

СОДЕРЖАНИЕ

1 Задание на практическую работу	3
2 Краткая теоретическая часть	4
2.1 Алгоритм встраивания ЦВЗ на основе межблочной разности коэффициентов ДКП	5
3 Программная реализация	10
3.1 Результаты работы программы	11
4 Результаты вычислительных экспериментов при отсутствии атак	13
5 Результаты вычислительных экспериментов при наличии атак.....	24
6 Выводы	33

1 Задание на практическую работу

Целью данной работы является приобретение навыков программной реализации алгоритмов встраивания цифровых водяных знаков в частотную область цифровых изображений.

В рамках практической работы необходимо выполнить следующее:

- 1) Написать программную реализацию одного из двух рассмотренных методов встраивания ЦВЗ в частотную область цифровых изображений по выбору студента;
- 2) Провести вычислительные эксперименты с полученной программной реализацией и сделать выводы об эффективности рассмотренного метода встраивания;
- 3) Подготовить отчет о выполнении работы.

2 Краткая теоретическая часть

Цифровой водяной знак (ЦВЗ) – это некоторая дополнительная информация, которая встраивается в цифровые изображения с целью их аутентификации или контроля целостности. ЦВЗ различаются по степени устойчивости к искажениям: хрупкие ЦВЗ разрушаются при любом изменении контейнера, полухрупкие могут выдержать некоторые разрешенные преобразования, а робастные ЦВЗ обнаружимы после существенных искажений изображения–контейнера.

Полухрупкие и робастные ЦВЗ представляют большую ценность для решения задач аутентификации цифрового контента и защиты авторства на него. Для этого необходимо использовать такие подходы ко встраиванию дополнительной информации, которые сохраняют следы водяного знака после различных атак на изображение, содержащее ЦВЗ. В литературе можно встретить анализ устойчивости алгоритмов встраивания ЦВЗ к самым разнообразным атакам, однако на практике не все из них являются актуальными. Вероятность того, что изображение с водяным знаком будет подвергаться зашумлению типа «соль-перец» или медианной фильтрации крайне мала. Реальные сценарии включают такие типичные операции обработки изображений, как кадрирование, масштабирование, изменение яркости и/или контрастности, JPEG-сжатие. Поэтому на практике наибольшей ценностью обладает робастность встраивания относительно данного перечня атак.

В общем случае частотная область изображений позволяет добиться большей робастности встраивания при сохранении визуальной незаметности, чем пространственная. Встраивание информации в частотную область цифровых изображений предполагает предварительную реализацию какого-либо частотного преобразования и последующее изменение полученных частотных коэффициентов. Наиболее широко распространены для задач сокрытия информации в цифровых изображениях такие преобразования, как дискретное косинусное преобразование (ДКП), дискретное преобразование Фурье (ДПФ), дискретное вейвлет-преобразование (ДВП).

2.1 Алгоритм встраивания ЦВЗ на основе межблочной разности коэффициентов ДКП

В последние годы большинство исследователей имеют дело с ДКП и ДВП. В работе [1] представлен алгоритм, скрывающий водяной знак в среднечастотных коэффициентах ДКП цифровых изображений. Один бит водяного знака внедряется в один блок изображения 8×8 путем изменения разности между соседними блоками. Возможные значения разности разбивают на интервалы, соответствующие нулевому или единичному биту встраиваемого ЦВЗ. Частотные коэффициенты изменяют так, чтобы их разность попадала в соответствующий интервал. Таким образом, общая идея данного алгоритма похожа на идею метода PVD в пространственной области.

Рассмотрим данный алгоритм подробнее.

В качестве изображения–контейнера берется полутоновое или полноцветное квадратное изображение разрешением $M \times M$. Цифровой водяной знак представляет собой бинарное изображение разрешением $\frac{M}{8} \times \frac{M}{8}$.

На этапе предобработки значения пикселей, находящиеся в интервале от 0 до 255 необходимо преобразовать таким образом, чтобы они находились в промежутке от -128 до 127 . Для этого от каждого значения пикселя необходимо отнять 128. Далее изображение разбивается на неперекрывающиеся блоки размером 8×8 пикселей, и к каждому блоку применяется ДКП.

Предобработка водяного знака заключается в некотором перемешивающем преобразовании. Популярным методом перемешивания битов ЦВЗ, представленного в виде квадратной матрицы порядка M , является преобразование Арнольда. Данное преобразование изменяет координаты пикселей, создавая запутанную версию исходного ЦВЗ. Преобразование Арнольда выполняется по следующей формуле (1):

$$\begin{bmatrix} i' \\ j' \end{bmatrix} = \begin{bmatrix} 1 & 1 \\ 1 & 2 \end{bmatrix} \begin{bmatrix} i \\ j \end{bmatrix} \bmod M, \quad (1)$$

где i, j – это исходные координаты пикселя, i', j' – новые координаты пикселя, M – порядок матрицы ЦВЗ.

Данное преобразование может выполняться в течение некоторого количества итераций для усиления перемешивающего эффекта.

При восстановлении исходной матрицы изображения соответствующая процедура, обратная преобразованию Арнольда, может быть определена как показано на формуле (2)

$$\begin{bmatrix} i \\ j \end{bmatrix} = \begin{bmatrix} 2 & -1 \\ -1 & 1 \end{bmatrix} \begin{bmatrix} i' \\ j' \end{bmatrix} \bmod M. \quad (2)$$

Для сохранения баланса между робастностью и незаметностью встраивания авторы алгоритма предлагают скрывать биты водяного знака в среднечастотных коэффициентах. Это объясняется тем, что использование низких частот приводит к заметной деградации изображения, в то время как высокочастотные коэффициенты не обладают достаточной устойчивостью к атакам на изображение, содержащее ЦВЗ. В частности, высокочастотные коэффициенты полностью теряются при JPEG-сжатии. Ниже на рисунке 1 показано, каким образом блок ДКП-коэффициентов делится на частотные области. Обход ДКП-коэффициентов осуществляется по зигзагу. Для сокрытия битов ЦВЗ используется только один (любой заранее определенный) среднечастотный коэффициент в каждом блоке.

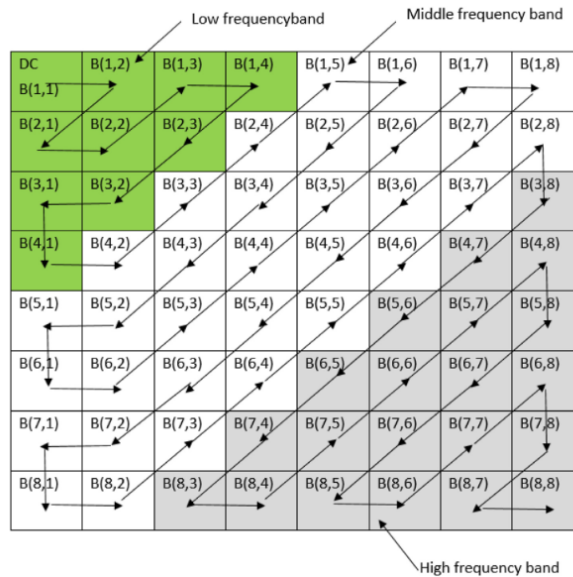


Рисунок 1 – Низкие, средние и высокие частоты коэффициентов ДКП

Перед непосредственной процедурой встраивания необходимо рассчитать значение медианы $Med(B_{P,Q})$ первых девяти AC-коэффициентов ДКП-блока $B_{P,Q}$. Медиана – это срединное значение упорядоченного по возрастанию набора чисел, то есть такое число, что половина из элементов набора не меньше него, а другая половина не больше.

Сила модификации блока $M(B_{P,Q})$ вычисляется следующим образом по формуле (3):

$$M(B_{P,Q}) = \begin{cases} |Z \times Med(B_{P,Q})|, & \text{если } |DC(B_{P,Q})| > 1000 \text{ или } |DC(B_{P,Q})| < 1, \\ \left| Z \times \frac{DC(B_{P,Q}) - Med(B_{P,Q})}{DC(B_{P,Q})} \right|, & \text{иначе,} \end{cases} \quad (3)$$

где Z – параметр алгоритма (по умолчанию $Z = 2$), $DC(B_{P,Q})$ – DC-коэффициент блока $B_{P,Q}$.

Для сокрытия битов ЦВЗ авторы вводят четыре разностных величины для установления соотношений между соседними блоками пикселей. При расчете разностной величины имеет значение порядок выбора блоков: LR – слева направо, UD – сверху вниз, RL – справа налево, DU – снизу вверх, показано в формуле (4).

$$\begin{aligned}\Delta_{LR} &= B_{P,Q}(x, y) - B_{P,Q+1}(x, y), \\ \Delta_{UD} &= B_{P,Q}(x, y) - B_{P+1,Q}(x, y), \\ \Delta_{RL} &= B_{P,Q}(x, y) - B_{P,Q-1}(x, y), \\ \Delta_{DU} &= B_{P,Q}(x, y) - B_{P-1,Q}(x, y).\end{aligned}\tag{4}$$

В представленных выражениях $B_{P,Q}(x, y)$ – это выбранный для встраивания бита ЦВЗ коэффициент, находящийся в блоке $B_{P,Q}$ в ячейке с координатами (x, y) . Из данного коэффициента вычитается соответствующий коэффициент соседнего блока, причем предусмотрено четыре разных способа выбора соседнего блока.

При реализации алгоритма расчет всех четырех разностей не требуется, необходимо выбрать один из представленных вариантов. Обозначим выбранную разность как Δ , а соседний для $B_{P,Q}$ блок – $\tilde{B}_{P,Q}$.

Встраивание битов ЦВЗ выполняется в зависимости от значения Δ и бита водяного знака. При встраивании используются два параметра T и K , которые определяют упомянутые ранее интервалы, соответствующие нулевому и единичному битам ЦВЗ. По умолчанию $T = 80$ и $K = 12$.

Если **бит ЦВЗ равен 1**, то возможны следующие варианты при встраивании информации:

$$1) \quad \Delta > T - K.$$

Тогда, пока $\Delta > T - K$, необходимо выполнять (5)

$$\begin{aligned}B_{P,Q}(x, y) &= B_{P,Q}(x, y) - M(B_{P,Q}), \\ \Delta &= B_{P,Q}(x, y) - \tilde{B}_{P,Q}(x, y).\end{aligned}\tag{5}$$

$$2) \quad K > \Delta > -\frac{T}{2}.$$

Тогда, пока $\Delta < K$, необходимо выполнять (6)

$$\begin{aligned}
B_{P,Q}(x, y) &= B_{P,Q}(x, y) + M(B_{P,Q}), \\
\Delta &= B_{P,Q}(x, y) - \tilde{B}_{P,Q}(x, y).
\end{aligned}
\tag{6}$$

$$3) \quad \Delta < -\frac{T}{2}.$$

Тогда, пока $\Delta > -T - K$, необходимо выполнять (7)

$$\begin{aligned}
B_{P,Q}(x, y) &= B_{P,Q}(x, y) - M(B_{P,Q}), \\
\Delta &= B_{P,Q}(x, y) - \tilde{B}_{P,Q}(x, y).
\end{aligned}
\tag{7}$$

Если **бит ЦВЗ равен 0**, то возможны следующие варианты при встраивании информации:

$$1) \quad \Delta > \frac{T}{2}.$$

Тогда, пока $\Delta \leq T + K$, необходимо выполнять (8)

$$\begin{aligned}
B_{P,Q}(x, y) &= B_{P,Q}(x, y) + M(B_{P,Q}), \\
\Delta &= B_{P,Q}(x, y) - \tilde{B}_{P,Q}(x, y).
\end{aligned}
\tag{8}$$

$$2) \quad -K < \Delta < \frac{T}{2}.$$

Тогда, пока $\Delta \geq -K$, необходимо выполнять (9)

$$\begin{aligned}
B_{P,Q}(x, y) &= B_{P,Q}(x, y) - M(B_{P,Q}), \\
\Delta &= B_{P,Q}(x, y) - \tilde{B}_{P,Q}(x, y).
\end{aligned}
\tag{9}$$

$$3) \quad \Delta < K - T.$$

Тогда, пока $\Delta \leq K - T$, необходимо выполнять (10)

$$\begin{aligned}
B_{P,Q}(x, y) &= B_{P,Q}(x, y) + M(B_{P,Q}), \\
\Delta &= B_{P,Q}(x, y) - \tilde{B}_{P,Q}(x, y).
\end{aligned}
\tag{10}$$

После того, как все биты ЦВЗ будут встроены в блоки изображения, необходимо выполнить обратное ДКП, округлить полученные значения до целых, а после прибавить к каждому из них 128, чтобы вернуть значения пикселей в диапазон от 0 до 255.

Для извлечения водяного знака все пиксели изображения уменьшаются на 128, и к блокам изображения применяется ДКП. Для среднечастотных коэффициентов блока, содержащих биты ЦВЗ, рассчитываются значения Δ (тем же способом, который был выбран при встраивании). Бит водяного знака W_i формируется следующим образом (11):

$$W_i = \begin{cases} 1, & \text{если } (\Delta < -T) \text{ или } ((\Delta > 0) \text{ и } (\Delta < T)), \\ 0, & \text{если } (\Delta > T) \text{ или } ((\Delta < 0) \text{ и } (\Delta > -T)). \end{cases} \quad (11)$$

На последнем этапе к извлеченному водяному знаку нужно применить обратное преобразование Арнольда, если оно использовалось на этапе встраивания для перемешивания водяного знака.

3 Программная реализация

Программная реализация алгоритма встраивания ЦВЗ на основе межблочной разности коэффициентов ДКП была выполнена на языке программирования python с использованием библиотек PIL [2] и Numpy [3]. Программа имеет три главные функции: main, insert и extract. Main отвечает за интерфейс и вызов функций insert и extract, insert – встраивание водяного знака, а extract – извлечение водяного знака. При работе с программой пользователь имеет возможность ввести следующие параметры:

- 1) Режим работы – встраивание или извлечение;
- 2) Выбор ЦВЗ для встраивания, так же реализована поддержка пользовательских ЦВЗ;
- 3) Использование преобразования Арнольда и выбор количества итераций данного преобразования;
- 4) Ввод значения параметра алгоритма Z;
- 5) Ввод параметров T и K;
- 6) Возможность оценки качества встраивания;
- 7) Возможность оценки качества робастности;
- 8) Построение гистограмм исходного изображения и стегоизображения.

Фото пользовательского интерфейса показано на рисунке 3.1. С полным кодом программы можно ознакомиться по ссылке: <https://github.com/il3241/watermark.git>

```
Введите название входного файла (файл должен быть сохранен в папке ./in_image):
cat.jpg
Что необходимо сделать с входным файлом:
1 - встроить ЦВЗ;
2 - извлечь ЦВЗ.
1
Выберите ЦВЗ
1 - ЦВЗ 12x12 (для изображений 256x256);
2 - ЦВЗ 16x16 (для изображений 512x512);
3 - ЦВЗ 8x8 (для изображений 64x64).
Введите название файла со своим ЦВЗ
1
Необходимо применить преобразование Арнольда?
1 - Да;
2 - Нет.
1
Введите кол-во итераций
3
Введите значение параметра алгоритма (значение по умолчанию 2):
2
Введите значения параметров T и K (значение по умолчанию 88 и 12):
88
12
Оценить качество встраивания?
1 - Да;
2 - Нет.
1
=====
MSE
5.367
=====
RMSE
2.304
=====
PSNR
46.883
=====
SSIM
0.983
=====
Оценить устойчивость встраивания к деструктивным воздействиям?
Да - 1
Нет - 2
1
=====
BER_0
0.0
=====
BER_Bit
0.75
=====
BER_Channel
0.915
=====
BER_contour
0.228
=====
BER_smooth
0.463
=====
BER_img_90
0.423
=====
BER_img_60
0.484
=====
BER_img_20
0.512
=====
Построить гистограммы?
Да - 1
Нет - 2
2
```

Рисунок 3.1 – Интерфейс программы

3.1 Результаты работы программы

Теперь перейдем к результатам работы программы. Воспользуемся изображением, представленным на рисунке 3.1.1.



Рисунок 3.1.1 – Контейнер для встраивания

Данное изображение имеет размер 64 на 64 пикселя. Поскольку сторона ЦВЗ должна быть в 8 раз меньше стороны контейнера, воспользуемся ЦВЗ размера 8 на 8 пикселей, такой ЦВЗ показан на рисунке 3.1.2.

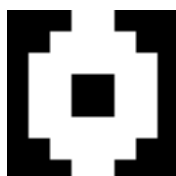


Рисунок 3.1.2 – ЦВЗ, используемый для встраивания

Запустим программу с настройками, представленными на рисунке 3.1.4, при этом необходимо отдельно заметить, что перед встраиванием ЦВЗ в течение трех итераций подвергается преобразованию Арнольда. ЦВЗ после применения данного преобразования представлен на рисунке 3.1.3.

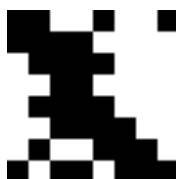


Рисунок 3.1.3 – ЦВЗ после трёх итераций преобразования Арнольда

```
Введите название входного файла (файл должен быть сохранен в папке /in_image)
64.png
Что необходимо сделать с входным файлом:
1 - встроить ЦВЗ;
2 - извлечь ЦВЗ.
1
Выберите ЦВЗ
1 - ЦВЗ 32x32 (для изображений 256x256);
2 - ЦВЗ 16x16 (для изображений 128x128);
3 - ЦВЗ 8x8 (для изображений 64x64).
3
Необходимо применить преобразование Арнольда?
1 - Да;
2 - Нет.
1
Введите кол-во итераций
3
```

Рисунок 3.1.4 – Настройки запуска программы для встраивания ЦВЗ

После выполнения программы мы получаем изображение со встроенным ЦВЗ, продемонстрированное на рисунке 3.1.5.



Рисунок 3.1.5 – Изображение со встроенным ЦВЗ

Теперь перейдем к обратному алгоритму. Возьмем изображение, в которое ранее было встроено ЦВЗ. Запустим программу с настройками, представленными на рисунке 3.1.6.

```
Введите название входного файла (файл должен быть сохранен в папке /in_image)
64em.png
Что необходимо сделать с входным файлом:
1 - встроить ЦВЗ;
2 - извлечь ЦВЗ.
2
При встраивании было применено преобразование Арнольда?
1 - Да;
2 - Нет.
1
Введите кол-во итераций:
3
```

Рисунок 3.1.6 – Настройки запуска программы для извлечения ЦВЗ

Поскольку при встраивании было использовано преобразование Арнольда, извлеченное ЦВЗ будет выглядеть, как на рисунке 3.1.7.



Рисунок 3.1.7 – ЦВЗ, полученный при извлечении из изображения

Для того чтобы вернуть его к исходному состоянию необходимо применить обратное преобразование Арнольда, после которого ЦВЗ будет выглядеть, как показано на рисунке 3.1.8.



Рисунок 3.1.8 – Извлеченный ЦВЗ, после обратного преобразования Арнольда

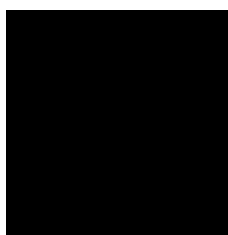
4 Результаты вычислительных экспериментов при отсутствии атак

Перейдем к вычислительным экспериментам с полученной программой. Изображение для встраивания представлено на рисунке 4.1.

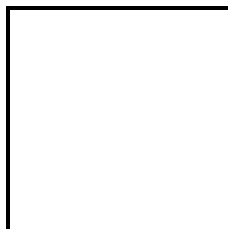


Рисунок 4.1 – Изображение для вычислительных экспериментов

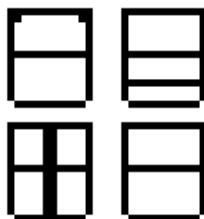
Для начала оценим незаметность встраивание, при отсутствии атак. При оценке будут собираться такие метрики, как MSE, RMSE, PSNR, SSIM. Проведем встраивание трех разных ЦВЗ, показанных на рисунках 4.2 – 4.4.



4.2 – ЦВЗ, состоящий только из черных пикселей



4.3 – ЦВЗ, состоящий только из белых пикселей



4.4 – ЦВЗ, из черных и белых пикселей

По очереди запустим программы с разными ЦВЗ, настройки запуска приведены на рисунках 4.5 – 4.7.

```

Введите название входного файла (файл должен быть сохранен в папке /in_image)
cat.png
Что необходимо сделать с входным файлом:
1 - встроить ЦВЗ;
2 - извлечь ЦВЗ.
1
Выберите ЦВЗ
1 - ЦВЗ 32x32 (для изображений 256x256);
2 - ЦВЗ 16x16 (для изображений 128x128);
3 - ЦВЗ 8x8 (для изображений 64x64).
Либо введите название файла со своим ЦВЗ
32b.png
Необходимо применить преобразование Арнольда?
1 - Да;
2 - Нет.
2
Введите значение параметра алгоритма (значение по умолчанию 2):
2
Оценить качество встраивания?
1 - Да;
2 - Нет.
1
=====
MSE
5.05
-----
RMSE
2.247
-----
PSNR
41.098
-----
SSIM
0.983
=====

```

Рисунок 4.5 – Настройки запуска программы для ЦВЗ из черных пикселей

```

Введите название входного файла (файл должен быть сохранен в папке /in_image)
cat.png
Что необходимо сделать с входным файлом:
1 - встроить ЦВЗ;
2 - извлечь ЦВЗ.
1
Выберите ЦВЗ
1 - ЦВЗ 32x32 (для изображений 256x256);
2 - ЦВЗ 16x16 (для изображений 128x128);
3 - ЦВЗ 8x8 (для изображений 64x64).
Либо введите название файла со своим ЦВЗ
32w.png
Необходимо применить преобразование Арнольда?
1 - Да;
2 - Нет.
2
Введите значение параметра алгоритма (значение по умолчанию 2):
2
Оценить качество встраивания?
1 - Да;
2 - Нет.
1
=====
MSE
5.151
-----
RMSE
2.27
-----
PSNR
41.012
-----
SSIM
0.984
=====

```

Рисунок 4.6 – Настройки запуска программы для ЦВЗ из белых пикселей

```

Введите название входного файла (файл должен быть сохранен в папке /in_image)
cat.png
Что необходимо сделать с входным файлом:
1 - встроить ЦВЗ;
2 - извлечь ЦВЗ.
1
Выберите ЦВЗ
1 - ЦВЗ 32x32 (для изображений 256x256);
2 - ЦВЗ 16x16 (для изображений 128x128);
3 - ЦВЗ 8x8 (для изображений 64x64).
Либо введите название файла со своим ЦВЗ
1
Необходимо применить преобразование Арнольда?
1 - Да;
2 - Нет.
2
Введите значение параметра алгоритма (значение по умолчанию 2):
2
Оценить качество встраивания?
1 - Да;
2 - Нет.
1
=====
MSE
5.014
-----
RMSE
2.239
-----
PSNR
41.129
-----
SSIM
0.983
=====

```

Рисунок 4.7 – Настройки запуска программы для ЦВЗ из комбинации белых и черных пикселей

Все метрики собранные при работе программы приведены в таблице 4.1.

Таблица 4.1 – Метрики, полученные при встраивании без атак.

	ЦВЗ чёрный	ЦВЗ белый	ЦВЗ из комбинации
MSE	5.05	5.151	5.014
RMSE	2.247	2.27	2.239
PSNR	41.098	41.012	41.129
SSIM	0.983	0.984	0.983

MSE – метрика показывающая разницу между значениями пикселей изображения до и после встраивания, соответственно, чем меньше это значение, тем незаметнее встраивание. RMSE – метрика являющаяся корнем из MSE, а следовательно для так же верно это утверждение. PSNR, метрика, так же характеризующая незаметность, при этом чем больше данное значение тем вставка незаметнее. Так же одной из метрик демонстрирующей незаметность встраивания является SSIM. Значение данной метрики лежит в диапазоне от 0 до 1, где 1 говорит о полном сходстве изображения до и после встраивания. Как можно заметить из таблицы, все показатели незаметности остаются примерно на одном уровне, независимо от того, какой из трех ЦВЗ встраивался. Из этого можно сделать вывод, что незаметность встраивания мало зависит от цвета пикселей встраиваемого ЦВЗ. Так же из этого можно сделать вывод, что преобразования Арнольда не сказываются на каких-либо характеристиках изображения, поскольку каждое преобразование можно рассмотреть, как новый ЦВЗ, а следовательно, из утверждения выше это не должно влиять на характеристики изображения. Подводя итог: преобразование Арнольда, только маскирует сам ЦВЗ. По этой причине все дальнейшие эксперименты будут проходить при отсутствии преобразования Арнольда. Извлечении ЦВЗ из всех ранее полученных изображений проходит без ошибок.

Теперь изменим значение параметра алгоритма Z , при условии отсутствия атак. Для эксперимента возьмем значения Z равные 1, 2 (значение по умолчанию), 10, 50, 100. Настройки запуска программы представлены на рисунке 4.8 – 4.12, в качестве ЦВЗ будет использован ЦВЗ, представленный на рисунке 4.4.

```

Введите название входного файла (файл должен быть сохранен в папке ./in_image)
cat.png
Что необходимо сделать с входным файлом:
1 - встроить ЦВЗ;
2 - извлечь ЦВЗ.
1
Выберите ЦВЗ
1 - ЦВЗ 32x32 (для изображений 256x256);
2 - ЦВЗ 16x16 (для изображений 128x128);
3 - ЦВЗ 8x8 (для изображений 64x64).
Либо введите название файла со своим ЦВЗ
1
Необходимо применить преобразование Ариольда?
1 - Да;
2 - Нет.
2
Введите значение параметра алгоритма (значение по умолчанию 2):
1
Оценить качество встраивания?
1 - Да;
2 - Нет.
1
-----
MSE
4.763
-----
RMSE
2.182
-----
PSNR
41.352
-----
SSIM
0.984
-----
Оценить устойчивость встраивания к деструктивным воздействиям?
Да - 1
Нет - 2
1
-----
BER_0
0.0

```

Рисунок 4.8 – Настройки запуска при параметре алгоритма равным 1

```

Введите название входного файла (файл должен быть сохранен в папке ./in_image)
cat.png
Что необходимо сделать с входным файлом:
1 - встроить ЦВЗ;
2 - извлечь ЦВЗ.
1
Выберите ЦВЗ
1 - ЦВЗ 32x32 (для изображений 256x256);
2 - ЦВЗ 16x16 (для изображений 128x128);
3 - ЦВЗ 8x8 (для изображений 64x64).
Либо введите название файла со своим ЦВЗ
1
Необходимо применить преобразование Ариольда?
1 - Да;
2 - Нет.
2
Введите значение параметра алгоритма (значение по умолчанию 2):
2
Оценить качество встраивания?
1 - Да;
2 - Нет.
1
-----
MSE
5.014
-----
RMSE
2.239
-----
PSNR
41.129
-----
SSIM
0.983
-----
Оценить устойчивость встраивания к деструктивным воздействиям?
Да - 1
Нет - 2
1
-----
BER_0
0.0

```

Рисунок 4.9 – Настройки запуска при параметре алгоритма равным 2

```

Введите название входного файла (файл должен быть сохранен в папке ./in_image)
cat.png
Что необходимо сделать с входным файлом:
1 - встроить ЦВЗ;
2 - извлечь ЦВЗ.
1
Выберите ЦВЗ
1 - ЦВЗ 32x32 (для изображений 256x256);
2 - ЦВЗ 16x16 (для изображений 128x128);
3 - ЦВЗ 8x8 (для изображений 64x64).
Либо введите название файла со своим ЦВЗ
1
Необходимо применить преобразование Ариольда?
1 - Да;
2 - Нет.
2
Введите значение параметра алгоритма (значение по умолчанию 2):
10
Оценить качество встраивания?
1 - Да;
2 - Нет.
1
-----
MSE
7.155
-----
RMSE
2.675
-----
PSNR
39.584
-----
SSIM
0.975
-----
Оценить устойчивость встраивания к деструктивным воздействиям?
Да - 1
Нет - 2
1
-----
BER_0
0.0

```

Рисунок 4.10 – Настройки запуска при параметре алгоритма равным 10

```

Введите название входного файла (файл должен быть сохранен в папке ./in_image)
cat.png
Что необходимо сделать с входным файлом:
1 - встроить ЦВЗ;
2 - извлечь ЦВЗ.
1
Выберите ЦВЗ
1 - ЦВЗ 32x32 (для изображений 256x256);
2 - ЦВЗ 16x16 (для изображений 128x128);
3 - ЦВЗ 8x8 (для изображений 64x64).
либо введите название файла со своим ЦВЗ
1
Необходимо применить преобразование Арнольда?
1 - Да;
2 - Нет.
2
Введите значение параметра алгоритма (значение по умолчанию 2):
50
Оценить качество встраивания?
1 - Да;
2 - Нет.
1
-----
MSE
39.97
-----
RMSE
6.322
-----
PSNR
32.113
-----
SSIM
0.87
-----
Оценить устойчивость встраивания к деструктивным воздействиям?
Да - 1
Нет - 2
1
-----
BER_0
0.009765625

```

Рисунок 4.11 – Настройки запуска при параметре алгоритма равным 50

```

Введите название входного файла (файл должен быть сохранен в папке ./in_image)
cat.png
Что необходимо сделать с входным файлом:
1 - встроить ЦВЗ;
2 - извлечь ЦВЗ.
1
Выберите ЦВЗ
1 - ЦВЗ 32x32 (для изображений 256x256);
2 - ЦВЗ 16x16 (для изображений 128x128);
3 - ЦВЗ 8x8 (для изображений 64x64).
либо введите название файла со своим ЦВЗ
1
Необходимо применить преобразование Арнольда?
1 - Да;
2 - Нет.
2
Введите значение параметра алгоритма (значение по умолчанию 2):
100
Оценить качество встраивания?
1 - Да;
2 - Нет.
1
-----
MSE
155.534
-----
RMSE
12.471
-----
PSNR
26.213
-----
SSIM
0.683
-----
Оценить устойчивость встраивания к деструктивным воздействиям?
Да - 1
Нет - 2
1
-----
BER_0
0.7734375

```

Рисунок 4.12 – Настройки запуска при параметре алгоритма равным 100

Параметры, собранные в ходе данных экспериментов представлены в таблице 4.2.

Таблица 4.2 – Метрики, полученные при встраивании без атак.

	Z = 1	Z = 2	Z = 10	Z = 50	Z = 100
MSE	4.763	5.014	7.155	39.97	155.534
RMSE	2.182	2.239	2.675	6.322	12.471
PSNR	41.352	41.129	39.584	32.113	26.213
SSIM	0.984	0.983	0.975	0.87	0.683

Проанализируем полученные результаты. Хотя при увеличении значения увеличивается заметность встраивания, при Z равном 1, 2 и 10, встраивание можно считать незаметным. ЦВЗ из полученных изображений извлекается без единой ошибки. Отдельно рассмотрим результаты при Z равном 50 и 100.

При значении параметра алгоритма равном 50, встраивание все еще можно считать незаметным, хотя показатели заметно ухудшились относительно значения по умолчанию. Но, что более значимо, при таком значении ЦВЗ извлекается с ошибками. Запустим программу со значениями, представленными на рисунке 4.13.

```

Введите название входного файла (файл должен быть сохранен в папке /in_image)
z50.png
Что необходимо сделать с входным файлом:
1 - встроить ЦВЗ;
2 - извлечь ЦВЗ.
2
При встраивании было применено преобразование Арнольда?
1 - Да;
2 - Нет.
2

```

Рисунок 4.13 – Настройки запуска для извлечения ЦВЗ, при параметре 50

При этом извлеченный ЦВЗ показан на рисунке 4.14.

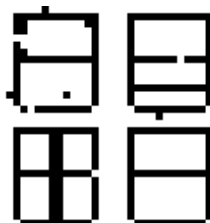


Рисунок 4.14 – Извлеченный ЦВЗ, при параметре 50

Как видно из рисунка 4.14, хотя ЦВЗ и был извлечен с ошибками, его еще можно распознать.

Теперь перейдем к значению алгоритма равном 100. При данном Z встраивание уже нельзя назвать незаметным. К примеру, значение SSIM опустилось к 0.683, что говорит о том, что изображения мало похожи. Попробуем извлечь ЦВЗ из полученного изображения. Настройки запуска программы приведены на рисунке 4.15.

```

Введите название входного файла (файл должен быть сохранен в папке /in_image)
z100.png
Что необходимо сделать с входным файлом:
1 - встроить ЦВЗ;
2 - извлечь ЦВЗ.
2
При встраивании было применено преобразование Арнольда?
1 - Да;
2 - Нет.
2

```

Рисунок 4.15 - Настройки запуска для извлечения ЦВЗ, при параметре 100

При этом извлеченный ЦВЗ показан на рисунке 4.16.



Рисунок 4.16 – Извлеченный ЦВЗ, при параметре 100

Как можно заметить, при таком значении ЦВЗ практически полностью уничтожается из-за большого количества ошибок, но структура все же остается. Из

данного эксперимента можно сделать вывод, что увеличение параметра Z увеличивает заметность и количество ошибок при извлечении ЦВЗ. В подтверждение слов о увеличении количества ошибок, для данного эксперимента были собраны значения BER, приведенные в таблице 4.3.

Таблица 4.3 – Значения BER.

	$Z = 1$	$Z = 2$	$Z = 10$	$Z = 50$	$Z = 100$
BER	0.0	0.0	0.0	0.009765625	0.7734375

Данная таблица подтверждает слова о том, что при параметрах с 1 до 10 извлечение проходит без ошибок, при параметре 50 с небольшим количеством ошибок, а при значении 100, исходный ЦВЗ извлечется с искажениями, хотя и сохранит общую структуру. Отдельно стоит отметить разницу между наличием ошибок и не читаемостью ЦВЗ. Поскольку ЦВЗ представляет из себя черно-белую картинку, привычная шкала для метрики BER не применима в контексте оценки читаемости извлеченного ЦВЗ. BER вычисляется по формуле (12):

$$BER = \frac{B_e}{B}, \quad (12)$$

где B_e – количество неверно извлеченных битов;

B – количество встраиваемых битов.

Из формулы вытекает следующее соотношение: чем больше количество ошибок, тем больше BER и соответственно BER равный 1 говорит о полной ошибочности извлеченного сообщения. Но в контексте данной ситуации BER равный единицы говорит о полной инверсии пикселей ЦВЗ. То есть при BER равной 1 будет извлечен исходный ЦВЗ, где чёрные пиксели заменятся на белые и наоборот. А следовательно, при значениях метрики близких к 0 или 1 извлеченный ЦВЗ будет читаем, а при значениях близких к 0.5 извлечь ЦВЗ будет невозможно.

Подводя итог эксперимента, чем больше параметр Z , а следовательно, и сила встраивания, тем сильнее меняются изображение, а следовательно встраивание становится более заметным. Так же отдельно стоит заметить, что при больших значениях Z ЦВЗ инвертируется по цвету.

Перейдем к измерению незаметности, при изменении параметров T (по умолчанию равен 80) и K (по умолчанию равен 12). В ходе данного эксперимента в качестве изображения для встраивания будет служить, показанное на рисунке 4.1, а ЦВЗ,

представленный на рисунке 4.4. Параметры в ходе данного эксперимента будут принимать значение приведенные в таблице 4.4

Таблица 4.4 – Значения Т и К.

Т	1000000000	40	12	80	80	80	12	80
К	12	12	12	80	6	0	80	12

Запустим программу с измененными параметрами Т и К. Все метрики, собранные в ходе данного эксперимента представлены в таблице 4.5.

Таблица 4.5 – Метрики, при измененных Т и К.

	Т = 1000000000, К = 12	Т = 40, К = 12	Т = 12, К = 12	Т = 80, К = 80	Т = 80, К = 6	Т = 80, К = 0	Т = 12, К = 80	Т = 80, К = 12
MSE	6.18	4.277	4.269	75.633	2.826	1.547	82.749	5.014
RMSE	2.486	2.068	2.066	8.697	1.681	1.244	9.097	2.239
PSNR	40.221	41.819	41.827	29.344	43.62	46.236	28.953	41.129
SSIM	0.982	0.985	0.987	0.8	0.992	0.997	0.771	0.983
BER	0.001	0.0	0.610	0.798	0.0	0.0918	0.194	0.0

Первое, что необходимо заметить, любое изменение параметров Т и К, повлияло на возможность извлечения ЦВЗ из стегоизображения. При сильном увеличении параметра Т в первом случае, по сравнению со случаем, где значения параметров взяты по умолчанию, увеличилась заметность данного встраивания, но все же осталась на довольно приемлемом уровне, и такое встраивание можно назвать незаметным. Так, к примеру параметр SSIM остался практически на том же значении. Обратимся к параметру BER, как видно из таблицы, он так же остался практически на нулевом уровне, попробуем извлечь ЦВЗ, при параметре Т = 1000000000, извлеченный ЦВЗ продемонстрирован на рисунке 4.23.

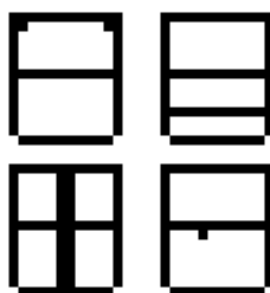
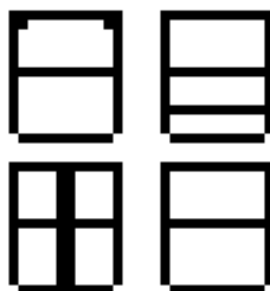


Рисунок 4.23 – ЦВЗ извлеченный из стегоизображения при Т = 1000000000, К = 12

Как видно из рисунка 4.23, извлечение действительно прошло практически без ошибок, из 1024 битов один был извлечен с ошибкой. Из этого измерения можно сделать вывод, что увеличение параметра Т в большей степени влияет на незаметность и в меньшей на возможность извлечения исходного ЦВЗ.

При значениях $T = 40$ и $K = 12$, параметры незаметности улучшились, относительно ситуации при значениях по умолчанию, а BER остался равным нулю, что говорит, о возможности извлечь исходный ЦВЗ без ошибок. Результат извлечения представлен на рисунке 4.24.



4.24 – ЦВЗ извлеченный из стегоизображения при $T = 40$, $K = 12$

Возможно, данные значения параметров T и K были бы более подходящими для данного встраивания, чем значения по умолчанию, но поскольку пока не исследовано устойчивость к атакам, при таких параметрах, об этом нельзя судить окончательно.

При значениях $T = 12$ и $K = 12$, параметры незаметности опять же улучшились относительно ситуации при значениях по умолчанию, но при этом значение параметра BER ухудшилось настолько, что извлечь исходный ЦВЗ, невозможно. Результат извлечения показан на рисунке 4.25.



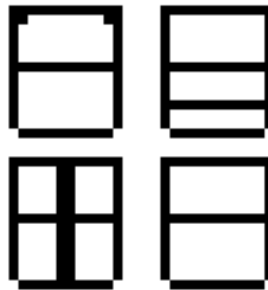
4.25 – ЦВЗ извлеченный из стегоизображения при $T = 12$, $K = 12$

При значениях $T = 80$ и $K = 80$ появилось сильно увеличение параметров MSE, RMSE и уменьшение PSNR и SSIM это говорит, что такое встраивание достаточно заметно. И ко всему возрос и параметр BER, теперь он принимает значение близкое к 1, что говорит о том, что извлеченный ЦВЗ если и сохранит структуру, то будет инвертирован по цвету. Результат извлечение представлен на рисунке 4.26.



4.26 – ЦВЗ извлеченный из стегоизображения при $T = 80$, $K = 80$

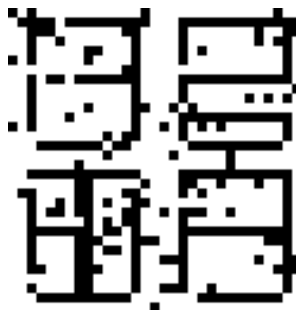
При значениях $T = 80$ и $K = 6$ параметры MSE, RMSE, PSN, SSIM показывают, что такое встраивание незаметнее, чем при значениях по умолчанию, а параметр BER не изменился, что говорит о возможности извлечения без ошибок. Результат извлечения показан на рисунке 4.27.



4.27 – ЦВЗ извлеченный из стегоизображения при $T = 80$, $K = 6$

Как и в ситуации, при $T = 40$, а $K = 12$, возможно такие параметры лучше подходят для данного случая встраивания, но пока не исследована их устойчивость к атакам, окончательно судить нельзя.

При значениях $T = 80$ и $K = 0$ параметры MSE, RMSE, PSN, SSIM показывают, что такое встраивание опять же незаметнее, чем при значениях по умолчанию. Но ухудшился параметр BER. Результат извлечения при таких значения показан на рисунке 4.28.



4.28 – ЦВЗ извлеченный из стегоизображения при $T = 80$, $K = 0$

Как видно из рисунка в извлеченном ЦВЗ узнаются контуры исходного, но все же ошибок достаточно много.

При значениях $T = 12$ и $K = 80$ параметры MSE, RMSE, PSN, SSIM показывают на сильную заметность такого встраивания, а параметр BER на трудность извлечения исходного ЦВЗ. Результат извлечения на рисунке 4.29. Как видно из рисунка извлечь исходный ЦВЗ не получится.



4.28 – ЦВЗ извлеченный из стегоизображения при $T = 12$, $K = 80$

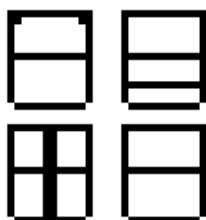
Из проведенного эксперимента можно сделать вывод, что параметры T и K необходимо подбирать в каждом конкретном случае. При этом уменьшение интервала (K , T) положительно сказывается на незаметности встраивания, так как в таком случае в ходе встраивания ЦВЗ блоки коэффициентов меньше изменяются, а следовательно, стегоизображение меньше отличается от исходного.

5 Результаты вычислительных экспериментов при наличии атак

Теперь перейдем к вычислительным экспериментам при наличии атак на стегоизображение, в качестве атак будут использоваться: JPEG сжатие при разных степенях сжатия, наложение фильтров blur, sharpen, contour, smooth. В качестве изображения для встраивания воспользуемся представленным на рисунке 5.1, а ЦВЗ представлено на рисунке 5.2.



Рисунок 5.1 – Изображение для вычислительных экспериментов



5.2 – ЦВЗ для вычислительных экспериментов

В ходе данного эксперимента будет исследована зависимость устойчивости к атакам от параметра алгоритма Z . При этом параметр будет принимать значения, приведенные в таблице 5.1

Таблица 5.1 – Значения Z .

Z	1	2	10	50	100
-----	---	---	----	----	-----

Запустим программу последовательно изменяя параметр алгоритма, заметим, что при значениях Z используемых в ходе данного эксперимента встраивание можно считать незаметным для человеческого глаза, поэтому для демонстрации наложения эффектов на изображения будут использоваться только стегоизображения созданные с параметром 2, они показаны на рисунках 5.3 – 5.8.



Рисунок 5.3 – Наложение эффекта blur



Рисунок 5.4 – Наложение эффекта sharpen



Рисунок 5.5 – Наложение эффекта contour



Рисунок 5.6 – Наложение эффекта smooth



Рисунок 5.6 – Наложение эффекта jpeg сжатие с параметром quality = 90



Рисунок 5.7 – Наложение эффекта jpeg сжатие с параметром quality = 60



Рисунок 5.8 – Наложение эффекта jpeg сжатие с параметром quality = 20

Все метрики, собранные в ходе данного эксперимента отображены в таблице 5.2. Поскольку изображение подвергалось атакам, собранные после их проведения метрики BER имеют название в формате: BER_attackname, метрика под названием BER_0 – это BER при отсутствии атак. Как уже было сказано ранее, чем ближе метрика BER к 0.5, тем извлеченный ЦВЗ меньше будет похож на исходный, чем ближе к 0, тем идентичнее исходному, а чем ближе к 1, тем опять же идентичнее исходному, но инвертированному по цвету.

Таблица 5.2 – Собранные в ходе эксперимента метрики

	Z = 1	Z = 2	Z = 10	Z = 50	Z = 100
BER_0	0.0	0.0	0.0	0.013	0.927
BER_blur	0.312	0.308	0.272	0.239	0.23
BER_sharpen	0.001	0.001	0.006	0.926	0.973
BER_contour	0.064	0.073	0.518	0.982	0.991
BER_smooth	0.002	0.001	0.001	0.005	0.012
BER_jpeg_100	0.141	0.137	0.1	0.035	0.021
BER_jpeg_60	0.498	0.495	0.488	0.495	0.475
BER_jpeg_20	0.541	0.542	0.53	0.526	0.525

При значении параметра алгоритма равном 1, исходный ЦВЗ возможно извлечь только после атаки фильтрами sharpen, contour, smooth, извлеченные ЦВЗ продемонстрированы на рисунках 5.9 - 5.11.

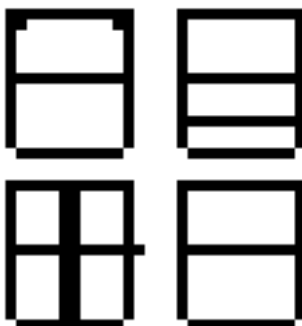


Рисунок 5.9 – ЦВЗ, извлеченный при параметре Z = 1, после атаки фильтром sharpen

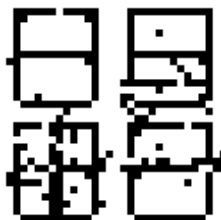


Рисунок 5.10 – ЦВЗ, извлеченный при параметре $Z = 1$, после атаки фильтром contour

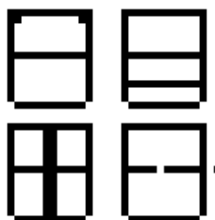


Рисунок 5.11 – ЦВЗ, извлеченный при параметре $Z = 1$, после атаки фильтром smooth
 Как видно из рисунков выше после данных атак, ЦВЗ извлекается практически без ошибок. Отдельно рассмотрим ЦВЗ, извлекаемый после атак jpeg сжатием, результат продемонстрирован на рисунках 5.12 - 5.14.

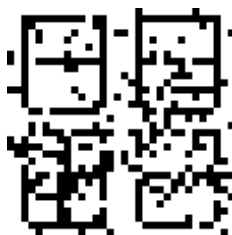


Рисунок 5.12 – ЦВЗ, извлеченный при параметре $Z = 1$, после атаки jpeg сжатием с
 quality = 100



Рисунок 5.13 – ЦВЗ, извлеченный при параметре $Z = 1$, после атаки jpeg сжатием с
 quality = 60



Рисунок 5.14 – ЦВЗ, извлеченный при параметре $Z = 1$, после атаки jpeg сжатием с
 quality = 20

Можно заметить, что при высоком качестве сжатия ЦВЗ извлекается с ошибками, но при этом сохраняет общую структуру и остается различим. При более низких качествах сжатия извлечь исходный ЦВЗ невозможно.

При значении параметра алгоритма равном 2, исходный ЦВЗ возможно извлечь также только после атаки фильтрами *sharpen*, *contour*, *smooth*. Стоит отметить, что при использовании параметра по умолчанию возросла устойчивость к таким атакам, как *blur*, *smooth* и *jpeg* сжатие, а к атаке *contour* устойчивость понизилась. Извлеченные, после проведенных атак, ЦВЗ не показаны, в виду малого того, что показатели мало отличаются от предыдущего опыта, но на рисунке 5.15 продемонстрирован ЦВЗ, извлеченный после атаки *jpeg* сжатием с высоким качеством для демонстрации динамики повышения устойчивости к данной атаке.



Рисунок 5.15 – ЦВЗ, извлеченный при параметре $Z = 2$, после атаки *jpeg* сжатием с $\text{quality} = 100$

При значении алгоритма равном 10 увеличилась устойчивость изображения к таким атакам, как *blur* и *jpeg* сжатие, однако устойчивость ко всем остальным ухудшилась. Извлечем ЦВЗ из изображения на которое была проведена атака *jpeg* сжатием с высоким качеством, результат продемонстрирован на рисунке 5.16.



Рисунок 5.16 – ЦВЗ, извлеченный при параметре $Z = 10$, после атаки *jpeg* сжатием с $\text{quality} = 100$

Как видно в данном случае уменьшилось количество ошибок и извлеченный ЦВЗ становится более различимым. Отдельно стоит заметить резко упавшую устойчивость к атаке *contour*, извлеченный ЦВЗ показан на рисунке 5.17.



Рисунок 5.17 – ЦВЗ, извлеченный при параметре $Z = 10$, после атаки *contour*

При параметре равном 50 повысилась устойчивость к атаке *jpeg* сжатием и *blur*, однако показатели, после таких атак, как *sharpen*, *contour* стали близкими к 1, как уже говорилось ранее, из за особенностей метрики BER в рамках данной практической работы

такие значения говорят об инвертации значений пикселей ЦВЗ, демонстрируется данный эффект на рисунках 5.18 и 5.19.



Рисунок 5.18 – ЦВЗ, извлеченный при параметре $Z = 50$, после атаки sharpen

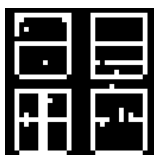


Рисунок 5.19 – ЦВЗ, извлеченный при параметре $Z = 50$, после атаки contour
Так же на рисунке 5.20 продемонстрирован ЦВЗ, извлеченный после атаки jpeg сжатием.

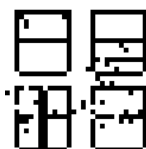


Рисунок 5.20 – ЦВЗ, извлеченный при параметре $Z = 50$, после атаки jpeg сжатием с
quality = 100

При параметре равном 100, наблюдается следующая ситуация, даже при отсутствии атак на изображение параметр BER близок к единице, что говорит об инвертации ЦВЗ. Устойчивость к атакам возросла, за исключением атаки smooth, однако значение остается приемлемым для извлечения ЦВЗ, показано на рисунке 5.21. Так же на рисунке 5.22 отдельно продемонстрирован ЦВЗ, извлеченный после атаки сжатием с высоким качеством.

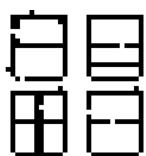


Рисунок 5.21 – ЦВЗ, извлеченный при параметре $Z = 100$, после атаки smooth

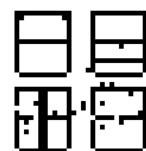


Рисунок 5.22 – ЦВЗ, извлеченный при параметре $Z = 100$, после атаки jpeg сжатием с
quality = 100

Увеличение параметра Z положительно сказывается на устойчивости изображения состроенным ЦВЗ к практически всем типам рассмотренных атак.

Теперь проведем эксперименты с изменением параметров Т и К. В процессе эти параметры будут принимать значения, показанные в таблице 5.3.

Таблица 5.3 – Значения Т и К.

Т	40	80	80	80
К	12	6	0	12

Поскольку при изменении этих параметров встраивание остается незаметным для человеческого глаза, внешний вид изображений подверженных атакам продемонстрирован на рисунках 5.3 – 5.8. В ходе эксперимента были получены данные, собранные в таблицу 5.4

Таблица 5.4 – Значения BER.

	Т = 40, К = 12	Т = 80, К = 6	Т = 80, К = 0	Т = 80, К = 12
BER_0	0.0	0.0	0.119	0.0
BER_blur	0.303	0.361	0.474	0.308
BER_sharpen	0.015	0.001	0.156	0.001
BER_contour	0.85	0.043	0.311	0.073
BER_smooth	0.018	0.004	0.262	0.001
BER_jpeg_100	0.134	0.258	0.446	0.137
BER_jpeg_60	0.496	0.5	0.507	0.495
BER_jpeg_20	0.54	0.532	0.542	0.542

При Т = 40, К = 12 извлечь ЦВЗ возможно после таких атак, как sharpen, smooth, jpeg сжатие с высоким качеством, при извлечении ЦВЗ после атаки contour, пиксели инвертируются. Извлеченные ЦВЗ продемонстрированы на рисунках 5.23 – 5.25.

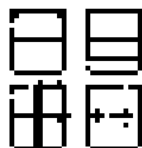


Рисунок 5.23 – ЦВЗ, извлеченный при Т = 40, К = 12, после атаки sharpen



Рисунок 5.24 – ЦВЗ, извлеченный при Т = 40, К = 12, после атаки contour

После данной атаки, извлеченный ЦВЗ имеет достаточное количество ошибок и инвертирован по цвету, но очертания остаются и ЦВЗ можно назвать различимым.

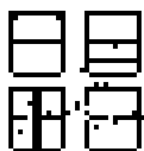


Рисунок 5.24 – ЦВЗ, извлеченный при $T = 40$, $K = 12$, после атаки smooth



Рисунок 5.26 – ЦВЗ, извлеченный при $T = 40$, $K = 12$, после атаки jpeg сжатием с $quality = 100$

При $T = 80$, $K = 6$ возможно извлечь ЦВЗ после тех же атак, что и в предыдущем случае. Отдельно стоит отметить, что ухудшилась стойкость к атаке сжатием, а также после атаки contour ЦВЗ не инвертируется, оба извлеченных ЦВЗ показаны на рисунках 5.27 и 5.28.

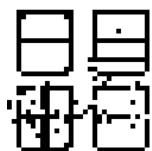


Рисунок 5.27 – ЦВЗ, извлеченный при $T = 80$, $K = 6$, после атаки contour



Рисунок 5.28 – ЦВЗ, извлеченный при $T = 80$, $K = 6$, после атаки jpeg сжатием с $quality = 100$

При $T = 80$, $K = 0$ возможно извлечь ЦВЗ только после атаки sharpen, показано на рисунке 5.29. Данное сочетание даёт крайне низкую робастность.



Рисунок 5.29 – ЦВЗ, извлеченный при $T = 80$, $K = 6$, после атаки jpeg сжатием с $quality = 100$

При $T = 80$, $K = 12$ — это сочетание параметров по умолчанию. ЦВЗ возможно извлечь после атак sharpen, contour, smooth, jpeg с качеством 100. ЦВЗ, извлеченный после атаки сжатием продемонстрирован на рисунке 5.30.

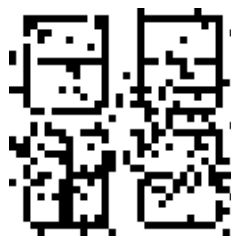


Рисунок 5.30 – ЦВЗ, извлеченный при $T = 80$, $K = 12$, после атаки jpeg сжатием с $quality = 100$

Исходя из данного и предыдущего эксперимента, где изменялись значения T и K , но без проведения атак, и оценки робастности можно сделать несколько выводов:

- 1) Уменьшение параметров T и K негативно сказывается на робастности и положительно на незаметности. В рамках первого эксперимента выдвигалось предположение, что значения параметров T и K равные 40 и 12, а также 80 и 6 более интересны для данного случая встраивания. Данное предположение на мой взгляд оказалось ошибочным, так как улучшение незаметности, которое даёт данное изменение, нивелируется ухудшением робастности.
- 2) Сильное уменьшение K до 0 также, хотя и дает преимущество в незаметности, но ухудшает робастность.

6 Выводы

В ходе данной практической работы был изучен алгоритм встраивания ЦВЗ на основе межблочной разности коэффициентов ДКП. Данный алгоритм обладает рядом преимуществ: гибкая настройка алгоритма, подбирая сочетания параметров можно осуществить встраивание практически под любые требования незаметности, робастности. Высокая незаметность встраивания, даже при параметрах по умолчанию.

7 ЛИТЕРАТУРА

[1] Ko H.-J., Huang C.-T., Horng G., Shiuh-Jeng W. Robust and blind image watermarking in DCT domain using inter-block coefficient correlation // Information Sciences. – 2020. – Vol. 517. – P. 128–147.

[2] Документация библиотеки PIL – URL:
<https://pillow.readthedocs.io/en/stable/index.html> Fridrich J. Steganography in Digital

[3] Документация библиотеки Matplotlib – URL:
<https://matplotlib.org/stable/index.html>