулировать цели.

Более вероятный сценарий развития событий начинается уже сейчас. ИИ всё больше и больше облегчает жизнь человеку в самых различных сферах. Так, в будущем появятся андроиды, которых нельзя будет внешне и по интеллекту отличить от человека. Для того чтобы избежать путаницы и различного рода недоразумений, необходимо будет присвоить андроидам отличительный признак (к примеру, горящую лампочку во лбу). При этом надо понимать, что это сходство будет лишь внешним и появление свободы воли у андроидов невозможно.

В то же время для многих людей, не понимающих принцип работы ИИ, использование андроидов в качестве рабочих, возможно, будет выглядеть как рабство. Данное обстоятельство может выступить в качестве основания массовых протестов. Например, уже сейчас людям жалко роботов. В социальных сетях все чаще и чаще можно встретить сообщения подобного плана: "Подарил маме робот-пылесос, но она его практически не использует, потому что ей его жалко". И это при том, что современные роботы-пылесосы пока даже внешне не похожи на живых существ. Зачастую люди склонны сочувствовать неживому объекту, приписывая ему качества живого существа: сознание и способность чувствовать боль и эмоции.

Представьте, что будет, если такие роботы начнут разговаривать и вести себя так же, как и люди. Должны ли мы их жалеть, должны ли мы верить в то, что все сказанные ими слова идут не из алгоритмов нейронных сетей, а из некой души, заточённой в микросхеме? Поверхностный ответ на этот вопрос: то, что ведёт себя так, будто у него есть сознание, обладает им. В то же время даже если андроиды и будут неотличимы от человека ни по внешности, ни по поведению, это всё ещё будет программа в искусственной плоти. Главной целью этой программы будет являться создание иллюзии того, что у андроида есть душа. Если же мы поверим в эту иллюзию, значит, алгоритм справился со своей задачей.

Разберем основные принципы работы ИИ. Есть входные сигналы, внутренние обрабатывающие "нейроны", их чаще всего много слоёв, и выходные сигналы (Рисунок 1). Сигнал передаётся слева направо, проходя через нейроны, каждый из которых получает сигналы и выдаёт результат согласно простейшим математическим формулам. Здесь всё подчинено принципам детерминизма, здесь нет места свободе воли или случайностям. Другими словами, при одинаковых входных данных одна и та же нейронная сеть выдаст всегда один и тот же результат. Человеческий мозг устроен таким же образом.

Ответим на вопрос: есть ли что-либо общее у месячных, опухоли мозга, злоупотребления сладостями и анаболическими стероидами? На первый взгляд, нет. В то же время именно эти факторы успешно использовались в суде для объяснения поведения убийц. Например, в случае со сладостями мужчине неимоверно сильно сократили срок за двойное убийство. Понятие "свобода воли" включает в себя ответственность за совершаемые действия. Получается, суд согласился с тем, что при определенных условиях мы можем частично или полностью терять свободу воли, а наши

поступки являются лишь результатом некоторых биохимических реакций, имевших место в нашем организме. Но кто сказал, что когда состояние мозга, уровень гормонов и сахара в крови в норме, то мы в ответе за свои поступки, а когда уровень сахара превышает допустимые значения, мы резко теряем управление над нашими действиями? Эта утрата контроля не может происходить резко: на самом деле, как представляется, ее вообще не происходит, поскольку свободы воли у нас не было уже изначально. К примеру, возьмём какое-либо простое действие. Пусть человек пытается согнуть палец. Мозг посылает сигнал, командуя мышцам сокращаться. Абсолютно незначительное, в целом безобидное действие, какие тут вообще могут быть последствия? Весьма серьезные, если в руках у человека пистолет, то есть действие всё-таки не такое уж безобидное.

Нейробиология позволяет найти конкретный нейрон в моторной коре, который отправил сигнал и заставил палец согнуться (Рисунок 2). Можно также найти нейроны в так называемой премоторной коре, которые послали сигнал в моторную, откуда нейрон послал сигнал в руку. Эти нейроны, в свою очередь, получили сигнал от фронтальной коры, а она от префронтальной, а та в свою очередь от участков мозга, которые отвечают за эмоции [1].

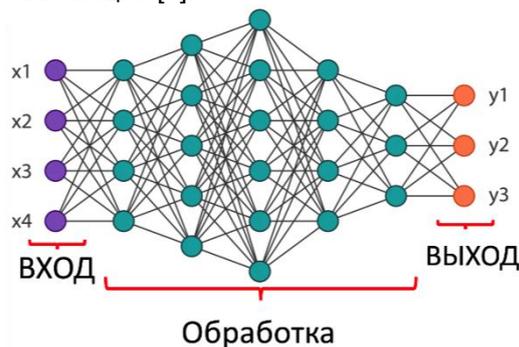


Рисунок 1 – Нейронная сеть

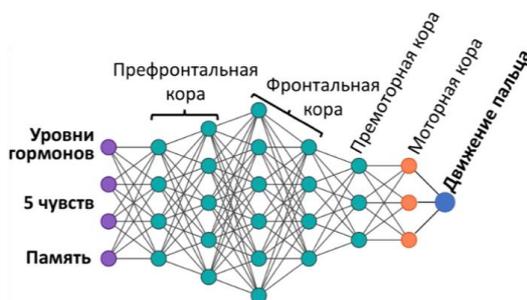


Рисунок 2 – Нейроны в мозге человека.

Зададимся весьма правомерным вопросом: существует ли нейрон, который стоит у истоков, который сработал бы сам, есть ли точка, где потенциал действия возник из ниоткуда, без внешних воздействий, или нейрон, который вдруг решил нарушить все известные физические законы, внезапно посылая сигнал без предшествующей этому причины? Пусть нам покажут нейрон, который способен на такое, и у нас появилась бы нейробиологическая основа свободы воли. Но пока этого не случилось. Не бывает причины без причины. Ничего в биологии нашего поведения не происходит просто вдруг. За каждым событием обнаруживается целая история [2].

Любой человек, какие бы у него ни были принципы, сидя за рулём машины, которая вот-вот неизбежно врежется в столб, направит машину левее, чтобы получить меньшие повреждения, чем человек, сидящий справа. Как бы нам ни казалось, что мы что-то решаем, мозг любого из нас при одинаковых параметрах и строении выдаст одно и то же действие. Случись у кого-либо такая же опухоль в отделе мозга, отвечающем за агрессию, как у убийцы, который ещё неделю назад был спокойным семьянином, и он бы тоже совершил преступление.

К настоящему времени было проведено большое количество невероятно важных исследований. Обратимся к одному из них. Так, ученые рассмотрели более 5000 судебных решений, принятых непосредственно на заседании, на предмет того, от чего зависит, отпустят ли заключенного досрочно или нет. Оказалось, что единственным показателем, который способствовал такому прогнозу – это то, сколько часов прошло с тех пор, как судья поел в последний раз. Если дело рассматривали сразу после обеденного перерыва, то шансы на досрочное освобождение составляли приблизительно 60 процентов, если же через два часа, то они стремились к нулю. Данное исследование подтверждено большим количеством фактического материала. Допускаю, что если попросить судью обосновать принятое решение о возможности досрочного освобождения, он вряд ли скажет про уровень сахара в крови, но приведет аргументы из области юридической литературы и практики.

Таким образом, мы принимаем огромное число решений без своего собственного ведома. Современная нейробиология убедительно доказывает, что когда мы совершаем обдуманый, как нам представляется на первый взгляд, выбор, обеспечиваемый наличием у нас свободы воли, оказывается, что на самом деле все уже было решено на несколько секунд раньше, а сознательная, рациональная часть мозга затем просто находит этому разумное объяснение *post factum*. Свободой воли, другими словами, не может обладать ни человек, ни ИИ.

Список использованных источников:

1. Курс "Биология поведения человека" [Электронный ресурс] / Стэнфордский университет. – Режим доступа: <https://youtube.com/playlist?list=PL8YZyma552VcePhq86dEkoHvoTpWPuauk> – Дата доступа: 05.06.2020.
2. Интервью с Робертом Сапольски [Электронный ресурс] / Р.Сапольски. – Режим доступа: <https://youtube.com/playlist?list=PL8YZyma552VcfR6U49lyqClcDfpO65OPp> – Дата доступа: 12.10.2020..

«ЭКСПЕРИМЕНТЫ ЛЕЙН» С ТОЧКИ ЗРЕНИЯ ФИЛОСОФИИ ЭКЗИСТЕНЦИАЛИЗМА

Бокун А.Г.

Белорусский государственный университет информатики и радиоэлектроники
г. Минск, Республика Беларусь

Дисько-Шуман М.Р. – канд. филос. наук

Аннотация. Данная работа содержит анализ произведения массовой культуры жанра аниме «Эксперименты Лейн». Для анализа был использован инструментарий философов-экзистенциалистов XX-го века.

Ключевые слова. Постмодерн, экзистенциализм, Сеть, Бог, существование, деконструкция.

Постмодернистская парадигма подарила европейской культуре массу интересных концепций, которые, хоть и являются по сути своей лишь деконструкцией идей и течений минувших парадигм, все-таки имеют некоторый общий дух, свое неповторимое и несвойственное ни одной из предыдущих парадигм настроение. Философия экзистенциализма XX века ни в коем случае не является исключением. В данной работе под первой волной экзистенциализма я буду подразумевать религиозный экзистенциализм, под второй – постмодернистский экзистенциализм, а сущность третьей волны я постараюсь раскрыть по ходу повествования.

Для отделения третьей волны от всех предыдущих, необходимо понять по какому принципу была отделена первая волна от второй. Собственно, это заметно уже по названиям: Кьеркегор был приверженцем религиозного экзистенциализма, который предполагал познание христианского Бога не посредством разума, тем самым признавая трансцендентную природу Его, а через существование или же экзистенцию. Экзистенциализму человека (= познанию Бога), по мнению Кьеркегора, ни в коем случае нельзя было мешать. Его философия предполагала полную свободу личности и выбора, что выводилось Кьеркегором напрямую из Писания, а также первичность Бога и существования перед сущностью. Данные позиции заложили фундамент для второй волны экзистенциализма, однако, как и многие другие идеи, были подвергнуты деконструкции. Наиболее важная деконструкция произошла в работе Жана-Поля Сартра «Экзистенциализм – это гуманизм». В ней он упразднил Бога как такового в прочтении философии экзистенциализма, сняв максимум Кьеркегора: Бог → Существование → Сущность и перейдя к своему положению: Существование → Сущность.

Весь экзистенциализм XX века строится на принципе первичности существования, что значит полное отрицание детерминизма и взятия человеком ответственности за свои поступки. Тезис религиозного экзистенциализма о полной свободе человека не претерпел серьезных изменений, однако теперь ее обеспечивает не Бог, а сам человек. В связи с этим появляется чувство заброшенности (Бог оставил человека) и отчаяния (человек несет ответственность за себя и свои действия).

В чем же была главная проблема нового экзистенциализма? На мой взгляд, самая важная ошибка Сартра – отрицание факта существования какого-либо детерминизма в принципе. Согласитесь, довольно забавно, что философия, имеющая претензию хоть на что-то, в XX веке не берет в расчет существование естественных наук (например, биологии).

И вот тут мы наконец-то переходим к произведению, упомянутому в названии работы. «Эксперименты Лейн» довольно нетипичное произведение, в нем крайне интересно раскрыта тема слияния разума человека с компьютером.

По началу кажется, что мы наблюдаем за жизнью обычной школьницы, однако вся эта обычность быстро уходит, когда мы осознаем, что в том мире граница между «реальностью» и Сетью

попросту перестала существовать. К этому осознанию нас подводят постепенно, от этого эффект некоего экзистенциалистского опуса становится лишь сильнее. Завязка аниме демонстрирует нам типичную «пограничную ситуацию» по Сартру, Лейн столкнулась со смертью: девочка из школы, в которой она учится, совершила суицид. Далее происходит первое значимое столкновение с Сетью, Ивакура получает сообщение от мертвой девочки, в котором та сообщает, что не погибла, а, используя словесный инструментарий данной работы, изменила способ своего существования, то есть перешла в Сеть. Это происшествие подвигает Лейн начать интересоваться компьютерами, с которыми она, как сообщается зрителю, раньше почти не взаимодействовала.

Вообще в сериале есть несколько таких моментов и чем ближе к финалу, тем понятнее становится, что это вовсе не альтернативные версии, а другие измерения существования оригинала. В этом аспекте заключается первая черта Ивакуры, как идеального с точки зрения экзистенциалистов существа – ее существование многомерно.

Наиболее наглядно это можно объяснить, проведя аналогию с христианским Богом. Он триедин, то есть существует одновременно как Бог-отец, Бог-сын и Святой дух, Лейн существует так же, отличие лишь в том, что Сеть позволила ей быть не триединой, а бесконечно единой, ее есть столько, сколько нужно и во всех ее Я содержится одинаково много ее. Это принципиальный момент, который наследуется экзистенциализмом от философии Кьеркегора.

Пройдя пограничную ситуацию и встречу с собственной экзистенцией, та часть существования Лейн, которую мы видим, как главную героиню аниме, можно сказать забрасывает в новый мир (на самом деле он остается прежним, но новая информация полностью меняет восприятие его Ивакурой) о котором конкретно она, как единица существования всеобщей Лейн, ничего не помнит. Однако помнит она или не помнит – не имеет никакого значения, ведь в игру вступает максима Сартра, по которой существование всеобщей Лейн предопределило сущность той, за которой мы наблюдаем. В итоге она делает, то что делает (вероятно она делала это уже много раз в прошлом), не совсем понимая истинного смысла своих поступков (существование слишком опередило сущность).

В аниме есть еще один довольно значимый персонаж Масами Эйри. Это талантливый сетевой инженер, который добрался до результатов экспериментов по использованию экстрасенсорных способностей детей.

На основе полученных результатов Эйри нашел способ создать цифровую копию сознания человека, которая постоянно синхронизировалась бы с физическим оригиналом с помощью резонанса Шумана. Подключение всех людей к Сети стало основной целью Масами, для этого он заключил сделку с крупной корпорацией, однако из-за разногласий был уволен. После увольнения он совершил суицид и стал, вероятно, первым человеком, уничтожившим оригинал своего сознания, оставив лишь цифровую копию. Эйри в сериале выступает в роли классического персонализированного божества, он постоянно апеллирует к пророчествам, его сторонники образовали хакерскую группу «Рыцари» (аналогия с тамплиерами).

Однако не стоит забывать, что «если бы кошки выдумали бога, то он был бы котом», вот и Эйри несмотря на свое сетевое могущество существует как человек, более того, взаимодействуя с Лейн, направляя ее по тому пути, который предлагало ей пророчество (Лейн должна была стать агентом, который переведет сознание всех людей в Сеть, то есть исполнит цель Эйри), он воспринимает ее как человека, хотя ему известно о ее истинной природе. Этот персонаж, как художественный образ, призван показать слабость старых премодернистских божеств перед принципиально новым идеальным существом третьей волны экзистенциализма.

В чем же заключается сущность третьей волны и чем она отличается от всех предыдущих? Как мы помним, все предыдущие экзистенциальные теории рушились из-за детерминизма. Кьеркегора от него защищал Бог, но Бога не стало и теория исчезла. Сартр просто начал его отрицать как постмодернист, за что был раскритикован, а его теория уничтожена. Лейн же справилась с детерминизмом просто перейдя в иной план существования – информационный. Математика абстрактна, внутри математики вас не достанет ни биологический, ни физический, ни любой другой вид детерминизма. Масами Эйри это только понимал, Ивакура же начала существовать внутри этого понимания, она познала это так, как Кьеркегор предлагал познавать Бога. Существовая в плане, лишенном детерминизма, Лейн смогла сама менять свою эссенцию и экзистенцию, то есть стала абсолютно ответственной и абсолютно заброшенной, ибо никто ей не мог противостоять. Любое противостояние заканчивалось тем, что Ивакура просто выписывала проблему из своего бытия (иногда даже не осознавая этого). Ее борьба с Эйри носила скорее характер смены, смены старого, слабого антибожества новым идеальным экзистенциалистским существом.

Подытожим. Собственно, мною было выделено основное отличие между волнами экзистенциализма – способ борьбы с детерминизмом. В результате анализа идеального существования Лейн в рамках третьей волны, можно сделать вывод, что современные экзистенциалисты, вероятно, нашли наиболее оптимальное решение извечной проблемы их философии.

Уничтожение естественного детерминизма невозможно при существовании в естественном (реальном) мире, который к тому же лишен метафизических существ (вроде Бога). В связи с этим необходимо открыть новый «мир идей», основанный на абстрактной математике и существовать

внутри его. На мой взгляд, это единственное рабочее решение проблем экзистенциализма как культурной и общественной идеи, его дискурс в принципе имеет смысл только в таком мире.

Список использованных источников:

1. *Serial Experiments Lain - The Nightmare of Fabrication/ ABE Yoshitoshi // Doujinshi-manga, 1998. – 18 с.*
2. *Экзистенциализм – это гуманизм/ Жан-Поль Сартр ; пер. с фр. М. Грецкого. - Москва : Изд-во иностр. лит., 1953. - 42, [2] с.*
3. *Болезнь к смерти / Кьеркегор Серен Обю; пер. с дат. С.А.Исаев. – Москва : Изд-во Республика., 1993. – 33 с.*

СОВРЕМЕННОЕ СЕКТАНТСТВО. МЕТОДЫ И ИНСТРУМЕНТЫ МАНИПУЛИРОВАНИЯ СОЗНАНИЕМ

Абдулин Д.М.

Белорусский государственный университет информатики и радиоэлектроники
Г. Минск, Республика Беларусь

Миськевич В.И., кандидат философских наук, доцент

Аннотация. В данной статье определяется понятия секта и ставится проблемы такого типа религиозной организации. На примере Церкви Саентологии осуществляется анализ методов и способов вербовки adeptов. Анализируются информационные ресурсы саентологии, а также используемые в речи сектантов слова и фразы. Подчеркивается вся опасность деструктивных культов, а также важность наступательной противосектантской проповеди в СМИ и Интернете.

Ключевые слова. Сектоведение, саентология, деструктивные секты.

В 21 веке, несмотря на доступность информации, продолжают множиться деструктивные культы, которые проповедуют антинаучные, ложные и опасные идеи. Апогей деструктивной деятельности сект наступает в тот момент, когда фанатики, одурманенные идеями своих “пророков”, совершают ритуальные убийства. Возникает вопрос: почему люди до сих пор вступают в секты, которые основаны на недоказуемых и часто корыстных речах проповедников культа? Цель моей работы состоит в том, чтобы ответить на этот вопрос. Однако прежде следует прояснить значение понятия религиозная секта.

Религиозная секта – это один из типов религиозных объединений. Возникает как оппозиция к тем или иным религиозным направлениям. В таком сообществе подчеркивается равенство всех членов, провозглашается принцип добровольности объединения, делается акцент на «обращение», предшествующие членству. Стоит понимать, что отдельные вышеназванные признаки могут быть обнаружены и у других религиозных объединениях, но их совокупность характерна именно для сект [1].

Для анализа сектантских инструментов работы с обществом, я рассмотрю такую популярную секту, как саентология. К данному культу относят себя много людей из разных стран, в том числе и большое число знаменитостей. Данная секта обладает высокой скоростью распространения, поэтому я выбрал саентологию в качестве примера.

Следует понимать, что для саентологов Интернет не является единственным методом для распространения своих идей. Так, в данном культе предается большое значение личным встречам «проповедников» с проявившими интерес людьми. Это связано с тем, что в ситуации межличностных контактов человеком проще манипулировать и внушать определенные идеи. Но прежде индивид должен побольше разузнать о существовании данной организации. Для этого есть большой штат квалифицированных манипуляторов, которые рассказывают людям про саентологию. Кроме того, существует и Интернет. В Сети потенциальный последователь может еще больше узнать о данной секте и, если его заинтересуют идеи саентологии, посетит мероприятия, которое организуются этой религиозной организацией. В век информационных технологий именно виртуальный вариант общения становится преобладающим. Поэтому неудивительно, что саентологи ведут постоянную пропаганду своего учения, критикуют противников и создают в глазах пользователей Интернета положительный образ своей организации.

Scientology.ru является официальным сайтом саентологии. Это русскоязычная версия, но существуют версии этого сайта и на других языках (английском, немецком и т.д.) Сайт выполнен очень качественно и выделяется приятным дизайном. Тут очень много креативных и интерактивных элементов, много красочных фотографий и качественных видеозаписей. Фильмы, снятые для саентологов, сделаны профессионально. Они довольно запоминающиеся и интересные. Многие работы для меня были по-настоящему увлекательными и информативными.

Материалы, представленные на сайте, в основном рассказывают о практической пользе саентологии. В них говорится о преимуществах данной секты, ее различных практиках и т.д. На scientology.ru можно узнать и про техники, связанные с личностным ростом. Лидеры секты хотят создать иллюзию пользы саентологии. Человеку обещают, что ему дадут знания, которые помогут ему в достижение материального и духовного успеха. Причем, когда раздают обещания, то редко говорят о самой секте. Например, на сайте представлено видео о том, как бороться с тревожностью. В видеоролике рассказывается об агрессивности окружающего мира, о людях, которые пытаются загнать вас в стрессовую ситуацию для достижения своих целей. А если человек пройдет курс, который предлагается Церковью Саентологии, то он сможет эффективно противостоять тревожности.[3]

Однако действительная цель таких фильмов заключается не в помощи человеку решить свои проблемы, а в желание продемонстрировать бескорыстие саентологии, в отличие от других предложений, которые ориентированы на манипулирование сознанием человека ради достижения корыстных целей кукловодами. После просмотра сайта саентологов возникает ощущение, что эта секта хочет сделать мир лучше, помочь людям и достигнуть светлого будущего. Лидеры этого религиозного объединения, мол, всячески стремятся помочь людям. Ведь только у них есть своя стратегия, которая успешно работает и цель организаторов рассказать про нее как можно большему числу людей, тем самым делая наш мир лучше, добрее и справедливее. Но, в действительности, это не так. Саентологи инвестируют значительные суммы денег и времени на рекламу своего имиджа и тренингов. То есть существует подспудная установка: ты должен отдавать часть своих доходов на развитие организации, но при этом большая часть этих пожертвований реально уходит в единоличное распоряжение руководителей секты. Веб-странички были бы бесполезны без слов. Именно фразы, слова и речи являются инструментами манипуляции сознанием людей. Все по Оруэллу. Как известно, в романе «1984» Большой Брат управлял сознанием подданных посредством новояза. Так, человек не сможет говорить о свободе, равенстве и братстве, если этих понятий нет в языке, и он их не знает.

Секты идут по пути Большого Брата. Лидеры культов придумывают новые слова и переопределяют старые. Причем делают они это по-настоящему профессионально.[2] Многие даже не замечают подмены понятий. Для саентологов привычно говорить про важность е-метра в работе одитора, которые являются ключевыми понятиями саентологии, но для простого человека являются непонятными и как бы значимыми. Индивидуума намного проще запутать, если использовать непонятные для него термины. С помощью новых понятий легко пропагандировать выгодные им идеи. Например, саентологи стремятся показать, что одитинг – это их разработка, которая помогает человеку справиться со своими проблемами. Суть одитинга в общении между консультантом и клиентом, у которого есть какая-то проблема. Одитинг очень похож на сеанс у психолога. Но саентологи отрицают пользу психологии. Они утверждают, что именно одитинг позволяет людям решать свои проблемы, а психологи лишь вредят. Очевидно, что это акт двоемыслия. Сеанс у психолога – зло, одитинг – добро. Психолог – злодей, а одитор – целитель. Но на самом деле эти понятия идентичны.

Человечество всегда во что-то верило: в Яхве, в Будду, в Аллаха, в духов или в коммунизм. Вера свойственна людям и попытки уничтожить ее является бессмысленными и тщетными, так как человек всегда будет искать замену запрещенной религии, даже если этот путь окажется тернистым. Но веру нельзя путать с фанатизмом. Поэтому нельзя оправдывать сомнительные религиозные организации, основанные на корыстных целях их лидеров. Нельзя говорить хорошо о тех людях, которые наживаются на внутреннем стремление человека к вере. Поэтому я считаю, что борьба против деструктивных сект необходима. Важно распространять знания о необходимости критического мышления и об опасности сект. Однако, при изучении материалов для данной публикации, я наткнулся на очень серьезную проблему: информация, в которой пропагандируется позитивный имидж секты намного качественнее и интереснее, чем статьи, в которых содержится критика подобного рода религиозных организаций. К тому же люди часто критикуют деструктивные секты, прибегая к дезинформации и непроверенным данным. А это не только не помогает, а, скорее, наоборот вредит в деле борьбы против «ловцов душ человеческих».

Исходя из проведенного мною анализа я считаю, что общество должно стремиться к демифологизации деструктивных культов и вести активную наступательную антисектантскую пропаганду, как через сферу образования, так СМИ и Интернет.

Список использованных источников:

1. Атеистический словарь / под ред. Новикова М.П. – М. : Политиздат, 1983. – С. 439.
2. URL: <https://www.scientology.ru/courses/solutions/overview.html>
3. Сектоведение. Тоталитарные секты. Опыт систематического исследования / А.Л. Дворкин, – 3-е изд., пераб. и доп.. – Н.Новгород: Христианская библиотека, 2006. – С. 813.

СОФИСТИКА И ЛОГИЧЕСКАЯ АРГУМЕНТАЦИЯ

Абаканович А.В.

Белорусский государственный университет информатики и радиоэлектроники
Г. Минск, Республика Беларусь

Миськевич В.И., кандидат философских наук, доцент

Аннотация. В тезисах рассматривается история возникновения софистики. Акцентируется вклад софистов в развитие логики и логического мышления. Разбираются некоторые древнегреческие софизмы и анализируются допущенные в них логические ошибки. Обосновывается важность логической культуры мышления в учебе, профессиональной деятельности и при взаимодействии с окружающими людьми. Подчеркивается необходимость уметь аргументированно излагать свою точку зрения. Описываются правила корректной аргументации.

Ключевые слова. Логика, логическая культура, софистика, аргументация.

Софистика – это философское течение, которое появилось в Древней Греции в середине V века до н.э. Последователи этого течения называли себя софистами или учителями мудрости. Изначально слово «софист» обозначало знающего и авторитетного в различных вопросах человека. Софисты учили верным приемам доказательства и опровержения, открыли ряд правил логического мышления, предприняли попытки систематизировать приемы умозаключений, одними из первых стали разрабатывать логические и психологические приемы убеждения и противостояния доводам оппонентов.

Известный софист Протагор Абдерский (ок. 480-380 гг. до н.э.) считал, что «человек есть мера всех вещей: существующих – насколько они существуют, несуществующих – насколько они не существуют», а раз так, то объективной истины не существует и «относительно каждой вещи можно выставить два противоположных суждения» [1, с. 316]. Другой известный софист того времени Горгий (ок. 480-380 гг. до н.э.) учил, «что ничто не существует, если существует, то оно непознаваемо, а если и познаваемо, то неизъяснимо» [1, с. 318]. Следовательно, потенциально, опровергать и доказывать можно все, что угодно.

Поздние софисты отошли от логических принципов своих предшественников и начали разрабатывать логические уловки, основанные на многозначности слов, омонимии, синонимии, подмене понятий, внешнем сходстве явлений, принятии ложных посылок за истинные, несоблюдение правил логического вывода, извлечении отдельного события из общей связи событий и других хитростях. Так образовалась риторическая практика, основанная на сознательном нарушении законов логики. Эту практику называли софистикой, а ее приемы – софизмами (буквально – «уловки»). Софисты утверждали, что так как о каждой вещи можно высказать несколько суждений, противоречащих друг другу, то и каждому доказательству можно противопоставить другое, противоположное, причем столь же хорошо обоснованное и убедительное. Так слово «софист» приобрело негативную окраску, возникло представление о софисте, как о алчном болтуне и лгуне, который говорит, будто все знает и может за недорогую плату в короткий срок и другого этому обучить.

Оценивая феномен софистики в целом, Гегель трактовал ее как первичную форму освоения логической проблематики. В самом деле, некоторые софизмы оказались парадоксами – важнейшей формой постановки логических проблем. Примером такого софизма может служить парадокс Эватла. У известного софиста Протагора был ученик Эватл, обучавшийся софистике и судебному красноречию. По заключенному между ними договору Эватл должен был заплатить за обучение 10 тысяч драхм, если он выиграет свой первый судебный процесс.

Однако, закончив обучение, Эватл не стал участвовать в судебных тяжбах и, соответственно, не совершал оплату за учебу. Это длилось очень долго, терпение Протагора иссякло, и он сам подал на своего ученика в суд. Таким образом, должен был состояться первый судебный процесс Эватла.

Протагор объяснил свой поступок следующим образом: «Каким бы ни было решение суда, Эватл должен будет заплатить. Если он выиграет свой первый процесс, то заплатит по договору, а если проигрывает, то заплатит по решению суда».

Эватл был не согласен: «Ни в том, ни в другом случае я не заплачу. Если я выиграю, то решение суда освободит меня от обязанности платить, а если проиграю, то в силу нашего договора я не должен платить, поскольку не выиграл свой первый процесс».

Как разрешилось дело двух софистов - неизвестно. Есть версия, что судьи отказались выносить вердикт, потому что любое их решение противоречило бы самому себе. [2, с. 261-264]

Рассмотрим еще некоторые примеры известных софизмов и укажем на скрытые в них логические ошибки.

«Что ты не потерял, то имеешь. Ты не потерял рога, следовательно, ты их имеешь». Этот софизм построен на двусмысленности выражения "то, что не терял". В первой посылке оно означает "то, что имел и не потерял", но во второй посылке это значение не подходит, потому что высказывание "Рога – это то, что ты имел и не потерял" является ложным. Следовательно, во второй посылке используется уже иное значение: "то, что не потерял, независимо от того, имел или нет".

«Знаете ли вы, о чем я сейчас хочу вас спросить?» – «Нет». – «Неужели вы не знаете, что лгать – нехорошо?» – «Конечно, знаю...» – «Но именно об этом я и собирался вас спросить, а вы ответили, что не знаете; выходит, что вы знаете то, чего вы не знаете». Ошибка данного софизма заключается в неоднозначности естественного языка. Первый вопрос можно трактовать двояко: «Знаете ли вы ответ на вопрос, который я задам?» или «Знаете ли вы вопрос, который я задам?».

«Сидящий встал; кто встал, тот стоит; значит, сидящий стоит». «Сидящий стоит» эквивалентно высказыванию "тот, кто сидит, стоит". Это высказывание ложно. Правильно было бы «сидевший стоит». В данном софизме ошибка произошла из-за потери контроля над временем причастия.

« $5 = 2 + 3$, но 2 чётно, а 3 нечётно, следовательно, 5 одновременно чётно и нечётно». Этот софизм основан на разъединении смысла целого.

Сегодня под софизмом понимаются логические ошибки, которые преднамеренно допускают в рассуждениях для введения собеседника в заблуждение. Сознательное использование софизмов является некорректным приёмом рассуждения и аргументации, поэтому обоснованно подвергается критике.

Необходимо уметь распознавать софизмы, чтобы не попасться на уловки софиста и не быть обманутым. Чтобы успешно справляться с софизмами, встречающимися в процессе дискуссии, необходимо хорошо знать предмет, о котором идет речь, обладать знаниями в области законов логики, уметь подмечать допускаемые оппонентом логические ошибки и убедительно раскрывать неправоту его аргументов. И первый помощник человека в этом деле – его логическая культура. Благодаря логической культуре мышления можно научиться грамотно строить свои рассуждения и отстаивать свою точку зрения верными методами. Очень часто понимание ошибок в софизме ведет к пониманию смысла в целом, помогает развивать логику и навыки правильного мышления.

Вместе с тем, софистика была важным этапом на пути формирования и становления логической культуры мышления и логики как науки. Логическая культура мышления – это прежде всего дисциплина ума, способность и умение человека излагать свои мысли аргументированно, соблюдая законы логики и правила логического вывода. Основы культуры логического мышления закладываются в школе, однако по большей части это происходит стихийно. В Белорусском государственном университете информатики и радиоэлектроники мы изучаем систематический курс логики в течении семестра, сочетая теорию с практикой. Полученные знания и интеллектуальные навыки оказываются очень полезны и востребованы в последующем: при изучении других учебных дисциплин, подготовке работ на студенческие научные конференции, восприятию и оценке информации из интернета и других источников.

Сегодня в обществе идет ориентация на цивилизованный диалог во всех сферах человеческой деятельности, что предполагает умение общаться с оппонентом, дискутировать, аргументированно излагать свою точку зрения и приводить правильные доводы, чтобы в итоге убедить противную сторону. Таким образом, современному человеку необходимо обладать навыками аргументации. Потребность в аргументации возникает тогда, когда по одному вопросу есть несколько точек зрения. Любой процесс аргументации предполагает наличие тезиса, то есть утверждения, истинность которого необходимо доказать, аргументов или доводов и демонстрации – логической связи между тезисом и аргументами. Аргументы – истинностные суждения, которые используются для доказательства тезиса. Они бывают двух видов: корректные и некорректные. Корректные – объективны и касаются сути доказываемого тезиса, это могут быть аксиомы, теоремы, законы, факты, определения. Некорректные аргументы касаются психологии человека и характерных черт личности, например, ссылка на авторитет, игра на публику, угрозы, обещания и заверения, игра на чувствах и т.д. Доказательства с такими аргументами считаются в логике неправильным и их следует избегать

Для корректной аргументации необходимо придерживаться некоторых правил:

1. Тезис должен быть точно, ясно и однозначно сформулирован.
2. Тезис должен оставаться одним и тем же на протяжении всего доказательства или опровержения.

3. Аргументы должны быть истинными, не противоречащими друг другу суждениями.
4. Число аргументов должно быть конечным и достаточным для доказательства тезиса.
5. Истинность аргумента должна быть доказана независимо от тезиса [3, с. 261, 285].

Таким образом, правильное логическое мышление и корректная аргументация должны быть неотъемлемой частью логической культуры каждого образованного человека и помогать ему при взаимодействии с окружающими людьми.

Список использованных источников:

1. Антология мировой философии. В 4-х томах. Т. 1, Ч. 1. Философия древности и средневековья / ред. В. В. Соколов, Ф. В. Асмус и др. – М.: «Мысль», 1969. – 576 с.
2. Геллий, Авл. Аттические ночи. В двух книгах. Книги I–X. / Авл Геллий. – СПб.: "Гуманитарная Академия", 2007-2008. – 448с.
- Малыхина, Г. И. Логика: учебник / Г. И. Малыхина. – Минск: Вышэйшая школа, 2013. – 335 с.

SOFT SKILLS – ГЛАВНОЕ ТРЕБОВАНИЕ СОВРЕМЕННОЙ ИТ-ИНДУСТРИИ

Климец А.А.

*Белорусский государственный университет информатики и радиоэлектроники
г. Минск, Республика Беларусь*

Малыхина Г. И. – к.ф.н., доцент

Доклад рассматривает возможные причины того, почему в современной ИТ-индустрии при найме специалистов всё больше обращается внимание на так называемые «мягкие навыки» потенциального сотрудника, и зачастую эти навыки становятся даже важнее профильных знаний.

Soft skills в переводе с английского языка – «гибкие навыки» или «мягкие навыки». В общем случае эти навыки не связаны конкретно с ИТ-индустрией или с любой другой конкретной профессией. В общем случае это непрофессиональные навыки, которые, тем не менее, важны для карьеры и влияют на то, насколько хорошо человек выполняет свою работу. О том, насколько важны эти навыки говорит активное обсуждение данного неологизма специалистами различного профиля, так как в современном мире абсолютное большинство людей не работают в одиночку, а так или иначе взаимодействуют с другими коллегами, клиентами, партнёрами или руководством. Работа с людьми предполагает умение ладить, договариваться, аргументировать свои доводы, доносить какую-либо информацию до других людей в понятной форме. ИТ-индустрия в данном докладе рассматривается как одно из самых прогрессивных и динамично развивающихся профессиональных сообществ, в котором о «мягких навыках» можно услышать, чуть ли не чаще всего на данный момент.

ИТ-индустрия прошла долгий и тернистый путь за десятилетия своего существования, и претерпела значительные изменения в том числе и в подходах к работе. Первые компании чаще всего занимались глубоко исследовательской деятельностью, разрабатывая нечто новое, для чего гораздо более важными были именно технические компетенции специалиста, хотя уже и в то время начали обращать внимание на важность «мягких навыков». Первые исследования в этой области начались также в США в 1960-е годы.

В этот период времени были формализованы традиционные методы разработки, основанные на создании спецификаций, тщательном планировании, и реализации согласно разработанному плану. Данный подход к работе применялся довольно долго, ставя профессиональные навыки выше личностных. Основной причиной этого является тот факт, что многие технологии разрабатывались именно в это время, закладывая фундамент, на котором и стоит ныне вся ИТ-индустрия. Технологии развивались не так стремительно, как это происходит сейчас, соответственно и информация устаревала не так быстро. От специалистов не требовалось постоянно быть в курсе последних разработок.

Ситуация изменилась в связи с глобальной информатизацией, активно начавшейся в девяностых годах прошлого века, а уже в конце девяностых сам термин «soft skills» полноценно вошёл в бизнес-среду. И сейчас уже невозможно представить успешное предприятие без каких-либо программ для учёта, хранения информации, управления предприятием, или веб-сайта. Сферы применения информационных технологий чрезвычайно разнообразны, и у каждой своя специфика, которую обязательно следует учитывать при создании продукта для конкретной сферы. В современной ИТ-индустрии требуется, чтобы ИТ-специалисты и представители бизнеса взаимодействовали как можно активнее. Без такого синергичного взаимодействия невозможно говорить о результативной работе. И в этих условиях личностные качества начинают брать верх над профессиональными компетенциями.

В силу этих событий в 2001 году на свет появился Agile Manifesto, определивший 4 принципа, на основе которых впоследствии и возникли гибкие методологии, чрезвычайно быстро захватившие всю индустрию. Эти принципы направлены на то, чтобы отказаться от каких-то жёстких правил, отчётности и т. п., предоставляя людям возможность свободно творить, кооперироваться и создавать нечто инновационное, находить наилучшие решения самых нестандартных задач. Помимо этого, благодаря гибким методологиям удалось установить тесное взаимодействие с заказчиком, неким представителем бизнеса, не специалистом в разработке ПО. Это позволило разработчику понять, что ему необходимо сделать, чтобы заказчик был доволен.

В этот же период кардинально снизился порог вхождения в ИТ-индустрию новых специалистов из-за появления большого количества простых технологий, не требующих специфических знаний или большого опыта. Однако, при этом, появление большого количества «родственных» технологий и решений потребовало более творческого подхода при создании чего-то нового, быстрой адаптации к постоянно меняющимся запросам современных пользователей.

Таким образом, современный ИТ-специалист — это прежде всего всесторонне развитая личность, а самыми важными «мягкими навыками» для ИТ стали: коммуникативные навыки, навыки самоорганизации, креативные навыки, умение работать с информацией, стрессоустойчивость.

ВСЕГДА ЛИ МЫ МЫСЛИМ?

Ермаков А. В.

*Белорусский государственный университет информатики и радиоэлектроники
г. Минск, Республика Беларусь*

Гуринович С. В., преподаватель философии

Работа описывает мышление человека с точек зрения философов различных эпох. Рассматриваются особенности мышления, присущие современной молодёжи, сущность и значение мышления.

Все мы считаем себя мыслящими существами. Это кажется нам логично вытекающим из самого определения понятия “человек”, однако в действительности всё куда сложнее. Задумавшись о сущности мышления, зададим вопросы: “Что такое мышление?” “Всегда ли мы мыслим?” и, наконец, “Зачем мыслить?”

В современном философском пространстве имеет место чёткое различие двух понятий - «сознание» и «мышление» [1, с. 35]: человек имеет сознание в качестве атрибутивного свойства. Оно многомерно и включает в себя разные порядки работы: стихийный и автономный, то есть складывающийся под воздействием внешних обстоятельств и работающий сам по себе, по собственным правилам. Знака равенства между сознанием и мышлением не существует. Мышление — особый вид деятельности сознания, который предполагает его автономную работу. Мышление — это свобода от всех внешних факторов, способность усомниться в их правильности и начать думать самостоятельно. Оно предполагает взятие на себя ответственности за свои идеи.

Мыслящий человек делает мир “своим” — он проводит границу между собой и Вселенной. Именно мышление, а не какой-либо другой режим работы сознания, выделяет человека среди всех остальных живых существ и из природы в принципе. Природа в миллион раз сильнее человека, но она *не знает* этого. Блез Паскаль в своём труде “Мысли” писал следующее: “Человек — всего лишь тростник, слабейшее творение природы, но он — тростник мыслящий... пусть даже его уничтожит Вселенная, человек всё равно возвышеннее, чем она, ибо сознаёт, что расстанется с жизнью и что слабее Вселенной, а она ничего не сознаёт. Итак, всё наше достоинство — в способности мыслить”. Однако современное поколение (достаточно чётко в философской литературе артикулировано как «компьютерное поколение» [2]) «страдает» игнорированием, отказом, «бегством от мышления».

О субъективных причинах подобной проблемы разума размышлял ещё в XVII наш соотечественник — просветитель, книжник, богослов Симеон Полоцкий. Особый интерес представляют его размышления о соблазнах и искушениях, которые встают на пути человека к мышлению. Он выделил из них 7 и назвал их «грехами» ума: невежество (отсутствие знаний и нежелание их получать); нерассудительность (совершение опрометчивых поступков под влиянием отрицательных эмоций); скоросудие, когда человек делает выводы быстро без тщательного обдумывания всех факторов; непостоянство ума, когда человек очень легко меняет одну точку зрения на другую; и, в противоположность — упрямство, когда человек не желает менять точку зрения, наглядно доказавшую свою ложность; мудрствование, когда человек, не желая проигрывать в споре,

начинает пускаться в хитроумные, далеко не всегда оправданные рассуждения; суетность, ведущая к знаниям бесполезным или недостойным; стремление к бесполезным знаниям. [3, с. 168]

Трудно называть невежественным современного молодого человека — как правило, он образован, многое знает, способен поддержать беседу и высказать мнение практически на любую тему. Однако он крайне редко стремится к глубоким знаниям о предмете и тем более не задумывается о *смысле* того, что он познаёт. Его познания поверхностны и разрозненны, он знает “ничего обо всём” и способен лишь произвести впечатление знающего.

Для компьютерного поколения характерны установки: доступ сразу ко всему и везде, гибкость и скорость. Подверженность этих людей «грехам» ума очень сильна, для них характерны суетность и нетерпение, сильно мешающие мышлению. В современном постоянно ускоряющемся мире у человека просто нет времени на то, чтобы что-то обдумать. Ему постоянно кажется, что он где-то не успевает, он берётся за что-то, бросает при первой же неудаче, берётся за что-то другое и самое главное — он не признаёт, что что-то делает неправильно, ни перед собой, ни тем более перед другими.

Объективные причины отказа от мышления, присущие субъекту техногенной цивилизации, были проанализированы немецким философом XX века М. Хайдеггером. М. Хайдеггер в своей речи на праздновании 175-й годовщины со дня рождения К. Крейцера [4] сказал: “Бездумность – зловещий гость, которого встретишь повсюду в сегодняшнем мире, поскольку сегодня познание всего и вся доступно так быстро и дешево, что в следующее мгновение полученное так же поспешно и забывается”. Эти слова были сказаны в 1955 году, и нельзя не признать, что они не перестали быть актуальными в современном мире. “В то, что называется мышлением, мы попадаем, когда беремся думать сами. Чтобы подобная попытка удалась, мы должны быть готовы учиться мысли”.

По М. Хайдеггеру, у современного человека преобладает вычисляющее мышление – оно постоянно и непрерывно калькулирует всё новые и новые возможности, рассчитывая на выгодный результат. Человек, являясь по своей природе мыслящим существом и обладая огромным потенциалом к мышлению – отвергает его, по выражению М. Хайдеггера, “спасаясь бегством от мышления”. Другой тип мышления – “осмысляющее раздумье” – то, что ищет смысл всех действий человека, хочет добраться до сути и углубить своё понимание без расчёта конечной выгоды – остаётся в подавленном, хотя и способном к росту состоянии.

В этом таится невероятная опасность, которая стоит перед человечеством. И она не только в том, что в постоянно ускоряющейся погоне за чем-то новым человек может не совладать с огромным потенциалом, который у него есть. Опасность не только в возможной физической катастрофе, опасность в потере человеком самой сущности человека. “Мы не останавливаемся, чтобы подумать, что с помощью технических средств готовится наступление на жизнь и сущность человека, с которым не сравнится даже взрыв водородной бомбы... Тогда человек отречётся и отбросит свою глубочайшую сущность, именно то, что он есть размышляющее существо”. Эта опасность уже наступает на человечество, и чтобы его спасти, нужно не забыть про свою человеческую сущность. Необходимо подвергать каждое своё действие, разработанное вычислительным мышлением, тщательному осмыслению.

В заключение можно утверждать, что вне зависимости от временных координат иметь мужество и умение мыслить означает возможность обретения самого себя в этом мире.

Список использованных источников:

1. Вязовкин, В. С. *Бытие и мышление* / В. С. Вязовкин, Т. М. Тузова - Минск, 2000. - 60 с.
2. Луков, Вал. А. *Будущее молодежи в обществе будущего* / Вал. А. Луков // *Человек*. - 2012. - 4. С 38-48.
3. Симеон Полоцкий. *Обед душевный* / Симеон Полоцкий // *Памятники философской мысли Белоруссии XVII – первой половины XVIII в.* / сост. С. А. Подокшин [и др.]. – Минск, 1991. – 319 с.
4. Хайдеггер, М. *Разговор на просёлочной дороге* / М. Хайдеггер. – Москва, 1991. – 192 с.

ФИЛОСОФСКОЕ ОСМЫСЛЕНИЕ ПРОБЛЕМ ПРОИСХОЖДЕНИЯ ВСЕЛЕННОЙ И ТЕОРИИ БОЛЬШОГО ВЗРЫВА

Фикова П.В., Петрова Т.А.

Белорусский государственный университет информатики и радиоэлектроники

г. Минск, Республика Беларусь

Шепетюк В.В. – кандидат философских наук

Тема научной работы посвящена исследованию философских аспектов ряда вопросов, связанных с изучением процессов происхождения Вселенной. Рассматривая различные исторические периоды философии, авторы реконструируют линию рассуждений мыслителей прошлого о проблемах сотворения космоса и окружающего мира. В завершении работы указываются основные философские идеи теории Большого взрыва как наиболее известного и авторитетного научного подхода, используемого в наше время, для описания процесса зарождения и развития Вселенной.

Космос всегда был для людей страшной и одновременно манящей загадкой, которая привлекала к себе внимание выдающихся умов человечества. Их целью была попытка объяснить суть происходящих в мире процессов, а также ответить на выходящие далеко за пределы человеческого опыта вопросы – как и когда возник мир, есть ли у него пределы, что будет со Вселенной через много лет? Эти и многие другие схожие по тематике вопросы были важны в прошлом, однако, и сегодня их актуальность только растет. Открыв в середине XX века для себя возможность пребывания на околоземной космической орбите, люди в настоящее время демонстрируют озабоченность очередным рывком по развитию технологий будущего, которые неизменно ассоциируются с реализацией уже более сложных космических проектов и миссий. Подтверждением тому являются инновационные проекты освоения космоса Илона Маска, амбициозные планы ведущих сверхдержав по колонизации Луны, международные исследовательские программы по изучению экзопланет и многое другое. Практическая значимость изучения проблем происхождения Вселенной, таким образом, связана с поиском путей выхода на новый уровень технико-технологического развития человеческой цивилизации, которая осмысливает себя и свое существование уже не только локально, но и глобально, ассоциируя себя со сложнейшими и масштабными процессами мироздания, что ведет к расширению знаний и совершенствованию разума новых поколений людей. В данной работе делается попытка обозначить некоторые значимые философские идеи, касающиеся темы происхождения Вселенной и теории Большого, так как это может рассматриваться в качестве инструмента по массовому просвещению современников с целью их необходимой информационной подготовки к обществу будущего, а также как способ недопущения деградации интеллекта людей и уменьшения их мировоззренческой компетентности.

Итак, вопросы рождения и развития космоса обсуждались еще в рамках космогонии древнеегипетской и месопотамской цивилизаций [1, с. 11]. Для них Вселенная была создана как динамическая сущность, которая дала начало всему существующему, включая людей и богов. Истории, которые древние ученые рассказывали о происхождении и эволюции Вселенной, были не научными, а мифологическими. Из-за того, что решающую роль играли боги, истории были теогониями в той же мере, что и космогониями. Общей чертой культур обеих цивилизаций была вера в то, что творческий процесс начался с недифференцированного водного хаоса, который боги впоследствии разделили на два или более отдельных царства, создав таким образом землю и небо как отдельные тела. Говорилось о первоначальном состоянии темноты и безжизненного однообразия, но оно не было объяснено. В любом случае, главной заботой ранних космогоний было не описание Вселенной, а объяснение существования ее обитателей в виде богов, людей и всего остального. Именно такую историю мы видим в месопотамской мифе о сотворении мира «Энума Элиш», датированном примерно 1500 годом до нашей эры, а также в гораздо более поздней «Теогонии», написанной греческим поэтом Гесиодом [2].

С появлением греческой натурфилософии и ее постепенным превращением в науку, астрономия превратилась в математическую дисциплину, сосредоточенную на наблюдениях за планетной системой. Римский поэт Тит Лукреций Кар, сторонник ранней атомистической философии, опирающейся на учения Демокрита и Эпикура в знаменитом тексте, составленном примерно в 50-м году до нашей эры, под названием «De Rerum Natura» («О природе вещей») [3], описал бесконечную вселенную состоящую исключительно из атомов в непрерывном движении. Эта вселенная имела конечный возраст. Неспособный представить себе мир без людей, он спросил: «Если не было происхождения небес и земли от поколения к поколению, и если они существовали вечно, то почему другие поэты, до Фиванской войны и разрушения Трои, не воспевали также другие подвиги

жителей Земли?» [3, с. 120] Лукреций предположил, что весь мир сравнительно современен и имеет недавнее происхождение, он не может быть вечным в прошлом, но должен иметь начало во времени.

Философская космология Лукреция полностью отличалась от гораздо более влиятельных идей Аристотеля, изложенных в его трактате "О небе" [4] и других работах. Вселенная Аристотеля находилась в устойчивом состоянии настолько, что была вечной. Он утверждал, что Вселенная в целом не только уникальна, но и бесконечна во времени в обоих направлениях. Другими словами, она была вечной и, следовательно, несотворенной, а также неразрушимой.

Христианская Вселенная была божественно создана из ничего и, следовательно, имела конечный возраст, что резко противоречило тому, чему учил Аристотель. В VI веке христианский философ Иоанн Филопон разработал ряд рациональных аргументов, основанных на концепции бесконечности, против ереси Аристотеля. Фома Аквинский, в свою очередь, указывал, что, поскольку Бог - существо вневременное, ему не нужно предшествовать своим действиям во времени. Непрерывное, но вневременное создание было в конце концов принято католической церковью под названием *creatio continua* в противоположность *creatio originans* [5].

В эпоху Просвещения наиболее новаторская космогония была опубликована в 1755 году И. Кантом под названием «Всеобщая естественная история и теория неба» [6], где говорилось о первобытном хаосе частиц, а затем, опираясь на принципы ньютоновской механики, указывалось, как хаотическое состояние естественным образом эволюционировало в сгущения, из которых образовалась Солнечная система и вся упорядоченная Вселенная. Сотворение, по Канту, то, что однажды началось, но никогда не прекратится, создавая новые вещи и новые миры.

В последующие периоды истории важными вехами в развитии космогонии было открытие в 1850-х годах второго закона термодинамики, разработка концепции энтропии Р. Клаузиуса, создание общей теории относительности А. Эйнштейна, модели нестационарной Вселенной А.А.Фридман в 1922 г., открытие новых галактик в 1924 г. Э. Хабблом и многие другие. К настоящему времени, несмотря на накопленный массив знаний, научные дискуссии и споры о механизмах и законах, лежащих в основе процессов развития Вселенной, продолжают. Вместе с тем наиболее известным и авторитетным научным подходом, используемым в наши дни, для описания процесса зарождения и развития Вселенной является теория Большого взрыва.

Не имея возможности представить здесь детальное описание теории Большого взрыва, ограничимся лишь указанием на тот факт, что согласно этой точке зрения расширяющаяся в настоящий момент Вселенная примерно 15 млрд. лет назад была сосредоточена в очень маленькой области. Предполагается, что плотность вещества Вселенной была сравнимой с плотностью атомного ядра, и вся Вселенная представляла собой огромную ядерную каплю. По каким-то причинам ядерная капля оказалась в неустойчивом состоянии и взорвалась. Предполагается, что расширение Вселенной после этого происходило с одинаковой скоростью, начиная с момента взрыва. Самая ранняя Вселенная имела температуру превышающую 10 млрд. градусов. Материя в форме атомов тогда существовать не могла. Атомы появились лишь когда возраст Вселенной достиг приблизительно трехсот тысяч лет. Молодая Вселенная состояла из очень плотной смеси частиц и античастиц разнообразных типов. Взаимодействуя между собой, они стали основой для формирования материи, которая в пределах далее расширяющейся Вселенной постепенно охлаждалась, достигая тех значений, когда она начинает изменять свои фундаментальные свойства и приобретает привычные нам формы.

Бесспорно, теория Большого взрыва захватывает воображение и почти никого не оставляет равнодушным. Важность обращения к ней при изучении курса изучения философии связана с необходимостью восполнения научно-культурной некомпетентности и неосведомленности многих современных людей, искренне полагающих, что они образованы и многое знают об окружающем мире. Фундаментальность мировоззренческих вопросов, поднимаемых в рамках теории Большого взрыва позволяет по-новому взглянуть на привычный порядок вещей и помогает человеку стать более зрелым при восприятии себя и пространства своей жизни, природы, планеты и всего того, что пока еще не доступно нашему повседневному опыту. Таким образом, обращение к теме происхождения и развития Вселенной, изучение теории Большого взрыва наделяют носителя такой информации ценными и полезными с точки зрения эволюции общества качествами, совершенствуя в нем способность широко мыслить, тренировать свое воображение, память, логику и интеллект, развивать пытливость разума и любознательность, умение интегрировать знания из различных областей науки и, в конечном счете, способность творить.

Список использованных источников:

1. Берковичи, Д. Происхождение всего. От Большого взрыва до человеческой цивилизации / Д. Берковичи. – Москва: Альпина Нон-фикшн, 2017. - 202 с.
2. Гесиод. Работы и дни: Теогония; Щит Геракла / Гесиод. - Изд. 2-е. - Москва: URSS, Либроком, 2012. - 282 с.
3. Лукреций. О природе вещей / Лукреций. - Москва: Мир книги, Литература, 2006. - 335 с.
4. Аристотель. О небе / Аристотель // Сочинения: в 4 т. – Москва, 1981. - Т. 3. – С.263 - 378.
5. Аквинский, Ф. Сочинения / Ф. Аквинский. - Изд. 6-е. - Москва: URSS, Либроком, 2013. – 229 с.
6. Кант, И. Всеобщая естественная история и теория неба / И. Кант. – Москва: Директ-Медиа, 2002. – 265 с.

СЕКЦИЯ «ИНФОРМАТИКА»

АЛГОРИТМ КОЛОРИЗАЦИИ ИЗОБРАЖЕНИЙ НА ОСНОВЕ ГЕНЕРАТИВНО-СОСТЯЗАТЕЛЬНЫХ НЕЙРОННЫХ СЕТЕЙ

Камзеев Н.А.

Белорусский государственный университет информатики и радиоэлектроники
г. Минск, Республика Беларусь

Калугина М.А. – канд. физ.-мат. наук, доцент

В работе рассмотрен один из подходов к решению задачи колоризации изображений. Описаны некоторые проблемы в этой области, реализована модификация модели с архитектурой на базе генеративно-состязательных нейронных сетей, позволяющая для обучения алгоритма использовать меньший объем данных, чем в базовой модели без потери качества результата.

Одним из наиболее впечатляющих приложений глубокого обучения является колоризация изображений. Колоризация представляет собой преобразование монохромного (черно-белого, сепия) изображения в цветное. Данный процесс применяется, например, для реставрации старых фотографий, киноплёнок и картин. Несколько лет назад задача колоризации требовала довольно много человеческих усилий: на воссоздание цвета старого черно-белого фильма мог уйти ни один год кропотливой работы, но сами идеи раскраски изображений были не очень сложными для понимания. Сейчас весь процесс от начала до конца может быть выполнен гораздо быстрее и эффективнее – с помощью методов искусственного интеллекта. Однако за счет сложного математического аппарата, использующегося в этих методах, алгоритм решения задачи колоризации кажется не таким простым, как при ручной раскраске. Применение математического аппарата обусловлено наличием того факта, что при преобразовании цветного изображения в его базовое одноканальное представление теряется очень много информации (два из трех цветовых измерений), а математический аппарат помогает ее частично восстанавливать, запоминая и затем воспроизводя семантику образов изображений: трава обычно имеет зеленый цвет, а небо – голубой. Таким образом, цель раскрашивания состоит не в том, чтобы восстановить фактический, истинный цвет, а скорее в том, чтобы произвести правдоподобную раскраску, которую пользователь может найти полезной, даже если она не соответствует оригинальному цвету изображения.

На данный момент особый успех в решении задач, связанных с обработкой изображений, имеют модели глубокого обучения. В одних статьях [1] к проблеме колоризации подходят как к задаче классификации и решают ее с помощью сверточных нейросетей (англ. CNN), уделяя внимание неопределенности задачи: автомобиль на изображении может принимать много различных и в то же время допустимых цветов, и без отсутствия дополнительных знаний нельзя отдать предпочтение какому-то определенному цвету. В других же исследованиях [2] подходят к колоризации как к задаче регрессии с некоторыми дополнительными настройками. У каждого подхода есть свои преимущества и недостатки, но в данной работе будет применена еще одна стратегия.

Генеративно-состязательные нейросети (англ. GANs) являются одной из самых интересных тем машинного обучения сегодня. Наиболее популярная сфера их применения – это генерация реалистичных фотографий, но они также дают потрясающие результаты в таких задачах, как изменение изображения с одного домена на другой (например лошадей на зебр) – CycleGAN, создание изображений с высоким разрешением из более низкого разрешения (англ. Super-resolution) – SRGAN, создание изображения по семантической окраске – Pix2Pix, создание изображений по заданному тексту – StackGAN, генерация музыки – GANSynth.

В проведенном исследовании за базовый алгоритм для изучения и последующего улучшения был взят алгоритм Pix2Pix [3]. Этот алгоритм предлагает общее решение для многих задач, связанных с преобразованием изображения (англ. Image-to-image translation), одной из которых как раз и является колоризация. В данном подходе используется функция потерь L1, которая делает задачу колоризации задачей регрессии, и функция потерь GAN, которая помогает решать задачу без размеченных данных. Но прежде, чем реализовывать алгоритм Pix2Pix и пытаться его модифицировать, нужно понимать архитектуру генеративно-состязательных сетей.

Генеративно-состязательная сеть состоит из двух нейросетей: генеративной (англ. Generator), которая пытается моделировать реальное распределение данных, и дискриминативной (англ. Discriminator), которая оценивает вероятность того, что выборка данных на входе данной модели принадлежит подлинному набору данных, а не сгенерированному.

В случае задачи колоризации генератор можно представить как модель, преобразующую одноканальные изображения в оттенках серого в двухканальные изображения с каналами *a (пространство зеленого и красного цвета) и *b (пространство желтого и голубого цвета). Дискриминатор принимает на вход два сгенерированных канала и объединяет их с одноканальным черно-белым изображением, тем самым получая цветное изображение в пространстве L^*a*b (L – отвечает за яркость), а затем оценивает вероятность того, что полученное изображение является

настоящим. Безусловно, дискриминатор нужно обучить на некотором наборе реальных изображений (трехканальные в цветовом пространстве L^*a^*b), которые не были произведены с помощью генератора.

Что касается оптимизационной функции, то она, как упоминалась ранее, состоит из двух функций потерь: cGAN-loss и L1-loss.

Первая помогает генерировать качественные и правдоподобные цветные изображения:

$$L_{cGAN}(G, D) = E_{x,y}[\log D(x, y)] + E_{x,z}[\log(1 - D(x, G(x, z)))], \quad (1)$$

где G – генератор, D – дискриминатор, x – черно-белое изображение, z – шум, подающийся на вход генератору, и y – двухканальное изображение, полученное генератором, или два канала реального изображения.

Вторая же минимизирует среднюю абсолютную ошибку предсказанных цветов:

$$L_{L1}(G) = E_{x,y,z}[|y - G(x, y)|_1]. \quad (2)$$

Если использовать только ее, то модель по-прежнему будет обучаться и решать поставленную задачу, но изображения будут получаться однотонными, с оттенками серого и коричневого.

Поэтому, объединив две функции потерь, можно получить следующую функцию для оптимизации:

$$G^* = \arg \min_G \max_D L_{cGAN}(G, D) + \lambda L_{L1}(G), \quad (3)$$

где λ — это коэффициент для регулирования влияния двух функций потерь на итоговую функцию.

Для получения хороших результатов с помощью данной архитектуры необходимо очень много данных. Если данных будет мало, то в результате окрашенные изображения будут иметь достаточно заметные аномалии. Однако, если обучить генератор на размеченных данных до того, как использовать его в итоговой модели, можно получить удивительные результаты. Предобучение состоит из двух ключевых частей. Первая – это использование уже обученной сети для классификации картинок – ResNet18 например. Вторая – весь генератор обучается, решая задачу колоризации, с функцией потерь L1.

В результате такой модификации для обучения всей состязательной сети было использовано всего 8000 изображений из набора данных COCO. Использование такого небольшого набора данных показывает достаточно хорошие результаты за счет того, что в скелете генератора используются сети, обученные на огромных наборах изображений из ImageNet, а исходный набор данных позволяет подстроить алгоритм под конкретную задачу.

Пример результатов работы итогового алгоритма приведен на рисунке 1.



Рисунок 1 – Результаты колоризации с помощью обученной модели: сверху – черно-белое входное изображение, снизу – сгенерированное цветное

Таким образом, разработанная модификация известного алгоритма Pix2Pix позволяет обучаться на меньшем наборе данных и в то же время дает отличные результаты. Идея модификации заключается в предварительном обучении генератора отдельно, используя для этого размеченные данные. Это позволяет устранить проблему, заключающуюся в том, что в начале обучения ни генератор, ни дискриминатор ничего не знает о задаче. В завершение отметим, что вся работа была реализована на языке программирования Python 3.8.

Список использованных источников:

1. Colorful Image Colorization / Richard Zhang, Phillip Isola, Alexei A. Efros // [arXiv:1603.08511v5](https://arxiv.org/abs/1603.08511v5) [cs.CV], 2016.
2. Old Image De-noising and Auto-Colorization Using Linear Regression, Multilayer Perceptron and Convolutional Neural Network / Junkyo Suh, Koosha Nassiri Nazif, Aravindh Kumar // <http://cs229.stanford.edu/proj2017/final-reports/5244272.pdf>, 2017.
3. Image-to-Image Translation with Conditional Adversarial Networks / Phillip Isola, Jun-Yan Zhu, Tinghui Zhou, Alexei A. Efros // [arXiv:1611.07004v3](https://arxiv.org/abs/1611.07004v3) [cs.CV], 2018.

СГЛАЖИВАЮЩИЕ СВОЙСТВА КЛЮЧЕЙ КОМБИНАЦИОННОГО УСТРОЙСТВА ВСЕВОЗМОЖНЫХ ПЕРЕСТАНОВОК

Кохновский С.И.

Белорусский государственный университет информатики и радиоэлектроники
г. Минск, Республика Беларусь

Иванюк А.А. – д-р техн. наук, профессор

В работе рассмотрено функционирование устройства всевозможных перестановок при работе с q -ми векторами, а также свойства ключей устройства, полученные в процессе анализа сопоставления результатов работы ключей на бинарных и q -х векторах.

Широта сферы применения аппаратных шифраторов всем хорошо известна: от блочных шифров до составного компонента сложных вычислительных систем. Предметом исследования является спроектированное запутывающее устройство, задающее отображение множества входных сигналов I на множество выходных O в зависимости от используемого ключа. На вход комбинационной схемы подаются векторы $input = (i_1, i_2, \dots, i_n), input \in I$ и $key = (k_1, k_2, \dots, k_m), key \in K$, причём $k_i \in \{0, 1\}, \forall i \in \overline{1, m}$. На выход схема возвращает вектор $output = (o_1, o_2, \dots, o_n), output \in O$ – перестановку входного вектора.

Введём расстояние по Хэммингу между векторами $input$ и $output$

$$D_H = d_{key}(input, output) = \sum_{i=1}^n |input_i - output_i|, \quad (1)$$

где $input_i \in \{0, 1\}, \forall i \in \overline{1, n}$ и $output_i \in \{0, 1\}, \forall i \in \overline{1, n}$.

А также среднее расстояние по Хэммингу

$$D_H^* = \frac{1}{n} \sum_{i=1}^n d_{key_i}(input, output), \quad (2)$$

причём $key_i \in K, \forall i \in \overline{1, m}$.

Расчёты D_H , а также D_H^* в зависимости от подаваемого вектора key приведены в работе [1].

Определим расстояние между q -ми векторами как

$$D_V = d'_{key}(input, output) = \sum_{i=1}^n |input_i - output_i|, \quad (3)$$

где $input_i = i, \forall i \in \overline{1, n}; output_i \in num, \forall i, num \in \overline{1, n}$.

В таблице 1 приведён расчёт D_V для q -го вектора размерности $n = 3$.

Таблица 1 – Векторное расстояние для q -х векторов размерности 3

<i>input</i>	123	123	123	123	123	123	123	123
<i>key</i>	000	001	010	011	100	101	110	111
<i>output</i>	123	213	132	312	213	123	231	321
D_V	0	2	2	4	2	0	4	4

Введём понятие качества, определяющее, насколько хорошо ключ перемешивает, сглаживает q -й или случайный бинарный вектор. Пусть ключ key_i называется более качественным, чем ключ key_j , если $d_{key_i}(input, output) > d_{key_j}(input, output)$. Также будет определено понятие качества и для q -х векторов: key_i более качественный, чем key_j , если $d'_{key_i}(input, output) > d'_{key_j}(input, output)$.

На рисунке 1 приведён график зависимости D_H^* и D_V от всевозможных ключей для устройства размерности 3.

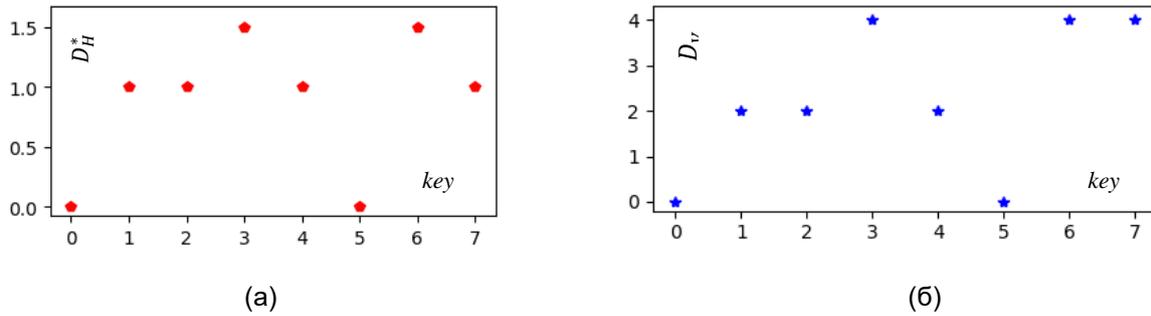


Рисунок 1 – Расстояние Хэмминга для бинарного вектора (а), векторное расстояние для q -го вектора (б), $n = 3$

Далее для большей иллюстративности будем рассматривать свойства устройства размерности $n = 4$. Пусть ключ j будет качественным, если $d_{key_j}(input, output) > (\max_{i=\overline{1,m}} d_{key_i}(input, output))/2$. Аналогично ключ j – качественный, когда $d'_{key_j}(input, output) > (\max_{i=\overline{1,m}} d'_{key_i}(input, output))/2$, тогда на рисунке 2 (график D_V масштабирован) наглядно отображено, что если ключ является качественным относительно D_V , то он также будет являться качественным относительно D_H^* .

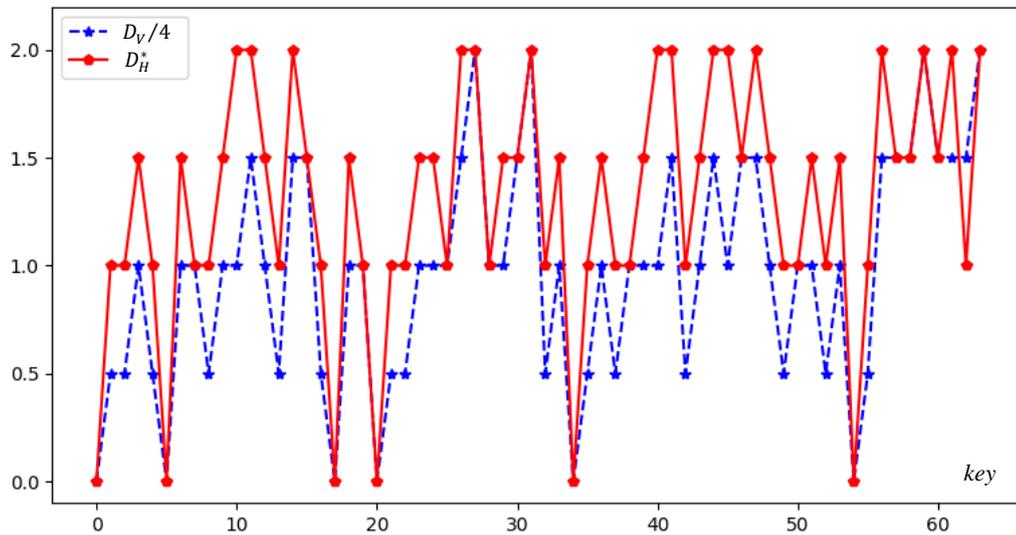


Рисунок 2 – Расстояние по Хэммингу для бинарного вектора D_H^* , векторное расстояние для q -го вектора, уменьшенное в 4 раза ($D_V/4$) для любого ключа устройства размерности $n = 4$

Причём если назвать каждое из значений 0, 0.5, 1, 1.5, 2 группой и в этом порядке присвоить им номера, то верно утверждение, что $|D_V/4 - D_H^*| \leq 1, \forall key \in K$. Это утверждение можно обобщить и на большие размерности. Тогда нормировочный коэффициент может быть рассчитан как максимум из векторных расстояний, разделённый на максимум из средних расстояний по Хэммингу, а константа в правой части неравенства может быть вычислена как $\max(d_{key_i}(input, output))/2, i = \overline{1,m}$ для всевозможных векторов $input$.

В результате анализа значений D_H^* и D_V , полученных на различных размерностях устройства, были сделаны следующие выводы:

- Средние значения расстояний по Хэммингу для бинарных векторов и значения векторного расстояния q -х векторов в зависимости от подаваемого ключа находятся в зависимости друг от друга.
- Для быстрой проверки качества ключа возможно рассчитать векторное расстояние D_V для определённого ключа. Если ключ окажется качественным, то его можно принимать за качественный и для бинарных векторов.

Список использованных источников:

1. Кохновский, С. И. Оценка ключей комбинационного устройства всевозможных перестановок [Электронный ресурс] / С. И. Кохновский, А. А. Иванюк // Информационные технологии и системы 2020 (ИТС 2020) : материалы междунар. науч. конф. – Минск : БГУИР, 2020. – 220 с. – С. 105–106. – Режим доступа : https://its.bsuir.by/m/12_130111_1_147692.pdf. – Дата доступа : 05.04.2021.

ОТСЛЕЖИВАНИЕ ОБЪЕКТОВ НА ВИДЕОЗАПИСИ ПРИ ПОМОЩИ МЕТОДОВ ГЛУБОКОГО ОБУЧЕНИЯ

Сидоренко К.А.

Белорусский государственный университет информатики и радиоэлектроники
г. Минск, Республика Беларусь

Калугина М.А. – канд. физ.-мат. наук, доцент

В докладе рассматриваются два алгоритма отслеживания объектов на видеозаписи, основанные на методах глубокого обучения. Описываются принципы реализации алгоритмов, производится их сравнение, выделяются достоинства и недостатки каждого алгоритма.

Задача отслеживания объектов на видео изучается и развивается уже не первый десяток лет. Необходимость в отслеживании объектов возникает, например, для решения следующих задач: отслеживание людей в толпе для обеспечения безопасности при проведении массовых мероприятий, учет количества посетителей в различных заведениях для подсчета важных статистик для бизнеса, отслеживание перемещения и поведения скота в фермерских хозяйствах, трекинг автомобилей в системах автоматического определения правонарушений и идентификации нарушителей. Несмотря на то, что в большинстве своем современные алгоритмы трекинга объектов обучены отслеживать преимущественно людей, транспортные средства и животных, можно обучить модель отслеживанию практически любых других объектов [1].

Необходимо понимать различия между задачами обнаружения (детектирования) и отслеживания объектов. В то время как алгоритмы обнаружения объектов лишь определяют цель на фото или видео, алгоритмы отслеживания связывают информацию, полученную с предыдущих кадров таким образом, чтобы не терять объект, а также при необходимости предсказывать траекторию его движения. Так, при использовании методов детектирования объектов для задачи трекинга, модель будет всякий раз терять цель, когда объект на некоторое время пропадает из видимости (например, автомобиль проезжает через мост). Однако, алгоритмы отслеживания включают в себя задачу определения объектов в каждом кадре, поэтому этап детектирования играет очень важную роль.

В работе рассматриваются два популярных алгоритма трекинга – SORT [2] и DeepSORT [3]. DeepSORT по своей сути является модификацией алгоритма SORT. SORT включает в себя следующие ключевые концепты:

1. *Обнаружение объекта.* На данном этапе происходит определение объектов в текущем кадре. В результате получаем положение и размер прямоугольника (ограничивающая рамка), который содержит в себе обнаруженный объект. Надо учитывать, что в задаче отслеживания объектов на видеозаписи критическую роль играет скорость работы алгоритмов, поэтому на каждой стадии необходимо выбирать наиболее быстродействующие алгоритмы. В качестве алгоритма детектирования будем использовать YOLO [4]. Данный алгоритм отличается высокой скоростью работы (последние версии алгоритма обрабатывают изображения со скоростью 70 кадров в секунду), что позволяет применять его и для видеозаписей.

2. *Оценка модели.* Задача данного этапа заключается в создании представления и модели движения объекта, используемых для передачи цели в следующий кадр. Здесь мы аппроксимируем межкадровые перемещения каждого объекта с помощью линейной модели постоянной скорости.

3. *Связывание данных.* На этом этапе выполняется связывание обнаруженных объектов в текущем кадре с уже захваченными целями. Для ограничивающей рамки каждой захваченной цели выбирается наиболее близкая ограничивающая рамка обнаруженного объекта в текущем кадре.

4. *Создание и удаление уникального идентификатора объекта отслеживания.* Когда в кадре появляется новый, ранее не обнаруженный объект, ему присваивается уникальный идентификатор, и объект начинает отслеживаться. Когда отслеживаемый объект пропадает из кадра, его отслеживание заканчивается. Однако, если тот самый объект вновь попадает в кадр, ему присвоится новый идентификатор. Данный факт может привести к некоторым неудобствам, например к ошибочному расчету статистики.

Обозначенную проблему решает алгоритм DeepSORT. Алгоритм реализует все описанные концепции алгоритма SORT, но дополнительно использует нейронную сеть для реидентификации объектов [5]. Такая сеть создает так называемый хеш-код для каждого отслеживаемого объекта, по которому его можно заново распознать. Этот хеш-код запоминается и в случае, если объект покидает зону видимости, а через некоторое время возвращается в кадр, он рассматривается как уже ранее отслеживаемый объект со старым идентификатором. Чтобы реидентификационная сеть работала хорошо, ее обучают по уникальным признакам объектов. Например, для людей это может быть одежда или лицо, если оно отчетливо видно на видео, для машин – марка автомобиля. В то же время, нейронная сеть реидентификации требует тонкой настройки и качественного обучения, что влечет за собой дополнительные расходы времени и вычислительных мощностей. Для сравнения,

для отслеживания определенных объектов в алгоритме SORT достаточно обучить нейронную сеть обнаружения объектов.

В качестве демонстрации и сравнения работы двух алгоритмов проверим их на одной и той же тестовой видеозаписи. На короткой (30 секунд) видеозаписи пешеход движется по улице и на одно мгновение скрывается за кадр. Проследим, как алгоритмы справятся с поставленной задачей. Сперва запустим алгоритм SORT.



Рисунок 1 – Пример работы алгоритма SORT

На рисунке 1 видно, что после того, как пешеход на некоторое время пропадает из обзора камеры, алгоритм SORT определяет этого пешехода как два разных, что можно понять по отличию цвета рамок. Теперь запустим алгоритм DeepSORT на той же видеозаписи.



Рисунок 2 – Пример работы алгоритма DeepSORT

Как видно из рисунка 2, DeepSORT хорошо справился с проблемой кратковременного исчезновения пешехода из кадра. Однако стоит отметить, что по времени работы DeepSORT несколько уступает своему «предку»: алгоритм SORT обработал видео на 5 секунд быстрее.

В итоге проделанной работы были изучены два алгоритма трекинга объектов – SORT и DeepSORT. Оба алгоритма обладают своим набором преимуществ, поэтому выбор конкретного алгоритма зависит от поставленной задачи.

Список использованных источников:

1. How to train to detect your custom objects [Электронный ресурс]. – Режим доступа : <https://github.com/AlexeyAB/darknet#how-to-train-to-detect-your-custom-objects>.
2. Simple Online and Realtime Tracking / A. Bewley [et al.] // arXiv:1602.00763v2 [cs.CV], 2017.
3. Simple Online and Realtime Tracking with a Deep Association Metric / N. Wojke [et al.] // arXiv: 1703.07402v1 [cs.CV], 2017.
4. You Only Look Once: Unified, Real-Time Object Detection / J. Redmon [et al.] // arXiv:1506.02640v5 [cs.CV], 2016.
5. Pedestrian Attribute Recognition: A Survey / X. Wang [et al.] // arXiv: 1901.07474v1 [cs.CV], 2019.

АРИФМЕТИКА БЕЗ СКОБОК

Марат О.И., Пешко А.А., Селивестров В.А., Чепиков П.А.

Белорусский государственный университет информатики и радиоэлектроники
г. Минск, Республика Беларусь

Стройникова Е.Д. – ст. преподаватель

В докладе рассмотрены история создания, общие сведения, суть алгоритма, преобразование из инфиксной нотации, практические реализации и основные характеристики арифметики без скобок, или обратной польской записи. Авторами была разработана программа на языке C++, которая позволяет делать вычисления с использованием обратной польской записи.

Сложные вычислительные задачи обычно требуют больших объемов вычислений, поэтому к программистам предъявлялось требование: приблизить форму записи математических выражений в коде программы к естественному языку математики. В результате наибольшее распространение получил метод трансляции с помощью обратной польской записи, которую предложил польский математик Я. Лукасевич.

История создания

Обратная польская запись (ОПЗ), или обратная польская нотация (ОПН), была разработана австралийским философом и специалистом в области теории вычислительных машин Чарльзом Хэмблином в середине 1950-х годов на основе польской нотации, которая была предложена в 1920 г. польским математиком Яном Лукасевичем. Работа Хэмблина была представлена на конференции в июне 1957 г. и опубликована в 1957 г. и в 1962 г. Первыми компьютерами, поддерживающими обратную польскую нотацию, были KDF9 от English Electric Company, который был анонсирован в 1960 г. и выпущен (появился в продаже) в 1963 г., и американский Burroughs B5000: анонсирован в 1961 г., выпущен в том же 1963 г. Один из проектировщиков B5000, Р. С. Бартон, позже написал, что разработал обратную польскую запись независимо от Хэмблина примерно в 1958 г. в процессе чтения книги по символической логике, до того как познакомился с работой Хэмблина. ОПН применялась в советском инженерном калькуляторе БЗ-19М (совместная разработка с ГДР), выпущенном в 1976 г. Все выпускаемые в СССР вплоть до конца 1980-х годов программируемые микрокалькуляторы, за исключением «Электроника МК-85», использовали ОПН – она проще реализовывалась и позволяла обойтись в программировании вычислений меньшим числом команд по сравнению с обычной алгебраической нотацией, а количество программной памяти в этих моделях всегда было критическим ресурсом (не более 105 ячеек при том, что команда занимала 1–2 ячейки). ОПН используется в современных российских программируемых калькуляторах «Электроника МК-152» и «ЭЛЕКТРОНИКА МК-161», что обеспечивает их совместимость с программами, разработанными для советских калькуляторов.

Общие сведения

Обратная польская запись представляет собой выражение, записанное в постфиксной форме, без скобок, по специальным правилам. Получение ОПЗ реализуется с использованием структур в виде стека и дерева. Получение ОПЗ с использованием стека может осуществляться очень просто – на основе алгоритма, предложенного Э. Дейкстрой, который ввел понятие стекового приоритета операций (таблица 1).

Таблица 1 – Стековый приоритет операций ОПЗ

Операция	Приоритет
(0
)	1
+, –	2
×, /	3
xx	4

Суть алгоритма

Исходное выражение, записанное в виде строки символов S, просматривается слева направо. Операнды переписываются в выходную строку В, операции обрабатываются с использованием стека, который первоначально пуст, на основе следующих правил:

1. Если в строке S встретился операнд, то его помещаем в строку В.
2. Если в строке S встретилась открывающая скобка, то ее помещаем в стек.
3. Если в строке S встретилась закрывающая скобка, то извлекаем из стека и записываем в строку В все операции до "(" и саму "(" , скобку ")" также извлекаем из стека; обе скобки игнорируются.

4. Если в строке S встретилась операция X, то выталкиваем из стека все операции, приоритет которых ниже X, после чего саму операцию X записываем в стек.

5. При достижении конца строки S анализируем стек и, если он не пуст, извлекаем и переписываем его элементы в выходную строку V.

Вычисление выражения, записанного в ОПЗ, может проводиться путем однократного просмотра, что является весьма удобным при генерации объектного кода программ. Отличительной особенностью обратной польской нотации является то, что все аргументы (или операнды) расположены перед операцией (оператором). Это позволяет избавиться от необходимости использования скобок. Например, выражение $3 \times (4 + 7)$ будет выглядеть как $3\ 4\ 7\ +\ \times$ (можно записать компактней: $4\ 7\ +\ 3\ \times$). Вычисления в обратной польской нотации основаны на стеке, это означает, что операнды достаются из стека и результат вычисления кладется обратно в стек. На первый взгляд это может выглядеть непонятно, однако, обратная польская нотация имеет важное преимущество – она невероятно проста для разбора, поэтому ее реализация на компьютере требует мало ресурсов и выполняется быстро.

Преобразование из инфиксной нотации

Эдсгер Дейкстра изобрел алгоритм для преобразования выражений из инфиксной нотации в ОПН. Алгоритм получил название «сортировочная станция» за сходство его операций с происходящим на железнодорожных сортировочных станциях. Инфиксная нотация – это форма математических записей, которую использует большинство людей (например, $3 + 4$, или $3 + 4 \times (2 - 1)$). Как и алгоритм вычисления ОПН, алгоритм сортировочной станции основан на стеке. В преобразовании участвуют две текстовых переменных: входная и выходная строки. В процессе преобразования используется стек, хранящий еще не добавленные к выходной строке операторы. Преобразующая программа читает входную строку последовательно символ за символом (символ – это не обязательно буква), выполняет на каждом шаге некоторые действия в зависимости от того, какой символ был прочитан.

Практические реализации

В качестве практического применения данной методики можно привести организацию байт-кода конфигураций прикладных решений системы 1С: Предприятие. Официального подтверждения компания 1С не дает, но использующие данную систему программисты на специализированных форумах приводят доказательства и алгоритмы, позволяющие декомпилировать исходные тексты.

Основные характеристики:

1. Вычисления выполняются слева направо.
2. Операнды предшествуют своему оператору. При выполнении операции они удаляются.
3. Когда операция выполнена, результат сам становится операндом (для следующих операторов).
4. В ОПН нет скобок и нет приоритетов операций.
5. Большинство вычислений в ОПН записываются короче, нежели в инфиксной нотации.

Программа

Авторами была разработана программа на языке C++, которая позволяет делать вычисления с использованием ОПН по принципу ввода данных и выводу результата на экран.

Список использованных источников:

1. Компиляторы: принципы, технологии и инструменты / А. В. Ахо [и др.] ; пер. с англ. – 2-е изд. – М. : Вильямс, 2008. – 1184 с.
2. Пратт, Т. Языки программирования: разработка и реализация / Т. Пратт, М. Зелковиц ; пер. с англ. – 4-е изд. – СПб. и др. : Питер, 2002. – 688 с.
3. Обратная польская запись [Электронный ресурс]. – 2001. – Режим доступа : http://www.univer.omsk.ru/students/m84/docs/pol_zap.html.
4. Обратная польская нотация [Электронный ресурс]. – 2021. – Режим доступа : <http://alqolist.manual.ru/math/misc/revpn.php>.
5. martiros: Обратная польская запись [Электронный ресурс]. – 2020. – Режим доступа : <http://martiros24.blogspot.com/2016/06/blog-post.html>.

ТРАНСФОРМАЦИИ ПРОСТРАНСТВА-ВРЕМЕНИ

Резунов М.В., Минько Д.А.

Белорусский государственный университет информатики и радиоэлектроники
г. Минск, Республика Беларусь

Сиротко С.И. – канд. физ.-мат. наук, доцент

В докладе рассматриваются области пространства-времени и трансформации между областями пространства-времени, связанными с соответствующими системами отсчета. Внимание также уделено проблеме категорий топологических пространств, преобразований друг в друга и самих в себя.

История физики как самостоятельной науки начинается с Галилея, который постулировал, что математика – это именно тот «язык, на котором написана книга природы». Из этого фундаментального принципа следует, что физические понятия существуют в соответствии с математическими понятиями.

Теоретический фундамент классической физики основал Ньютон в XVII в., создав для физики того времени новые математические понятия, называемые сегодня математическим анализом. Физическое пространство ньютоновской физики с точки зрения математики является евклидовым геометрическим пространством. Более двух тысяч лет считались очевидными аксиомы евклидовой геометрии.

На рубеже XIX–XX вв. в физике и математике возник кризис. В физике кризис был связан с созданием двух фундаментальных теорий: общей теории относительности и квантовой теории. В математике аналогичный кризис связывают с развитием неевклидовой геометрии [1].

В современной физике термин «пространство-время» как континуум появился на основании анализа теории относительности Минковским. Данный термин связан с представлением «четырёхмерного» многообразия картины мира, неразрывной взаимосвязанности пространства и времени [2].

В докладе рассматривается модель с областями пространства-времени с разными свойствами и законами и переходами между ними. Переходы представлены в виде топологических трансформаций – обратимых непрерывных преобразований, соответствий свойств и законов одних областей другим [3].

Теоретико-множественное определение топологического пространства формулируется как множество с выделенной системой подмножеств, которая удовлетворяет соответствующим аксиомам. При таком подходе топологические пространства и преобразования рассматриваются отдельно. Теория категорий представляет альтернативный подход, в рамках которого объекты в совокупности с преобразованиями объектов друг в друга и самих в себя образуют категорию. Категория топологических пространств состоит из всех топологических пространств и всех непрерывных обратимых преобразований пространств друг в друга и самих в себя [1, 3].

В докладе рассматривается пространство-время как категория, в которой область пространства-времени и связанная с ним система отсчета является объектом, а трансформация – морфизмом. Основное внимание уделено именно трансформациям [4].

Всевозможные трансформации можно разделить на три вида: пространственные, временные и пространственно-временные. Временные трансформации могут быть описаны с помощью отображения переменной времени: $T_1 \rightarrow T_2$. В качестве примера можно привести замедление в k раз, нелинейное изменение скорости течения времени, остановку времени, течение времени вспять.

Пространственные трансформации связаны со всевозможными обратимыми преобразованиями пространства. К ним относятся три простейшие пространственные трансформации, соответствующие простым матричным преобразованиям: смещение, поворот и масштабирование. В общем случае пространственные преобразования могут быть описаны с помощью биекций трех переменных пространства:

$$f(x, y, z): X \leftrightarrow Y.$$

Пространственно-временные трансформации отражают взаимосвязь и взаимозависимость пространства и времени и могут быть заданы всевозможными обратимыми преобразованиями. Пространственно-временные преобразования являются самыми общими преобразованиями и могут быть представлены как биекции четырех переменных пространства и времени:

$$f(x, y, z, t): X \leftrightarrow Y.$$

Основной целью данной научной работы является изучение трансформаций областей пространства-времени, а также рассмотрение с точки зрения категорий. Ставится задача применения результатов исследования на практике, разработка теоретической базы и алгоритма для трансформаций областей пространства-времени виртуального мира, создание движка и на его основе – приложения, реализующего алгоритм.

В практической части исследования разработан движок и игровое приложение. Движок реализован на языке программирования C++ с использованием Qt Framework для оконной оболочки. Движок использует прямые вызовы OpenGL, что предоставляет широкий спектр возможностей: эффективное управление ресурсами и производительностью, создание программируемого конвейера обработки графической информации с помощью вершинных и фрагментных шейдеров.

Движок может быть разделен на две составляющие: основную часть, разработанную на C++, и графическую часть, разработанную на OpenGL Shading Language (GLSL). Основная часть реализует следующие задачи:

- 1) ядро движка;
- 2) передачу данных в графическую часть на выполнение шейдеров;
- 3) взаимодействие с файловой системой, управление ресурсами;
- 4) физику игрового мира.

Графическая часть в свою очередь отвечает за:

- 1) формирование конвейера обработки графической информации;
- 2) отображение объектов в преобразованном пространстве;
- 3) вывод изображения на экран.

На основе игрового движка создано интерактивное приложение, реализующее алгоритм преобразований пространства-времени. Реализованы простейшие пространственные трансформации: смещение, поворот и масштабирование, а также более сложное – спиралевидное вращение. Спиралевидное вращение представляет собой вращение вдоль оси, при этом угол вращения прямо пропорционален расстоянию от наблюдателя до точки пространства.

На рисунке 1 представлено положение наблюдателя, который находится в исходном, не трансформированном пространстве. Затем, при перемещении наблюдателя вперед, пространство постепенно расширяется по оси X, что продемонстрировано на рисунке 2. При дальнейшем движении наблюдателя вперед пространство подвергается трансформации спиралевидного вращения, что видно на рисунке 3. На рисунке 4 показан взгляд из центра сцены в сторону, обратную взгляду на предыдущих рисунках 1–3.



Рисунок 1 – Исходное пространство



Рисунок 2 – Растянутое пространство



Рисунок 3 – Спиралевидная трансформация

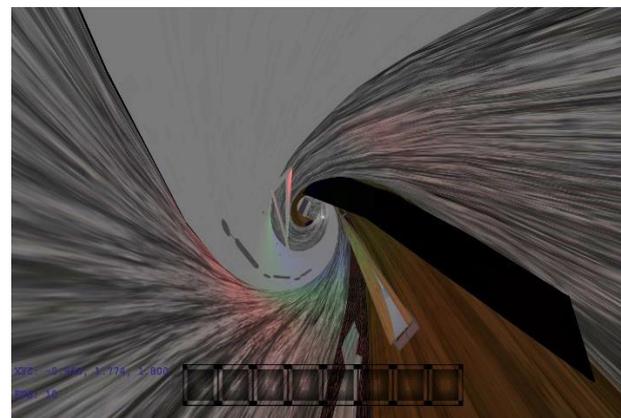


Рисунок 4 – Взгляд в обратную сторону

Список использованных источников:

1. Родин, А. В. Теория категорий и поиски новых математических оснований физики / А. В. Родин // Вопросы философии. – 2010. – №7. – С. 67–81. URL: http://vphil.ru/index.php?option=com_content&task=view&id=164.
2. Андреев, В. М. Проблема континуума «Пространство – время» / В. М. Андреев // Ученые записки Таврического национального университета им. В. И. Вернадского. Серия «Философия. Социология». – 2008. – Т. 21 (60), №2. – С. 52–60. URL: <https://cyberleninka.ru/article/n/problema-kontinuuma-prostranstvo-vremya>.
3. Монастырский, М. И. Бернхард Риман. Топология. Физика / М. И. Монастырский. – М. : Янус-К, 1999. – 184 с.

4. Кондратьев, Г. В. Возможные применения теории категорий в информационных науках / Г. В. Кондратьев // Труды НГТУ им. П. Е. Алексеева. – 2013. – №2 (99). С. 339–345. URL: <https://cyberleninka.ru/article/n/vozmozhnye-primeneniya-teorii-kategoriy-v-informatsionnyh-naukah>.

ОНТОЛОГИИ И ЯЗЫКИ ДЛЯ ПРЕДСТАВЛЕНИЯ МАТЕМАТИЧЕСКИХ ЗНАНИЙ В SEMANTIC WEB

Стаселько И.Д., Позняков Т.Д., Протасов А.П.

*Белорусский государственный университет информатики и радиоэлектроники
г. Минск, Республика Беларусь*

*Анисимов В.Я. – канд. физ.-мат. наук, доцент
Теслюк В.Н. – канд. мат. наук, доцент*

Предоставлена информация об использовании математических знаний совместно с технологиями Web 2.0 и Semantic Web.

Ключевые слова: математические знания, семантические технологии, искусственный интеллект, семантические сети, Web 2.0.

Обзор состояния математики в сети Интернет должен учитывать веб-сайты, которые находятся в повседневном употреблении: реферативные службы, такие как Math Zentralblatt и MathSciNet, arXiv – предварительный сервер печати, библиотеки формализованного и проверенного алгоритмом математического содержания, такие как Mizar Mathematical Library (MML). Эти сайты облегчили доступ к математическому материалу. Однако они предлагают ограниченный уровень взаимодействия и не облегчают пользование, и средства автоматического восстановления, использования и адаптивного представления знания через автоматизированных агентов ограничены.

Относительно сети Интернет в целом эта проблема и проблемы Семантической паутины (Semantic Web) были решены приложениями Web 2.0. В данной работе рассматриваются проблемы, связанные с получением и обработкой математических знаний в сети Интернет, и предложения по их преодолению с использованием новой комбинации Web 2.0 и Semantic Web. После того, как недавно математики освоили технологию Web 2.0, возросло число математиков, которые начали использовать данную технологию для того, чтобы совместно развивать новые идеи, а также использовать как новый поток для публикации полученных знаний. Типичные представители указанных технологий – блог исследований и энциклопедия Wiki. Такие блоги отражают традиционную математическую практику публикации кратких обзоров ранее изданного материала.

Исследователи также считают блоги полезными для того, чтобы собрать раннюю связь между предварительными результатами, тогда как традиционные государственные публикации, предназначенные для экспертной оценки, представляют зрелые идеи, скрывая альтернативы, которые когда-то рассмотрели и затем от них отказались, – такой метод работы сильно препятствует сотрудничеству между малочисленными исследовательскими группами. Успешное сотрудничество среди математиков, не знающих друг друга прежде, началось в блогах и имело результатом последующие совместные статьи. По сравнению с блогами исследования форум MathOverflow, где пользователи могут отправить свои проблемы и давать решения проблем других пользователей, предлагает больше помощи с меньшими рисками. Основой репутации этого форума является то, что он действует как качественное моделирование традиционных научных процессов публикации и экспертной оценки. Для развития идей, появившихся из обсуждений блога, или для создания постоянных, коротких, связанных описаний тем Wiki считают наиболее подходящей.

Также следует упомянуть сервис Tricky, который является хранилищем общих математических методов Wiki, работающий на Web 2.0. Wiki, которая собирает существующие математические знания для образовательных и ознакомительных целей, известна более широко. PlanetMath является математической энциклопедией и насчитывает более, чем 8000 записей на время данных исследований. Wikipedia, имеющая 15 миллионов статей на более чем 250 языках, также затрагивает математику. Предназначаясь для широкой аудитории, она опускает большинство формальных доказательств, но включает чистое математическое знание в более широкий контекст, а также включает, например, историю математики, биографию математиков и информацию об областях ее применения. Отсутствие доказательств частично компенсировано благодаря технически подобной ProofWiki или PlanetMath, содержащими более 2500 доказательств.

Наконец, Connexions, управляемый более традиционной системой работы с контентом, является открытым веб-хранилищем, специализированным на программном обеспечении учебного курса. Connexions продвигает более 17000 небольших модулей курса многократного использования его стандарту свободного содержания так, чтобы не только автор, но и другие пользователи могли свободно объединить их в сборники, такие как примечания к определенному курсу, приблизительно

4000 от математики и статистики и приблизительно 6000 от науки и техники. Разработчики установленных баз знаний также начинают использовать технологию Web 2.0 – например, недавно разработанный прототипный макет Wiki frontend для Mizar Mathematical Library (MML) – это крупная библиотека формализованного и проверенного машиной математического контента. Wiki намеревается осуществить общую поддержку и улучшение MML.

Сайты на Web 2.0 требуют крупных инвестиций рабочей силы для компилирования сборников знаний. Повторного использования знаний с применением машинного интеллекта, например из других сетевых коллекций знаний, не происходит. Различные базы знаний технически отделены друг от друга из-за форматов документа, которые подходят для одиночного представления данных, такие как XHTML с LATEX формулами.

Единственным способом присоединяться к другим базам знаний является гиперссылка. Методы доказательств, собранные в Tricky, не могут быть автоматически применены к проблемам разработки в блоге исследования, поскольку ни в одном из них недостаточно формул. С другой стороны, сообщество Polymath не имеет достаточно автоматизированных инструментов проверки, но оно полагается на краудсорсинговый принцип. Также плохо поддерживается поиск математической информации, являющийся предпосылкой для поиска знаний к применению и повторному использованию.

Например, Википедия указывает теорему Пифагора и регистрирует ее в категории «Статьи с доказательством» и «Математические теоремы». Представление формул в формате LATEX не поддерживает поиск функциональных структур.

Несмотря на то, что Википедия не может искать формулы, поиск эквивалентных выражений не привел бы к теореме, если они явно не указаны в статье. Если смотреть со стороны классификации, это в любом случае непонятно для машины (хотя, очень вероятно, понятно для человека), содержит ли статья доказательство конкретной теоремы и правильно ли оно.

Следовательно, сотрудники Polymath должны были искать предыдущие публикации и опровержения с доказательством и ключевым словом. Формализованные хранилища, такие как MML, специализированные поисковые системы, также поддерживают внутреннее повторное использование знаний, формулируя новые математические понятия для уже существующих и доказывая новые теоремы, которые уже были доказаны. Также находятся в дефиците математические сайты на Web 2.0 с автоматизированными услугами по рассуждению и вычислению. Интерактивное вычисление доступно в математических системах электронного обучения, таких как ActiveMath или MathDox, где авторы документа прежде, чем издать в цифровых библиотеках общего назначения и в среде сообщества, достаточно сформулировали основную математику в отдельных инструментах редактирования. Мэшапы, которые являлись движущей силой развития Web 2.0, сейчас едва используются для математических задач.

В начале 2000-х, когда XML все больше использовался для математики, особенно для формул, которые были первыми стандартными блоками Семантической паутины, такие как RDFS, приблизились к стандартизации. Эти события вызвали интерес к появившемуся междисциплинарному сообществу математического управления знаниями, состоящему из программистов, опытных математиков и цифровых исследователей библиотек, цель которых состоит в том, чтобы «развивать новые и лучшие способы управления математическими знаниями, используя сложные программные средства» [1]. Или, более конкретно, «служить математикам, ученым и инженерам, которые производят и используют математическое знание; педагогам и студентам, которые преподают и изучают математику; издателям, которые предлагают математические учебники и распространяют новые математические результаты; и библиотекарям и математикам, которые каталогизируют и организуют математическое знание» [1].

Список использованных источников:

1. Marchiori, M. *The Mathematical Semantic Web* / M. Marchiori // *Mathematical Knowledge Management : Second International Conference, Bertinoro, Italy, February 16–18, 2003* / Springer-Verlag. – Berlin, Heidelberg, 2003. – P. 216–223.

ЧЕГО МОЖНО ОЖИДАТЬ ОТ SEMANTIC WEB И ДРУГИХ СЕМАНТИЧЕСКИХ ТЕХНОЛОГИЙ В 2021 ГОДУ

Стаселько И.Д., Позняков Т.Д., Протасов А.П.

*Белорусский государственный университет информатики и радиоэлектроники
г. Минск, Республика Беларусь*

Летохо А.С. – ассистент, Стаховский С.М. – ст. преподаватель

Предоставлена информация о возможном развитии технологии Semantic Web и ее интегрировании в различные отрасли.
Ключевые слова: графы знаний, семантические технологии, искусственный интеллект, семантические сети.

Графы знаний уже получили большое внимание в качестве основы для машинного обучения, глубокого обучения и бизнес-кейсов использования искусственного интеллекта. Можно ожидать, что эта тенденция будет продолжаться. Так же улучшается понимание того, что графы знаний-онтологии обеспечивают «зонтичное наложение для перекрестных переходов через изолированные информационные ресурсы». Семантические технологические подходы обеспечивают мощную основу для машинного обучения и искусственного интеллекта. При контролируемом обучении приходится от 60 до 80 процентов усилий на очистку входных данных и последующее обозначение их в качестве основы для обучения. Мы можем получить эти чистые метки в соответствующих учебных наборах «бесплатно» с помощью семантических технологий и графиков знаний.

Среди семантических технологий искусственного интеллекта предприятий в текущем году будут и те, которые касаются здравоохранения. За последние два года некоторые технологии начали использоваться в производстве. В 2021 году мы увидим рост числа приложений искусственного интеллекта, которые помогут спасти жизни людей, предоставляя ранние предупреждающие признаки надвигающихся заболеваний. Некоторые заболевания будут предсказаны на годы вперед, используя генетические данные пациентов, чтобы понять будущие биологические проблемы, такие как вероятность раковых мутаций, и начать профилактическую терапию до того, как болезнь возьмет верх.

Все больше систем ответов на вопросы, таких как доменные чат-боты, будут полагаться на семантическую базу знаний для проведения рассуждений. Для этого требуется более интеллектуальная система контроля качества, чтобы выйти за рамки вопроса об одном прыжке на графе знаний и зрелости графовой технологии реального времени. eBay ShopBot, голосовая активируемая система для покупок из огромного каталога eBay (Alexa, Google Home), общаясь с агентом, благодаря обработке естественного языка и огромной базе знаний, может отличить потребности одного клиента в сумке для спортивного снаряжения от потребностей другого клиента в сумке для ноутбука.

Также предполагается использование графов знаний, которые фокусируются на беглой обработке естественного языка для конкретной области. Например, пользователь хочет сделать ставку на футбольный матч через игровое приложение, но есть вероятность, что компьютер распознает это как: «Я хочу поставить леди на канониров для побелки в субботу днем». Общая система нейролингвистического программирования может быть в состоянии расшифровать это, но для декодирования требуется специализированный граф.

Сочетание искусственного интеллекта и семантических графов знаний будет применяться для преобразования работ ученых, технологов и политиков в интерактивную систему ответов, которая использует реальный голос человека для ответа на вопросы. Цифровые персонажи искусственного интеллекта будут динамически связывать информацию из различных источников – таких, как книги, научные статьи, заметки и интервью в средствах массовой информации – и превращать разрозненную информацию в систему знаний, с которой люди могут взаимодействовать в цифровом виде. Эти цифровые персонажи искусственного интеллекта также могут быть использованы, пока человек еще жив, чтобы расширить доступность своих знаний.

Есть еще один пример использования графов семантических знаний, который не получил такого большого внимания ранее, как, вероятно, будет в 2021 году. В игровой индустрии, например в игре Phasmophobia, где игра распознает речь человека для прохождения сюжета. Пример использования семантических сетей заключается в том, что человек способен генерировать объекты внутриигрового мира своей речью. Например, при команде «Создай призрак», игра анализирует речь, после чего создает нужный объект. На данный момент это доступно только на английском языке. То есть использование графов знаний будет существенно расширено в различных областях, включая обнаружение и исследование взаимосвязей данных, семантическую совместимость и особенно управление данными.

Действительно, общеорганизационные графы знаний, которые поддерживают модели коллекций сущностей и их семантических типов, свойств и взаимосвязей, могут сыграть большую роль в решении проблем в ключевых областях предприятий.

Будучи гибкими, эволюционирующими, семантическими и интеллектуальными, графики знаний поддерживают комплексное (сверху вниз, снизу вверх и в середине) управление данными, которое описывают как деятельность по управлению активами, ориентированную на жизненный цикл.

Среда управления данными должна представлять активы и их роль на предприятии с использованием открытого, расширяемого и «умного» подхода. Графики знаний являются основополагающими для управления данными, поскольку они преобразуют разнообразные данные в каталог, фиксируя их технический и деловой контекст и значение через связи между всеми активами в экосистеме предприятия. «Графики знаний – это идеальная, и, возможно, единственная жизнеспособная основа для соединения корпоративных хранилищ метаданных» [1].

Дебют Amazon сервиса графических баз данных AWS Neptune в мае 2020 года привлек внимание к графическим базам данных для создания интерактивных графических приложений. Пользователи бета-теста использовали его, чтобы построить все, начиная от графики до социальных сетей с рекомендованным движком. Tom Sawyer Perspectives был выбран командой Amazon Neptune

для поддержки сервиса Graph Database с интегрированным решением для создания приложений для визуализации и анализа данных и соединений. Большинство приложений для визуализации графов показывают сетевые диаграммы только в двух измерениях, но неестественно манипулировать графами на плоском экране компьютера в 2D. Современная виртуальная реальность добавит, по крайней мере, два измерения к визуализации графов, что создаст более естественный способ манипулировать сложными графами, включив больше глубины и временного промежутка для понимания информации в перспективе времени.

Графические базы данных сильно выделяются, начиная с 2020 года. На семинаре W3C по веб-стандартизации графических данных обсуждались такие вопросы, как введение стандартов для обмена информацией между мирами RDF и графов свойств.

В настоящее время существует целевая группа из нескольких отраслевых поставщиков графовых баз данных, которые объединяются, работая над стандартным языком графовых запросов. В состав таких групп входят Neo4j, Oracle, TigerGraph и другие.

Похоже, что все больше и больше отраслей промышленности применяют семантические технологии, часто используя графы знаний. Например, компания для заказа такси в любую точку Вашего города Uber использует графики знаний для своих приложений. Можно предположить, что это помогает лучше понять своих клиентов и то, что они делают – другими словами, пользовательский контекст для повышения уровня удовлетворения своих клиентов.

По словам генерального директора и главного консультанта компании Working Ontologist LLC, понятие графа знаний в банке превращается из конкурентного преимущества в обязательное.

Если рассматривать сельское хозяйство как новое пространство для использования семантических технологий, в мире существует множество проектов, которые работают над этим, и все они объединены AGROVOC (контролируемый словарь, охватывающий все области деятельности и знаний, попадающие в сферу интересов Продовольственной и сельскохозяйственной организации Объединенных Наций).

В заключение можно сказать, что семантические сети – неотъемлемая часть в дальнейшей разработке и улучшении искусственного интеллекта и систем анализа данных, что в перспективе приведет к эволюции сфер, которые затрагивают использование высоких технологий.

Список использованных источников:

1. Allemang, D. *What Is the Semantic Web?* / D. Allemang, J. Hendler // *Semantic Web for the Working Ontologist: Effective Modeling in RDFS and OWL* / D. Allemang, J. Hendler. – 2nd Edition. – Morgan Kaufmann Publishers, 2011. – Chapter 1. – P. 1–12.

СРАВНИТЕЛЬНЫЙ АНАЛИЗ ИНТЕРПОЛЯЦИОННЫХ МНОГОЧЛЕНОВ ЛАГРАНЖА И НЬЮТОНА

Макарич Д.А., Богомаз К.С.

*Белорусский государственный университет информатики и радиоэлектроники
г. Минск, Республика Беларусь*

Анисимов В.Я. – канд. физ.-мат. наук, доцент

Целью данной научной работы является рассмотрение поведения многочленов Лагранжа и Ньютона от двух переменных и многих переменных, проведение их сравнительного анализа, а также демонстрация использования интерполяции в современных технологиях.

Во многих областях науки выполняются измерения. Если эти измерения нарисовать на графике, они будут показаны как простые несвязанные точки. Такие данные называются дискретными. Обработка и анализ данных станут проще, если их можно описать с помощью непрерывной функции или линии, соединяющей дискретные точки данных вместе. Для этого необходимо создать линию между двумя точками. Этот процесс называется интерполяцией. В данной работе мы познакомимся с интерполяцией на примере многочленов Лагранжа и Ньютона.

Интерполяция как таковая активно вошла в нашу жизнь вместе с появлением первых сенсорных экранов. Данная математическая теория используется для минимизации потери качества при увеличении картинки на экране смартфона/цифровой камеры/ноутбука и т. д. Обратимся к определению: «Интерполяция камеры – это искусственное увеличение разрешения изображения. Именно изображения, а не размера матрицы. То есть это специальное программное обеспечение, благодаря которому снимок с разрешением 8 Мп интерполируется до 13 Мп или больше (или меньше)». Если рассматривать это с технической точки зрения, то при увеличении изображения мы «скругляем» неровности и делаем контуры изображения более плавными.

Многочлен Лагранжа удобен, когда значения функций меняются, а узлы интерполяции неизменны. Число арифметических операций, необходимых для построения многочлена Лагранжа, пропорционально n^2 и является наименьшим для всех форм записи.

У многочлена Лагранжа есть один главный недостаток: при добавлении нового узла нам придется перестраивать весь многочлен. Эту проблему призван решать многочлен Ньютона.

В практической части проведен анализ на примере многочленов Лагранжа и Ньютона от одной и двух переменных. С помощью программного продукта Maple построены графики и выведены формулы многочленов. На языке Python реализована программа для подсчета итогового вида многочлена и построения графика по заданным точкам.

Список использованных источников:

1. Тараник, В. А. Применение «Интерполяционного многочлена Лагранжа» для функций со многими переменными / В. А. Тараник. – Красноярск : СибГТУ, 2008. – 64 с.
2. Задорожный, А. Г. Построение сплайнов с использованием библиотеки OpenGL : учеб. пособие / А. Г. Задорожный, Д. С. Киселев. – Новосибирск : НГТУ, 2019. – 88 с.
3. A Simple Expression for Multivariate Lagrange Interpolation [Электронный ресурс]. – 2007. – Режим доступа : http://evoq-eval.siam.org/Portals/0/Publications/SIURO/Vol1_Issue1/A_Simple_Expression_for_Multivariate.pdf?ver=2018-03-30-130233-050.
4. Erik Meijering PhD FIEEE. A Chronology of Interpolation: From Ancient Astronomy to Modern Signal and Image Processing [Электронный ресурс]. – 2002. – Режим доступа : <https://imagescience.org/meijering/research/chronology/>.

РОБАСТНЫЕ АЛГОРИТМЫ С РАЗРЕЖИВАНИЕМ ДЛЯ ВЕРОЯТНОСТНОГО ТЕМАТИЧЕСКОГО МОДЕЛИРОВАНИЯ

Изгачёв И.Ю., Аниховский М.А.

*Белорусский государственный университет информатики и радиоэлектроники
г. Минск, Республика Беларусь*

Стержанов М.В. – канд. техн. наук, доцент

В данной работе рассмотрены обобщенный EM-алгоритм и робастный алгоритм с разреживанием для решения задач тематического моделирования, рассмотрены тематическое моделирование и его стандартный критерий качества – перплексия.

Тематическое моделирование (topic modeling) – способ семантического анализа коллекции текстовых документов, первое описание которого датируется 1998 годом. С помощью тематического моделирования можно определить, какие темы содержатся в большой коллекции текстовых документов, какие слова характеризуют определенную тему, к каким темам относится каждый документ. Согласно вероятностной тематической модели, коллекция текстовых документов представляет собой распределение на множестве терминов, а отдельный документ – распределение на множестве тем. Тематическое моделирование можно использовать, например, для определения авторства документов.

Предположение о том, что лишь определенные слова в текстах относятся к каким-либо темам, можно формализовать как робастную тематическую модель. Такая модель может быть представлена как вероятностное сочетание тематической, шумовой и фоновой компоненты.

Тематическая компонента – термины, которые характеризуют текст как принадлежащий к определенной теме. Шумовая компонента – это термины, которые относятся к темам, слабо представленным в текущей коллекции. Также к шумовой компоненте относят слова, характерные только для конкретного документа. Фоновая компонента – часто употребляемые слова, а также стоп-слова, не отброшенные на предварительной стадии получения ключевых терминов. Стоп-слова – это слова, связывающие части речи в предложении. Они должны пропускаться автоматически, т. к. не несут в себе смысловой нагрузки и не относятся к каким-либо темам.

Перплексия (perplexity) – это стандартный критерий качества тематических моделей. Это мера несоответствия терминов, наблюдаемых в документах коллекции, текущей модели. Чем меньше перплексия, тем лучше модель подходит для предсказания появления терминов.

Рассмотрим обобщенный EM-алгоритм с эвристиками сглаживания, сэмплирования, робастности и разреживания, который позволяет при различных сочетаниях этих эвристик получать известные тематические модели PLSA (probabilistic latent semantic analysis), LDA (latent Dirichlet allocation), SWB (special words with background), а также новые.

Возьмем упрощенный робастный алгоритм, который хорошо сочетается с эвристикой разреживания. Указанный алгоритм не требует ни дополнительных вычислений, ни хранения матрицы параметров шума. Такая робастная модель без сглаживания позволяет разреживать искомые распределения на 99 % без ухудшения качества перплексии модели.

Робастные алгоритмы с разреживанием являются лучшими по критерию контрольной перплексии. Указанный алгоритм не требует введения априорных распределений Дирихле. Расчет

сглаживания для таких алгоритмов не является необходимым, т. к. не приводит к значительному уменьшению показателя перплексии.

Список использованных источников:

1. К. В. Воронцов. Вероятностное тематическое моделирование [Электронный ресурс]. – 2013. – Режим доступа : <http://www.machinelearning.ru/wiki/images/2/22/Voron-2013-ptm.pdf>.
2. Chemudugunta, Ch. Text Mining with Probabilistic Topic Models: Applications in Information Retrieval and Concept Modeling / Ch. Chemudugunta. – LAP LAMBERT Academic Publishing, 2010. – 140 p.

ГРАФОВЫЕ БАЗЫ ДАННЫХ И ИХ СВОЙСТВА

Зорко П.А., Кулевич А.О., Хрол К.А.

Белорусский государственный университет информатики и радиоэлектроники
г. Минск, Республика Беларусь

Пилецкий И.И. – канд. физ.-мат. наук, доцент

Большая часть мировых данных хранится в базах данных (БД).

База данных – это информационная модель, позволяющая в упорядоченном виде хранить данные о группе объектов, обладающих подобным набором свойств [1]. Такой способ организации данных позволяет эффективно извлекать их. Данные можно просматривать, вставлять, обновлять и удалять по мере необходимости.

В настоящее время большой интерес приобретают графовые БД, т. к. они обеспечивают представление, более близкое к структуре реальных данных. Базы данных такого типа относятся к NoSQL базам данных.

Графовые базы данных обладают рядом преимуществ и достоинств по сравнению с другими БД: они, как и RDBS, обладают свойствами OLTP (Online Transaction Processing) & OLAP (Online Analytical Processing), поддерживают транзакции ACID (atomic, consistent, isolated и durable), чего не обеспечивает ни одна NoSQL БД.

Графовые технологии являются основой для построения интеллектуальных приложений, для применения алгоритмов искусственного интеллекта [2]. Еще одним отличием графовых баз данных является то, что они явно описывают зависимости между узлами данных, в то время как другие базы данных связывают данные неявными связями.

Главным отличием графовых баз данных от сетевых является то, что в сетевых базах данных нет возможности связывать произвольные записи друг с другом, что затрудняет работу с графо-ориентированными наборами данных, в то время как графовые базы данных предоставляют такую возможность.

Кроме того, сетевые базы данных используют записи с predetermined числом полей, а графовые базы данных благодаря их гибкой структуре допускают произвольные пары ключ/значение как в узлах, так и в отношениях [3].

Графовые базы данных могут использоваться в самых разнообразных приложениях. Некоторые популярные области применения графовых баз данных включают:

- Социальные сети.
- Рекомендации по продукту в реальном времени.
- Сетевые схемы разнообразных физических объектов.
- Обнаружение мошенничества (в банках и сетях).
- Управление доступом.
- Поиск цифровых активов на основе графиков.
- Управление основными данными.
- Различные области искусственного интеллекта.

Использование графовых баз данных очень перспективно из-за простоты, наглядности и интуитивности графовой модели. Они часто не имеют схемы, что обеспечивает гибкость. Благодаря этому графовым моделям не требуется предварительно моделировать задачу в мельчайших подробностях, что очень удобно из-за быстро меняющихся бизнес-требований.

Присущая графам возможность расширения и модификации означает, что можно добавлять новые виды связей, новые узлы, новые метки и новые подграфы в существующую структуру, не нарушив при этом существующих запросов и функционала приложения.

Одной из веских причин выбора графовых баз данных является большой прирост производительности при работе со взаимосвязанными данными по сравнению с реляционными базами данных и NoSQL-хранилищами. Они могут более естественно масштабироваться до больших

наборов данных, поскольку обычно им не нужны операции объединения, которые часто могут быть дорогостоящими.

В отличие от других моделей, графовая модель данных может обеспечивать визуализацию с использованием обходов графа. Таким образом, предоставляется возможность быстро исследовать и просматривать данные графа, чтобы получить ценную информацию.

Графовые базы данных создаются и управляются с помощью графовых систем управления базами данных (СУБД). В настоящее время разработаны различные графовые СУБД.

Первой и одной из популярнейших на данный момент платформ является платформа Neo4j Graph. Она включает компоненты, которые позволяют разрабатывать приложение с поддержкой графов.

Графическая платформа Neo4j оптимизирована для хранения, картографирования, анализа и обхода сетей связанных данных [4].

С помощью интуитивно понятного отображения данных и связей между ними Neo4j обеспечивает возможность использования в реальном времени таких технологий, как: искусственный интеллект, машинное обучение; управление основными данными; управление идентификацией и доступом. Механизм хранения Neo4j использует массивы фиксированного размера для хранения данных графа и может обеспечить доступ к данным за время $O(1)$ [5].

В Neo4j используется SQL-подобный язык запросов Cypher. Cypher – это декларативный язык запросов для взаимодействия с графовой БД. Это мощный, оптимизированный для графов язык, который понимает и использует преимущества хранимых соединений.

Он состоит из предложений, ключевых слов и выражений, таких как предикаты и функции (например: MATCH, WHERE, LIMIT, AND, p.firstName = "Vsevolod").

```
MATCH (p:Professor)
WHERE p.lastName = "Petrov"
RETURN p
```

На основе полученных знаний авторами была разработана собственная графовая база данных, которая описывает наборы связей между группами специальности «Информатика и технологии программирования», преподавателей, которые ведут занятия у данных групп, и соответствующие им кафедры.

Графовое представление данной БД приведено на рисунке 1.

В данном графе узлы – это объекты, а дуги – отношения между объектами. В разработанной базе данных используются узлы трех типов: преподаватель (Professor); группа (Group); кафедра (Department).

Для каждого узла-преподавателя указаны его ФИО, ученая степень, электронная почта, рабочий номер телефона и рабочая аудитория.

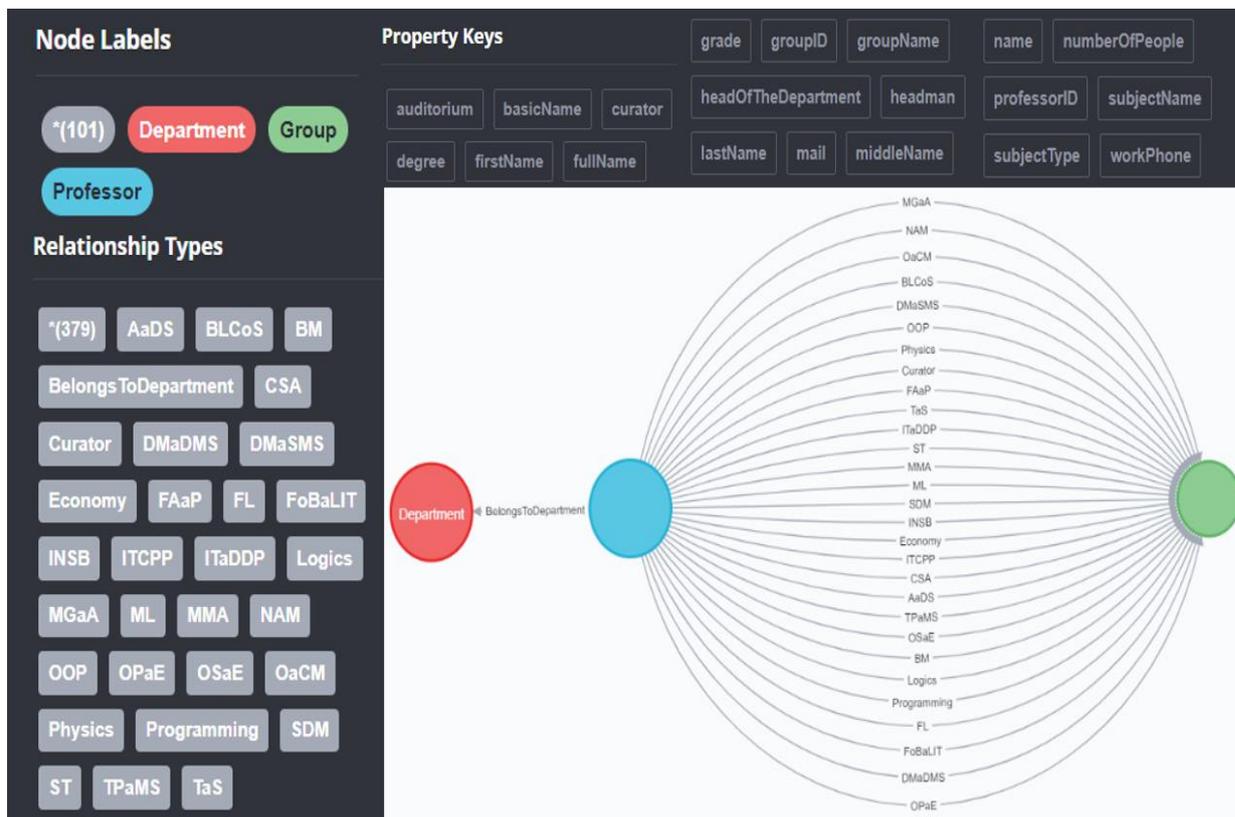


Рисунок 1 – Графовая база данных и ее представление

Для каждого узла-группы дополнительно указаны курс, ФИО старосты, количество человек в группе, а для групп 1–2 курсов – ФИО куратора группы.

Для каждого узла-кафедры дополнительно указаны ФИО заведующего кафедрой, а также полное название кафедры.

Результатом научной работы стала разработанная база данных, которую можно легко интегрировать в различные приложения. Так, например, она может быть использована в приложении, отслеживающем расписание групп. Можно оперативно получать данные о сотрудниках и студентах.

Список использованных источников:

1. База данных – Википедия [Электронный ресурс]. – 2021. – Режим доступа : https://ru.wikipedia.org/wiki/База_данных.
2. Интеллектуальная система комплексного анализа данных интернет-источников / М. П. Батура [и др.] // *BIG DATA and Advanced Analytics = BIG DATA и анализ высокого уровня* : сб. материалов VI Междунар. науч.-практ. конф., Минск, 20–21 мая 2020 г. : в 3 ч. Ч. 1 / редкол. : В. А. Богуш [и др.]. – Минск : Бестпринт, 2020. – С. 220–241.
3. Робинсон, Я. Графовые базы данных: новые возможности для работы со связанными данными / Я. Робинсон, Дж. Вебер, Э. Эфрем ; пер. с англ. Р. Н. Рагимова ; науч. ред. А. Н. Кисилев. – 2-е изд. – М. : ДМК Пресс, 2016. – 256 с.
4. *Graph Database Platform | Graph Database Management System | Neo4j* [Электронный ресурс]. – 2021. – Режим доступа : <https://neo4j.com>.
5. *History of Databases and Graph Database – Bitnine Global Inc.* [Электронный ресурс]. – 2020. – Режим доступа : <https://bitnine.net/blog-graph-database/history-of-databases-and-graph-database/>.

ОСОБЕННОСТИ РАБОТЫ С ПЛАТОЙ COSMOS+ OPENSSED ПРИ ПЕРЕНОСЕ НА СОВРЕМЕННЫЕ АППАРАТНЫЕ И ПРОГРАММНЫЕ ПЛАТФОРМЫ

Зайцев И.Д.

*Белорусский государственный университет информатики и радиоэлектроники
г. Минск, Республика Беларусь*

Заливако С.С. – канд. техн. наук, доцент

В работе были рассмотрены проблемы, возникающие при переносе проекта Cosmos+ OpenSSD на актуальную в настоящее время версию инструментов фирмы Xilinx (Vivado 2020.1, Vitis 2020.1). В Xilinx Vivado была рассмотрена проблема несоответствия частот IP-ядер после их обновления. В Xilinx Vitis была рассмотрена проблема добавления новой аппаратной платформы и выбора корректных исходных файлов FTL (Flash Translation Layer). Отдельно была произведена сборка хоста и настройка его операционной системы (ОС).

В настоящее время твердотельные накопители информации (NAND Flash SSD – Solid State Drive) широко применяются во многих компьютерных системах наряду с традиционными накопителями на жестких магнитных дисках (HDD – Hard Disk Drive). Твердотельные накопители в сравнении с традиционными жесткими дисками имеют ряд преимуществ: более быстрое время доступа, меньшее энергопотребление и меньшую массу. Также стоит отметить, что из-за отсутствия движущихся элементов твердотельные накопители имеют лучшую устойчивость к вибрациям. К недостаткам можно отнести: более высокую стоимость на единицу хранения информации, уменьшенную пропускную способность во время выполнения процедур обслуживания диска (например сборки мусора, выравнивания износа и т. п.), а также большее ограничение на число циклов стирания для каждого блока устройства [1].

SSD устройства поддерживают три основные операции: чтение, запись и стирание. Операции чтения и записи работают в пределах страницы, в то время как операция стирания работает в пределах блока (блок является совокупностью страниц) [2]. Каждая страница в SSD имеет одно из трех доступных состояний: действительная (valid), недействительная (invalid) и свободная/стертая (free/erased). Операция записи может быть применена только к странице, находящейся в стертом состоянии.

Современное встроенное программное обеспечение должно включать в себя таблицу соответствия адресов (L2P – Logical to Physical), сборщик мусора (GC – Garbage Collector), выравниватель износа (WL – Wear-leveler) и т. д. Таким образом, особенности устройства NAND Flash памяти требуют создания дополнительных модулей, что приводит к усложнению разработки микропрограммного обеспечения современных SSD устройств. В связи с этим при разработке используют готовые платформы, с помощью которых осуществляется прототипирование аппаратных решений и программных алгоритмов.

Примером такой платформы служит проект Cosmos+ OpenSSD [3] – проект с открытым исходным кодом, который позволяет проводить исследования в области SSD технологий. Проект является модульным, т. е. программные и аппаратные компоненты являются взаимозаменяемыми. В качестве отладочной платы используется Cosmos/Cosmos+ FPGA Platform (FPGA – Field-Programmable Gate Array), на которой происходит проектирование аппаратуры микроконтроллера

SSD. Работа с проектом осуществляется при помощи программного обеспечения компании Xilinx. Данный проект изначально был разработан для версии Xilinx Vivado 2014.4. В связи с этим для переноса проекта на современную версию Xilinx Vivado 2020.1 требуются модификации аппаратного и программного обеспечения.

Ядром отладочной платы является система на кристалле Zynq-7000, содержащая программируемую логическую интегральную схему XC7Z045-FFG900-3. Вычислительные единицы представлены двумя ядрами ARM Cortex – A9. Оперативная память типа DDR3 объемом 1 Гб. Модуль памяти представляет собой печатную плату форма-фактора SO-DIMM, состоящую из 8 микросхем NAND Flash памяти. Каждая микросхема объемом 128 Гб. Размер страницы составляет 18048 байт. Размер запасной области (spare area) – 1664 байта. Проект Cosmos+ OpenSSD поддерживает 2 модуля памяти.

Целью данной работы является анализ и обобщение изменений для переноса проекта на актуальную версию средств разработки Xilinx. Опишем особенности трех этапов работы над проектом: сборки аппаратной платформы, создания проекта и загрузки промежуточного микропрограммного обеспечения, а также сборки хоста и настройки его ОС. Для работы над проектом была выбрана среда автоматизированного проектирования Xilinx Vivado 2020.1 и среда разработки промежуточного микропрограммного обеспечения Xilinx Vitis 2020.1. Необходимо было также удостовериться, что материнская плата хоста поддерживает NVMe протокол [4]. Для Windows 7 необходимо скачать OpenFabrics Alliance (OFA) NVMe драйвер. В свою очередь, Windows 8/8.1 и Windows 10 поддерживают NVMe протокол по умолчанию.

На первом этапе производится сборка проекта аппаратного обеспечения контроллера твердотельного накопителя OpenSSD. В связи с тем, что IP-ядра проекта нуждаются в обновлении, тактовую частоту необходимо настроить вручную. Для этого необходимо выбрать блок ZYNQ7 Processing System (5.5) и выставить частоты, как показано на рисунке 1. Перед началом моделирования необходимо провести валидацию дизайна (Design Validation). После стадии синтеза (Synthesis) и размещения ресурсов FPGA в логические ячейки и трассировки (Implementation) конфигурационный файл FPGA сгенерируется корректно. В папке проекта Xilinx Vivado появится файл с расширением *.xsa. На этом этапе также возможно изменить существующую или создать свою аппаратную платформу контроллера твердотельного накопителя.

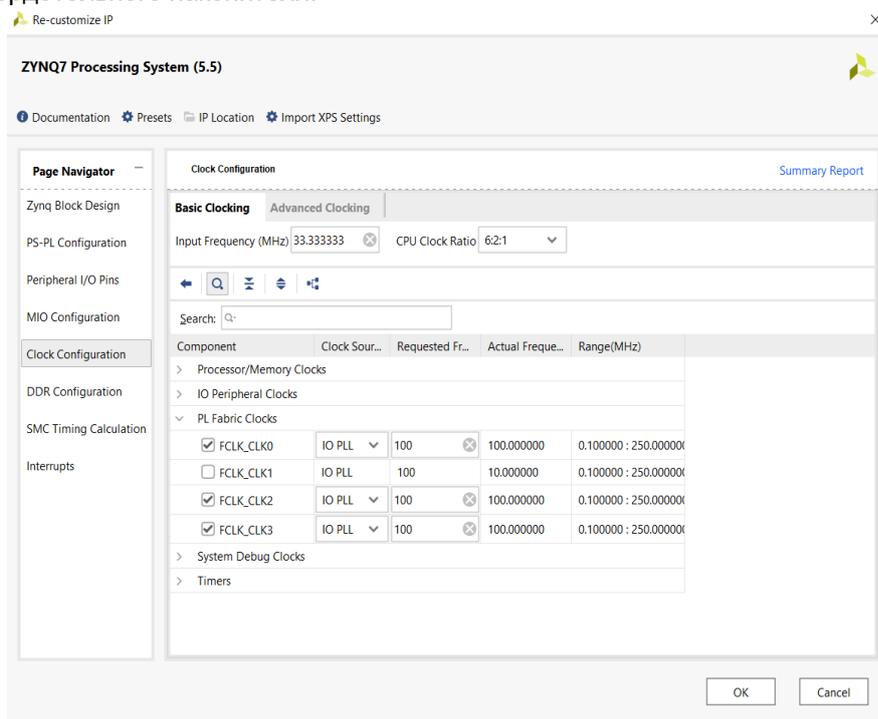


Рисунок 1 – Корректировка частот в IP-компоненте ZYNQ7 Processing System

На втором этапе происходит работа с микропрограммным обеспечением. В среде Xilinx Vitis необходимо создать новый проект (Create Application Project), добавить новую аппаратную платформу (во вкладке Create a new platform from hardware (XSA) нужно будет выбрать файл с расширением *.xsa в папке проекта Xilinx Vivado). После создания проекта необходимо добавить исходный код на языке C из проекта GreedyFTL-3.0.0. FTL представляет собой программный слой, который обслуживает запросы записи и чтения хоста, а также реализует дополнительные процедуры (GC, WL и т. д.). На этом этапе возможно редактировать или создавать свои модули для микропрограммного обеспечения. Это может быть необходимо для экспериментов с разработанными алгоритмами различных процедур работы диска. После второго этапа происходит загрузка конфигурационного файла в память FPGA и микропрограммного обеспечения во Flash память.

На третьем этапе, после подключения платы к компьютеру, в диспетчере устройств должен появиться накопитель с названием NVMe Cosmos+ OpenSSD. Далее необходимо открыть панель управления → администрирование → управление компьютером → управление дисками, выбрать накопитель и создать составной том. В результате появится диск, с которым можно работать. Далее необходимо установить программу теста производительности (benchmark) и осуществлять оценку производительности полученного накопителя.

В ходе проделанной работы были рассмотрены проблемы переноса проекта Cosmos+ OpenSSD на новую версию инструментов фирмы Xilinx. Для запуска проекта в новых версиях необходимо было провести корректировку частот, провести валидацию дизайна и сгенерировать конфигурационный файл FPGA в Xilinx Vivado. В Xilinx Vitis была добавлена новая аппаратная платформа и исходные файлы FTL. В конце была произведена сборка и настройка хоста. В результате чего был настроен твердотельный накопитель, который корректно отображается в ОС и в дальнейшем может быть использован для оценки производительности аппаратных решений и программных алгоритмов.

Список использованных источников:

1. Tavakkol, A. MQSim: A Framework for Enabling Realistic Studies of Modern Multi-Queue SSD Devices / A. Tavakkol [et al.] // 16th USENIX Conference on File and Storage Technologies, Oakland, USA, Feb. 12–15, 2018 / USENIX Association. – 2018. – P. 49–65.
2. Van Houdt, B. Performance of garbage collection algorithms for flash-based solid state drives with hot/cold data / B. Van Houdt // Performance Evaluation. – 2013. – Vol. 70, iss. 10. – P. 692–703.
3. OpenSSD: open-source solid-state drive project for research and education [Электронный ресурс]. – Режим доступа : <http://www.openssd.io/>. – Дата доступа : 04.04.2021.
4. NVMe specification [Электронный ресурс]. – Режим доступа : <https://nvmexpress.org/>. – Дата доступа : 04.04.2021.

КРИПТОГРАФИЧЕСКИЙ АЛГОРИТМ RSA И АЛГОРИТМЫ СОЗДАНИЯ ЦИФРОВОЙ ПОДПИСИ НА ЕГО ОСНОВЕ

Герасимёнок Н.А., Готченя Д.Г., Лазук И.С.

Белорусский государственный университет информатики и радиоэлектроники
г. Минск, Республика Беларусь

Стройникова Е.Д. – ст. преподаватель

В современных реалиях мы не то что каждый день, а чуть ли не каждый час используем Интернет, а, следовательно, передаем огромный поток данных. А там, где есть передача данных, сразу возникает необходимость в защите этих данных, следовательно, и в шифровании. Остановимся на рассмотрении алгоритма шифрования RSA. В ходе данной работы была реализована консольная программа на языке программирования C++ (в среде Visual Studio Code).

Алгоритм RSA

Первый этап – генерация ключей:

1) Выбираем два больших случайных простых числа, приблизительно заданного размера (например 2048 бит каждое).

2) Находим их произведение, именуемое модулем, которое обычно обозначается n , где

$$n = p \cdot q.$$

3) Вычисляем мультипликативную арифметическую функцию Эйлера

$$\varphi(n) = (q - 1) \cdot (p - 1).$$

4) После этого для открытого ключа необходимо выбрать целое число e ($1 < e < \varphi(n)$), взаимно простое со значением функции $\varphi(n)$.

5) Для генерации закрытого ключа необходимо вычислить число d , оно, в свою очередь, должно удовлетворять сравнению

$$d \cdot e \equiv 1 \pmod{\varphi(n)},$$

d называется секретной экспонентой.

После проведенных действий получаем пару ключей:

- открытый ключ $\{e, n\}$, который публикуется в открытом доступе;
- закрытый ключ $\{d, n\}$, который остается у отправителя и держится в секрете.

Второй этап – работа протокола:

Выбираем целое число m для шифрования, чтобы его зашифровать, воспользуемся открытым ключом:

$$c = E(m) \equiv m^e \pmod{n}.$$

Полученное c и есть зашифрованное число. Для того чтобы из c обратно получить исходное число m , необходимо использовать закрытый ключ:

$$m = D(c) \equiv c^d \pmod{n}.$$

Алгоритмы цифровой подписи

Также криптосистема RSA позволяет реализовать аутентификацию, создав цифровую подпись. Данная подпись подтверждает целостность отправляемой информации и может быть использована вместо стандартной подписи от руки.

Алгоритм с открытым текстом

Первый этап – генерация цифровой подписи:

- 1) Возьмем открытый текст m .
- 2) Для создания подписи, обозначаемой s , необходимо воспользоваться сгенерированным ранее закрытым ключом:

$$s = S_A(m) \equiv m^d \pmod{n}.$$

- 3) Передаем пару $\{s, m\}$, состоящую из подписи и открытого текста.

Второй этап – проверка неизменности сообщения с помощью электронной подписи:

- 1) Используя полученную подпись и открытый ключ, сгенерируем прообраз текста m , обычно обозначаемый m' :

$$m' = P_A(s) \equiv s^e \pmod{n}.$$

- 2) Сравниваем прообраз m' и сам текст m , тем самым выясняя подлинность подписи и целостность текста.

Алгоритм с закрытым текстом

Данный способ предоставляет большую защиту, однако требует дополнительных вычислений.

Алгоритм:

- 1) Отправитель шифрует исходный текст алгоритмом RSA с помощью открытого ключа получателя.
- 2) Далее отправитель должен сгенерировать цифровую подпись с помощью своего закрытого ключа.
- 3) Шифруем полученную подпись открытым ключом получателя.
- 4) Отправляем зашифрованный текст и подпись.
- 5) Получатель расшифровывает текст и подпись с помощью своего закрытого ключа.
- 6) С помощью расшифрованной подписи и открытого ключа получателя создаем прообраз исходного текста.
- 7) Сравниваем прообраз и исходный текст.

Вывод

Алгоритм RSA является классическим примером шифров с открытым ключом, он надежно выполняет свою работу. С его помощью можно как шифровать текст, так и создавать цифровую подпись, что делает его довольно универсальным. В основе алгоритма RSA лежит использование односторонних функций, что на данный момент обеспечивает хорошую защиту. Однако с каждым годом вычислительные мощности стремительно растут, открываются новые математические методы разложения больших чисел на простые множители, что в перспективе ослабляет данный способ шифрования. При наличии достаточной выборки текста, зашифрованного чистым алгоритмом RSA, используя частотный анализ, возможно «обойти» данный шифр. Также скорость шифрования

оставляет желать лучшего, поэтому в наше время RSA используется в связке с другими, более быстрыми, алгоритмами шифрования.

Модифицированный алгоритм RSA

На самом деле, можно довольно легко модифицировать алгоритм шифрования RSA, чтобы он смог «сопротивляться» частотному анализу.

Алгоритм:

- 1) После перевода символов в числа добавляем в начало еще одно случайное число.
- 2) Начиная уже со второго числа, переопределяем его значение следующим образом: берем существующее значение и складываем его с предыдущим, а далее берем остаток от деления на n .
- 3) Новый полученный набор чисел шифруем открытым ключом и передаем зашифрованное сообщение.
- 4) Для расшифровки используем закрытый ключ, осталось получить исходные числа.
- 5) Чтобы «распутать», идем с начала, минуя первое число, и рассматриваемому числу присваиваем остаток от деления на n от разности самого числа с предшествующим ему. Первое число не используем, т. к. соответствующего ему символа в исходном сообщении не было.
- 6) Далее из чисел восстанавливаем текст, действуя по стандартному алгоритму.

Данным набором действий устраняется однозначное соответствие между зашифрованным числом и буквой в исходном сообщении. Также добавленное в начало случайное число окончательно убирает эту связь.

Ссылка на программную реализацию:

<https://github.com/LaRtik/RSAsignature>.

Список использованных источников:

1. Фергюсон, Н. *Практическая криптография* / Н. Фергюсон, Б. Шнайер. ; пер. с англ. – М. [и др.] : Диалектика, 2005. – 421 с.
2. *Преимущества и недостатки алгоритма шифрования RSA* [Электронный ресурс]. – 2017–2021. – Режим доступа : https://studwood.ru/1685074/informatika/preimuschestva_nedostatki_algoritma_shifrovaniya.
3. *Использование криптосистемы RSA в настоящее время* [Электронный ресурс]. – 2007. – Режим доступа : <http://www.cyberguru.ru/algorithms/cryptography/rsa.html?start=10>.
4. *RSA* – Википедия [Электронный ресурс]. – 2021. – Режим доступа : <https://ru.wikipedia.org/wiki/RSA#%D0%98%D1%81%D1%82%D0%BE%D1%80%D0%B8%D1%8F>.

Научное издание

КОМПЬЮТЕРНЫЕ СИСТЕМЫ И СЕТИ

**57-я научная конференция
аспирантов, магистрантов и студентов**

Сборник тезисов докладов

(Минск, 19-23 апреля 2021 года)

В авторской редакции

Ответственный за выпуск *Е.В. Калабухов*

Компьютерная верстка: *С.К. Матусевич*
А.В. Гончарова