ВОРОНЕЖСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ

Мультиплексирование скрытых каналов при реализации стеганографического встраивания в цифровые изображения с использованием некриптографических хеш-функций

Дрюченко М.А.,
к.т.н, доц. каф ТО и ЗИ, ФКН

Воронеж 2023

Задачи

- Разработать алгоритм стеганографического скрытия данных в цифровые изображения, позволяющий встраивать сообщения, длина которых существенно превышает число модифицируемых элементов носителя.

- Предложить модификацию алгоритма, позволяющую реализовать мультиплексирование скрытых каналов при сохранении или увеличении пропускной способности алгоритма.

- Исследовать базовые показатели качества работы алгоритма и провести сравнение с современными алгоритмами адаптивной пространственной стеганографии.

Алгоритм встраивания данных

Шаг 1. Загрузка файла-сообщения M и контейнера I, разбиение контейнера на непересекающиеся блоки

$b_i \in I, \quad b_i \cap b_j = \varnothing, \quad \forall i \neq j, \quad i, j = \overline{1,(W \cdot H)/(w \cdot h)},$

$w \times h$ – размер обрабатываемых блоков,

$W \times H$ – размер контейнера.

Шаг 2. Выбор очередного блока $b_i \in I$, оценка его гладкости с целью определения оптимальной длины n скрываемого в блок слова.

Шкала гладкости

$$c_i = \begin{cases} 0, & d_i < 0{,}15, \quad n = 0; \\ 1, & 0{,}15 \le d_i < 0{,}3, \quad n = 4; \\ 2, & 0{,}3 \le d_i < 2, \quad n = 8; \\ 3, & 2 \le d_i < 7, \quad n = 12; \\ 4, & d_i \ge 7, \quad n = 16 \end{cases}$$

$$d_i = \frac{1}{3} \sum_{ch = R,G,B} D \quad grad \quad b_{i,ch}$$

а) *б)*

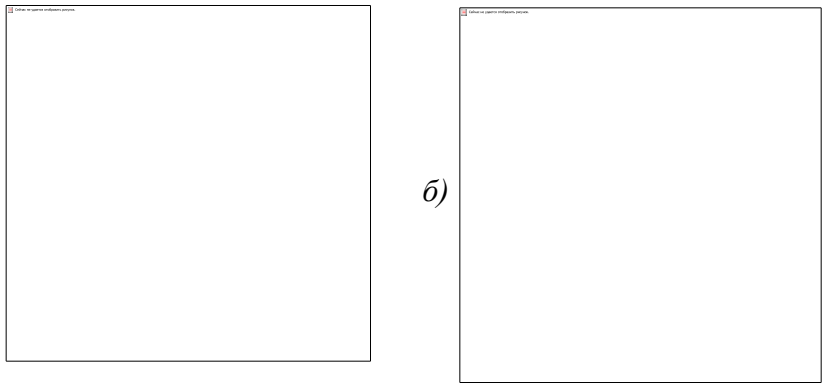

Рис.1. Исходное (а) и размеченное по уровням гладкости блоков (б) изображение «Lena»

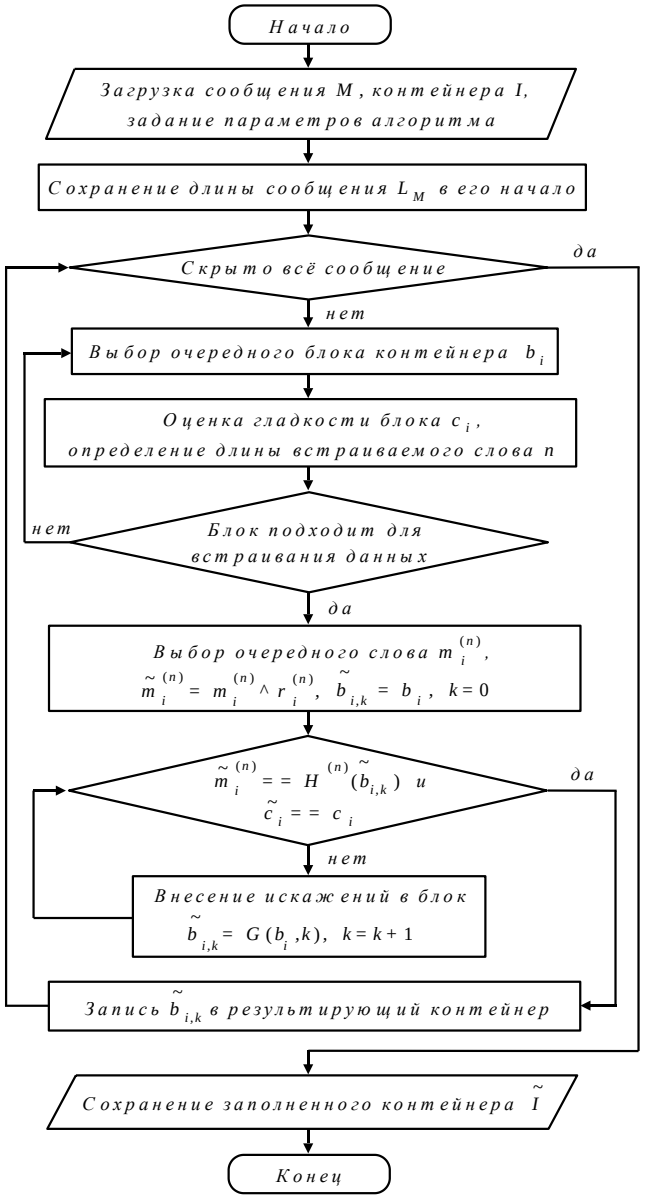

Рис.2

Алгоритм встраивания данных

Шаг 3. Выбор очередного n-битного слова $m_i^{(n)} \in M$ и его рандомизация $\tilde{m}_i^{(n)} = m_i^{(n)} \oplus r_i^{(n)}$, $r_i^{(n)}$ – выход ГПСЧП

Шаг 4. В цикле поиск подходящего варианта искажения блока $\tilde{b}_{i,k}$ такого, что

$$H^{(n)}\left(\tilde{b}_{i,k}\right) = \tilde{m}_i^{(n)}, \quad \left\| b_i - \tilde{b}_{i,k} \right\| \to \min, \quad c_i = \tilde{c}_i.$$

Для модификации элементов b_i применяется функция G

$$\tilde{b}_{i,k} = G\left(b_i, k\right) = \left(b_i(1) + \psi_{1,k}, b_i(2) + \psi_{2,k}, \cdots, b_i(L_b) + \psi_{L_b,k}\right).$$

Элементы вектора $\Psi_k = \left(\psi_{1,k}, \cdots \psi_{L_b,k}\right)^T$ могут принимать значения $0, \pm 1, \pm 2, \cdots \pm \lambda$, λ – максимальное значение, добавляемое к значению цвета пикселя (обычно $\lambda \le 3$), $L_b = 3 \cdot w \cdot h$ – число доступных для модификации элементов блока для полноцветных контейнеров.

В качестве H рассматривалась некриптографическая хеш-функция Murmur3.

С учетом возможных коллизий для H количество векторов Ψ_k должно в несколько раз превосходить размерности пространства решений 2^n.

Найденная подходящая модификация $\tilde{b}_{i,k}$ записывается в результирующий контейнер.

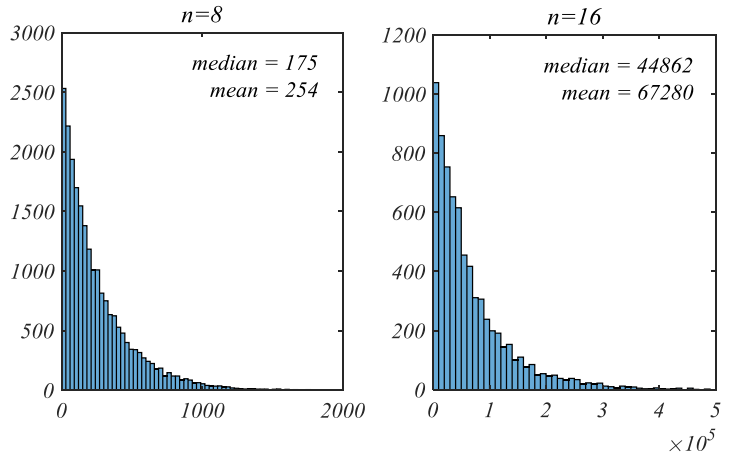

Рис.3. Гистограммы распределения числа итераций искажений блоков при встраивании слов разрядностью 8 и 16 бит

Шаг 5. Сохранение заполненного контейнера $\tilde{I}$.

Алгоритм извлечения данных

Шаг 1. Загрузка заполненного контейнера $\tilde{I}$.

Шаг 2. Выбор в соответствие с ГПСЧП на $\tilde{I}$ очередного блока $\tilde{b}_i$.

Шаг 3. Оценка гладкости блока $\tilde{b}_i$ и определение длины извлекаемого слова n. Если блок гладкий ($\tilde{c}_i = 0$), то он пропускается и осуществляется возврат к шагу 2.

Шаг 4. Извлечение и декодирование очередного слова

$$\tilde{m}_i^{(n)} = H^{(n)}\left(\tilde{b}_i\right), \quad m_i^{(n)} = \tilde{m}_i^{(n)} \oplus r_i^{(n)}.$$

Шаг 5. Сохранение извлеченного сообщения M.

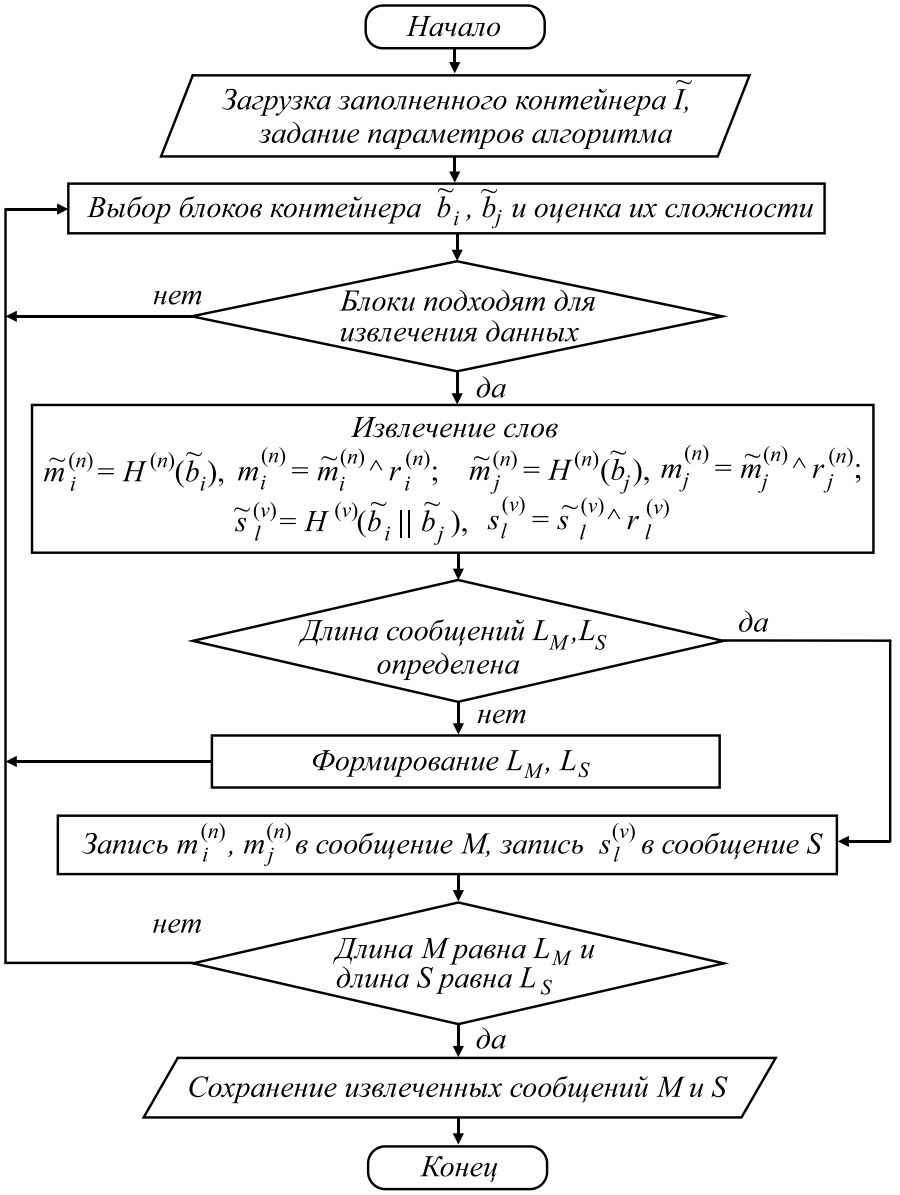

Рис.4

Алгоритм извлечения данных

Табл. 1. Среднее число искаженных вариантов блока, каждый из которых обеспечивает корректное извлечение n-битного слова

Размер и параметры искажения блока	$n = 4$	$n = 8$	$n = 12$	$n = 16$
3 3, $\lambda = 1$, $N_{corr} \le 3$	1503	88	7	0,6
3 3, $\lambda \le 2$, $N_{corr} \le 3$	8352	526	34	3
3 3, $\lambda = 1$, $N_{corr} \le 4$	39381	2575	167	14
4 4, $\lambda = 1$, $N_{corr} \le 3$	8852	536	36	2
4 4, $\lambda \le 2$, $N_{corr} \le 3$	47405	3072	192	11
4 4, $\lambda = 1$, $N_{corr} \le 4$	480102	30896	1867	146
5 5, $\lambda = 1$, $N_{corr} \le 3$	34446	2 068	121	7
5 5, $\lambda \le 2$, $N_{corr} \le 3$	184329	11594	719	42
5 5, $\lambda = 1$, $N_{corr} \le 4$	3166088	197318	12328	748

Мультиплексирование скрытых каналов

- Классические варианты мультиплексирования скрытых каналов, как правило, предполагают выделение в контейнере непересекающихся подмножеств модифицируемых элементов с последующим встраиванием в них информации, предназначенной для различных пользователей.

- Предложенный алгоритм использует **одно и то же подмножество элементов контейнера для встраивания нескольких независимых сообщений**.

- Рассмотрим вариант создания двух скрытых каналов C_M и C_S, используемых для передачи независимых сообщений $M = m_1^{(n)} \,||\, m_2^{(n)} \,||\cdots m_{N_M}^{(n)}$ и $S = s_1^{(v)} \,||\, s_2^{(v)} \,||\cdots s_{N_S}^{(v)}$.

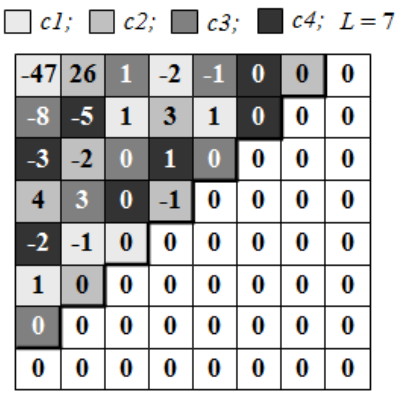

Рис.5. Разметка блока контейнера при классическом варианте мультиплексирования скрытых каналов

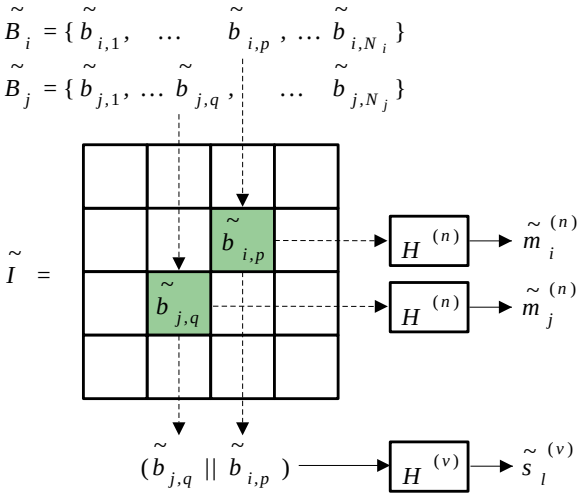

Рис.6. Выбор искаженных блоков, обеспечивающих извлечение данных из первого и второго скрытых каналов

Обобщенная схема алгоритма встраивания данных при создании двух скрытых каналов

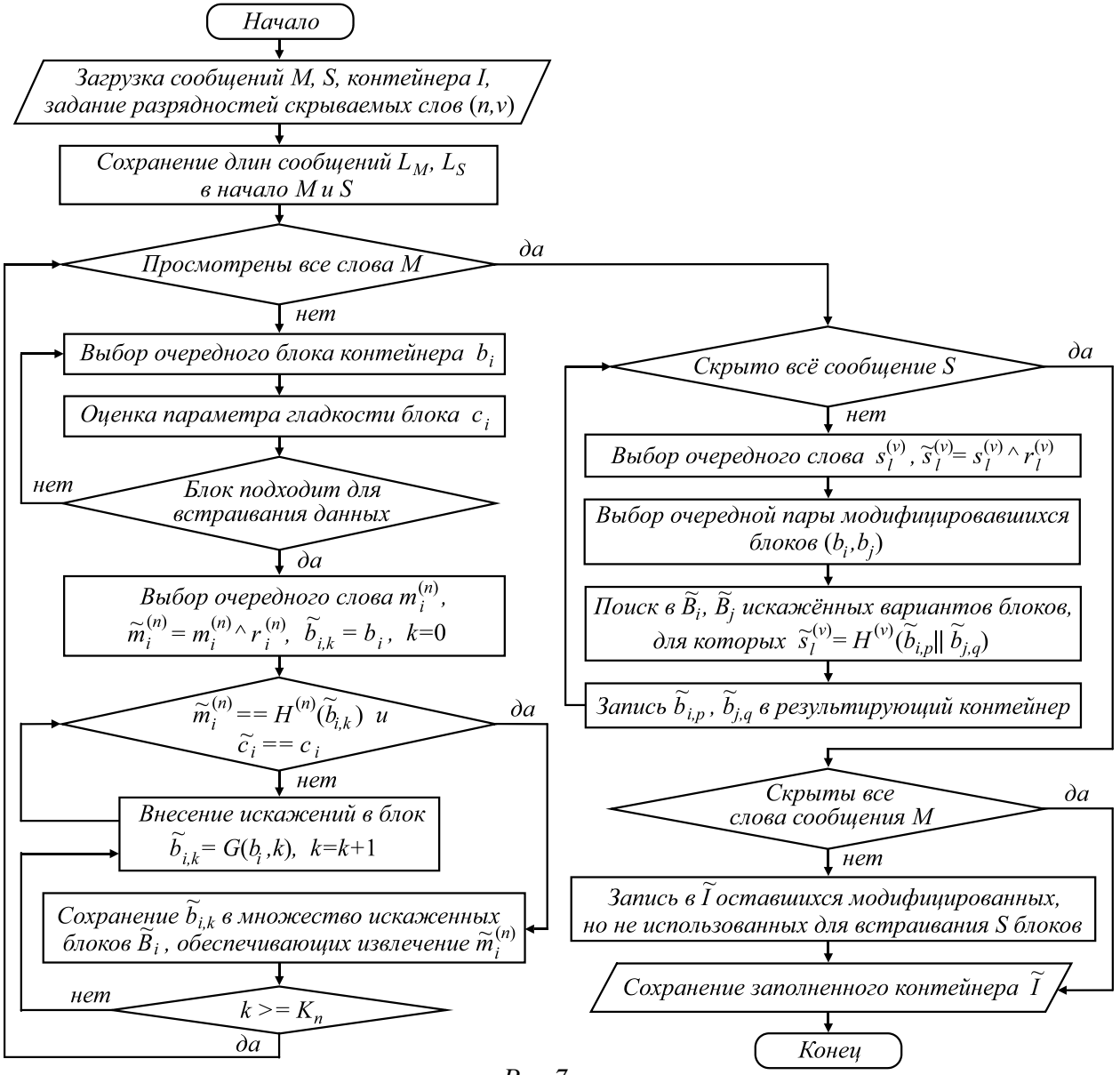

Рис.7

Мультиплексирование скрытых каналов

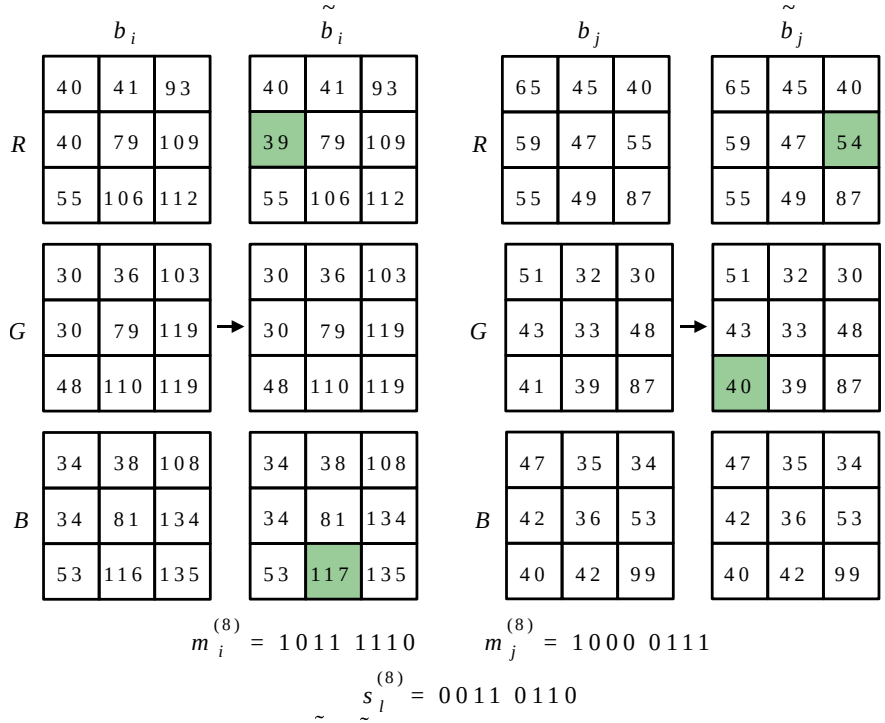

Рис.8. Пример пары искаженных блоков $\tilde{b}_i$, $\tilde{b}_j$, обеспечивающих извлечение тройки слов $m_i^{(8)}$, $m_j^{(8)}$, $s_l^{(8)}$

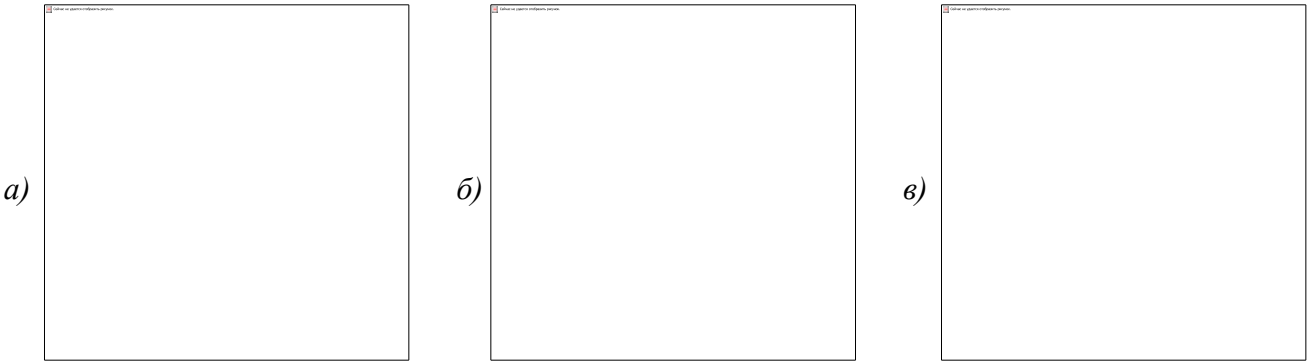

Рис.9. Исходный контейнер (а), маркированные контейнеры, созданные без мультиплексирования (б) и в режиме мультиплексирования скрытых каналов (в) (n=8, v=8)

Результаты экспериментальных исследований

- Оценка зависимостей **среднего числа модифицируемых элементов и уровня их искажений** в блоках различного размера **от длин встраиваемых в данные блоки слов**.
- Тестирование на выборке из 50 RGB-изображений размером 512 512 из базы PPG-LIRMM-COLOR.
- Скрываемые сообщения – псевдослучайные последовательности чисел требуемой длины, формируемые алгоритмом MT19937.

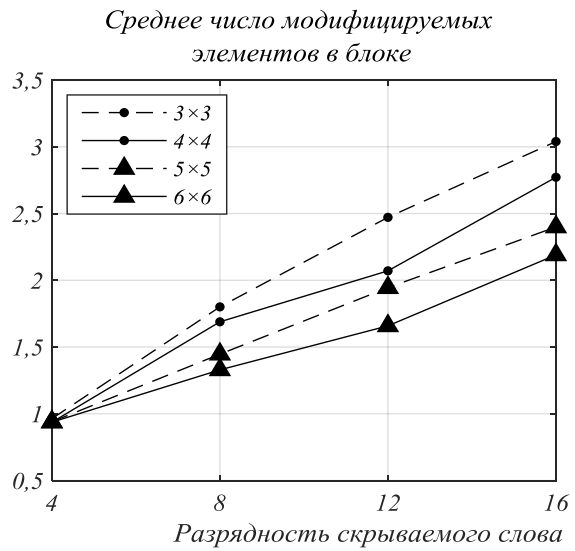

Рис.10. Зависимость среднего числа модифицируемых элементов блока от разрядности скрываемых слов

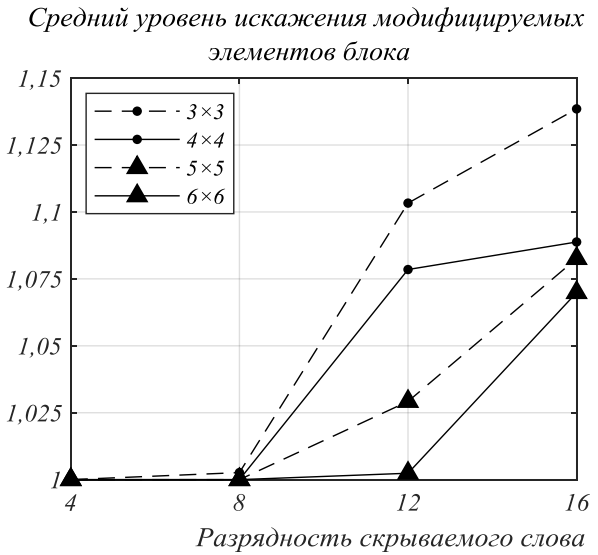

Рис.11. Зависимость среднего уровня искажения модифицируемых элементов блока от разрядности скрываемых слов

Результаты экспериментальных исследований

- Оценка **степени искажения заполненных контейнеров** с использованием метрик пикового отношения сигнала к шуму, индекса структурного сходства и максимальной абсолютной ошибки.

- Сравнение с современными алгоритмами адаптивной пространственной стеганографии WOW и S-UNIWARD.

- Параметры алгоритма, использующего некриптографические хеш-функции для извлечения скрытых данных (AHX): размер блока 4 4 пикс, $\lambda \leq 2$, $N_{corr} \leq 3$.

Табл. 2. Результаты сравнения алгоритмов стегоскрытия по показателям искажения заполненных контейнеров

Наименование алгоритма	PSNR	SSIM	MAE
	α = 0,1		
WOW	66,433	0,9999	1
S-UNIWARD	66,778	0,9999	1
AHX	69,846	0,9999	1
	α = 0,5		
WOW	59,319	0,9996	1
S-UNIWARD	59,032	0,9994	1
AHX	62,988	0,9999	1
	α = 1		
WOW	55,348	0,9986	1
S-UNIWARD	54,796	0,9973	1
AHX	59,959	0,9997	2

Результаты экспериментальных исследований

- Оценка **времени создания маркированных контейнеров** проводилась для распараллеленного варианта алгоритма скрытия (многопоточная реализация на языке C++, компилятор MinGW gcc-7.3.0, CPU Intel Core i5-10400 2.90ГГц, 16Гб ОЗУ).

Табл. 3. Среднее время (в секундах) создания заполненного контейнера размером 512 512 пикселей при 100% его заполнении скрытой информации

Разрядность встраиваемых слов	Тип функции *H*	
	Murmur3	CRC-8, CRC-16-CCITT
n = 4	0,018	0,019
n = 8	0,041	0,075
n = 12	0,375	1,896
n = 16	3,386	27,303
Определяется автоматически	0,759	5,732

Табл. 4. Среднее время (в секундах) создания заполненного контейнера размером 512 512 пикселей при 100% его заполнении скрытой информации в режиме мультиплексирования скрытых каналов

Разрядность встраиваемых слов	Размер блоков 4 4, $\lambda \le 2$, $N_{corr} \le 4$
n = 4, *v* = 4	0,637
n = 4, *v* = 8	0,828
n = 4, *v* = 12	1,470
n = 4, *v* = 16	4,055
n = 8, *v* = 4	0,607
n = 8, *v* = 8	1,785
n = 8, *v* = 12	7,621
n = 8, *v* = 16	28,869

Выводы

- Описанные принципы встраивания/извлечения данных являются **универсальными** и могут применяться как в пространственном, так и в частотном представлении контейнеров различных форматов.

- Предложенный алгоритм имеет **настраиваемую ПС**, зависящую от размеров искажаемых блоков, разрядности встраиваемых слов, характера содержимого самого контейнера, определяющего процент пропускаемых блоков или блоков, для которых длина встраиваемых слов автоматически уменьшается.

- Алгоритм способен обеспечить **малое соотношение числа фактически модифицируемых бит контейнера к числу встраиваемых бит** сообщения порядка 0,06 (в то время как для большинства классических стегоалгоритмов данный показатель находится на уровне от 0,5 и выше).

- **Малый уровень искажений** маркированных контейнеров (при объеме полезной нагрузки $\alpha \leq 0,5$ фиксируются лучшие значения PRNR и SSIM, чем для алгоритмов WOW и S-UNIWARD).

- Предложен вариант мультиплексирования скрытых каналов, отличительной особенностью которого является использование **одного и того же подмножества элементов, подвергающихся одинаковым искажениям при встраивании различных сообщений**.