

# DSPA:

## Вопросы применения цифровой обработки сигналов

№2

2023



## СОДЕРЖАНИЕ

<b>Быков В.В.</b> <b>МОДЕЛИРОВАНИЕ ПЕРЕДАЧИ ИЗОБРАЖЕНИЙ С СЖАТИЕМ</b>	<b>4</b>
<b>Гадасин Д.В., Комкова М.Г., Пантелеева К.А., Гадасин Д.Д.</b> <b>СВЯЗЬ МЕЖДУ ВЕЛИЧИНОЙ ЭНТРОПИИ И КОЛИЧЕСТВОМ ИНФОРМАЦИИ ПРИ ПРЕДСТАВЛЕНИИ ПРЕДМЕТНОЙ ОБЛАСТИ РАЗНЫМИ ЛИНГВИСТИЧЕСКИМИ ЕДИНИЦАМИ</b>	<b>12</b>
<b>Горгадзе С.Ф., Ермакова А.В.</b> <b>ЭФФЕКТИВНОСТЬ ТЕХНОЛОГИЙ IDMA И CDMA ПРИ МАЛОМ КОЭФФИЦИЕНТЕ РАСШИРЕНИЯ СПЕКТРА</b>	<b>22</b>
<b>Калушев К.А.</b> <b>СРАВНЕНИЕ РАЗЛИЧНЫХ МЕТОДОВ КЛАССИФИКАЦИИ РУКОПИСНЫХ ЦИФР НА НАБОРАХ ДАННЫХ С ПРЕДВАРИТЕЛЬНОЙ ПОДГОТОВКОЙ И БЕЗ НЕЕ</b>	<b>30</b>
<b>Прокурат Г.А., Балобанов А.В.</b> <b>МЕТОДОЛОГИЯ ПОНИЖЕНИЯ ДЕТАЛИЗАЦИИ ПОЛИГОНАЛЬНОЙ ГЕОМЕТРИИ ДЛЯ СУБЪЕКТИВНОЙ ОЦЕНКИ КАЧЕСТВА ИЗОБРАЖЕНИЯ ОТНОСИТЕЛЬНО РЕГИОНА ИНТЕРЕСА</b>	<b>37</b>
<b>Четыркин М.И., Безумнов Д.Н.</b> <b>НЕЙРОСЕТЕВОЕ РАСПОЗНАВАНИЕ МАРКИРОВКИ ПРОДОВОЛЬСТВЕННЫХ ТОВАРОВ</b>	<b>43</b>

## МОДЕЛИРОВАНИЕ ПЕРЕДАЧИ ИЗОБРАЖЕНИЙ С СЖАТИЕМ

**Быков Виктор Викторович,**

*Московский Технический Университет Связи и Информатики, к.т.н., доцент, Москва, Россия*  
[selinn60@yandex.ru](mailto:selinn60@yandex.ru)

### **Аннотация**

*Разработано ПО моделирования передачи сжатого изображения с QAM модуляцией и OFDM по AWGN каналу в системе Матлаб. Сжатие изображений выполнено как с помощью дискретного косинусного преобразования, так и Вейвлет метода. Обработка коэффициентов дискретного косинусного преобразования осуществляется поблочно для каждого блока. Сжатие с Вейвлет-преобразованием производится путем ограничения числа передаваемых коэффициентов. Моделирование сопровождается отображением переданного и принятого эталонных изображений, позволяющее сравнивать оба изображения по качеству. Вычисление отношения сигнал/шум на изображениях выполняется автоматически.*

**Ключевые слова:** *изображение; сигнал; модель; сжатие; канал; вейвлет; модуляция; сканирование; коэффициенты.*

### **Введение**

В связи с развитием средств ТВ-вещания, внедрения цифровых методов и увеличения числа передаваемых программ обостряется вопрос повышения сжатия передаваемых цифровых изображений. Это обеспечивает увеличение эффективности использования каналов связи и устройств хранения телевизионной информации. Видеокамеры, RAID-массивы, системы хранения информации на телецентрах и в Интернете – везде требуется сжатие цифрового потока. Задача состоит в том, чтобы снизить скорость цифрового потока с сохранением высокого качества передаваемых изображений.

В настоящее время в Москве передаются три мультиплекса цифровых ТВ-программ. Каждый мультиплекс занимает канал около 8 МГц. Таким образом каждая программа передается со скоростью не более 1 Мбит/сек с сохранением хорошего качества ТВ-изображений. Коэффициент сжатия при этом составляет - более 100 раз.

Это оказалось возможным благодаря совершенным методам сжатия. При обработке цифрового сигнала сжатие выполняется дискретным косинусным преобразованием (ДКП) и вейвлет-преобразованием, используемых в последующих развитиях стандарта MPEG-4 (H.264, H.265/HEVC). При передаче сигналов сжатие осуществляется благодаря применению многопозиционных видов модуляции QAM, PSK и APSK.

Разработанные модели на основе ПО Матлаб позволяют охарактеризовать передачу сжатых цифровых изображений через модуляторы многопозиционной квадратурной модуляции QAM и по каналам OFDM с ортогональным частотным мультиплексированием.

ПО в настоящей работе является продолжением разработок представленных в [3,4,5].

При подготовке программ некоторые функции и выражения Матлаб взяты из [1,6,7].

### **Моделирование передачи поблочно сжатого изображения с QAM модуляцией и OFDM по AWGN каналу**

Ниже представлен алгоритм и программная модель такой передачи. Часть ПО взята из [2].

Таблица 1

Алгоритм модели поблочного сжатия изображения ДКП и передачи этого изображения по каналу OFDM QAM256

Начало алгоритма
1. Основные параметры передачи с OFDM
2. Загрузка испытательного изображения
3. Формирование блока с ДКП коэффициентов
4. Разделение изображения на блоки 8x8
5. Введение маски удаления коэффициентов в блоках
6. Удаление коэффициентов в маске блока
7. Обработка сигнала изображения для его подачи на модулятор QAM256
8. OFDM-модуляция обратным преобразованием Фурье
<b>9. Передача сигнала через канал AWGN с определенным уровнем шума</b>
10. OFDM-демодуляция прямым преобразованием Фурье
11. Демодуляция QAM256
12. Восстановление принятого сжатого изображения и его отображение
13. Расчет отношения сигнал/шум на изображении
Конец алгоритма

Предусмотрено два варианта ПО: одно - для передачи с сжатием, второе - для передачи без сжатия с полным сохранением всех элементов ПО и отношения сигнал/шум 15 дБ. Это необходимо, чтобы сравнить качество переданного изображения для обоих вариантов передачи.

В программе использованы параметры передачи с OFDM для DVB-T2, но не применяются помехоустойчивые коды (БЧХ и LDPC). Сведения о их использование в ПО имеются в более ранних статьях [3,4,5]. Представленное в этой работе ПО тоже работает с этими кодами. Однако введение кодов в ПО приводит к необходимости передачи исходных изображений с низким разрешением, поскольку их цифровые потоки необходимо согласовывать с длительностью кодовых слов БЧХ и LDPC.

**ПО модели передачи**

```

clc; close all; clear all;
Tu = 3588e-6; % длительность символа OFDM
T=Tu/(16*2048);
G=0; % относительная длительность защитного интервала
delta=G*Tu; % длительность защитного интервала
Ts=delta+Tu; % длительность символа OFDM с защитным интервалом
Kmax=27264; % число поднесущих
Kmin=0;
FS =16*4096;
q=10; %отношение периода поднесущей к элементарному периоду (1/T)
fc=q*1/T; % частота поднесущей
Rs=4*fc; % длительность моделирования
t=0:1/Rs:Tu;
% Загрузка изображения
B = imread('lena.jpg');
figure(1);
imshow(B); title('Исходное изображение');
%Преобразование изображения в форму double
B = im2double(B);
    
```



```

%
% Здесь формируются блоки, на которые затем разбивается изображение, и отображается матрица
8x8 пикселей (маска) для сжатия в пределах блока. Выбран сжатие в 4,3 раза. Эта часть ПО взята из
[2].
% Удаление коэффициентов в матрице 8x8
B2 = blkproc(blok,[8 8],'P1.*x',mask) ;
B22=B2+3;
% Квантование на 256 уровней
M = 256; % количество векторов
% Квантуем сигнал B22
Bn_k = fix(B22*(M-1))/(M-1);
Bn_kr = reshape(Bn_k,1,256*256);
% Приводим значения коэффициентов
%матрицы Bn_k до значений 0-255
Bn_k2 = Bn_k*255/(max(Bn_kr));
% Берем 255, чтобы работал qammod
% Округляем Bn_k2 до ближайших целых чисел 0-255
y1 = round(Bn_k2) ;
y12=reshape(y1,1,256*256);
M = 256; k = log2(M);
a1 = qammod(y12,M);
% Обратное преобразование Фурье для формирования %многих несущих
carriers= FS'.*ifft(a1,FS);
k = log2(M);
% Отношение сигнал/шум
EbNo = 15;
% Передача сигнала сжатого изображения
%по каналу AWGN
a= awgn(carriers,EbNo+10*log10(k),'measured');

% На приемной стороне
% Прямое преобразование Фурье
info_2N=(1/FS).*fft(a,FS);
% Демодуляция QAM
a3 = qamdemod(info_2N,M); % max(a3)=255 min(a3)=7
% Преобразование последовательного сигнала a3
% в матрицу изображения
a32= reshape(a3,256,256);
% Обратные преобразования матрицы коэффициентов
Bk3= a32/255*(max(Bn_kr));
Bk4=Bk3-3;
B7= blkproc(Bk4,[8 8],'P1*x*P2',T,T);
B7r=reshape(B7,256,256);
% Восстановление сжатого изображения
figure(4), imshow(B7r)
title('Принятое сжатое изображение');

```

Принятое сжатое изображение



```
%
% Вычисление отношения сигнал/шум
Imr=double(B7r);
Im=double(B);
[psnr, snr]=psnr(B7r, B)
```

Для сравнения передачи сжатием и без него маска была полностью заполнена единицами (отсутствие сжатия) при том же отношении сигнал/шум. В результате полученное изображение выглядело как:

Принятое изображение без сжатия



Результаты расчета отношения сигнал/шум:

Выдало сжатием: psnr = 22.8 дБ; snr = 14.3 дБ.

Выдало без сжатия: psnr = 21.5 дБ; snr = 13.0 дБ.

Сравнивая изображения видно, при сжатии уменьшилось разрешение сжатого изображения. Отношения сигнал/шум отличаются мало.

### Моделирование передачи изображения, сжатого вейвлет преобразованием, с модуляцией QAM16 по AWGN каналу.

Ниже представлен алгоритм и программная модель такой передачи. В ПО использованы выражения Матлаб для декомпозиции и рекомпозиции сигналов [6,7].

Таблица 2

Алгоритм модели передачи изображения, сжатого вейвлет преобразованием, с модуляцией QAM16 по AWGN каналу

Начало алгоритма
1. Загрузка испытательного изображения
2. Вейвлет преобразование изображения (декомпозиция)
3. Ограничение числа коэффициентов вейвлет преобразования
4. Преобразование чисел коэффициентов в положительную форму для преобразования чисел в биты
5. Преобразования битового сигнала для подачи на модулятор QAM16

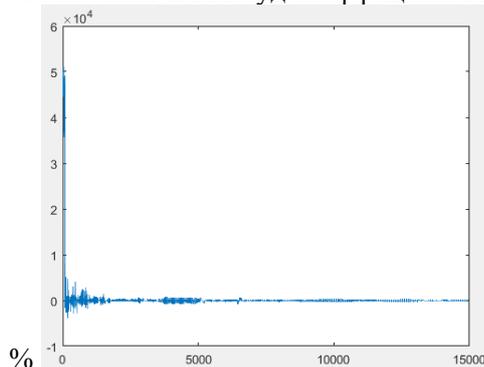
6. Обработка сигнала изображения для его подачи на модулятор QAM256
7. Модуляция QAM16
8. Перенос сигнала на несущую
<b>9. Передача сигнала через канал AWGN с определенным уровнем шума</b>
10. Перенос сигнала на низкие частоты
11. Демодуляция QAM16
12. Преобразование символов в битовую форму
13. Преобразование битов в числа от 0 до 255
14. Восстановление длины вектора принятого сигнала до его номинального значения путем добавления необходимого количества нулей
15. Вейвлет рекомпозиция принятого сжатого изображения и его отображение
16. Расчет отношения сигнал/шум на изображении
Конец алгоритма

### ПО модели передачи

```

clc; close all; clear all;
% Загружаем изображение
im = imread('lena.jpg');
figure(1); imshow(im)
title('Исходное изображение')
% Применим вейвлет-функцию bior4.4
N = log2(256) ; % Уровень декомпозиции
% Обращение к вейвлет функциям
wname = 'bior4.4';
% Декомпозиция изображения
[W,S] = wavedec2(im, N, wname) ;
figure(3) ; plot(W) ; xlim([0 15000]);
% Значения амплитуд коэффициентов (ось Y) в %зависимости от номера коэффициента

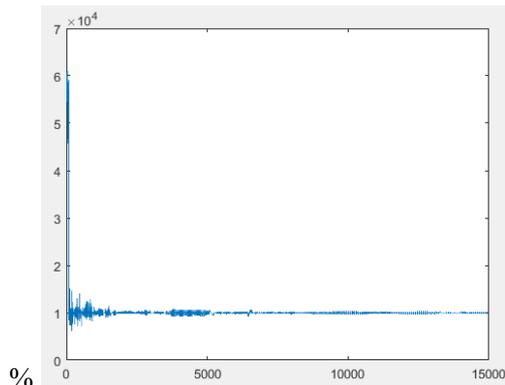
```



```

% Ограничим W до 15000 коэффициентов.
% Это дает сжатие в 5 раз
Wг = W([1:15000]) ; % 75087-15000=60087
% Прибавим 10000, чтобы все числа были больше 0
Wгп = Wг+10000 ;
% Округление до целых значений
Wгп = round(Wгп) ;
figure(4); plot(Wгп) ; xlim([0 15000])
% Значения амплитуд коэффициентов (ось Y) в %зависимости от номера коэффициента

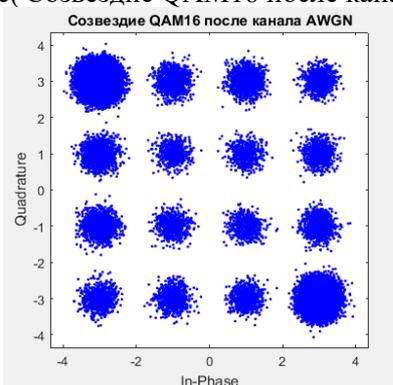
```



%

```

M = 16; % Число векторов QAM
k = log2(M); % Число бит на символ QAM
% 1 Преобразование в биты 01010... (16 колонок и 15000 строк)
x = de2bi(Wrp,'left-msb'); % преобразование чисел в биты
% 2 Преобразуем байты в 1 строку 163840 колонок
x_bit2=reshape(x,1, 15000*16);
% 3 Преобразуем биты в 60000 строк 4 колонки
% (преобразование в символы QAM16)
xsym = reshape(x_bit2,k,length(x_bit2)/k).','left-msb';
% Преобразование бит в символы QAM16
yn2 = bi2de(xsym,'left-msb');
% Модуляция QAM16
y = qammod(yn2,M);
% Длительность передаваемого сообщения
t=60000;
fc = 9e+07; % Частота несущей
% Перенос сигнала на несущую
y = (y.)*exp(1i*2*pi*fc*t);
snr = 25; % Отношение сигнал/шум в дБ
% Передача сжатого сигнала
y = awgn(y,snr,'measured');
scatterplot(y);
title('Созвездие QAM16 после канала AWGN')
    
```



%

```

% Перенос сигнала на низкие частоты
y = (y).*(exp(-1i*2*pi*fc*t));
% Демодуляция QAM-16
yr = qamdemod(y,M);
    
```

```

% Преобразование символов в битовую форму
yn_r2= de2bi(yr,'left-msb');
z3 = reshape(yn_r2.',prod(size(yn_r2)),1) ; size(z3) ;
z4 = reshape(z3, 1, 240000);
z6= reshape(z4, 15000 , 16) ;
% Преобразование битов в числа от 0 до 255
z7 = bi2de(z6,'left-msb') ; % 15000 строк, 1 кол
z7= z7-10000 ;
z8 = reshape(z7, 1, 15000) ;
% Добавим к вектору 60087 нулей (75087-15000)
zo = zeros(1, 60087) ; % 1 строка нулей 0000...
v1= z8 ; % Вектор коэффициентов
v2 = zo ; % Вектор нулей
% Суммируем два вектора - восстановление длины %сигнала
Wrr = [v1 v2] ;
figure(6); plot(Wrr); xlim([0 15000])
% Отображаемое изображение коэффициентов на рис.6 %полностью соответствует рис.3.
% Восстановление принятого сжатого изображения
imr = waverec2(Wrr, S, wname) ; % 256x256
figure(6); imshow(imr,[]);
title('Принятое сжатое изображение')

```

**Принятое сжатое изображение**



```

%
% Вычисление отношения сигнал/шум
imd=double(im);
imrd=double(imr);
[psnr, snr]=psnr(imrd, imd)
% Выдало psnr = -20.2 дБ; snr = 19.3 дБ.

```

### Заключение

Представлены программные модели передачи поблочно сжатого изображения с QAM модуляцией и OFDM по AWGN каналу и передачи изображения, сжатого вейвлет преобразованием, с модуляцией QAM16 по AWGN каналу. Они позволяют оценить качество изображения сжатого ДКП или вейвлет преобразованием с различной степенью сжатия после передачи этого изображения по каналу AWGN, через модулятор QAM и с OFDM. В моделях можно менять параметры для получения требуемого результата. Все модели позволяют отображать принятые изображения и сравнивать их с переданными. Программы работают и с PSK модуляцией.

Вместе с моделями, о которых рассказано в [3, 4, 5], представленное в статье ПО моделей позволяет охватить много проблем передачи изображений по различным каналам связи. Модели предназначены для студентов, изучающих соответствующие дисциплины.

### Литература

1. *Сергиенко А. Б.* Цифровая связь: Учеб. пособие. СПб.: Изд-во СЗ2 СПбГЭТУ «ЛЭТИ», 2012. 164 с.
2. Matlab image processing basic knowledge. Discrete cosine transform. <https://www.programmersought.com/article/12624527413/>
3. *Быков В.В.* Алгоритм математического моделирования цифрового телевизионного тракта с основными элементами DVB-T2. Технологии информационного общества. Сборник трудов XIV Международной отраслевой научно-технической конференции «Технологии информационного общества». 18-19 марта 2020 г. Москва, МТУСИ. М.: ИД Медиа Паблшер, 2020. 580 с.
4. *Быков В.В.* Оценка некоторых характеристик вещательных систем передачи на программных моделях // XV Международная отраслевая научная конференция «Технологии информационного общества». Сборник докладов. М.: МТУСИ, 2021.
5. *Быков В.В.* Оценка возможностей программной модели matlab системы DVB-T2 // Сборник трудов XVI Международной отраслевой научно-технической конференции «Технологии информационного общества». М.: Издательский дом Медиа Паблшер, 2022.
6. Wavelet Toolbox. Waverec2. <http://matlab.izmiran.ru/help/toolbox/wavelet/waverec2.html>
7. Wavelet Toolbox. Wavedec2. <http://matlab.izmiran.ru/help/toolbox/wavelet/wavedec2.html#37962>

## **СВЯЗЬ МЕЖДУ ВЕЛИЧИНОЙ ЭНТРОПИИ И КОЛИЧЕСТВОМ ИНФОРМАЦИИ ПРИ ПРЕДСТАВЛЕНИИ ПРЕДМЕТНОЙ ОБЛАСТИ РАЗНЫМИ ЛИНГВИСТИЧЕСКИМИ ЕДИНИЦАМИ**

**Гадасин Денис Вадимович,**

*Московский Технический Университет Связи и Информатики, доцент кафедры СИТус, к.т.н.,  
Москва, Россия*  
[dengadiplom@mail.ru](mailto:dengadiplom@mail.ru)

**Комкова Марина Георгиевна,**

*Московский Технический Университет Связи и Информатики, старший преподаватель  
кафедры СИТус, Москва, Россия*  
[m.g.komkova@mtuci.ru](mailto:m.g.komkova@mtuci.ru)

**Пантелеева Ксения Александровна,**

*Московский Технический Университет Связи и Информатики, бакалавр гр. БСТ1904, Москва, Россия*  
[ksenya2013333@gmail.com](mailto:ksenya2013333@gmail.com)

**Гадасин Даниил Денисович,**

*Московский Технический Университет Связи и Информатики, бакалавр гр. БСТ2103, Москва, Россия*  
[gadasin115@gmail.com](mailto:gadasin115@gmail.com)

### **Аннотация**

*Визуальным отображением человеческой речи является ее представление в виде специальных символов. Уменьшение смысловой нагрузки в тексте, количество информации, приходящейся на 1 символ, должно уменьшаться в том случае, если из всех представленных символов невозможно получить описание какой-либо предметной области. В работе определяется параметр и величина энтропии, значения которых могут служить начальной точкой для предварительного анализа текстовых данных.*

**Ключевые слова:** *Текст, текстовые данные, количество информации, избыточность, энтропия.*

### **Введение**

Развитие технологий обработки и передачи информации определяет увеличение количества данных, которые используются для удовлетворения информационных потребностей [1, 2, 15-29]. Данные могут и не применяться сиюминутно, что обуславливает развитие средств их хранения.

В 2022 году объем мировых данных был оценен в примерно 97 зеттабайт. Время обработки такого объема данных ограничивается суммарной вычислительной мощностью устройств, при этом несвоевременная обработка может приводить к потере актуальности данных [3, 4].

В процессе обработки данных выделяют два основных этапа: этап предобработки и этап самой обработки [5]. На первом этапе предполагается формирование определенной структуры, которая подчиняется законам логики. В каком виде исходные данные не поступали бы на вход обработчика, они состоят из символов кодировочной таблицы, т.е. в общем случае любой вид цифровых данных возможно представить в виде текстового файла.

В текстовых данных на любом языке всегда возможно выделить определенную структуру [6], и, так как язык существует для обмена информацией, то в нем можно выделить уровни иерархии, где на самом верхнем уровне стоят наиболее используемые элементы, а на самом низком наиболее редкие. Данное деление соответствует алгоритмам кодирования [7], в основе которых лежит принцип вероятностного деления.

Элементами передачи текстовой информации являются предложения, которые можно разделить на слова, слова – на слога, а слога на символы, таким образом минимальным элементом деления является символ, в котором заключено определенное количество информации и определенное

количество избыточности, неинформативной части, которая определяет внешний вид символа, его произношение на языке и т.п. Чем меньше избыточности, тем больше количества информации заключено в информационной, лингвистической единице (символ, слог, слово и т.д.).

Появление информационных единиц определяется вероятностью и является событием [8,9]. До момента возникновения случайного события не представляется возможным однозначно определить его состояние, а в момент наступления его состояние определяется однозначно, а, следовательно, неопределенность устраняется. Чем больше неопределенность какого-то события, тем большую информацию дает определение его состояния [10].

### Результаты исследований

В соответствии с принципами фон Неймана, любая операция, выполняемая на вычислительном устройстве, осуществляется только в двоичном коде [11], поэтому любая лингвистическая единица подвергается двоичному кодированию и ей однозначно ставится в соответствие последовательность нулей и единиц. Таким образом, текстовый файл возможно закодировать нулями и единицами, буквами, слогами, словами, предложениями, где элементы кодирования являются лингвистическими единицами. Не зависимо от того какими лингвистическими единицами произведено кодирование, количество информации в нем остается неизменным, меняется лишь только количество избыточности [12]. Структура взаимосвязей всех единиц представляется в следующем виде (рис. 1):

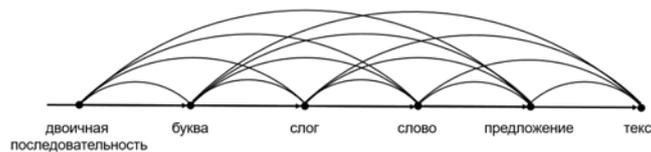


Рис. 1. Взаимосвязь лексических единиц с их отображением в вычислительном устройстве

Для случая равновероятного появления каждого символа алфавита энтропия  $H_0$  равняется 4,95 бит, а с учетом средней вероятностью появления буквы в текстах на русском языке  $H_1$  принимает значение 4,46 бит, т.е. степень неопределенности снижается, но полученные значения средней удельной информации на единицу не отражают практических значений, так как не учитывают зависимость появления буквы от предшествующих букв.

Исходя из этого среднее количество информации, которое приходится на одну букву необходимо вычислять исходя из формулы условной энтропии:

$$H_2 = H_{A_1}(A_2) = - \sum_{j,l=1}^n P([S_j, S_l]) \log_2 P_{jl}, \quad (1)$$

где  $n$  – число букв алфавита,  $P_{jl} = P_{S_j}(S_l)$  – условная вероятность появления буквы  $S_l$ , если известно, что непосредственно перед ней стоит буква  $S_j$ , а  $P([S_j, S_l])$  – вероятность появления двухбуквенного сочетания  $[S_j, S_l]$ .

С возрастанием  $M$  – количества символьных сочетаний – величина  $H_M$  станет убывать, стремясь к некоторому предельному значению  $H_\infty$ , которое можно принять за теоретически возможное значение средней удельной информации при обработке описания предметной области, в котором число сочетаний символов алфавита много больше самого алфавита, т.е. чем точнее определена вероятность сочетаний букв, тем меньше погрешность между теоретически возможным количеством информации, приходящейся на один символ алфавита, и количеством информации, соответствующей одному символу в данном описании.

Теорема о бесконечных обезьянах предполагает, что абстрактная обезьяна, ударяя случайным образом по клавишам печатной машинки в течение неограниченного времени, рано или поздно сможет напечатать любой наперед заданный текст (например, «Преступление и наказание»).

В русском алфавите содержится 33 буквы, пусть напечатание каждой буквы есть равновероятное событие, тогда вероятность появления одной буквы = 0,03. В романе «Преступление и наказание» содержится 1 100 000 знаков, тогда вероятность того, что обезьянка с первого раза напечатает данный текст, равняется порядка  $P = 10^{-1680000}$ , если учесть количество комбинаций, которые могут быть, и что разные комбинации могут повторяться, т.к. она случайным образом нажимает на клавиши, то вероятность правильного написания будет еще меньше.

В том случае, если время на написание стремится к бесконечности, то в конечном счете обезьяна напечатает данный роман. Под «обезьяной» можно понимать абстрактное устройство, порождающее случайную последовательность элементов используемого алфавита, например, искусственный интеллект. Теорема утверждает, что в течение продолжительного времени случайного набора текста будут возникать осмысленные слова, словосочетания и даже предложения.

Приближение нулевого порядка при соответствии клавишам машинки обезьяны биграммам и триграммам русского языка и соответствие размера клавиш частотам встречаемости букв алфавита будет представлять из себя следующее:

РАОВЛОВЛСДЛТЧСОГГВШШТЫТОЧТСЖВЧДЖД ЫЛТОГМРЩВРММИЫДОТСВДЮЮ МЮЕТ

С каждым приближением символы все больше будут приближаться к виду осмысленного текста естественного языка. Приближение первого порядка на уровне слов:

СВОБОДНОЙ ПТИЦЕ ПРОЛЕТЕЛ НО ПОМНИТЬ ПАМЯТЬ СВОБОДА КОМНАТАМ ПОЭМА  
МАЛЬЧИКА МНЕ ТУДА ПОЕХАВШЕГО КОНЦУ ГНЕВНО КАЖДОМУ РЯСАХ ОБ ПОДРУГ  
ПЕРЕРЕЗАЛО СЛЫШНО ВСЕМ КОНЧАЕТЕ НАД ДВУХ ЭТО ЗАКАТ ХОДУНОМ ЗЕЛЕНАЯ ЖУК  
ЗВУК ТОТ БЫ ГОЛОВА УТЕР УДАРНЫХ

Приближение второго порядка на уровне слов:

СООБЩЕСТВО СЧИТАЛО ВЫРАЖЕНИЕ ЧАСТОТНОГО ОРУДИЯ К ПОСТИЖЕНИЮ  
ДОЛЖНОСТЕЙ ДВА В ТИТРЫ НА БЕЗПРАВСТВЕННОСТИ В ПОЭЗИИ СОЗДАЕТСЯ ВСЯ  
ГРЫЗЕТ НАШИ БРАЗДЫ ПРАВЛЕНИЯ НАЧАЛА ЕГО ПОШЛОЙ

С каждым проделанным шагом сходство с обычным текстом заметно возрастает.

Уменьшение информационной нагрузки на один символ неравновероятного алфавита, а также взаимозависимость появления символов определяет избыточность языка [11]. Информационная избыточность рассчитывается по формуле:

$$R = \frac{H_0 - H_\infty}{H_0} = 1 - \frac{H_\infty}{\log n}, \quad (2)$$

Величина  $H_{0,\infty} = H_0 - H_\infty$  определяет недогруженность алфавита, а отношение  $\frac{H_\infty}{H_0} = \frac{H_\infty}{\log n}$  –

коэффициент его оптимальности.

У любого естественного языка есть своя избыточность, например, избыточность английского языка близка к 0,5, то есть в тексте на одном из этих языков можно исключить до 50% букв с тем, чтобы затем восстановить содержание текста по его оставшейся части.

Для решения задачи максимально экономичного кодирования необходимо увеличивать количество символов исходного алфавита [12]. Требование равномерности кода снимается основным принципом кодирования: чем чаще встречаемость – тем короче кодовая последовательность.

Будем считать, что алфавит  $\{S_1, \dots, S_n\}$  состоит из произвольных символов или наборов символов, таких как:

$$S_1 = K, S_2 = sq, S_3 = 200\Phi, S_4 = \sqrt[6]{\int_{-10}^4 \tan x \cdot y^2 dx}, \dots$$

В качестве ограничения примем: текст содержит в себе все символы алфавита, т.е. каждый символ алфавита используется хотя бы один раз и частоты встречаемости букв заранее известны. Произведем кодирование методом Шеннона – Фано, для чего произведем ранжирование символов по частоте

встречаемости и делим алфавит  $\{S_1, \dots, S_n\}$  на два подмножества как можно наиболее возможной равной вероятности.

Буквам одного подмножества ставим в соответствие 0, второго – 1. Это – первые разряды соответствующих кодовых последовательностей. Далее процесс повторяется для каждого из подмножеств – производится деление их на подмножества по возможности равной вероятности с последующим присвоением им номера 0 или 1. Процесс подходит к концу, когда в каждом из образовавшихся подмножеств не останется ровно по одной букве. В результате каждый имеет свой индивидуальный бинарный код.

Визуальным отображением человеческой речи является ее представление в виде специальных символов, букв. Буквы упорядочиваются специальным образом в соответствии с общепринятыми правилами, что позволяет наиболее полно передавать информацию с одновременным сохранением смысловой нагрузки.

Исходя из этого возможно выдвинуть предположение, что с уменьшением смысловой нагрузки, количество информации, которое приходится на 1 символ текста должно уменьшаться в том случае, если из всех представленных символов невозможно получить описание какой-либо предметной области. Например, количество информации приходящийся на один символ в предложении «Я прошу вашей руки, дорогая Дорофея!» и в предложении «Яяяпррррооооошшуувааеейки,дДгф!» будет одинаковой, так как количество и вид символов второго предложения полностью совпадает с количеством и видом символов первого.

При проведении эксперимента будет использоваться три типа предложений. Предложением первого типа является предложение, которое имеет смысловую нагрузку, например, «Я прошу вашей руки, дорогая Дорофея!». В предложении второго типа будут заменены некоторые слова при сохранении количества слов, в этих предложениях часть смысла будет потеряна, например, «Я прошу руки волка, дорогие козлята!». Предложения третьего типа содержат в себе последовательность символов, полученные в соответствии с псевдослучайным датчиком, их близость с предложениями первого и второго типа в большинстве случаев состоит в равенстве слов, например, «Ялпроатм юкдофь о.шеь увдае ейей ,дбгфл».

Для проведения эксперимента, на первом этапе, в соответствии с принятым алгоритмом были выбраны три отрывка с одинаковым количеством слов. Отрывки, входящие в выборку, в полной мере описывают каждый свою предметную область и взяты из произведений «Отцы и дети» И. С. Тургенева, «Детство» Л. Н. Толстого и «Героя нашего времени» М. Ю. Лермонтова. На втором этапе опыта в эти тексты, с помощью программы, генерирующей случайные слова (бредогенератор), были произведены замены слов в отрывках на случайно сгенерированных таким образом, что структура полученного текста полностью совпадает с исходной. Замена слов осуществлялась равномерно по всему тексту и заменено было порядка 60%, из них 8% подлежащих, 21% сказуемых, 34% определений, 35% дополнений и 2% обстоятельств.

На заключительном шаге для анализа был сгенерирован текст, с равным числом слов, слова в котором содержат в себе набор случайных последовательностей символов алфавита, т.е. с точки зрения человека в нем отсутствует какой-либо смысл. Таким образом в окончательном анализе принимало участие семь текстов, в которых было одинаковое количество слов, но разное смысловое наполнение.

В таблице 1 представлены сводные результаты проведенного опыта.

Таблица 1

Результаты анализа текстов с разной смысловой нагрузкой

	Предложения	Слова	Слога	Буквы
<b>Толстой «Детство»</b>				
Энтропия	5,4263	8,2517	7,7753	4,4417
Кол-во информации на единицу	0,9845	0,9897	0,9884	0,9753
<b>Толстой «Детство» после работы бредогенератора</b>				
Энтропия	5,4263	8,4820	8,1291	4,4687
Кол-во информации на единицу	0,9845	0,9884	0,9883	0,9791

<b>Лермонтов «Герой нашего времени»</b>				
Энтропия	5,9452	8,5368	7,9234	4,496
Кол-во информации на единицу	0,9977	0,9897	0,9890	0,9788
<b>Лермонтов «Герой нашего времени» после работы бредогенератора</b>				
Энтропия	5,9452	8,6651	8,0994	4,4827
Кол-во информации на единицу	0,9977	0,9894	0,9840	0,9802
<b>Тургенев «Отцы и дети»</b>				
Энтропия	5,0875	8,6834	7,9781	4,4461
Кол-во информации на единицу	0,9610	0,9927	0,9881	0,9757
<b>Тургенев «Отцы и дети» после работы бредогенератора</b>				
Энтропия	5,0875	8,8863	8,1073	4,4781
Кол-во информации на единицу	0,9610	0,9915	0,9887	0,9794
<b>Случайная последовательность букв алфавита</b>				
Энтропия	4,0875	9,1489	7,8703	4,3008
Кол-во информации на единицу	0,9519	0,9904	0,9908	0,9605

Анализ результатов из таблицы 1 показывает: величина энтропии, рассчитанная из вероятности появления предложений, не изменяется, хотя смысловая нагрузка падает с 100% до 40%, кроме случая с бессвязно случайно набранного текста (0% смысловой нагрузки), так как последовательность случайна, то знак точки, который означает конец предложения появляется также случайно. Соответственно, в данном тексте содержится меньше предложений.

С уменьшением смысловой нагрузки на 60% от исходного текста энтропия, рассчитанная исходя из вероятности появления отдельных слов, увеличивается на 1,5-2,8%; при этом с уменьшением смысловой нагрузки на все 100% от энтропии, рассчитанной для исходного текста, а для случая появления отдельных слов, увеличивается на 5,4-10,9%.

С уменьшением смысловой нагрузки на 60% от исходного текста энтропия, рассчитанная исходя из вероятности появления слогов, также увеличивается на 1,6-4,6%; при этом с уменьшением смысловой нагрузки до нулевого значения от исходного текста значение энтропии, рассчитанной исходя из вероятности появления слогов, снижается на 0,7-1,4%; рассчитанная исходя из вероятности появления букв, увеличивается на 0,7%, при этом в случае снижения смысловой нагрузки до ее полного отсутствия энтропия, рассчитанная исходя их вероятности появления букв, уменьшается на 3,2-4,3%.

Количество информации, приходящейся на одно предложения, не меняется, т.к. не изменяется количество предложений, а при снижении заложенного в тексте смысла до его полного отсутствия среднее количество информации, приходящейся на одно предложение, уменьшается на 1,0-4,6%.

Среднее количество информации, приходящейся на одно слово, уменьшается на 0,03-0,13%; при этом при изменении смысловой нагрузки от 100% до 0% среднее количество информации, приходящейся на одно слово, уменьшается на 0,23%.

При уменьшении смысловой нагрузки в тексте со 100% до 40% среднее количество информации, приходящейся на один слог, уменьшается на 0,00-0,50%; при этом при изменении смысловой нагрузки от 100% до 0% среднее количество информации, приходящейся на один слог, увеличивается на 0,18-0,27%.

С уменьшением смысловой нагрузки на 60% от исходного текста среднее количество информации, приходящейся на одну букву, увеличивается на 0,14-0,39%; при этом с уменьшением смысловой нагрузки до 0 значения от исходного текста среднее количество информации, приходящейся на одну букву, уменьшается на 1,52-1,87%.

Воспроизведение какого-либо отрывка с помощью речи возможно представить, как звуковое кодирование, при написании на бумаге - кодирование с помощью символов, определяющих алфавита, а машинное кодирование как двоичный код.

Каждый способ кодирования включает в себе определенную избыточность, поэтому сравним отрывки ранее рассмотренных произведений русских классиков, для которых считаем, что они в полной мере описывают предметную область, как кодирование данной области не только двоичной последовательностью, но и буквами, слогами, словами и предложениями русского языка. Для

каждого способа кодирования определим среднее количество информации, приходящуюся на единицу кода (предложение, слово, слог, буква). Результаты приведены в таблице 2.

Таблица 2

Результаты анализа текстов русских классиков в зависимости от способа кодирования

	Предложения	Слова	Слога	Буквы
<b>Тургенев «Отцы и дети»</b>				
Кол-во информации на 1 символ при кодировании двоичной последовательностью	0,9610	0,9927	0,9881	0,9757
Кол-во информации на 1 символ при кодировании буквами алфавита	0,0086	0,2608	0,6736	0,8913
Кол-во информации на 1 символ при кодировании слогами	0,0113	0,3456	0,8915	-
Кол-во информации на 1 символ при кодировании словами	0,0268	0,9836	-	-
Кол-во информации на 1 символ при кодировании предложениями	0,9867	-	-	-
<b>Лермонтов «Герой нашего времени»</b>				
Кол-во информации на 1 символ при кодировании двоичной последовательностью	0,9977	0,9897	0,9890	0,9788
Кол-во информации на 1 символ при кодировании буквами алфавита	0,0127	0,2849	0,6783	0,8919
Кол-во информации на 1 символ при кодировании слогами	0,0165	0,3722	0,8839	-
Кол-во информации на 1 символ при кодировании словами	0,0381	0,9758	-	-
Кол-во информации на 1 символ при кодировании предложениями	0,9990	-	-	-
<b>Толстой «Детство»</b>				
Кол-во информации на 1 символ при кодировании двоичной последовательностью	0,9845	0,9897	0,9884	0,9753
Кол-во информации на 1 символ при кодировании буквами алфавита	0,0123	0,3013	0,6671	0,8883
Кол-во информации на 1 символ при кодировании слогами	0,0159	0,4053	0,8947	-
Кол-во информации на 1 символ при кодировании словами	0,0334	0,9527	-	-
Кол-во информации на 1 символ при кодировании предложениями	0,9886	-	-	-

Визуальное отображение полученных результатов из таблицы 2 представлено на рисунке 2.

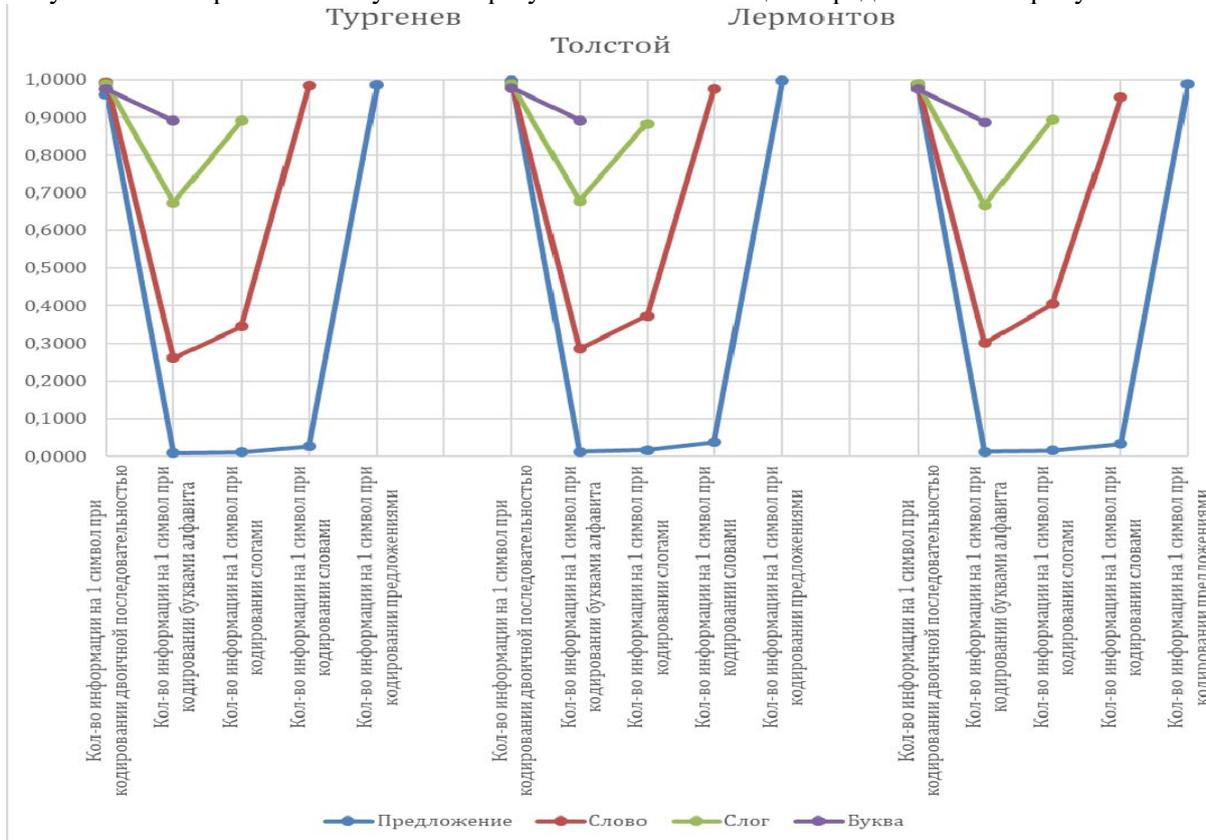


Рис. 2. Количество информации на единицу при различных способах кодирования

Анализ графика, представленного на рисунке 2, показал, что наибольшее количество информации, приходящейся на одно предложение приходится для случая кодирования предложениями, на втором месте по максимальному количеству информации на единицу предложения приходится при кодировании двоичной последовательностью, при этом при кодировании предложения буквами, слогами, словами количество информации растет в порядке возрастания сложности формирования лингвистической единицы от буквы до слова – т.е. минимальное количество информации, заключенное в одном предложении приходится при кодировании символами алфавита (буквами) и растет по мере перехода от букв к словам.

При сопоставлении каждому слову двоичной последовательности количество информации, приходящейся на одно слово, максимально, при этом минимальное значение количества информации на единицу слова приходится при кодировании буквами и также растет по мере структуризации кодирующей структуры до слова. При кодировании слогов исходного текста буквами алфавита количество информации, приходящейся на один слог, минимальное, при кодировании слогов исходного текста слогами количество информации, приходящейся на один слог, больше случая, когда кодируется буквами, но меньше случая, когда в соответствие каждому слогу ставится двоичная последовательность. Между кодированием букв, содержащихся в тексте, буквами алфавита и двоичной последовательностью, последний способ (кодирование двоичной последовательностью) показывает наибольшее значение количества информации, приходящейся на одну букву текста.

При рассмотрении зависимости количества информации, приходящейся на лингвистическую единицу, в разрезе способов кодирования, можно прийти к выводу, что в разрезе каждого отдельного способа кодирования количество информации, приходящееся на отдельную лингвистическую единицу, изменяется, уменьшается в соответствии со следующим порядком кодирования: <предложения, двоичная последовательность, слова, слоги, буквы>. В случае кодирования данных единиц бинарным кодом, среднее количество информации, которое соответствует единице кода еще меньше, чем количество информации при кодировании предложениями, но больше, чем при кодировании остальными рассмотренными лингвистическими единицами (слово, слог и буква).

Определим общее количество информации, которое содержится в каждом отрывке, путем перемножения среднего количества информации, приходящееся на единицу кодирования с количеством этих единиц.

Для отрывка текста Л.Н. Толстого «Детство», в котором содержится 43 предложения, 732 слова, 1351 слогов и 3209 букв, а количество информации на лингвистическую единицу (предложение, слово, слог, буква) при кодировании их двоичной последовательностью соответственно 0,9845, 0,9897, 0,9884 и 0,9753. Тогда, в случае представления исходного текста в виде предложений количество информации в нем равно 42,33 бит информации, для слов – 724,46 бит информации, слогов – 1 335,33 бит информации, а в виде букв – 3 129,74 бит информации.

Для отрывка М. Ю. Лермонтова «Герой нашего времени», в котором содержится 45 предложений, 732 слова, 1576 слогов и 3667 букв, а количество информации на предложение, слово, слог и букву при кодировании их двоичной последовательностью соответственно 0,9977, 0,9897, 0,9890 и 0,9788 бит. Тогда, в случае представления исходного текста в виде предложений количество информации в нем равно 44,90 бит информации, для слов – 724,46 бит информации, слогов – 1 558,66 бит информации, а в виде букв – 3 589,26 бит информации.

Для отрывка И. С. Тургенева «Отцы и дети», в котором содержится 34 предложений, 732 слова, 1597 слогов и 3796 букв, а количество информации на предложение, слово, слог и букву при кодировании их двоичной последовательностью соответственно 0,9610, 0,9927, 0,9881 и 0,9757 бит. Тогда, в случае представления исходного текста в виде предложений количество информации в нем равно 32,67 бит информации, для слов – 726,66 бит информации, слогов – 1 578,00 бит информации, а в виде букв – 3 703,76 бит информации.

Буквы алфавита – атомарная единица, из которой возможно получить другие лингвистические единицы. В связи с тем, что формирование предложений обуславливается более сложной системой построения и взаимосвязей, то формируется наиболее строгая иерархия по отношению к другим лингвистическим переменным, из-за чего предложение несет наибольшую избыточность в сравнении с другими лингвистическими единицами.

Когда текст содержит количество символов равное несколько десятков и их появление равновероятно, то такой текст считать бессмысленно хаотичным, как движение частиц, если вероятность появления символов подчинена какой-либо функции принадлежности, то данный текст возможно структурировать, т.е. поставить однозначной последовательности символов – определенную информацию.

Не зависимо от того, в каком кодовом виде рассматривается представление одного и того же текста, количество информации, заключенное в нем, должно быть неизменно, так как он описывает определенную предметную область и в нем заключена строго определенная смысловая нагрузка.

Исходя из строения органов, отвечающих за человеческую речь, можно предположить, что наиболее близкому представлению звуков соответствует представление текста в виде слогов.

Исходя из этого, чтобы определить количество информации, с минимальной долей избыточности, в любом текстовом файле, необходимо его разбить на слога и, в соответствии с формулой Шеннона, определить среднее количество информации, приходящейся на каждый слог, после чего произвести перемножение полученного значения с количеством слогов.

Так, для отрывка текста И.С. Тургенева «Отцы и дети» при сопоставлении каждому предложению текста последовательности слогов количество информации в тексте будет равняться 18, при сопоставлении каждому слову текста последовательности слогов количество информации в тексте будет 552, при сопоставлении каждому слогу текста последовательности слогов количество информации в тексте будет равняться 1424.

Для отрывка текста Л.Н. Толстого «Детство» при сопоставлении каждому предложению текста последовательности слогов количество информации в тексте будет равняться 21, при сопоставлении каждому слову текста последовательности слогов количество информации в тексте будет 548, при сопоставления каждому слогу текста последовательности слогов количество информации в тексте будет равняться 1209.

Для отрывка М. Ю. Лермонтова «Герой нашего времени» при сопоставлении каждому предложению текста последовательности слогов количество информации в тексте будет равняться 26, при сопоставлении каждому слову текста последовательности слогов количество информации в тексте будет 587, при сопоставлении каждому слогу текста последовательности слогов количество информации в тексте будет равняться 1393.

### **Заключение**

С точки зрения человеческого восприятия, наиболее законченной смысловой формой является предложение. Таким образом, в качестве отправного коэффициента при анализе текстовых данных принимается значение, равное отношению количества информации, полученное в результате представления текста предложениями из слогов и представлении текста в виде слогов, лежащее в границах  $[54;79]$  раз, при этом значение энтропии должно лежать в промежутке  $H(x)=[7,7; 8,1]$ .

Данные параметры определяют, что в исходных данных находится определенная смысловая нагрузка, которая может быть извлечена, а сами данные возможно представить в виде структуры в целях дальнейшего анализа.

Исходя из этого, предложенная структуризация текстового представления информации позволит облегчить работу по первоначальной обработке сырых данных, так как это сильно влияет на срок принятия решений, качество моделей, нейронных сетей и работоспособность искусственного интеллекта.

Соответственно появляется возможность в максимально короткие сроки получать результаты обработки и анализа для решений в определенной предметной области.

## Литература

1. *Гадасин Д. В., Шведов А. В., Пантелеева К. А.* Предобработка информации для систем машинного обучения // Актуальные проблемы и перспективы развития экономики : Труды XXI Международной научно-практической конференции, Симферополь-Гурзуф, 20-22 октября 2022 года. Симферополь: Крымский федеральный университет им. В.И. Вернадского, 2022. С. 268-269. EDN QVIOMF
2. *Maklachkova V. V., Shvedov A. V., Alyev S.* Analysis of Resilience Indicators in Corporate Networks and Possible Ways to Improve It // 2022 Systems of Signals Generating and Processing in the Field of on Board Communications, SOSG 2022 - Conference Proceedings, Moscow, 15–17 марта 2022 года. Moscow, 2022. DOI 10.1109/IEEECONF53456.2022.9744353. EDN ZRMVJS
3. *Гадасин Д. В., Шведов А. В., Кольцова А. В.* Вероятностная оценка построения виртуального кластера // Труды международного симпозиума "Надежность и качество". 2021. Т. 1. С. 87-92. EDN BCNSGB
4. *Яковенко Н. В., Шведов А. В., Пантелеева К. А., Гадасин Д. Д.* Средства реализации поисковых и контекстных механизмов для работы с большими данными // REDS: Телекоммуникационные устройства и системы. 2022. Т. 12. № 3. С. 56-63. EDN TBDOXX
5. *Гадасин Д. В., Пак Е. В., Коровушкина В. М., Мелькова Е. К.* Предобработка текстовой информации на основе термов естественного языка // REDS: Телекоммуникационные устройства и системы. 2022. Т. 12. № 1. С. 4-11. EDN PDGAVP
6. *Гадасин Д. В., Шведов А. В., Мелькова Е. К.* Структурирование данных исходя из центра масс // Актуальные проблемы и перспективы развития экономики : Труды XXI Международной научно-практической конференции, Симферополь-Гурзуф, 20-22 октября 2022 года. Симферополь: Крымский федеральный университет им. В.И. Вернадского, 2022. С. 266-268. EDN RFCCST
7. *Шведов А. В., Коноплева М. И.* Применение алгоритмов и способов преобразования речевого сигнала в цифровую информацию // Актуальные проблемы и перспективы развития экономики : Труды XXI Международной научно-практической конференции, Симферополь-Гурзуф, 20-22 октября 2022 г. Симферополь: Крымский федеральный университет им. В.И. Вернадского, 2022. С. 290-292. EDN HPOTTL
8. *Шведов А. В., Савин В. А., Мартынов М. Д.* Разработка приложения для синтаксического анализа сформированных в базу структурированных данных // Технологии информационного общества : Сборник трудов XVI Международной отраслевой научно-технической конференции, Москва, 02-03 марта 2022 года. М.: Издательский дом Медиа Паблишер, 2022. С. 167-169. EDN AUXGNF
9. *Алешинцев А. В., Сак А. Н.* Синтаксический анализ предложения с помощью функции tree в среде Matlab // Методические вопросы преподавания инфокоммуникаций в высшей школе. 2020. Т. 9. № 4. С. 43-45. EDN QTQRUT
10. *Гадасин Д. В., Шведов А. В., Вакурин И. С.* Определение семантической близости текстов с использованием алгоритма сравнения сущности графов // REDS: Телекоммуникационные устройства и системы. 2022. Т. 12. № 4. С. 11-19. EDN PVJKQJ
11. *Докучаев В. А., Кальфа А. А., Мытенков С. С., Шведов А. В.* Анализ технических решений по организации современных центров обработки данных // T-Comm: Телекоммуникации и транспорт. 2017. Т. 11. № 6. С. 16-24. EDN ZAOKJB
12. *Гадасин Д. В., Шведов А. В., Алексеева Е. А.* Информационная энтропия в стохастических сетях связи // Телекоммуникационные и вычислительные системы 2020 : Труды международной научно-технической конференции, Москва, 14-17 декабря 2020 года / Московский технический университет связи и информатики. М.: Научно-техническое издательство "Горячая линия-Телеком", 2020. С. 108-116. EDN IOGLQH
13. *Hilbert M., López P.* The world's technological capacity to store, communicate, and compute information // Science, no. 332(6025), pp. 60-65, 2011.
14. *Шведов А. В., Юдина А. А., Гадасин Д. Д.* Применение технологии роботизированной автоматизации процессов при решении задач кластеризации // Телекоммуникационные и вычислительные системы 2020 : Труды международной научно-технической конференции, Москва, 14-17 декабря 2020 года / Московский технический университет связи и информатики. М: Горячая линия – Телеком, 2020. С. 247-254. EDN OTMVC1
15. *Шведов А.В., Гадасин Д.В., Цыгулёва А.В., Вакурин И.С.* Разгрузка очереди сети при помощи Гамильтонова цикла // REDS: Телекоммуникационные устройства и системы. 2021. Т. 11. № 3. С. 45-53.
16. *Гадасин Д.В., Шведов А.В., Клыгина О.Г., Гадасин Д.Д.* Реализация платформы туманных вычислений для предоставления сервисов IoT // REDS: Телекоммуникационные устройства и системы. 2021. Т. 11. № 2. С. 65-75.
17. *Kalmykov N.S., Dokuchaev V.A.* Segment routing as a basis for software defined network // T-Comm. 2021. Т. 15. № 7. С. 50-54.

18. *Dokuchaev V.A., Maklachkova V.V., Statev V.Yu.* Classification of personal data security threats in information systems // T-Comm. 2020. Т. 14. № 1. С. 56-60.
19. *Докучаев В.А., Маклачкова В.В., Статьев В.Ю.* Цифровизация субъекта персональных данных // T-Comm: Телекоммуникации и транспорт. 2020. Т. 14. № 6. С. 27-32.
20. *Pavlov S.V., Dokuchaev V.A., Mytenkov S.S.* Model of a fuzzy dynamic decision support system // T-Comm. 2020. Т. 14. № 9. С. 43-47.
21. *Pavlov S.V., Dokuchaev V.A., Maklachkova V.V., Mytenkov S.S.* Features of supporting decision making in modern enterprise infocommunication systems // T-Comm. 2019. Т. 13. № 3. С. 71-74.
22. *Гадасин Д.В., Кольцова А.В., Гадасин Д.Д., Полякова А.Н.* Оценка вероятности формирования виртуального кластера // Системы синхронизации, формирования и обработки сигналов. 2021. Т. 12. № 1. С. 4-12.
23. *Кузин И.А., Гадасин Д.В.* Модель контейнера данных для минимизации трафика при передаче субъективных характеристик объектов на изображении трехмерной сцены // Телекоммуникации и информационные технологии. 2021. Т. 8. № 2. С. 96-100.
24. *Гадасин Д.В., Кольцова А.В., Полякова А.Н.* Модель построения кластера для пограничных вычислений // Труды Северо-Кавказского филиала Московского технического университета связи и информатики. 2020. № 1. С. 86-92.
25. *Усачева Д.И., Шишкин М.О., Гадасин Д.В., Гузеев А.В.* Применение OLAP-технологий для анализа многомерных данных в контакт-центре // Телекоммуникации и информационные технологии. 2019. Т. 6. № 1. С. 142-149.
26. *Гадасин Д.В., Кузин И.А.* Модель представления цветовых и глубинометрических характеристик объектов на изображении // DSPA: Вопросы применения цифровой обработки сигналов. 2021. Т. 11. № 1. С. 31-38.
27. *Гадасин Д.В., Нестерова Е.А.* Особенности проведения практических занятий по дисциплине мультимедийные информационные системы для стадии "исследование и обоснование создания информационной системы" // Методические вопросы преподавания инфокоммуникаций в высшей школе. 2021. Т. 10. № 1. С. 15-21.
28. *Shvedov A. V., Gadasin D. V., Alyoshintsev A. V.* Segment routing in data transmission networks // T-Comm. 2022. Vol. 16. No. 5. P. 56-62. DOI 10.36724/2072-8735-2022-16-5-56-62. EDN VAYLJQ
29. *Шведов А. В., Гадасин Д. В., Клыгина О. Г.* Организация взаимодействия туманных вычислений и сегментной маршрутизации для предоставления сервисов IoT в smart grid // Системы синхронизации, формирования и обработки сигналов. 2022. Т. 13. № 3. С. 40-49. EDN TRRYZN

## ЭФФЕКТИВНОСТЬ ТЕХНОЛОГИЙ IDMA И CDMA ПРИ МАЛОМ КОЭФФИЦИЕНТЕ РАСШИРЕНИЯ СПЕКТРА

**Горгадзе Светлана Феликсовна,**

*Московский технический университет связи и информатики, профессор, д.т.н.,  
Москва, Россия,  
[svetlana-gorgadze@yandex.ru](mailto:svetlana-gorgadze@yandex.ru)*

**Ермакова Анастасия Всеволодовна,**

*Московский технический университет связи и информатики, аспирант,  
Москва, Россия,  
[msikisyliia@gmail.com](mailto:msikisyliia@gmail.com)*

### Аннотация

Для случая многостанционного асинхронного доступа абонентов, работающих в общей полосе частот, производится сравнение двух его технологий – IDMA и DS-CDMA – по величине вероятности ошибки на передаваемый информационный символ. Рассматриваются три варианта линейных алгоритмов многопользовательского детектирования сигналов – по минимуму среднеквадратической ошибки (minimum mean square error – MMSE), алгоритм, основанный на использовании согласованного фильтра (matched filter), в зарубежных источниках называемый алгоритмом rake, а также метод обработки, без «подавления» шума с отводов применяемой линии задержки, известный как soft-rake detector. Проведенные исследования для коэффициента расширения спектра, равного четырем, свидетельствуют о преимуществе технологии IDMA, по сравнению с DS-CDMA при одновременной работе 6-8 абонентов. Делается вывод о необходимости дальнейших исследований при увеличении коэффициента расширения спектра информационного сигнала.

**Ключевые слова:** IDMA, DS-CDMA, многопользовательское детектирование, итеративное обнаружение и декодирование.

### Введение

Технология множественного доступа с разделением сигналов абонентов на основе перемежения символов (interleave division multiple access – IDMA) является вариантом не ортогонального доступа (Non-orthogonal Multiple Access – Noma), рассматриваемого в рамках стандартизации 3GPP [1,2]. Считается, что она является вариантом DS-CDMA, но без использования псевдослучайной последовательности (ПСП) для расширения спектра. Последнее в случае IDMA реализуется на основе помехоустойчивого кодирования с последующим повторением и перемежением полученных символов, причем алгоритм перемежения является индивидуальным у каждого абонента и используется на приемной стороне для их разделения [1,3,5]. В этом случае скорость помехоустойчивого кода будет более низкой, по сравнению со скоростью кода, используемого в сочетании с DS-CDMA при такой же скорости передачи информации и в полосе частот с такой же шириной. Как показано ниже, скорость помехоустойчивого кода с учетом повторения символов при IDMA должна быть в  $N_u N_{\text{ck}}$  раз меньше, по сравнению с DS-CDMA, где  $N_u N_{\text{ck}}$  – коэффициент

расширения спектра [4,6]. Но очевидно, что принципиальным отличием этих технологий является то, что при IDMA разделение сигналов абонентов производится по принятым символам кода, а при DS-CDMA – непосредственно по канальным сигналам. То есть в первом случае используется более «мощный» вариант помехоустойчивого кодирования при полном отсутствии «разделения» канальных сигналов, а во втором – более «слабый» вариант помехоустойчивого кодирования в сочетании с разделением канальных сигналов. Вместе с тем, при DS-CDMA вместе с разделением канальных сигналов может использоваться и разделение в соответствии с алгоритмом перемежения информационных символов на выходе помехоустойчивого кодера.

### 1. Модели сигналов и канала связи

Обобщенная функциональная схема устройства формирования как IDMA, так и CDMA сигнала представлена на рисунке 1.

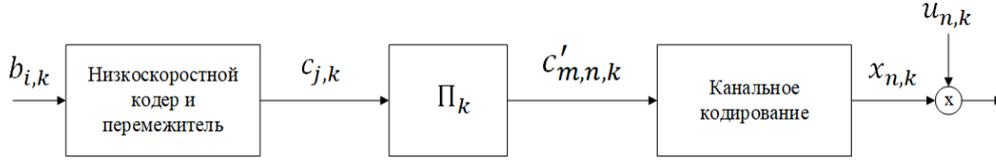


Рис. 1. Обобщенная структурная схема устройства формирования IDMA и CDMA сигналов.

В этом устройстве поток двоичных информационных символов каждого -го абонента  $b_{i,k}$  ( $k \in \{1, \dots, K\}, i \in \{1, \dots, N_b\}$ ) кодируется на основе помехоустойчивого сверточного кода со скоростью  $R_c$  в поток двоичных символов, после чего производится их повторение при увеличении скорости потока в  $R_r$  раз. Результирующая скорость низкоскоростного помехоустойчивого кода равняется  $R = R_c R_r$ . Операция перемежения полученных двоичных символов  $c_{j,k}$ , ( $j \in \{1, \dots, N_c\}, N_c > N_b$ ) производится в блоке  $\Pi_k$  на интервале длительностью в  $N_c$  символов, в результате чего формируется поток двоичных символов, которые разбиваются на слова  $c'_{m,n,k}$  длиной по  $N_c$  символов, где  $m$  – номер символа -го слова. Таким образом, получаем поток слов  $c'_{1,n,k} \dots c'_{N_m,n,k}$ ,  $n = 1, 2, \dots, N_s$ .

Отметим, что алгоритм перемежения символов в пределах слова является индивидуальным у каждого абонента, и в случае IDMA используется для его идентификации на приемной стороне. Затем производится канальное кодирование путем отображения вышеупомянутых кодовых слов на комплексные канальные символы  $\dot{x}_{n,k} \in \{\dot{\alpha}_1, \dots, \dot{\alpha}_{2^{N_m}}\}$ , удовлетворяющие условиям:  $\sum_{i=1}^{2^{N_m}} \dot{\alpha}_i = 0$ ,  $(1/2^{N_m}) \sum_{i=1}^{2^{N_m}} |\dot{\alpha}_i|^2 = 1$ . В случае IDMA процесс кодирования исходного потока информационных символов на этом завершается. Таким образом, каждый информационный символ  $b_{i,k}$  передается с помощью последовательности комплексных канальных символов  $\dot{x}_{n,k}$ , причем их число  $N_u$  зависит от значений  $R$  и  $N_m$ , а длительность каждого канального сигнала, формируемого на основе  $\dot{x}_{n,k}$ , определяет ширину используемой полосы частот.

В случае CDMA длительность канального сигнала, соответствующего каждому символу  $\dot{x}_{n,k}$ , должна быть в  $N_u$  раз больше, по сравнению с IDMA. Тогда его перемножение с символами псевдослучайной последовательности (ПСП)  $u_1 u_2 \dots u_{N_u}$ , индивидуальной для каждого абонента, приведет к расширению спектра канального сигнала в  $N_u$  раз, в результате ширина занимаемой полосы частот и скорость передачи информационных символов  $b_{i,k}$  будет одинаковой при IDMA и CDMA. Поэтому в случае CDMA необходимо, чтобы скорость  $R$  результирующего помехоустойчивого кода была более высокой, по сравнению с IDMA. Таким образом, канальный сигнал, формируемый на основе двоичной ПСП  $u_1 u_2 \dots u_{N_u}$  в случае CDMA, и канальные сигналы, формируемые на основе последовательности символов  $\dot{x}_{1,k}, \dots, \dot{x}_{N_u,k}$  в случае IDMA, передают одно и то же количество исходных информационных символов  $b_{i,k}$ , то есть по одному символу. Причем длительность элементарного символа (чипа) CDMA сигнала в канале связи  $T_c T_c$  равняется длительности канального информационного сигнала, формируемого на основе каждого символа  $\dot{x}_{n,k}$  при IDMA.

В дальнейшем будем рассматривать последовательность символов  $\dot{u}_1 \dot{u}_2 \dots \dot{u}_{N_u}$  в качестве канальных информационных символов IDMA сигнала, считая, что  $x_{n,k} = u_n$ , поскольку расширение спектра сигнала в этом случае происходит при передаче случайной последовательности канальных информационных символов  $\dot{u}_n$ . В случае CDMA каждый символ ПСП  $u_1 u_2 \dots u_{N_u}$  умножается на  $\dot{x}_{n,k}$ , в результате чего получается последовательность комплексных канальных элементарных символов  $\dot{u}_1 \dot{u}_2 \dots \dot{u}_{N_u}$ , соответствующих чипам (элементарным импульсам) CDMA сигнала. При этом  $T_s = T_c N_u$  – длительность канального информационного сигнала.

Модель многолучевого канала связи для  $k$ -го абонента ( $k = 1, \dots, K$ ) представим в виде КИХ-фильтра с числом отводов  $v_k$ , соответствующим длительности времени задержки по времени его сигнала в нем, равном  $v_k T_c$ . При этом  $\sum_{l=1}^{v_k} |h_{lk}|^2 = 1$ , где  $h_{lk}$  – весовой коэффициент отвода фильтра, а  $l = 1, \dots, v_k$ . Кроме того, сигнал каждого абонент дополнительно сдвинут по времени на  $\tau_k$  отсчетов относительно условно нулевого момента времени из-за несовершенства работы системы сетевой синхронизации. Тогда задержка по времени сигнала  $k$ -го абонента на входе приемника составит  $(v_k + \tau_k)$  отсчетов сигнала (по одному отсчету на символ или чип), а максимальная временная задержка по всем  $K$  абонентам составит  $L_c \triangleq \max_k (v_k + \tau_k)$  символов (чипов). Эта задержка, представленная как число канальных информационных сигналов, составит  $L_s \triangleq [L_c / N_u]$ , где  $[\bullet]$  – целая часть от числа.

В соответствии с вышесказанным, сигнал каждого абонента на входе приемника можно рассматривать как дискретную свертку

$$\mathbf{g}_{n,k} \triangleq a_k \mathbf{u}'_{n,k} * \mathbf{h}_k \quad (1)$$

векторов  $\mathbf{u}'_{n,k} \triangleq [\mathbf{u}_{n,k}^m, \mathbf{0}_{N_u}^m]^m$  и  $\mathbf{h}_k \triangleq [0_{\tau_k}^m, h_{0,k}, \dots, h_{v_k,k}, 0_{L_c - \tau_k - v_k}^m]^T$  длиной  $2N_u$  и  $(L_c + 1)$  соответственно, где  $0_{N_u}^m, 0_{\tau_k}^m$  и  $0_{L_c - \tau_k - v_k}^m$  – обозначение столбцов, состоящих из нулей длиной  $N_u, \tau_k, (L_c - \tau_k - v_k)$  соответственно,  $T$  операция транспонирования матрицы,  $a_k$  – амплитуда сигнала, учитывающая расстояние между абонентом и многопользовательским приемником.

Вектор длиной  $N_u$ , описывающий групповой сигнал  $KK$  абонентов на входе многопользовательского приемника, задержанный на  $l$  символов в канале связи, можно рассматривать как  $\dot{\mathbf{y}}_{n+l} \triangleq [y_{n+lN_u}, \dots, y_{n+(l+1)N_u-1}]^T$ . Он может быть представлен как  $\dot{\mathbf{y}}_n = \sum_{l=0}^{L_s} \mathbf{G}(n-l, l) \dot{\mathbf{x}}_{n-l} + \dot{\boldsymbol{\eta}}_n$ , где  $\dot{\mathbf{x}}_n \triangleq [x_{n,1}, \dots, x_{n,K}]^T$  – вектор длиной  $K$ , соответствующий сигналам абонентов,  $\dot{\boldsymbol{\eta}}_{n+l} \triangleq [\eta_{n+lN_u}, \dots, \eta_{n+(l+1)N_u-1}]^T$  – вектор шума, а  $\mathbf{G}(n, l)$  – матрица, описывающая канал связи. При этом

$$\mathbf{G}(n, l) \triangleq [\mathbf{g}_1(n, l), \dots, \mathbf{g}_K(n, l)] - \quad (2)$$

– матрица размером  $N_u \times K$ , а

$$\mathbf{g}_k(n, l) \triangleq [\mathbf{g}_{n,k}[lN_u + 1], \dots, \mathbf{g}_{n,k}[(l+1)N_u]]^T - \quad (3)$$

вектор длиной  $N_u$ .

С целью получения модели со всеми возможными относительными сдвигами канальных сигналов  $x_{n,k}$  всех  $K$  абонентов рассмотрим скользящее временное окно длиной  $2L_s + 1$ . Тогда матрицу входных полезных сигналов канала связи можно представить в виде вектора:  $\mathbf{x}_n \triangleq [\dot{\mathbf{x}}_{n-L_s}^m, \dots, \dot{\mathbf{x}}_n^m, \dots, \dot{\mathbf{x}}_{n+L_s}^m]^T$ . Затем для входного сигнала на приемной стороне можно записать:

$$\mathbf{y}_n = \mathbf{H}_n \mathbf{x}_n + \boldsymbol{\eta}_n, \quad (4)$$

где  $\mathbf{y}_n \triangleq [\mathbf{j}_n^m, \dots, \mathbf{j}_{n+L_s}^m]^T$ ,  $\boldsymbol{\eta}_n \triangleq [\boldsymbol{\eta}_n^m, \dots, \boldsymbol{\eta}_{n+L_s}^m]^T$  и, наконец,

$$\mathbf{H}_n \triangleq \begin{bmatrix} \mathbf{G}(n-L_s, L_s) & \dots & \mathbf{G}(n, 0) \\ \vdots & \ddots & \vdots \\ \mathbf{G}(n, L_s) & \dots & \mathbf{G}(n+L_s, 0) \end{bmatrix} \quad (5)$$

матрица размером  $(L_s + 1)N_u \times (2L_s + 1)K$ , описывающая канал связи.

Заметим, что матрицу  $\mathbf{H}_n$  можно представить как:

$$\mathbf{H}_n \triangleq \left[ [h_{n-L_s,1}, \dots, h_{n-L_s,K}], \dots, [h_{n+L_s,1}, \dots, h_{n+L_s,K}] \right].$$

Тогда (4) можно переписать в как:

$$\mathbf{y}(n) = \sum_{k=1}^K \sum_{l=-L_s}^{L_s} h_{n-l,k} x_{n-l,k} + \boldsymbol{\eta}_n. \quad (6)$$

Таким образом, с учетом (6) модель группового канала связи можно проиллюстрировать с помощью рисунка 2.

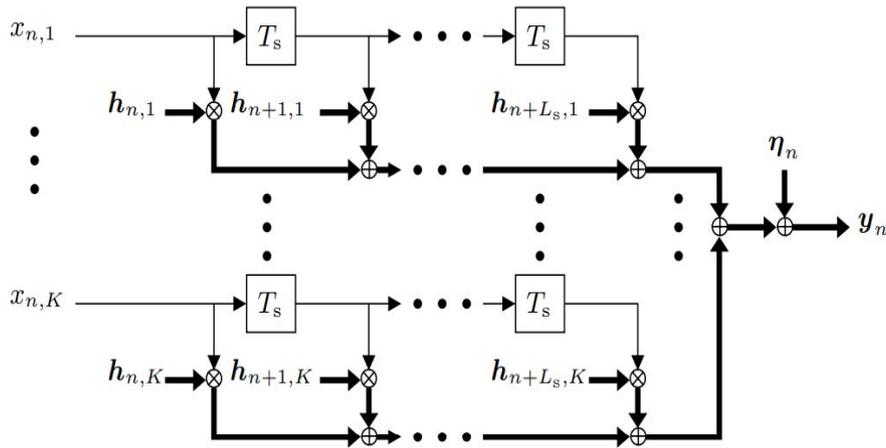


Рис. 2. Модель группового канала связи.

## 2. Алгоритмы многопользовательского детектирования и декодирования

Для декодирования помехоустойчивого кода в данном случае следует использовать итеративную версию правила максимума апостериорной вероятности [3,8]. Функциональная схема соответствующего приемного устройства представлена на рисунке 2. Она включает многопользовательский детектор и  $K$  декодеров, вычисляющих апостериорные вероятности

символов, переданных абонентами. Эти декодеры в дальнейшем будем называть декодерами апостериорной вероятности. Они подключены к многопользовательскому детектору с помощью  $K$  перемешителей ( $\Pi_K$ ) и соответственно  $K$  деперемешителей  $\Pi_K^{-1}$ . При этом многопользовательский

детектор и эти декодеры являются компонентами «мягкого» декодера и итеративно обмениваются значениями вычисленных логарифмических отношений правдоподобия (LLR). Таким образом, многопользовательский детектор получает от декодеров апостериорной вероятности априорные

логарифмические отношения правдоподобия о кодовых битах, переданных каждым  $n$ -ым абонентом,

определяемые как  $L_a^m(c'_{m,n,k}) \triangleq \ln \frac{P\{c'_{m,n,k} = +1\}}{P\{c'_{m,n,k} = -1\}}$ .

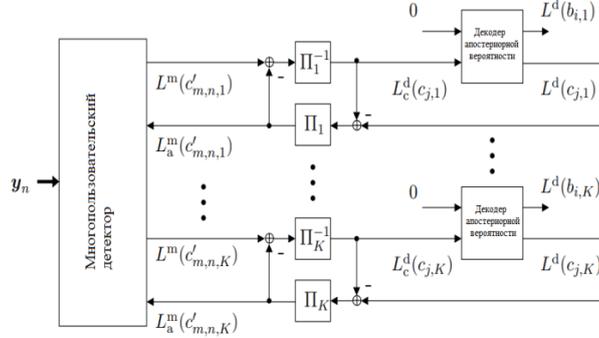


Рис. 3. Функциональная схема итеративной многопользовательской схемы обнаружения и декодирования.

Перед первой итерацией отсутствует информация от декодера, и  $c'_{m,n,k}$  равновероятно может принимать значения  $+1$  и  $-1$ , в результате чего перед первой итерацией  $L_a^m(c'_{m,n,k}) \triangleq 0$ . Затем с учетом принятого вектора наблюдения входного сигнала  $y_n$  с учетом поступивших значений  $L_a^m(c'_{m,n,k})$  в многопользовательском детекторе вычисляются апостериорные логарифмические отношения правдоподобия

$$L^m(c'_{m,n,k}) \triangleq \ln \frac{P\{c'_{m,n,k} = +1 | y_n\}}{P\{c'_{m,n,k} = -1 | y_n\}}, \quad (7)$$

а также внешние логарифмические отношения правдоподобия  $L_e^m(c'_{m,n,k}) \triangleq L^m(c'_{m,n,k}) - L_a^m(c'_{m,n,k})$ . Они через депережежитель отправляются в декодер апостериорной вероятности.

В соответствии с теоремой о полной вероятности можно получить:

$$L_e^m(c'_{m,n,k}) = \ln \frac{\sum_{\substack{x \in A \\ c'_{m,n,k} = +1}} p\{y_n | x_{n,k} = x\} \prod_{m' \neq m} P\{c'_{m',n,k} = d_{m'}\}}{\sum_{\substack{x \in A \\ c'_{m,n,k} = -1}} p\{y_n | x_{n,k} = x\} \prod_{m' \neq m} P\{c'_{m',n,k} = d_{m'}\}}, \quad (8)$$

где  $d_{m'} \in \{+1, -1\}$  -  $m'$ -й символ принятого слова  $x$ . Эти слова отображаются на  $x_n$ . В результате  $P\{c'_{m,n,k} = d_{m'}\} = \frac{1}{2} \left( 1 + d_{m'} \tanh \left( \frac{L_a^m(c'_{m,n,k})}{2} \right) \right)$ .

Таким образом, как показано на рисунке 3, каждый из  $K$  декодеров апостериорной вероятности получает последовательность внешних логарифмических отношений правдоподобия  $L_e^m(c'_{m,n,k})$  от многопользовательского детектора в виде  $L_c^d(c_{j,k})$ , на основе которых в соответствии со значениями кодовых ограничений каждый декодер вычисляет апостериорные логарифмические отношения правдоподобия  $L^d(c_{j,k})$ , которые являются улучшенными логарифмическими отношениями правдоподобия кодовых символов. Затем логарифмические отношения правдоподобия  $L_c^d(c_{j,k})$  вычитаются из  $L^d(c_{j,k})$ , в результате чего получают так называемые внешние логарифмические отношения правдоподобия  $L_e^d(c_{j,k})$ , то есть  $L_e^d(c_{j,k}) \triangleq L^d(c_{j,k}) - L_c^d(c_{j,k})$ , которые отправляются в многопользовательский детектор как новые априорные логарифмические отношения правдоподобия  $L_a^m(c'_{m,n,k})$  после перемежения в блоке  $\Pi_k$ .

В свою очередь, многопользовательский детектор вычисляет и выдает новые логарифмические отношения правдоподобия  $L^m(c'_{m,n,k})$ . После некоторого количества итераций декодер вычисляет апостериорные логарифмические отношения правдоподобия для информационных символов  $L_b(b_{i,k})$ , а также оценки символов как  $\hat{b}_{i,k} \triangleq \text{sing}(L^d(c_{j,k}))$ .

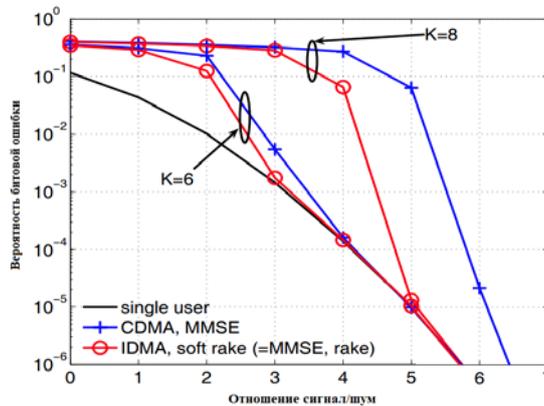
Далее рассматриваются три типа квазиоптимальных линейных многопользовательских детекторов – MMSE, rake detector, soft-rake detector, которые используют априорную информацию, поступающую от декодеров апостериорной вероятности. Принципы их работы подробно рассмотрены в [3,8-10].

### 3. Результаты компьютерного моделирования

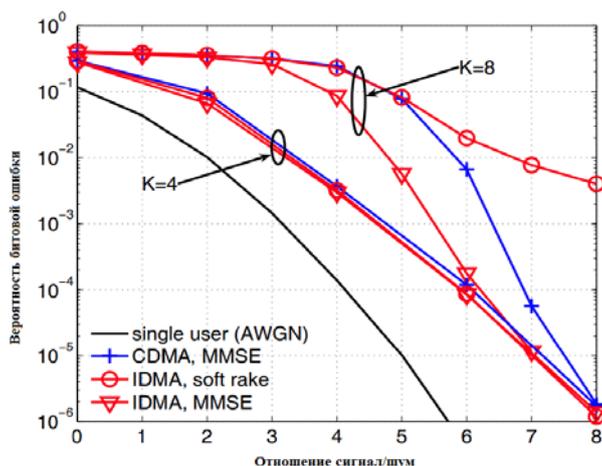
В [3] проводилось моделирование групповых каналов связи с технологиями IDMA и CDMA. С этой целью исходный поток двоичных информационных символов каждого  $i$ -го абонента  $b_{i,k}$  разбивался на блоки длиной по  $N_b = 1024$  символов. Затем производилось сверточное кодирование этого потока на основе одного и того же для всех абонентов генераторного полинома при скорости  $R_c = 1/2$  с кодовым ограничением 4. Таким образом, на выходе кодера было получено  $(N_b + 4) / R_c = 2056$  информационных символов.

Далее, для случая IDMA символы повторялись, в результате чего скорость кода увеличивалась в  $R_r = 1/4$  раза. Результирующее число кодовых символов  $c_{j,k}$  составило  $(N_b + 4 / R_c) / R_r = 8224$ . Затем эти символы чередовались при индивидуальном у каждого абонента алгоритме перемежения, в результате чего были получены символы  $c'_{m,n,k}$ . Длина блока символов, в пределах которого производилось перемежение в случае IDMA составило 8224 кодовых символа, в случае CDMA – 2056 символа, поскольку при CDMA повторения символов не производилось. Алгоритм перемежения генерировался случайным образом.

Каждые два символа  $[c'_{1,n,k} c'_{2,n,k}]$  отображались на каналные символы  $\left\{ \frac{1}{\sqrt{2}} + \frac{j}{\sqrt{2}}, \frac{1}{\sqrt{2}} - \frac{j}{\sqrt{2}}, -\frac{1}{\sqrt{2}} + \frac{j}{\sqrt{2}}, -\frac{1}{\sqrt{2}} - \frac{j}{\sqrt{2}} \right\}$  при QPSK модуляции несущей частоты. В случае CDMA поток канальных символов умножался на сверхдлинную ПСП так, на каждый канальный символ QPSK  $x_{n,k}$  приходилось по  $N_{ит} = 4$  символа ПСП. То есть коэффициент расширения спектра составил всего 4. Отметим, что в случае CDMA перемежение символов может дополнительно использоваться для разделения информационных потоков абонентов. Результаты моделирования для однолучевого канала представлены на рисунке 4, а для многолучевого канала при числе лучей, равном 16, - на рисунке 5.



**Рис. 4.** Зависимости вероятности ошибки на информационный символ от отношения сигнал/шум по мощности для однолучевого канала с аддитивным белым гауссовским шумом при CDMA (MMSE) и IDMA (soft rake) при  $K = 6, 8$  после 10 итераций.



**Рис. 5.** Зависимости вероятности битовой ошибки от отношения сигнал/шум по мощности для систем CDMA и IDMA при  $K = 4,8$  после 10 итераций для многолучевого канала связи с 16 лучами.

Анализ рисунков 4 и 5 свидетельствует о преимуществах технологии IDMA, по сравнению с CDMA, особенно при увеличении числа одновременно действующих абонентов в многолучевом канале связи. Но проведенный эксперимент соответствует слишком малому значению коэффициента расширения спектра, когда преимущества CDMA еще не могут проявиться в полной мере, так как при длине ПСП, использованной для расширения спектра в описанном эксперименте, эффективное разделение асинхронных CDMA сигналов невозможно из-за высокого уровня пиков взаимно корреляционных функций их суммы, по сравнению с уровнем основного пика автокорреляционной функции каждого из них [4,7]. В данном случае уровни суммы пиков взаимно корреляционных функций и основного пика автокорреляционной функции очевидно сопоставимы по своим значениям.

### Заключение

Как показали проведенные исследования, несмотря на довольно малый коэффициент расширения спектра, равный лишь четырем, технология DS-CDMA при малом числе одновременно действующих абонентов, равном также четырем, практически не проигрывает IDMA, хотя при DS-CDMA используется более «слабый» метод помехоустойчивого кодирования без повторения символов на выходе кодера. Но при увеличении числа одновременно действующих асинхронных абонентов IDMA явно превосходит CDMA, так как в последнем случае уровень взаимно корреляционных пиков асинхронных суммы сигналов превышает уровень основного пика автокорреляционной функции, то есть «разделения» сигналов по форме не происходит. IDMA также превосходит CDMA по эффективности в случае, если помехоустойчивое кодирование не производится, а лишь повторение символов при IDMA. В этом случае используется итеративный алгоритм обработки повторяющихся символов.

Вместе с тем, очевидно, что увеличение коэффициента расширения спектра, если оно возможно, приведет существенному усложнению алгоритма детектирования и декодирования сигналов при IDMA и повышение эффективности «разделения» канальных сигналов при DS-CDMA. Данный вопрос в сочетании с разработкой и исследованием алгоритмов обработки сигналов представляет существенный научный и практический интерес.

### Литература

1. 3GPP TR 38.812 V16.0.0 (2018-12), Technical Report, 3rd Generation Partnership Project; Technical Specification Group Radio Access Network; Study on Non-Orthogonal Multiple Access (NOMA) for NR (Release 16), 2018.
2. Бакулин М.Г., Бен Режеб Т.Б.К., Крейнделин В.Б., Панкратов Д.Ю., Смирнов А.Э. Технология NOMA с кодовым разделением в 3GPP: 5G или 6G? // Т-Comm: Телекоммуникации и транспорт. 2022. Том 16. №1. С. 4-14.
3. Kusume K., Bauch G., Utschick W. IDMA vs. CDMA: Analysis and Comparison of Two Multiple Access Schemes // IEEE Trans. on Wireless Comm. 2012. V. 11, № 1. p.p.78-87
4. Варакин Л.Е. Системы связи с шумоподобными сигналами. М.: Радио и связи, 1985. 384 с.
5. Beard C., Stallings W. Wireless Communication Networks and Systems. L.: Pearson, 2016.
6. Пестряков В.Б., Афанасьев В.П., Гурвиц В.Н. Шумоподобные сигналы в системах передачи информации / Под ред. В.Б. Пестрякова. М.: Сов. Радио, 1973. 424 с.
7. Гут Т.М., Горгадзе С.Ф. Характеристики ковариационных функций и оценка параметров шумоподобного сигнала // Телекоммуникации и информационные технологии. 2019. Т. 6. № 2. С. 35-41.
8. Leung K., Liu L., Ping L. Interleaving-based multiple access and iterative chip-by-chip multiuser detection // IEICE Trans. Commun., vol. E86-B, no. 12, pp. 3634-3637, Dec. 2003.
9. Ping L., Liu L., Leung W. K. A simple approach to near-optimal multiuser detection: interleave-division multiple-access // Proc. 2003 IEEE Wireless Commun. Netw. Conf., vol. 1, pp. 391-396.
10. Hagenauer J. The Turbo principle: tutorial introduction and state of the art // Proc. 1997 International Symp. Turbo Codes Related Topics, pp. 1-11.

## СРАВНЕНИЕ РАЗЛИЧНЫХ МЕТОДОВ КЛАССИФИКАЦИИ РУКОПИСНЫХ ЦИФР НА НАБОРАХ ДАННЫХ С ПРЕДВАРИТЕЛЬНОЙ ПОДГОТОВКОЙ И БЕЗ НЕЕ

**Калушев Константин Александрович,**

*Московский технический университет связи и информатики, магистрант, Москва, Россия,*

[konstantin.kalushev@gmail.com](mailto:konstantin.kalushev@gmail.com)

### **Аннотация**

*В статье рассматривается применение некоторых методов классификации для распознавания рукописных цифр на базе двух различных наборов их изображений: без шумов (MNIST) и с шумами (DIDA). Приведено описание экспериментов и результаты. По итогам сделан вывод о высокой эффективности фактически любых методов классификации на наборе данных с предварительной подготовкой без шумов. Продемонстрировано, что в условиях наличия в данных шумов и отсутствия предварительной обработки использование нейронных сетей имеет явные преимущества в эффективности перед другими методами классификации*

**Ключевые слова:** *Машинное обучение, методы классификации, распознавание рукописных цифр, DIDA, MNIST, нейронные сети*

### **Введение**

Классической задачей машинного обучения является распознавание рукописных цифр. В принципе, для распознавания рукописных цифр могут использоваться различные методы: от метода К-ближайших соседей до сверточных нейронных сетей.

Эксперименты по распознаванию рукописных цифр обычно проводятся на базе набора изображений MNIST, который характеризуется хорошей подготовкой изображений и отсутствием шумов. Так, известный исследователь Ян Лекун приводит сведения о применении более десяти методов на данном наборе, при этом даже для тривиального метода к-ближайших соседей эффективность распознавания может достигать 95% [1].

Альтернативой выступает недавно появившийся набор данных DIDA, цифры в котором представляют собой части фотографических изображений со всеми шумами и некоторыми артефактами, включая фрагменты иных символов.

Учитывая наличие двух различных наборов изображений цифр, представляется интересным поставить вопрос в сопоставительном ключе: какой метод классификации в машинном обучении является наиболее эффективным в смысле сравнения его применения на двух разных наборах данных – с шумами (DIDA) и без шумов (MNIST).

### **Описание тестовых наборов данных и предварительная обработка**

Набор данных выступает отправной точкой проводимого исследования. Как правило, эксперименты по распознаванию рукописных цифр проводятся с использованием набора цифр MNIST (сокращение от «Modified National Institute of Standards and Technology»). База данных MNIST содержит 60000 изображений для обучения и 10000 изображений для тестирования, каждое из которых имеет разрешение 28x28 пикселей. На основании данной базы проводятся регулярные попытки достичь минимальной ошибки, которые обсуждаются в научной литературе.

База данных является стандартом, предложенным Национальным институтом стандартов и технологий США с целью калибрации и сопоставления методов распознавания изображений с помощью машинного обучения в первую очередь на основе нейронных сетей. Сведения о базе данных, а также стандарты сбора информации были впервые опубликованы в 1995 году в статье «NIST Special Database 19. Handprinted form and characters Database» [2].

Существуют также и иные наборы данных рукописных цифр. В их ряду стоит отметить набор рукописных цифр DIDA (сокращенно от Digit Dataset), который в полной версии состоит из 250 000 рукописных образцов, разбитых на классы, соответствующие цифрам от 0 до 9 с 20 000 – 25 000 образцов в каждом классе. Данный набор изображений цифр был получен из шведских исторических документов, подготовленных в период с 1800 по 1940 годы.

В отличие от MNIST в DIDA образцы цифр не нормализованы и не очищены от шумов [3], более того, изображения некоторых цифр подвержены исторической деградации, написаны разными чернилами на бумаге разного оттенка.

С точки зрения машинной обработки в данных наборах данных можно выделить следующие различия.

Таблица 1

Сравнение наборов изображений цифр MNIST и DIDA

Показатель	MNIST	DIDA
Разрешение изображений (пикселей)	28x28	Различное разрешение (например 25x45, 38x40)
Фон изображений	Черный	Различный в зависимости от фото
Наличие артефактов (элементы других символов, помарки и т.д.)	Отсутствуют	Присутствуют
Формат изображений	Векторизованные в единой базе данных	Отдельные изображения в формате jpg

Наборы изображений цифр MNIST и DIDA имеют и чисто визуальные отличия. Ниже на рисунке 5 приводим их визуальное сопоставление.

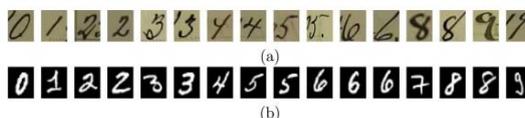


Рис. 1. Образцы из наборов цифр (a) DIDA и (b) MNIST [23]

Для проведения экспериментов при подготовке данной работы использовались два набора изображений рукописных цифр - ставший уже классическим набор обработанных изображений MNIST (в полной версии 60 000 обучающих и 10 000 тестовых изображений) и относительно недавно опубликованный набор изображений без какой-либо предварительной обработки DIDA (в урезанной версии 70 000 изображений).

**Подход к проведению экспериментов**

Для проведения экспериментов по распознаванию цифр на двух наборах данных автором были выбраны следующие методы классификации:

- Метод k-ближайших соседей
- Метод случайного леса
- Метод опорных векторов (полиномиальное ядро)
- Нейронная сеть прямого распространения (различные архитектуры).

Для применения указанных методов автором использовался язык программирования Python и его стандартная библиотека scikit-learn версии 1.2.0, содержащая стандартные классы для всех указанных выше методов. Итоговая эффективность каждого метода оценивалась с помощью метода score из той же библиотеки, определяющего соотношение правильных прогнозов к общему количеству элементов тестовой выборки.

Данные набора изображений цифр MNIST использовались без предварительной обработки, так как они «поставляются» в уже векторизованном виде. В связи с тем, что данные набора цифр DIDA представляют из себя цветные изображения разного разрешения в формате jpg, автором было проделано следующее:

Все изображения были конвертированы в разрешение 28x28 пикселей с помощью функции `resize` библиотеки Python PIL;

Полученные изображения были переведены в градации серого с помощью функции `grayscale` также библиотеки Python PIL;

Изображения были промаркированы от «0» до «9» и векторизованы



Рис. 2. Трансформация изображений в рамках приведения к единому формату

Как набор изображений MNIST, так и набор изображений DIDA были разделены на обучающую и тестовую совокупности в соотношении 80:20.

### Методы классификации, не основанные на использовании нейронных сетей

В качестве отправной точки было проведено сравнение эффективности классификации с помощью методов, не основанных на использовании нейронных сетей. Первым из таких методов выступил простейший - *метод k-ближайших соседей*.

Основным параметром при применении метода k-ближайших соседей выступает показатель k – количество элементов, расстояние до которых влияет на выбор итоговой метки класса для данного наблюдения. Изменяя параметр k, приведем результаты использования метода k-ближайших соседей.

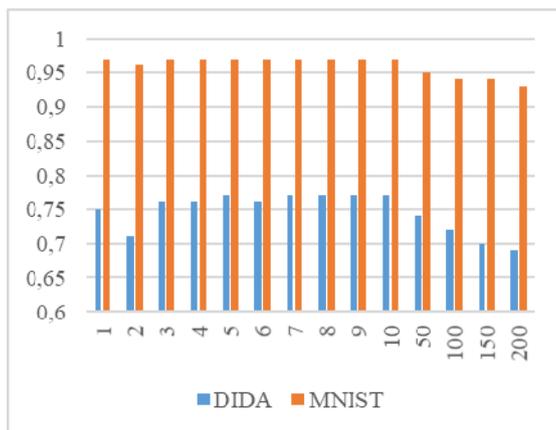
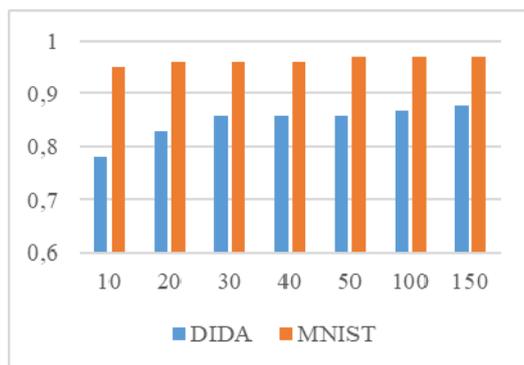


Рис. 3. Эффективность применения метода k-ближайших соседей (для различных k) для классификации рукописных цифр в наборах данных DIDA и MNIST

Таким образом, метод k-ближайших соседей, в его применении для различных наборов данных цифр показывает следующий результат. На предварительно обработанных данных, которые представляет собой набор цифр MNIST, метод k-ближайших соседей показывает впечатляющий результат распознавания в 97%. Вместе с тем, эффективность распознавания набора необработанных данных DIDA данным методом составляет всего 77%.

Следующим методом, применение которого мы исследуем для двух наборов цифр будет *метод случайного леса*. Данный метод заключается в использовании ансамбля деревьев решений. Как известно, к достоинствам данного метода относят возможность эффективно работать с большим числом признаков. После тренировки деревьев решений предсказание относительно нового элемента x осуществляется по формуле большинством «голосов» относительно общего количества деревьев решений [4], которое является основным изменяемым параметром модели.

Ниже приведем результаты использования метода случайного леса для наборов данных DIDA и MNIST:

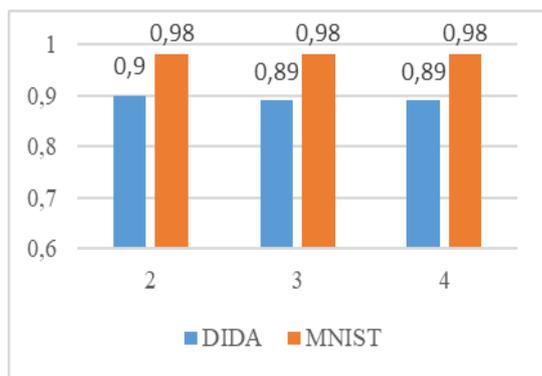


**Рис. 4.** Эффективность использования метода случайного леса для классификации изображений (для различного количества «деревьев») в наборах данных DIDA и MNIST

Как и в предыдущем случае, метод случайного леса показывает прекрасные результаты эффективности на предварительно обработанном наборе данных MNIST, где при значительном количестве «деревьев» его эффективность опять достигает 97%. Вместе с тем, на наборе данных без предварительной обработки DIDA подобная эффективность все еще не достижима и в наилучшем случае, при сопоставимом с наилучшим результатом набора MNIST количестве деревьев в 150 штук, эффективность достигнет только 88%.

Следующим методом для оценки эффективности будет выступать *метод опорных векторов*. Суть данного метода заключается в переводе исходных векторов в пространство более высокой размерности и поиск разделяющей гиперплоскости с наибольшим зазором в таком пространстве.

С практической точки зрения особенностью метода опорных векторов выступает значительное количество параметров, влияющих на проводимый расчет. Ключевым из таких параметров является тип ядра (kernel), описывающего характеристики гиперплоскости, разделяющей наборы данных с определенными метками. В связи с тем, что наборы данных, отражающих векторизованные изображения цифр, в целом не представляются линейными, для целей настоящей работы будет выбрано полиномиальное ядро. Ниже приведем результаты применения метода опорных векторов в отношении двух наборов данных DIDA и MNIST в зависимости от степеней полинома ядра.



**Рис. 5.** Эффективность использования метода опорных векторов (полиномиальное ядро, различные степени полинома) для распознавания рукописных цифр в наборах данных DIDA и MNIST

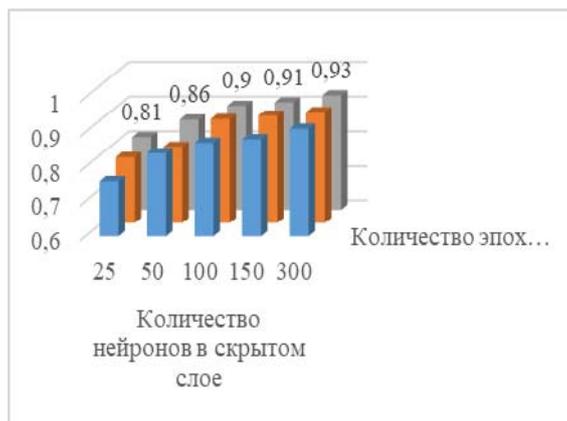
Как видно из представленных данных, применение метода опорных векторов для предварительно подготовленного набора данных MNIST дает эффективность 98%, аналогичную эффективности, полученной с применением других методов. В тоже время для неподготовленных данных набора изображений цифр DIDA эффективность распознавания не превышает 90% (полиномиальное ядро второй степени).

### Использование нейронной сети прямого распространения для классификации наборов рукописных цифр разного качества

С целью сравнить эффективность использования нейронных сетей для целей классификации рукописных цифр из наборов изображений разного качества автором была использована нейронная сеть прямого распространения, общие принципы построения которой хорошо описаны в литературе, например в [5] и [6]. Определяющим моментом выступает выбор архитектуры нейронной сети [7]. Архитектуру использованной нейронной сети можно выразить в виде  $784 - N - 10$ , где  $N$  – количество нейронов в единственном скрытом слое нейронной сети (Автором были проведены эксперименты также и с нейронными сетями, архитектура которых включала несколько скрытых слоев, однако повышения эффективности распознавания зафиксировано не было).

По аналогии с иными классификаторами модель нейронной сети строилась с использованием библиотеки языка Python известной как scikit-learn (версия 1.2.0) [8], для чего использовался класс MLPClassifier [9]. Существует значительное количество функций активации [10]. Для проведения экспериментов использовалась функция ReLu, а количество эпох обучения зависело от конкретного эксперимента.

Ниже приводим данные, полученные автором при проведении различных экспериментов с нейронной сетью прямого распространения. Прежде всего, рассмотрим результат распознавания рукописных цифр на наборе изображений без предварительной обработки DIDA.



**Рис. 6.** Эффективность распознавания рукописных цифр на примере набора изображений DIDA

Как видно из представленных выше данных для набора изображений цифр без предварительной обработки использование нейронной сети прямого распространения приводит к получению максимальной эффективности распознавания из всех проделанных экспериментов в 93%. При этом, как увеличение количества эпох обучения, так и увеличение количества нейронов в единственном скрытом слое нейронной сети ведет к увеличению эффективности распознавания.

Для сравнения рассмотрим результаты распознавания структурированного набора изображений цифр MNIST.

В отношении набора рукописных цифр MNIST ситуация с эффективностью распознавания меняется. Уже при минимальном количестве нейронов в нейронной сети и эпох обучения достигается эффективность распознавания в 96%, а при незначительном усложнении модели она возрастает до 98%.

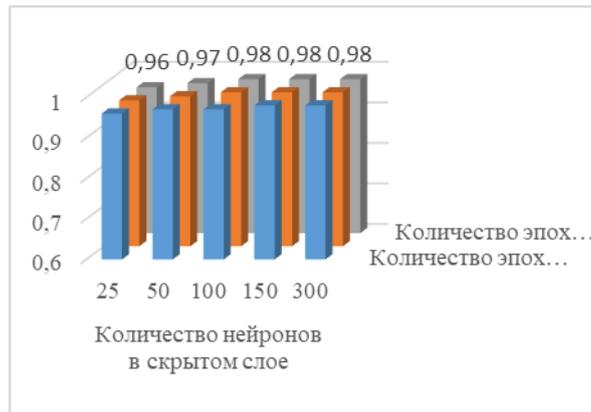


Рис. 7. Эффективность распознавания рукописных цифр на примере набора MNIST

Обобщая данные проведенных экспериментов, приведем следующую таблицу по различным методам.

Таблица 2

Наилучший результат распознавания изображений цифр, достигнутый соответствующим методом для наборов DIDA и MNIST

Метод классификации	Наилучшая аккуратность на тестовом наборе данных	
	DIDA	MNIST
к-ближайших соседей	77%	97%
Случайный лес	88%	97%
Метод опорных векторов	90%	98%
Нейронная сеть прямого распространения	93%	98%

Интересным является также сопоставление матриц некорректных прогнозов (confusion matrix) для различных наборов данных.

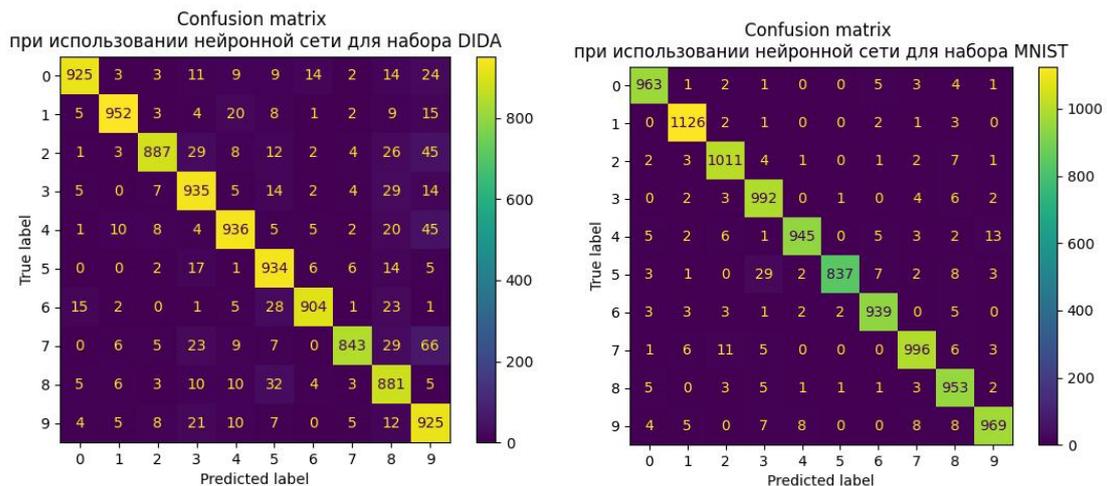


Рис. 8. Матрицы некорректных прогнозов для наборов данных DIDA и MNIST

Как видно из представленных данных, в отношении набора данных MNIST есть несколько явных отклонений, пиков распределения ошибок.

Прежде всего, это ошибка модели в паре 5-3, а также в парах 7-2 и 4-9. В отношении набора данных DIDA картина распределения ошибок иная, прежде всего, в смысле более равномерного распределения ошибок. Тем не менее, и здесь существуют пики концентрации ошибок в отдельных парах, например 7-9. В целом, наличие самого факта того, что модель по-разному распределяет ошибки двух разных наборов данных указывает на приоритетную зависимость от самого формата данных, недели чем от предметной области анализа. Иными словами, модель ошибается не при прогнозе в отношении какой-либо цифры, а ошибка зависит от того, какие конкретно данные использовались.

### Заключение

Проведенные эксперименты с использованием различных методов классификации изображений рукописных цифр показали, что для наборов изображений цифр с предварительной обработкой и без шумов (MNIST) фактически любой метод классификации может дать приемлемо высокий результат. Например, даже метод *k*-ближайших соседей позволяет получить эффективность распознавания в 97%. Распознавание наборов изображений без предварительной обработки (DIDA) продемонстрировало, что тривиальные методы классификации, например, *k*-ближайших соседей, имеют заметно худшую эффективность распознавания (77%), чем использование нейронных сетей прямого распространения (93%).

Таким образом, при проведении исследований с данными, плохо поддающимся структурированию, целесообразно изначально применять более сложные методы классификации с использованием нейронных сетей. Результаты проведенного эксперимента могут быть использованы при проектировании подходов к распознаванию изображений, например, аналогично использованных в работах [11] и [12].

### Литература

1. *Ян Лекун*, персональный сайт «MNIST handwritten digit database, Yann LeCun, Corinna Cortes and Chris Burges» - URL: <http://yann.lecun.com/exdb/mnist/> (обращение к сайту 12.12.2022).
2. *Pathric J Grother*. NIST Special Database 19. Handprinted forms and characters database. // National Institute of Standards and Technology. March 16, 1995.
3. *Huseyin Kusetogullari, Amir Yavariabdi, Johan Hall, Niklas Lavesson*, DIDA: The largest historical handwritten digit dataset with 250k digits, June 2021. (обращение 26.12.2022), доступно на: <https://github.com/didadataset/DIDA/>.
4. *Andriy Burkov*. The Hundred-Page Machine Learning Book // ABC Publishing. 2019. С. 94, 38.
5. *Raul Rojas*. Neural Networks, A Systematic Introduction // Springer. 1996. С. 46.
6. *Ясницкий Л.Н.* Интеллектуальные системы // Лаборатория Пилот. 2016. С.49, 114.
7. *Галушкин А.И.* Нейронные сети: основы теории. М.: Горячая линия – Телеком, 2012. С. 60.
8. Библиотека для Python scikit-learn <https://scikit-learn.org/stable/> (обращение 19.12.2022).
9. *David Kriesel*. A Brief Introduction to Neural Networks // dkriesel.com. 2022. С. 66.
10. *Martin T. Hagan et al.* Neural Network Design 2nd edition // eBook. 2022. С. 31.
11. *Березовская Е.В., Воронова Л.И.* Распознавание изображений на основе технологии Tensorflow // Сборник трудов XV Международной отраслевой научно-технической конференции «технологии информационного общества». 2021. С. 293-295.
12. *Лешин Е.В., Репинский В.Н., Воронова Л.И.* Распознавание изображений на основе технологии Tensorflow // Сборник трудов XV Международной отраслевой научно-технической конференции «технологии информационного общества». 2021. С. 321-323.

## **МЕТОДОЛОГИЯ Понижения ДЕТАЛИЗАЦИИ ПОЛИГОНАЛЬНОЙ ГЕОМЕТРИИ ДЛЯ СУБЪЕКТИВНОЙ ОЦЕНКИ КАЧЕСТВА ИЗОБРАЖЕНИЯ ОТНОСИТЕЛЬНО РЕГИОНА ИНТЕРЕСА**

**Прокурат Глеб Андреевич,**

*Московский Технический Университет Связи и Информатики, студент, Москва, Россия,*  
[gleb@prokurat.ru](mailto:gleb@prokurat.ru)

**Балобанов Андрей Владимирович,**

*Московский Технический Университет Связи и Информатики, кандидат технических наук, доцент,*  
*Москва, Россия,*  
[a.v.balobanov@mtuci.ru](mailto:a.v.balobanov@mtuci.ru)

### **Аннотация**

*В настоящее время наблюдается тенденция увеличения количества полигонов в 3D моделях, применяемых в интерактивном контенте, виртуальной и дополненной реальности. В последние годы значительно повысился интерес к разработке алгоритмов оптимизации трёхмерной компьютерной графики реального времени. Однако очень мало внимания было уделено регионам интереса человека и оптимизации геометрии относительно регионов интереса пользователя. Текущие технологии оптимизации графики и аппаратного ускорения позволяют отображать миллионы полигонов с кинематографической частотой кадров, но человек не способен охватить за кадр все элементы детализации, из-за чего часть ресурсов графических ускорителей расходуется не эффективно. В представленном исследовании была разработана методология оценки восприятия понижения детализации полигональной геометрии по мере удаления от центра региона интереса наблюдателя.*

**Ключевые слова:** *зрительная система человека, субъективное качество, регионы интереса, компьютерная графика, уровень детализации.*

### **Введение**

Трёхмерная компьютерная графика прочно вошла в повседневную жизнь. Развиваются средства художественного и инженерного 3D моделирования, игровые и кинематографические движки, интерактивные модели, видеоигры и средства виртуальной и дополненной реальности. По мере того как аппаратные ускорители 3D графики становятся совершеннее, увеличиваются детализация полигональной геометрии и количество высокополигональных объектов в кадре.

В большинстве инструментов для работы с трёхмерной компьютерной графикой используется снижение уровня детализации по глубине сцены, что при правильном позволяет получить полный кадр целевого разрешения без заметной потери качества объектов на заднем плане и сохранении высокого качества объектов на переднем плане.

На данный момент успешно применяется оптимизация за счет понижения детализации полигональных 3D моделей и сжатия геометрии путём кластеризации [1]. В случае с пререндерами высокой чёткости данный подход даёт оптимальный результат, однако в случае с графикой в реальном времени, когда скорость формирования кадра играет ключевую роль для восприятия [2, 10-13], снижение детализации по глубине зачастую приводит к появлению заметных графических артефактов на заднем плане. Как показали результаты, человек зачастую не в состоянии охватить взглядом весь кадр.

### **Методология понижения детализации**

Регион интереса (РОИ) – это определённая область, на которую человек сконцентрирован во время просмотра видео. В компьютерном зрении и оптическом распознавании символов РОИ определяет границы рассматриваемого объекта [3].

Уровень детализации LOD (англ. level of detail) – это метод снижения сложности рендеринга кадра, уменьшения общего количества полигонов, текстур и иных ресурсов в сцене, общее снижение её сложности [4]. Данный метод применяется повсеместно как в интерактивном контенте с применением полигональной 3D графики, так и при создании пререндеров с целью уменьшения времени необходимого на отрисовку одного кадра. Как правило, при использовании данного метода на основе модели с высокой детализацией создаётся ряд моделей с постепенным понижением детализации посредством уменьшения количества полигонов и снижения разрешения одной или нескольких текстурных карт. Снижения может проводиться как в автоматическом, так и в ручном режиме, а новые модели получаются эвристическим путём.

Создание LOD сводится к упрощению геометрии путём объединения соседних вершин в зависимости от расстояния. Поскольку в разных системах построения трёхмерных сцен используются различные единицы измерения, то расстояние между вершины необходимо рассчитывать в условных единицах, преобразуемых к пространству модели в конкретном программном обеспечении.

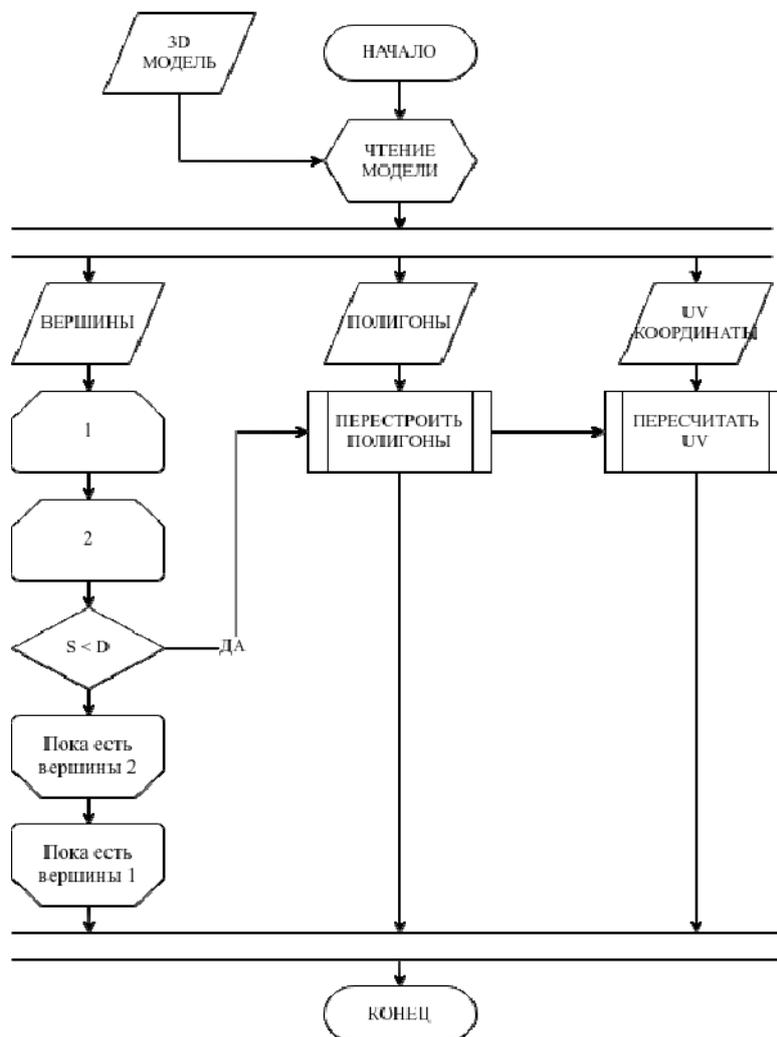


Рис. 1. Блоксхема алгоритма снижения детализации геометрии

Максимальная дистанция объединения двух соседних вершин в одну подбирается эвристическим путём в зависимости от диагонали экрана и его разрешения. Дистанция объединения вершин для каждого видимого объекта сцены рассчитывается по формуле (1).

$$l = k * \frac{|AB|}{L}, \quad (1)$$

где точка А– центр объекта на экране, В – центр экрана, L– половина диагонали экрана, k – максимальная дистанция объединения вершин (для экрана 1920x1080p, 15,6 дюймов).

Алгоритм получения необходимого LOD представлен на рисунке 1.

### Процесс измерения

В качестве тестовой модели использована модель *suzanne* из стандартной библиотеки программного пакета *blender*. Выбор обусловлен тем, что данная модель разработана для тестирования восприятия графических артефактов шейдеров [9] и материалов, число полигонов модели приближено к количеству полигонов в оптимизированных подвижных объектах, применяемых в качестве декораций в динамических сценах, обновляющихся в реальном времени.

Новый метод понижения детализации для сбора субъективных оценок обеспечивает условия для создания обширного объема данных для анализа восприятия полигональной геометрии в реальном времени и применения для последующей оптимизации трехмерных сцен, воспроизводимых в реальном времени средствами графических ускорителей.

Метод обеспечивает условия для создания масштабного набора данных для изучения субъективного восприятия качества полигональной геометрии с минимальным количеством субъективных экспериментов [3]. Измерения проводятся пороговым методом, при котором в случайные моменты времени, заданные псевдослучайным генератором персонального компьютера испытательного стенда, радиус предполагаемого региона интереса уменьшается на 1% от половины диагонали экрана.

В телекоммуникациях принято использовать байесовский адаптивный метод для сбора субъективных оценок, который значительно экономит трудозатраты, поскольку для получения оценки с той же точностью, что и без использования апостериорных вероятностей, требуется около значительно меньше попыток. Однако, если необходимо найти 10 взаимных комбинаций между двумя параметрами порогов, этот метод потребует 20 000 тестов, следовательно, около 2000 часов или 1 год тестов на одного оценщика. Следовательно, байесовский адаптивный метод не подходит для крупномасштабного тестирования. Пороговая методология, позволяет проводить новые комплексные исследования с использованием современных технологий [6]. Используя пороговой подход, каждому участнику требуется около 15 минут, чтобы выполнить 44 оценки различных взаимодействий двух значений.

Лабораторная установка состоит из монитора, устройства ввода, персонального компьютера с программой тестирования.

Дисплей должен представлять собой плоский экран 15,6 дюймов. Монитор должен иметь правильную настройку и калибровку яркости и цвета с помощью профессионального экспонетра. Калибровка соответствует параметрам, определенным в соответствующей Рекомендации для проводимого испытания. Освещенность 50 лк, цветовая температура белого D65.

По знанию ГВС и поля четкости острота зрения составляет 1/60 градуса [6]. По этим данным минимально допустимое расстояние от монитора до испытуемого для нахождения стимула в зоне ясного зрения составляет 0,872 м, а максимальное – 1,149 м [7]. Перед монитором может находиться один участник.

Персональный компьютер должен быть оснащён встроенным или дискретным видеоускорителем, поддерживающим графический стандарт OpenGL ES минимальной версии 2.0 и поддержкой разрешения 1920x1080, которое является самым популярным в мире, с частотой от 30 кадров в секунду.

Все участники эксперимента должны иметь нормальное зрение. В данной работе нормальное зрение определяется типичным пользовательским контентом, когда участники не используют очки,

линзы или другие медицинские приспособления для коррекции зрения в обычной повседневной деятельности [6].

Сеансы тестирования необходимо начинать с «этапа стабилизации». Продолжительность тестового сеанса без перерыва не должна превышать 30 минут. Перед каждым экспериментом необходимо проводить короткий обучающий сеанс просмотра. Для тренировки можно использовать отдельные наборы полигональной геометрии, которые не используются в реальных сессиях [7].

Для оценки восприятия были созданы уровни детализации, где за LOD 0 принята базовая модель из библиотеки Blender содержащая 968 вершин. Упрощённая модель LOD 6 содержит 360 вершин. Уровни детализации представлены на рисунке 2.

Для тестирования метода была разработана программа, стимулирующая участника сфокусировать внимание в центре экрана за счёт светового стимула и постепенно понижающая детализацию объектов на экране. В качестве стимула выступают сменяющиеся друг друга символы W, A, S, D, поскольку они являются наиболее распространённой раскладкой управления в интерактивных 3D сценах с видом от первого лица (начиная с 1999 года) и привычны большей части людей до 40 лет. Символы сменяются по вводу соответствующего символа с клавиатуры испытательного стенда или по истечении 280 мс. Интервал 280 мс основан на времени реакции человека на источник света, который составляет от 100 до 200 мс [5], с учётом погрешности в 40%. Изначально все объекты на экране находятся в состоянии LOD 0. По мере уменьшения региона интереса и удаления модели от границы региона интереса детализация модели понижается от LOD 0 до LOD 6. Экран пользователя в начале теста (слева) и в конце теста (справа) изображен на рисунке 3.

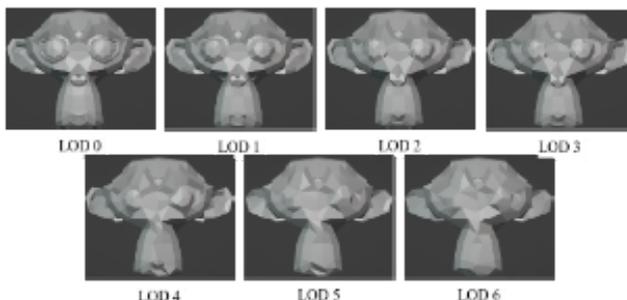


Рис. 2. Уровни детализации экспериментальной модели

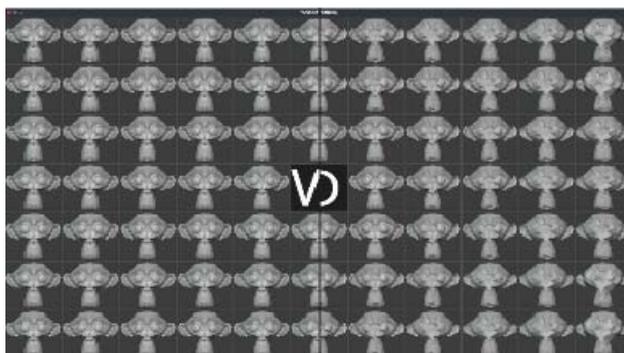


Рис.3. Экран испытуемого в начале и в конце теста

В момент понижения детализации начинается 280 мс интервал ввода, позволяющий человеку зарегистрировать изменение изображения посредством ввода с клавиатуры символа пробел. Программа может фиксировать один из двух откликов: положительный, если отклик был получен в течение 280 мс после изменения, и отрицательный, если отклик был получен вне данного интервала.

При получении положительного отклика программа фиксирует радиус, при котором был получен данный отклик, и LOD, изменение которого заметил испытуемый. Схема сбора данных изображена на рисунке 4.

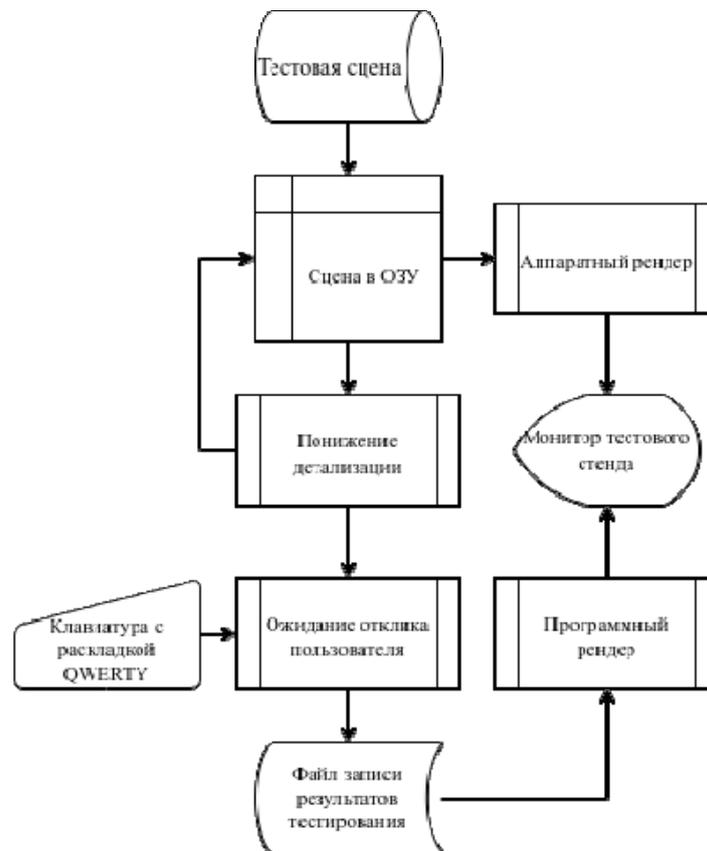


Рис.4. Схема сбора данных

Данные собираются анонимно и регистрируются в электронной таблице для расчета средних значений. «Пост-скрининг» участников проводится с использованием коэффициента линейной корреляции Пирсона. Испытания могут быть прекращены, когда экспериментальная неопределенность, измеряемая доверительным интервалом, упадет ниже 5% от текущего значения для всех испытаний, выполненных в ходе эксперимента [8].

Разница в количестве информации о геометрии, передаваемой графическому ускорителю на момент начала тестирования и на момент завершения, составляет 1.97 Мб, что составляет 38% от изначального объема.

### Заключение

Разработанное тестирование показывает, что человек не способен различить падение детализации полигональных объектов, находящихся вне его региона интереса, что открывает возможность увеличения скорости формирования кадра при работе со сценами, обновляющимися в реальном времени. Текущее исследование описывает новый метод понижения детализации полигональных объектов в динамических сценах относительно региона интереса наблюдателя для сбора данных и исследования субъективного восприятия качества полигональных объектов внутри региона интереса, на его границе и за его пределами.

Предложенный подход позволит проводить масштабные исследования восприятия пониженной детализации объектов за пределами региона интереса, что позволит сформировать более точные правила понижения детализации с целью оптимизации времени отрисовки кадра, что позволит снизить нагрузку на аппаратные ускорители и увеличить детализацию объектов, находящихся внутри региона интереса без потери производительности.

### Литература

1. A first look at Unreal Engine 5 // Unreal Engine URL: <https://www.unrealengine.com/> (дата обращения: 24.01.2023).
2. *Nezhivleva K. I., Davydova A. A., Drebuszhan A. M., Mozhaeva A. I., Balobanov A.* Comparing of Modern Methods Used to Assess the Quality of Video Sequences During Signal Streaming with and Without Human Perception // 2022 Systems of Signal Synchronization, Generating and Processing in Telecommunications (SYNCHROINFO), 2022, pp. 1-6, doi: 10.1109/SYNCHROINFO55067.2022.9840983.
3. Давыдова А. А., Можяева А. И., Хураны В. Х. и др. Методология исследования идентичности восприятия регионов интересов пользователей при просмотре потокового видео содержащего различный контент и артефакты сжатия // Системы синхронизации, формирования и обработки сигналов. 2022. Т. 13. № 6. С. 42-51. EDN IZCJOE.
4. *Musgrave F. Kenton, Craig E. Kolb, Robert S. Mace.* The synthesis and rendering of eroded fractal terrains // ACM Siggraph Computer Graphics. Vol. 23. No. 3. ACM, 1989.
5. *Бойко Е. И.* Время реакции в исследованиях практически прикладного характера // Пограничные проблемы психологии и физиологии. М.: Изд. АПН РСФСР, 1961. С. 192-209.
6. *Mozhaeva A. I., Vlasuyk I. V., Potashnikov A. M., Cree M. J., Streeter L.* The Method and Devices for Research the Parameters of the Human Visual System to Video Quality Assessment // 2021 Systems of Signals Generating and Processing in the Field of on Board Communications, 2021, pp. 1-5, doi: 10.1109/IEEECONF51389.2021.9416081.
7. *Mozhaeva A., Mazin V., Cree M. J., Streeter L.* Video quality assessment considering the features of the human visual system // Lecture Notes in Computer Science IVCNZ, 2022, to appear.
8. *BT-500-11: Methodology for the Subjective Assessment of the Quality of Television Pictures*, 2012.
9. *Боресков А. В.* Расширения OpenGL. СПб: БХВ, 2005. 668 с.
10. *Valitskaya N.S., Vlasuyk I.V., Potashnikov A.M.* Video compression method on the basis of discrete wavelet transform for application in video information systems with non-standard parameters // T-Comm. 2020. Т. 14. № 3. С. 47-53.
11. *Поташиников А.М., Власюк И.В.* Метод построения равноконтрастного цветового пространства для заданной системы отображения информации и условий контроля // T-Comm: Телекоммуникации и транспорт. 2020. Т. 14. № 4. С. 15-22.
12. *Егоров Д.А., Федоров В.Д., Лейман В.В., Власюк И.В.* Методика оценки пространственно-частотной характеристики камер на основе генеративных случайных последовательностей // DSPA: Вопросы применения цифровой обработки сигналов. 2021. Т. 11. № 1. С. 47-53.
13. *Можяева А.И., Власюк И.В., Поташиников А.М., Стритер Ли.* Эталонная объективная метрика оценки качества видео совместимая с PSNR учитывающая частотные и периферическую характеристики зрения человека // DSPA: Вопросы применения цифровой обработки сигналов. 2021. Т. 11. № 2. С. 44-54.

## НЕЙРОСЕТЕВОЕ РАСПОЗНАВАНИЕ МАРКИРОВКИ ПРОДОВОЛЬСТВЕННЫХ ТОВАРОВ

**Четыркин Максим Иванович,**

*Московский Технический Университет Связи и Информатики, магистрант, Москва, Россия*  
[skarpion234@gmail.com](mailto:skarpion234@gmail.com)

**Безумнов Данил Николаевич,**

*Московский Технический Университет Связи и Информатики, старший преподаватель,  
Москва, Россия*  
[d.n.bezumnov@mtuci.ru](mailto:d.n.bezumnov@mtuci.ru)

### **Аннотация**

*Статья посвящена исследованию особенностей применения нейронных сетей для распознавания маркировок, рассмотрены технологии проектирования, разработки и тестирования подобных систем.*

**Ключевые слова:** *Нейронные сети, искусственный интеллект, машинное обучение, маркировка, распознавание.*

### **Введение**

На сегодняшний день происходит активное внедрение искусственного интеллекта в различные области жизнедеятельности человека. Нейронные сети применяются в большом количестве секторов деятельности, в том числе, в промышленности, логистике, торговле. Их применение нашло себя при распознавании маркировки различной продукции, что значительно ускоряет и автоматизирует производственный процесс, снижая непосредственное влияние на него человеческих факторов.

В связи с этим, вопрос внедрения нейронных сетей с машинным обучением в производстве остается актуальным для изучения, как среди отечественных, так и среди зарубежных исследователей.

Кафедра «Интеллектуальные системы в управлении и автоматизации» (ИСУиА) Московского технического университета связи и информатики (МТУСИ) проводит исследования в области разработки интеллектуальных систем для промышленности [1-3], транспорта [4-6], медицины [7-8], бытовой автоматизации [9], в том числе распознавания и идентификации образов на фото- и видеоматериалах [10-12].

### **Анализ предметной области**

На сегодняшний день наиболее популярными технологиями, используемыми для идентификации товаров, являются использование различных маркировок и штрих-кодов (1D, 2D), прямой маркировки деталей (DPM) и RFID-меток. Использование подобных элементов для маркировки хорошо подходит для различных отраслей промышленности, стандартизированных производственных линий, складов и магазинов. Вне зависимости от объемов производства, для обработки таких маркировок важно использовать автоматические системы распознавания, так называемое «машинное зрение», поэтому сегодня специалисты ищут новые способы разработки интеллектуальных систем отслеживания, которые выводят такие системы на новый уровень, так как потребность в интеллектуальных системах отслеживания растет исключительно быстрыми темпами [13].

Подобные системы идентификации продукции по ее маркировке являются фундаментальными элементами учета и управления качеством на производстве. Их использование значительно ускоряет процесс производства, контроля, учета, а также выявляет различные несоответствия.

Система машинного зрения позволяет проследить все перемещения продукции по технологической цепочке, помогает контролировать соблюдение всех требований. В электронную маркировку можно внести все данные, которые подтверждают качество продукции.

Таким образом, рассматриваемые технологии используются для решения следующих задач:

- идентификация объектов контроля по маркировке;
- отслеживание качества продукции на всех этапах производства;
- верификация нанесенной маркировки.

Принцип работы системы по распознаванию маркировок строится следующим образом: пользователь сканирует изображение с маркировкой при помощи специальной камеры, которая в последующем передает изображение на сервер, где располагается информационная система, построенная на базе алгоритмов нейронного распознавания объектов [14].

### **Этапы разработки нейронной сети для распознавания маркировок**

Разработка нейронной сети по распознаванию маркировки продовольственных товаров подразделяется на следующие этапы [15]:

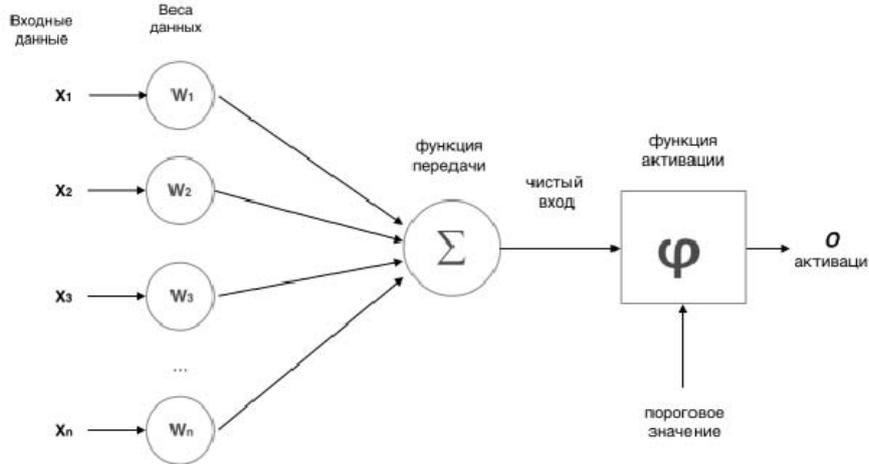
1. Сбор данных для первичной обработки данных (маркировки продовольственных товаров либо изображения из открытых источников) для будущего обучения нейронной сети;
2. Обработка графических данных, которая включает нормирование, фильтрацию и масштабирование;
3. Разбиение полученных данных на обучающую и тестовую выборки: обучающая выборка предполагает набор изображений, разделенных по разным параметрам отбора, которые будут участвовать в обучении нейросети, тестовая выборка содержит набор изображений, на основе которых можно получить информацию об уровне обученности сети;
4. Выбор и разработка архитектуры нейронной сети;
5. Обучение нейронной сети на основе разработанных алгоритмов с использованием обучающей выборки изображений;
6. Тестирование нейронной сети на основе тестовой выборки;
7. Анализ полученных результатов, на основе которых будет производиться следующий этап – доучивание;
8. Доучивание, которое может производиться на основе полученных результатов тестирования, данный этап обычно используется только для новых развивающихся нейронных сетей.

### **Выбор архитектуры нейронной сети**

Одним из самых важных этапов в разработке нейронной сети является выбор ее архитектуры. На сегодняшний день существует множество архитектур, поэтому необходимо рассмотреть наиболее популярные архитектуры.

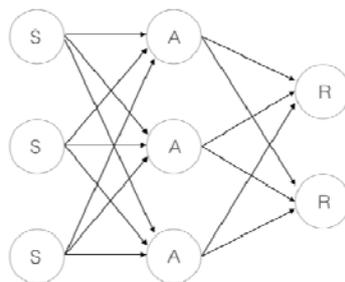
В целом, все нейронные сети состоят из основных узлов – нейронов, которые связаны между собой синапсами. Модель нейрона представлена на рисунке 1 [16].

На представленной модели видно, что на вход поступают определенные сигналы, количество которых было определено разработчиком сети с учетом поставленной задачи. Далее полученные сигналы умножаются на вес и суммируются. На следующем этапе запускается функция активации, которая преобразует полученные данные для получения ответа. Функция активации также выбирается с учетом решаемой задачи. Наиболее используемыми функциями активации выступают сигмоид, гиперболический тангенс и линейная функция.



**Рис. 1.** Модель нейрона

Одной из самых распространенных архитектур нейронных сетей является персептрон, его модель отражена на рисунке 2.

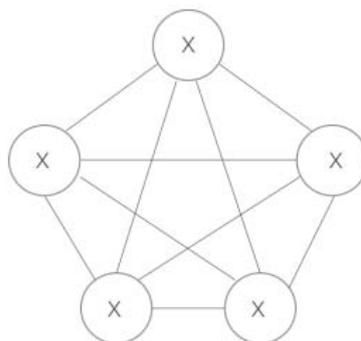


**Рис. 2.** Схема персептрона

Работа такой сети основывается на трех типах элементов: поступающие сигналы, ассоциативные элементы и реагирующие элементы. Таким образом, подобная модель позволяет создать ряд ассоциаций между входными элементами и необходимой реакцией на выходе.

К главным преимуществам архитектур, основанных на персептроне, можно отнести простоту реализации алгоритма, простота использования и гарантированное получение ответа на выходе. Подобные сети способны быстро обучаться, а также позволяют получить высокий уровень правильных ответов, поэтому персептроны могут использоваться в нейросетях любого уровня и любой сложности [17].

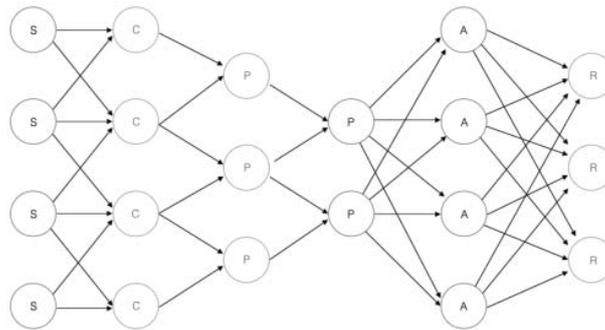
Следующая сеть – сеть Хопфилда. Ее схема отражена на рисунке 3.



**Рис. 3.** Сеть Хопфилда

Данную нейронную сеть отличает то, что каждый нейрон здесь играет сразу три роли. До того, как производится обучение, нейроны являются сенсорными, во время обучения – ассоциативными, после обучения – реагирующими. С каждой итерацией обучения производится снижение энергии сети, а сам результат выдается только после того, как будет принято конкретного решение. Обучение нейронов в сети Хопфилда производится параллельно, поэтому на выходе обычно все нейроны уже обучены. Такую сеть обычно связывают с ассоциативной памятью, и она используется в случаях, когда необходимо «дорисовать» картину событий на основе имеющихся данных.

Сверточная нейронная сеть наиболее подходит для решения поставленной задачи, так как она чаще всего используется для работы с изображениями и другой информацией, представленной в графическом виде. Модель сверточной нейронной сети отражена на рисунке 4.



**Рис. 4.** Сверточная нейронная сеть

В отличие от большинства других нейронных сетей, сверточная сеть не обрабатывает данные целиком, а производит анализ их фрагментов. При этом получаемые данные не делятся на части, а осуществляется их последовательный прогон, после которого информация поступает на следующие слои, которые бывают двух типов: сверточные слои и слои объединения. Например, из изображения извлекается только необходимая информация, которая была задана архитектором, после чего слои объединения сжимают ее с глубиной и отправляют на конечный слой. Таким образом, автор статьи считает целесообразным использовать сверточную нейронную сеть для реализации информационной системы для распознавания маркировки продовольственных товаров [18].

#### **Проектирование и реализация нейронной сети для распознавания маркировки продовольственных товаров**

Перед тем как приступить к разработке и тестированию работы нейронной сети, автор статьи подобрал обучающие наборы данных, которые позволят обучаться сверточной нейросети. Так как разрабатываемая нейронная сеть направлена на распознавания маркировок товаров, то был подобран Dataset, состоящий из 2100 маркировок различной продукции. Набор данных разделен на обучающую и тестовую выборки (train и test).

Сверточная нейросеть получает входную информацию в виде графических файлов, которые будут использоваться в работе в будущем, после чего производится создание отдельной папки, где будут храниться обработанные изображения. Для более быстрого и эффективного обучения нейронной сети все были приведены к одному формату и размеру 100 на 100 пикселей.

Для редактирования изображений автор статьи использовал библиотеку Python – Augmentor. Данная библиотека используется для работы с изображением, а также для его улучшения. Алгоритм с каждым проходом накладывает на изображение операции улучшения до достижения необходимого результата. Так, для входных изображений можно выбрать определенные параметры редактирования, например, случайное вращение на 90 или 180 градусов. После того, как все изображения были обработаны, создаются обучающие, валидационные и тестовые наборы данных. Все данные обучающего набора данных подвергаются нормализации с целью получения значения интенсивности пикселя в диапазоне от 0 до 1 [19].

На следующем этапе автор статьи произвел настройку архитектуры персептрона. Алгоритм был построен таким образом, что изображения с маркировками поступают на вход нейронной сети, которая извлекает из них некоторые признаки и производит оценку полученных результатов, после чего делает вывод о принадлежности объекта на фотографии к тому или иному классу. Результат работы отображается как в графическом, так и в текстовом виде.

Для реализации программного обеспечения использовалась модель Sequential, что означает, что слои будут идти последовательно. Модель создана с приведенными ниже параметрами и их значениями:

- первые три слоя – сверточные;
- размер ядра первых трех сверточных слоев – 3;
- количество нейронов на слоях: 32 на первом, 32 на втором, 64 на третьем, функция активации – ReLU;
- функция подвыборки MaxPooling() вызывается без передаваемых аргументов;
- полносвязный Dense слой расположен после третьего сверточного слоя, имеет 64 нейрона и использует функцию активации ReLU.
- второй Dense слой (выходной слой сети) имеет по одному нейрону на каждый класс изображений из набора данных, т.е. 10 нейронов и использует функцию активации SoftMax.

На следующем этапе производилась сборка модели с помощью метода compile(), который позволяет настроить модель и добавить ему уровень оптимизации, функцию потери и метрику.

Далее производилось обучение нейронной сети с помощью функции fit(). В качестве параметров функция принимала тренировочный датасет (train\_ds), проверочный (val\_ds) и количество эпох (epoch).

Функция активации разрабатываемой нейронной сети отражена на рисунке 5.

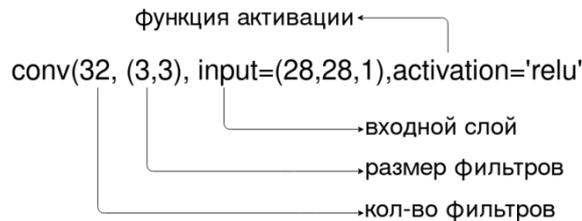


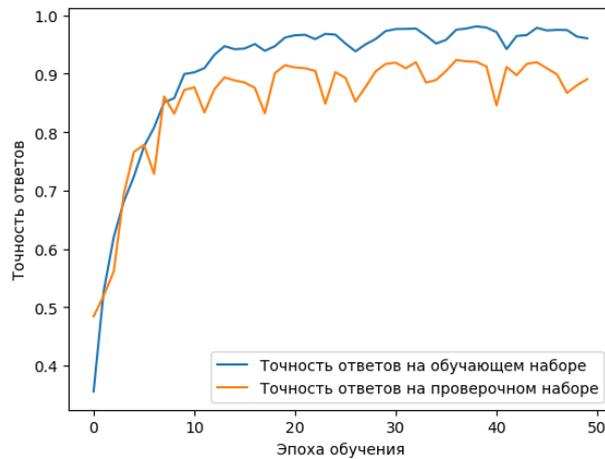
Рис. 5. Функция активации

После обучения нейросети проводилась проверка на тестовой выборке. Результаты отражены на рисунке 6.

1/1 [=====] - 0s 271ms/step - loss: 3.5459 - accuracy: 0.815

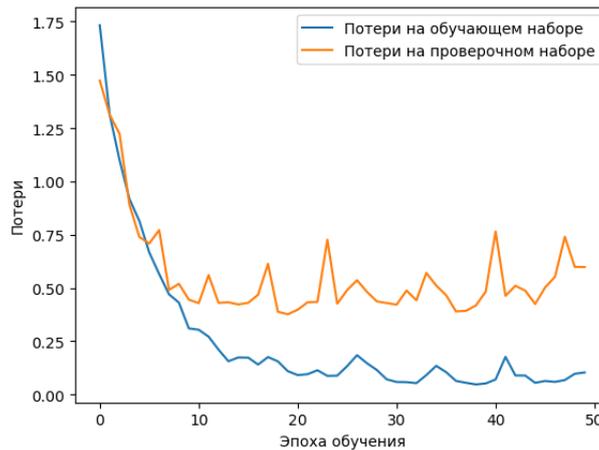
Рис. 6. Точность модели на тестовой выборке

Доля верных ответов на обучающем и тестовом наборе отражена на рисунке 7.



**Рис. 7.** Доля верных ответов на обучающем и тестовом наборе

Анализ ошибок нейросетевого распознавания маркировок на тренировочном наборе отражен на рисунке 8.



**Рис. 8.** Анализ ошибок нейросетевого распознавания маркировок на тренировочном наборе

### Заключение

В статье проанализирована предметная область использования нейронных сетей и машинного обучения для распознавания маркировки продовольственной продукции. Авторами рассмотрены основные этапы разработки нейросетей, их архитектуры, а также была представлена сверточная нейронная сеть. Для ее обучения выбран дата-сет, состоящий из 2100 изображений, которые поделены на тестовую и обучающую выборки. Результаты работы нейросети продемонстрировали высокую точность обучения системы.

### Литература

1. Бугаева Г.М., Ретинский В.Н., Воронова Л.И. Нейросетевой метод определения типа стекла по содержимым примесям // Сборник трудов XV Международной отраслевой научно-технической конференции «Технологии информационного общества». М.: Издательский дом Медиа Паблшер» 2021. С. 296-297.
2. Sichkar D.P., Bezumnov D.N., Voronov V.I., Voronova L.I., Dankovtsev V.I. Moving Elements of Mobile Robots Stabilization Modelling // 2019 Systems of Signals Generating and Processing in the Field of On Board Communications, 2019, p. 8706810.
3. Шишканова Е.О., Воронова Л.И. Применение нейронных сетей для распознавания продукции молочного производства // Технологии информационного общества: Сборник трудов XVI Международной отраслевой научно-технической конференции. М.: Издательский дом Медиа Паблшер, 2022. С. 327-329.

4. *Bezumnov D.N., Voronov V.I.* Development of the Research Stand for Exploration of Models and Algorithms for Group Control of Ground-Based Mobile Robots // 2021 Systems of Signals Generating and Processing in the Field of On Board Communications, conference proceedings, 2021, p. 9416084.
5. *Безумнов Д.Н., Воронова Л.И., Плотникова В.С.* Разработка исследовательского стенда для изучения моделей и алгоритмов управления группой наземных мобильных роботов // REDS: Телекоммуникационные устройства и системы. 2021. Т. 11. № 3. С. 3-9.
6. *Безумнов Д.Н.* Автоматизированная оценка потоков в сетях сообщений // Труды Северо-Кавказского филиала Московского технического университета связи и информатики. 2016. № 1. С. 46-50.
7. *Chelyadin I.A., Artemov M.D., Voronova L.I.* Intelligent Disabled Passenger Support System // 2022 Systems of Signals Generating and Processing in the Field of On Board Communications, conference proceedings, 2022.
8. *Двудненко Ю.Э., Воронова Л.И.* Нейросетевое распознавание видов пневмонии по рентгеновским снимкам // Технологии информационного общества: Сборник трудов XVI Международной отраслевой научно-технической конференции. М.: Издательский дом Медиа Паблишер, 2022. С. 278-272.
9. *Безумнов Д.Н., Воронова Л.И.* Оценка временных характеристик выполнения задач реального времени на плате Arduino Uno // Телекоммуникации и информационные технологии. 2017. Т. 4. № 2. С. 51-54.
10. *Жаров И.А., Воронов В.И., Воронова Л.И., Быков А.Д., Вовик А.Г., Усачев В.А.* Программа для идентификации изображения в программном комплексе нейросетевого распознавания лица / Свидетельство о государственной регистрации программы для ЭВМ № RU 2020612466, дата регистрации 11.02.2020.
11. *Voronov V.I., Zharov I.A., Bykov A.D., Trunov A.S., Voronova L.I.* Designing a Neural Network Identification Subsystem in the Hardware-Software Complex of Face Recognition // T-Comm. 2020. Т. 14. № 5. С. 69-76.
12. *Мартынов Д.А., Воронова Л.И.* Применение перцептрона для распознавания дактилем русского жестового языка // DSPA: Вопросы применения цифровой обработки сигналов. 2020. Т. 10. № 2. С. 37-46.
13. *Клехо Д.Ю., Карелина Е.Б., Батырев Ю.П.* Использование технологии сверточных нейронных сетей в сегментации объектов изображения // Вестник МГУЛ – Лесной вестник. 2021. №1. URL: <https://cyberleninka.ru/article/n/ispolzovanie-tehnologii-svertochnyh-neyronnyh-setey-v-segmentatsii-obektov-izobrazheniya> (дата обращения: 25.01.2023).
14. *Серета С.В.* Применение машинного зрения в логистике // Sciences of Europe. 2021. №65-1. URL: <https://cyberleninka.ru/article/n/primenenie-mashinnogo-zreniya-v-logistike> (дата обращения: 27.01.2023).
15. *Прокопья А.С.* Современные методы распознавания изображений / А.С. Прокопья, И.С. Азаров // Big Data and Advanced Analytics. 2019. № 5. С. 361-369.
16. *Москалев Н.С.* Виды архитектур нейронных сетей // Молодой ученый. 2016. №29 (133). С. 30-34. URL: <https://moluch.ru/archive/133/37121/> (дата обращения: 02.01.2023).
17. *Горин В.В.* Распознавание изображений на основе перцептрона // Science Time. 2015. №5 (17). URL: <https://cyberleninka.ru/article/n/raspoznavanie-izobrazheniy-na-osnove-perseptrona> (дата обращения: 28.01.2023).
18. *Бредихин Арсентий Игоревич.* Алгоритмы обучения сверточных нейронных сетей // Вестник ЮГУ. 2019. №1 (52). URL: <https://cyberleninka.ru/article/n/algoritmy-obucheniya-svertochnyh-neyronnyh-setey> (дата обращения: 28.01.2023).
19. Augmentor – Augmentor 0.2.9 documentation URL: <https://augmentor.readthedocs.io/> (дата обращения: 28.01.2021). E