

Муниципальное общеобразовательное учреждение

Гимназия «Дмитров»

Исследовательский проект по информатике

Невидимые письма: как спрятать текст внутри картинки

(стеганография и младшие биты пикселей)

Выполнила: ученица 7 класса

Красилова Ариана

Руководитель проекта:

(ФИО руководителя)

г. Дмитров, 2026 год

Содержание

Введение	3
1.1. Проблема и актуальность	3
1.2. Гипотеза, цель и задачи	4
1.3. Методы исследования	4
2. Теоретическая часть	5
2.1. Из чего состоит картинка	5
2.2. Что такое младший бит	5
2.3. Как спрятать и достать письмо	6
2.4. Чем измерять заметность: PSNR	6
3. Практическая часть	7
4. Результаты и их анализ	8
5. Заключение	11
Список источников	12

Введение

1.1. Проблема и актуальность

Иногда нужно передать сообщение так, чтобы посторонний даже не догадался, что оно вообще существует. Обычный шифр прячет смысл письма, но сам факт переписки виден: все понимают, что перед ними тайное сообщение. А можно поступить хитрее — спрятать текст внутри обычной фотографии. Снаружи это просто картинка, и никто не станет искать в ней письмо.

Наука о том, как прятать сам факт сообщения, называется **стеганографией** (от греческих слов «скрытый» и «пишу»). Этим приёмом пользуются очень давно, а сегодня проще всего прятать текст именно в цифровых картинках, потому что в них миллионы пикселей, и каждый можно чуть-чуть изменить незаметно для глаза.

Тема актуальна, потому что мы каждый день обмениваемся фотографиями в мессенджерах, и полезно понимать, что в картинке может скрываться больше, чем видно, и почему такое письмо иногда «ломается» при пересылке.

1.2. Гипотеза, цель и задачи

Гипотеза. Если изменять в каждом пикселе только самые младшие биты, можно спрятать в картинку целое письмо так, что глаз не заметит разницы. Но чем больше битов задействовать ради объёма, тем сильнее портится картинка, и в какой-то момент подмена становится видна.

Цель работы — показать на практике, как спрятать текст внутри картинки, и измерить, сколько текста помещается и когда тайник становится заметным.

Задачи:

1. разобраться, из чего состоит цифровая картинка и что такое младший бит;
2. описать способ, как спрятать письмо в пиксели и достать обратно;
3. сделать программу, которая прячет и читает письма прямо в браузере;
4. провести эксперимент: измерить вместимость тайника и заметность при разном числе битов;
5. проверить, переживёт ли письмо сжатие картинки, как при отправке «фото» в мессенджере;
6. сделать выводы и дать рекомендации.

1.3. Методы исследования

- изучение темы по интернет-источникам;
- создание программы-эмулятора (обработка изображений в браузере);

- компьютерный эксперимент с измерением вместимости и показателя PSNR;
- сравнение и анализ полученных данных.

2. Теоретическая часть

2.1. Из чего состоит картинка

Цифровая картинка — это сетка из множества точек, пикселей. Цвет каждого пикселя задаётся тремя числами от 0 до 255: сколько в нём красного, зелёного и синего цвета (система RGB). Смешивая их, можно получить любой оттенок. Например, картинка размером 300×200 содержит 60 000 пикселей, а значит 180 000 отдельных чисел-каналов.

2.2. Что такое младший бит

Каждое число от 0 до 255 компьютер хранит в двоичном виде — восемью битами (нулями и единицами). Например, число 200 записывается как **11001000**. Самая левая цифра отвечает за 128, а самая правая — всего за 1. Эту крайнюю правую цифру называют **младшим битом**.

Если поменять только младший бит, число 200 превратится в 201 или останется 200 — то есть цвет изменится на одну двести пятьдесят шестую. Такую крошечную разницу человеческий глаз не различает. Зато в этот младший бит можно записать один бит нашего письма. А в картинке таких битов — миллионы.

Одна буква в кодировке UTF-8 занимает 1–2 байта, то есть 8–16 битов. Значит, чтобы спрятать одну букву, нужно изменить младшие биты 8–16 каналов. В картинке 300×200 (180 000 каналов) поместится больше десяти тысяч русских букв — целое сочинение.

2.3. Как спрятать и достать письмо

Текст сначала превращают в длинную цепочку битов. В начало добавляют «заголовок» — число, которое говорит, сколько букв в письме. Затем биты письма по очереди записывают в младшие биты пикселей: первый бит — в младший бит красного канала первого пикселя, второй — в зелёный, и так далее.

Чтобы достать письмо, нужно знать правило: сколько младших битов использовано. Программа читает эти биты в том же порядке, собирает из них буквы и восстанавливает текст. Тот, кто правила не знает, видит просто картинку.

2.4. Чем измерять заметность: PSNR

Чтобы не спорить «видно или не видно» на глаз, заметность измеряют числом. Я использовала показатель **PSNR** — он показывает, насколько сильно изменённая картинка отличается от исходной, и измеряется в децибелах (дБ). Правило простое: **чем**

больше децибел, тем меньше заметна разница. Если PSNR выше 40 дБ, глаз разницы практически не видит; если ниже — подмену уже можно заметить.

3. Практическая часть

Для исследования сделана программа — страница, которая открывается в браузере и всё считает прямо на компьютере, без интернета (это важно: тайные картинки никуда не отправляются). В программе четыре части: «Спрятать письмо», «Прочитать письмо», «Эксперимент» и справка.

Как прячется письмо

Я выбираю картинку, пишу текст и указываю, сколько младших битов использовать. Программа прячет письмо и показывает рядом три картинки: исходную, картинку с письмом и «разницу», усиленную во много раз, чтобы изменения стало видно.

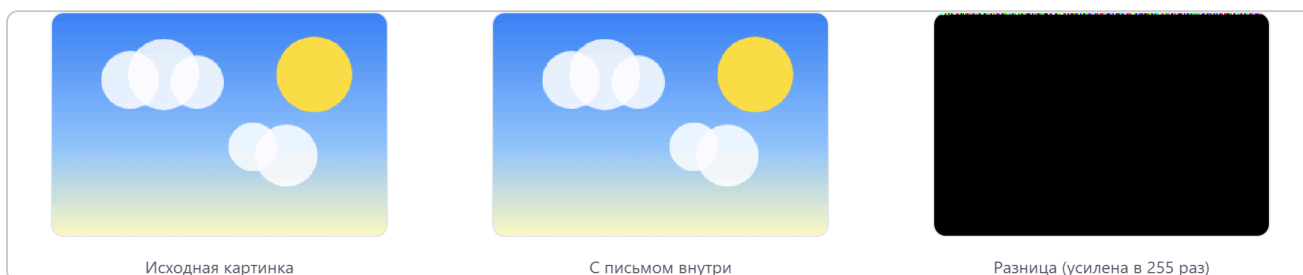


Рисунок 1. Слева — исходная картинка, в центре — та же картинка с письмом внутри (на глаз они одинаковые), справа — разница, усиленная в 255 раз. Видно, что изменилась только узкая полоска сверху, где и спрятано письмо.

Как проводился эксперимент

Я брала одну и ту же картинку и прятала в неё максимально длинное письмо, используя от 1 до 8 младших битов каждого цвета. Для каждого случая программа измеряла: сколько символов поместилось (вместимость), показатель PSNR и самый большой сдвиг яркости у пикселя. Все данные можно сохранить в файл.

Отдельно я проверяла, переживёт ли письмо сжатие картинки — такое же, какое происходит при отправке «фото» в мессенджере.

4. Результаты и их анализ

4.1. Вместимость и заметность

Результаты эксперимента собраны в таблице 1.

Битов на цвет	Вместимость, символов	Изменение картинки, PSNR, дБ	Макс. сдвиг яркости	Заметно глазом?
1	12 196	51.1	1	не видно
2	24 395	43.7	3	почти не видно
3	36 594	37.5	7	заметно
4	48 792	30.9	15	хорошо видно
5	60 991	25.2	31	хорошо видно
6	73 190	18.5	63	хорошо видно
7	85 389	12.4	127	хорошо видно
8	97 587	10.2	223	хорошо видно

Таблица 1. Вместимость тайника и заметность изменений при разном числе младших битов.

Из таблицы видно две противоположные вещи. С одной стороны, **вместимость растёт** ровно во столько раз, сколько битов мы берём: при 1 бите помещается около 12 000 символов, при 4 битах — почти 49 000, при 8 битах — около 98 000. С другой стороны, картинка при этом **всё сильнее портится**: показатель PSNR падает с 51 дБ (при 1 бите) до 10 дБ (при 8 битах), а максимальный сдвиг яркости вырастает с 1 до 223 из 255.

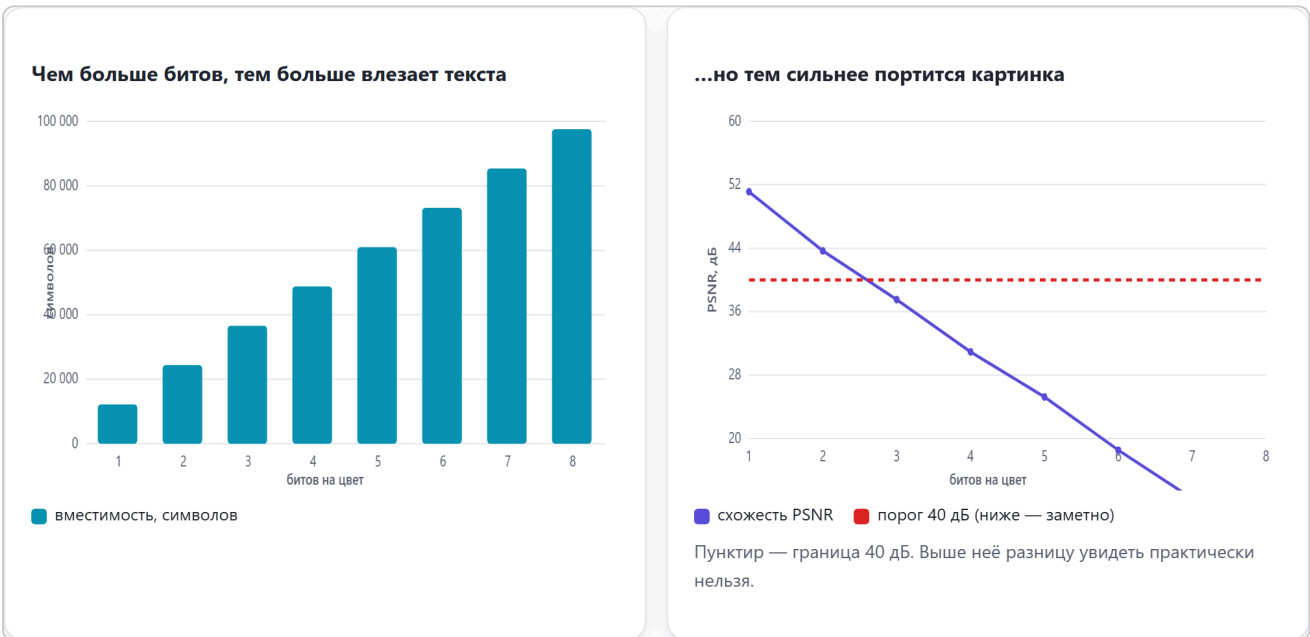


Рисунок 2. Слева — чем больше битов, тем больше влезает текста (линейный рост). Справа — тем сильнее портится картинка; пунктир показывает границу 40 дБ, ниже которой разница заметна.

Если сопоставить оба показателя, видно «золотую середину»: невидимыми (PSNR выше 40 дБ) остаются только варианты с 1 и 2 битами. Начиная с 3 бит разница уже заметна, а при 4 и больше — хорошо видна. При этом даже одного бита хватает, чтобы спрятать целое письмо. Значит, лучший выбор — 1–2 бита: и места достаточно, и подмена невидима.

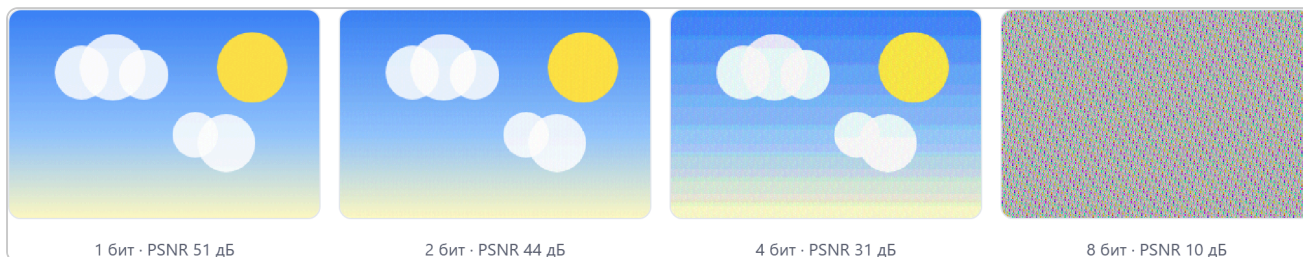


Рисунок 3. Одна и та же картинка при 1, 2, 4 и 8 использованных битах. При 1–2 битах она чистая, при 8 — покрыта «шумом».

4.2. Проверка «как в мессенджере»

Когда картинку отправляют **как фото**, мессенджер сжимает её, чтобы она весила меньше, и при этом слегка меняет пиксели. В опыте после такого сжатия письмо прочитать не удалось — вместо текста получился мусор. Причина в том, что сжатие огрубляет как раз мелкие, «незаметные» детали, а именно в них спрятано письмо.

Вывод из этой проверки: тайник нельзя пересылать «как фото». Если отправить ту же картинку **как файл** (без сжатия), пиксели сохранятся точно, и письмо дойдёт целым. Это объясняет, почему для скрытой передачи подходят только форматы без потерь — PNG, BMP.

5. Заключение

В ходе работы я:

- разобралась, из чего состоит картинка и что такое младший бит;
- описала способ прятать текст в младшие биты пикселей и доставать его обратно;
- сделала программу-эмулятор, которая прячет, читает письма и измеряет заметность;
- провела эксперимент с разным числом битов и проверила устойчивость к сжатию.

Гипотеза подтвердилась. В младшие биты картинки действительно можно спрятать целое письмо незаметно для глаза: при 1–2 битах показатель PSNR выше 40 дБ, и разницу не видно. И так же подтвердилась вторая часть гипотезы: чем больше битов, тем больше объём, но тем заметнее подмена — начиная с 4 бит картинка портится явно.

Я узнала и важное ограничение: стеганография прячет сам факт письма, но не шифрует его — кто знает правило, прочитает текст. Поэтому на практике письмо сначала шифруют, а уже потом прячут, и тогда тайна защищена дважды.

Что можно изучить дальше: как прятать письмо не подряд, а «рассыпая» по всей картинке; как объединить шифрование и стеганографию; можно ли прятать текст в звук или видео.

Список использованных источников

1. Стеганография — статья, Википедия. URL: <https://ru.wikipedia.org/wiki/Стеганография>
2. Цифровая стеганография. Метод наименее значащего бита (LSB) — обзорные материалы. URL: https://ru.wikipedia.org/wiki/Цифровая_стеганография
3. RGB (цветовая модель) — статья, Википедия. URL: <https://ru.wikipedia.org/wiki/RGB>
4. Пиковое отношение сигнала к шуму (PSNR) — статья, Википедия. URL: https://ru.wikipedia.org/wiki/Пиковое_отношение_сигнала_к_шуму
5. Двоичная система счисления — статья, Википедия. URL: https://ru.wikipedia.org/wiki/Двоичная_система_счисления