Дмитрий А. Мельников[1], Антон А. Абрамов[2], Пётр А. Кейер[3]
[1, 3]Федеральный исследовательский центр «Информатика и управление» РАН,
Вавилова ул., 44, корп. 2, г. Москва, 119333, Россия
[2]Национальный исследовательский ядерный университет «МИФИ»,
Каширское ш., 31, г. Москва, 115409, Россия
[1]e-mail: mda-17@yandex.ru, https://orcid.org/0000-0003-4515-9712
[2]e-mail: mvpcvp@gmail.com, https://orcid.org/0000-0002-4088-6606
[3]e-mail: pkeyer@ipiran.ru, http://orcid.org/0000-0001-9124-1528

РЕАЛИЗАЦИЯ СПОСОБА ЗАЩИТЫ НЕПОДВИЖНЫХ ИЗОБРАЖЕНИЙ В «КВАНТОВОМ МИРЕ»

DOI: http://dx.doi.org/10.26583/bit.2019.2.02

Аннотация. В статье рассматривается способ защиты статических (неподвижных) изображений, основанный на перестановке точек изображения (ТИ), вписанных в ребра правильной шестиугольной пирамиды. В отличие от классических методов генерации перестановок, в которых элементарная операция перестановки – это перестановка двух элементов, в предлагаемом способе перестановок элементарная операция параметрическая, т.е. зависит от входных параметров алгоритма и может затрагивать перестановку более двух ТИ. В предыдущих работах в качестве такой операции был выбран поворот шестигранника. В данной статье рассмотрено несколько возможных вариаций этой элементарной перестановки, и на основе анализа (критерия) количества реализуемых перестановок исходной последовательности ТИ (ПТИ) был выбран наиболее подходящий вид элементарной перестановки. Предложен способ защиты статических (неподвижных) изображений с использованием описанной в статье перестановки. Проведён анализ стойкости такого способа защиты от атак типа «полный перебор» и предложен вариант, использующий избыточность графического представления данных. Прикладной анализ указанного способа показал его способность противостоять современным криптоаналитическим атакам в условиях применения квантовых вычислительных средств.

Ключевые слова: *квантовый криптоанализ, пиксель, неподвижное изображение, шестигранник, контурный шестигранник, правильная пирамида, перестановка.*
Для цитирования: МЕЛЬНИКОВ, Дмитрий А.; АБРАМОВ, Антон А.; КЕЙЕР, Пётр А.. РЕАЛИЗАЦИЯ СПОСОБА ЗАЩИТЫ НЕПОДВИЖНЫХ ИЗОБРАЖЕНИЙ В «КВАНТОВОМ МИРЕ». Безопасность информационных технологий, [S.l.], v. 26, n. 2, p. 21-43, 2019. ISSN 2074-7136. Доступно на: <https://bit.mephi.ru/index.php/bit/article/view/1197>. Дата доступа: 03 june 2019. doi:http://dx.doi.org/10.26583/bit.2019.2.02.

Dmitry A. Melnikov[1], Anton A. Abramov[2], Peter A. Keyer[3]
[1 ,3]Federal Research Center «Computer Science and Control» of Russian Academy of Sciences,
Vavilov str., 44/2, Moscow, 119333, Russia
[2]National Research Nuclear University MEPhI (Moscow Engineering Physics Institute),
Kashirskoe shosse, 31, Moscow, 115409, Russia
[1]e-mail: mda-17@yandex.ru, https://orcid.org/0000-0003-4515-9712
[2]e-mail: mvpcvp@gmail.com, https://orcid.org/0000-0002-4088-6606
[3]e-mail: pkeyer@ipiran.ru, http://orcid.org/0000-0001-9124-1528

Implementation of A Static Images Protection Method in “Quantum World”

DOI: http://dx.doi.org/10.26583/bit.2019.2.02

Abstract. This study deals with the mechanism of protection of static (fixed) images, based on the permutation of image points (IP), inscribed in the edges of a regular hexagonal pyramid. In contrast to the classical methods of generating permutations, in which the elementary permutation operation is a permutation of two elements, in the proposed method the elementary operation is parametric, i.e. depends on the input parameters of the algorithm, and can affect the permutation of more than two IP. In the previous

paper the rotation of hexagon was chosen as elementary permutation. In the present paper several possible modifications of the elementary permutation are considered, and on the basis of the analysis (criterion) of the number of realizable permutations of the initial sequence of IP (SIP) the most acceptable type of modification for the elementary permutation was chosen. The mechanism of protection of static (fixed) images using the permutation is proposed. The analysis of the persistence of this protection mechanism from attacks such as «brute force» is given, and an option that uses the redundancy of the graphical presentation is proposed. Applied analysis of given mechanism has demonstrated its ability to withstand modern cryptanalytic attacks in the application of quantum computing.
Keywords: quantum cryptanalysis, pixel, static image, hexagon, contour hexagon, regular pyramid, permutation.
For citation. A. MELNIKOV, Dmitry A.; ABRAMOV, Anton A.; KEYER, Peter A.. Implementation of A Static Images Protection Method in "Quantum World". IT Security (Russia), [S.l.], v. 26, n. 2, p. 21-43, 2019. ISSN 2074-7136. Available at: <https://bit.mephi.ru/index.php/bit/article/view/1197>. Date accessed: 03 june 2019. doi:http://dx.doi.org/10.26583/bit.2019.2.02.

ВВЕДЕНИЕ

Во многих работах, например, [1–5], анализируется проблема стойкости различных криптографических систем в условиях «квантового мира», т.е. широкого распространения вычислительных устройств, которые используют явления квантовой механики для передачи и обработки данных. В частности, в [5] сделан вывод о том, что ассиметричные криптографические системы в нынешнем виде прекратят своё существование по причине криптографической уязвимости, а симметричные криптографические системы потребуют значительного увеличения размеров (длин) секретных ключей (табл. 1 [6]).

Таблица 1. Влияние квантовых вычислений на общие криптоалгоритмы

Криптоалгоритм	Вид ключа	Назначение	Результат широкомасштабного использования квантовых компьютеров
AES	Симметричный ключ	Шифрование	Необходимы более длинные ключи
SHA-2, SHA-3	Не определён	Хэш-функции	Необходимы более длинные выходные последовательности
RSA	Асимметричный ключ	Электронная подпись, формирование ключей	В дальнейшем безопасность не обеспечит
ECDSA, ECDH (криптография на основе эллиптических кривых)	Асимметричный ключ	Электронная подпись, обмен ключами	В дальнейшем безопасность не обеспечит
DSA (криптография на основе конечных полей)	Асимметричный ключ	Электронная подпись, обмен ключами	В дальнейшем безопасность не обеспечит

В настоящей статье представлен усовершенствованный способ защиты статических (неподвижных) изображений, впервые предложенный в 2004 году [7], который обеспечит защищённость данных (изображений) в условиях применения квантового криптоанализа. В [7–8] изложены основные принципы указанного способа, основанного на процедуре скремблирования исходной последовательности точек изображения (ПТИ) и её геометрической перестановки.

Любое статическое изображение представляет собой последовательность точек (пиксел), каждая из которых имеет свой цвет, диапазон цветовой гаммы зависит от конкретного способа кодирования цвета. Ввиду того, что любое изображение обладает большой информационной избыточностью, оно подвергается процедуре сжатия для хранения в памяти или передается по каналам связи.

Один из надежных способов защиты статических изображений заключается в использовании процедуры зашифрования, реализуемой следующим образом. Последовательность точек изображения, представленная в двоичной форме, суммируется по *mod* 2 с выходной последовательностью генератора гаммы. При этом необходимо отметить важную особенность результирующей последовательности: она полностью сохраняет порядок следования элементов в исходной ПТИ.

В [7–8] показано, что для создания надёжного алгоритма защиты статических изображений необходимо решить две задачи:

1) исключить какие-либо корреляционные (статистические) связи между соседними и близлежащими точками изображения;

2) исключить какие-либо рекуррентные свойства поточного (блочного) шифра.

Для решения первой задачи можно использовать линейный или нелинейный скремблер (аддитивный или самосинхронизирующийся) на основе конечного автомата, который обладает хорошими статистическими свойствами. Процедура скремблирования устраняет корреляционные связи между точками изображения. Ввиду того, что скремблер представляет собой конечный автомат, проскремблированная исходная ПТИ будет обладать рекуррентными свойствами, которые определяются генераторным полиномом автомата. Измененная в ходе процедуры скремблирования ПТИ будет сохранять естественный порядок следования ТИ, что снижает стойкость такого способа защиты статических изображений. Поэтому, необходимо исключить указанные рекуррентные свойства измененной ПТИ, а также естественный порядок следования в ней ТИ. Это можно сделать с помощью перестановки ТИ проскремблированной исходной ПТИ.

1. Способы геометрической перестановки

1.1 Геометрическая перестановка на основе шестигранников с квадратной решёткой

В работе [7] предполагается, что пиксели ПТИ представлены в форме квадратной решётки с нанесённой на неё треугольной решёткой, внутри каждого треугольника оказывается пара пикселей (рис. 1).

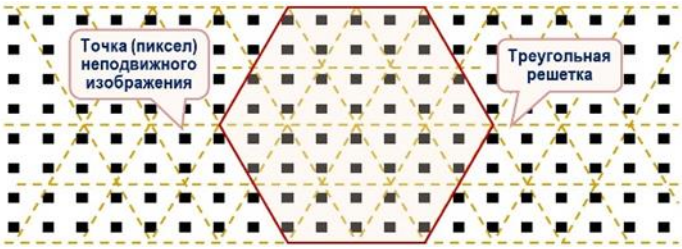

Рис. 1. ПТИ с нанесённой на них треугольной решеткой
(Fig. 1. Image Points (pixels) with triangle grid)

Такое размещение пикселей внутри шестигранника не удобно, т.к. идея перестановки ТИ заключается в повороте шестигранника на некоторый угол, после чего ТИ

считываются построчно в новом порядке и получается результирующая ПТИ (рис. 2), а при использовании треугольной решетки разные грани шестигранника содержат разное количество пикселей, что делает не возможным осуществить поворот.

Поворот шестигранника, не нарушающий структуру ПТИ, возможен только на допустимые углы, причём кратные 60°. Возможные углы поворота шестигранника:

$$\Phi = \{0^{\circ};\ 60^{\circ}; 120^{\circ}; 180^{\circ}; 240^{\circ}; 300^{\circ}\}. \quad (1)$$

Грани шестигранника состоят из ТИ с фиксированными позициями, поэтому поворот на произвольный угол просто невозможен. Причина этого в том, что определение новых позиций ТИ, будет крайне затруднительным. В частности, это показано на рис. 2 при использовании квадратной решётки, когда подряд идущие ТИ, отмеченные на рисунке как строка ТИ, после поворота шестигранника не попадают в ячейки исходной треугольной решетки, т.е. не могут быть однозначно в ней размещены без дополнительных условий.

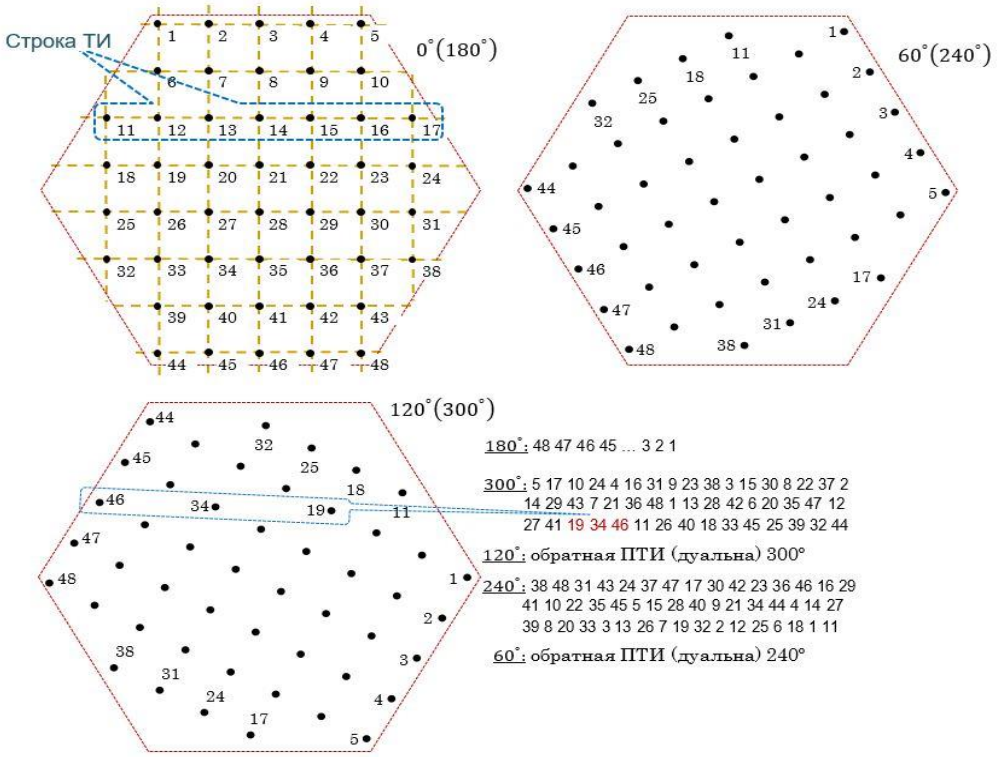

Рис. 2. Результирующие ПТИ после поворотов шестигранника с ТИ, встроенными в квадратную решётку
(Fig. 2. The resulting sequence of image points after different rotations of the hexagon in the square grid)

1.2 Геометрическая перестановка на основе шестигранников с треугольной решёткой

В [9] предложен способ представления элементов ПТИ в виде шестигранника на основе треугольной решётки (рис. 3). В таком представлении наименьшим элементарным шестигранником является шестигранник из 7 ТИ.

Расположение ТИ в узлах треугольной решётки обеспечивает очень важное геометрическое свойство – лучевую (радиальную) симметрию, которая при повороте шестигранника не нарушается. Введем обозначение H_n для шестигранника, где H – показывает, что это шестигранник, а индекс n показывает количество ТИ в стороне

шестигранника. Обозначим строку шестигранника, горизонтальную подпоследовательность ТИ внутри шестигранника, как l_i^n (i – порядковый номер строки), где l – обозначает, что это строка, n – количество элементов в стороне шестигранника, i – порядковый номер строки внутри шестигранника, начиная с верхней. Пример ПТИ при различных поворотах шестигранника H_4 со стороной $n = 4$ ТИ представлен на рис. 4. Количество строк в шестиграннике H_n составляет:

$$L_n = 2n - 1, \tag{2}$$

где n – количество элементов в стороне шестигранника.

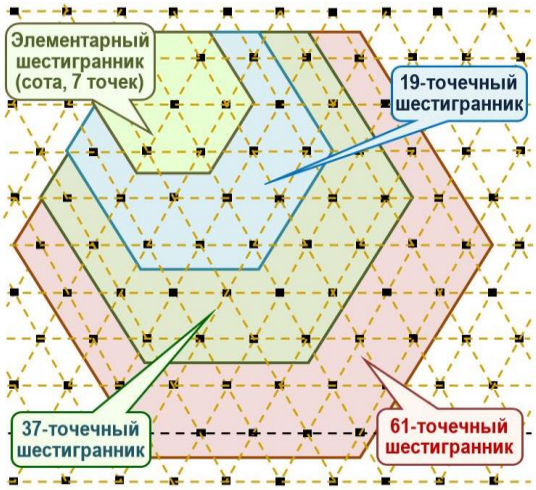

Рис. 3. Элементарные 7-, 19-, 37- и 61-точечные шестигранники (пиксели структурированы в форме треугольной решётки)
(Fig. 3. The elementary 7-, 19-, 37- and 61-points hexagon)

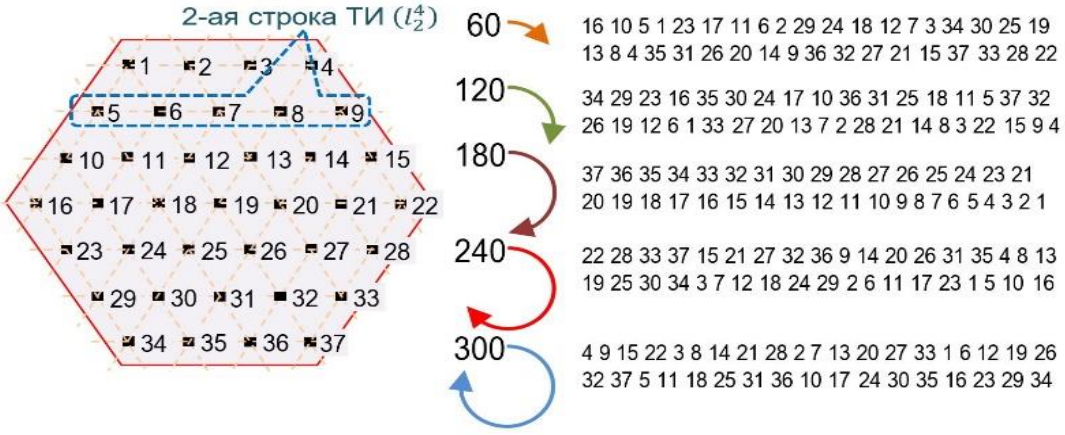

Рис. 4. Пример шестигранника H_4 и результирующие ПТИ после ротации шестигранника на допустимые углы
(Fig. 4. Example of the hexagon $H_4$and the resulting Sequence of Image Points after rotation on different angles)

Введем S_n – число ТИ, входящих в H_n, которое определяется по следующей формуле:

$$S_n = 7 + 6 \times \sum_{i=3}^{n}(i-1)\,, \text{ где } n \geq 3\,. \quad (3)$$

Например, для шестигранника H_7 общее число ТИ, входящих в этот шестигранник, будет равно:

$$S_7 = 7 + 6 \times [(3-1) + (4-1) + (5-1) + (6-1) + (7-1)] = 7 + 6 \times 20 = 127\,.$$

Для нахождения числа ТИ, входящих в H_n, выведем формулу в общем виде. Рассмотрим H_7, который можно представить как последовательность чисел: 7 8 9 10 11 12 13 12 11 10 9 8 7. Каждое из чисел указывает на число ТИ в каждой строке (число строк равно L_n) указанного шестигранника. Тогда формула вычисления количества ТИ в шестиграннике будет иметь вид:

$$S_n = (2n-1) + \sum_{i=0}^{n-2} 2(n+i), \quad (4)$$

где $n \geq 2$. Общая формула для определения числа ТИ, вписанных в шестигранник H_n:

$$S_n = 3n^2 - 3n + 1\,. \quad (5)$$

Множество ТИ в шестиграннике H_n представляет объединение строк l_i^n:

$$H_n = \bigcup_{i=1}^{L_n} l_i^n = l_1^n \cup \ldots \cup l_{L_n}^n\,. \quad (6)$$

Из (6) видно, что процедура объединения строк l_i^n в H_n не определяет порядок объединения самих строк. В этой связи введём оператор «сцепления/присоединения» (*concatenation*):

$$\Big\|_{i=1}^{\overset{L_n}{\Leftarrow}} l_i^n \quad ; \quad \Big\|_{i=1}^{\overset{L_n}{\Rightarrow}} l_i^n \quad ; \quad \Big\|_{i=1}^{\overset{L_n}{\Leftrightarrow}} l_i^n \quad , \quad (7)$$

где стрелка (влево) «⇐» – последовательное сцепление (присоединение) строк справа; стрелка (вправо) «⇒» – последовательное сцепление (присоединение) строк слева; стрелка (влево-вправо) «⇔» – произвольный порядок сцепления последовательных соседних строк шестигранника.

Теперь можно записать множество H_n как упорядоченную ПТИ:

$$H_n = \begin{cases} \Big\|_{i=1}^{\overset{L_n}{\Leftarrow}} l_i^n = l_1^n \parallel l_2^n \parallel \cdots \parallel l_{L_n-1}^n \parallel l_{L_n}^n \\ \Big\|_{i=1}^{\overset{L_n}{\Rightarrow}} l_i^n = l_{L_n}^n \parallel l_{L_n-1}^n \parallel \cdots \parallel l_2^n \parallel l_1^n \\ \Big\|_{i=1}^{\overset{L_n}{\Leftrightarrow}} l_i^n = l_1^n \parallel l_{L_n-1}^n \parallel \cdots \parallel l_1^n \parallel l_{L_n}^n \end{cases} \quad (8)$$

1.3 Контурные шестигранники

Введем *контурный шестигранник* h_n размерности n, состоящий из ТИ, которые содержат грани шестигранника H_n. Например, шестигранник H_4 на рис. 5 состоит из трёх контурных шестигранников h_4, h_3 и h_2 и центрального элемента O_{19} (вырожденного шестигранника с одной точкой $h_1 = H_1$). Количество ТИ в h_n:

$$s_n = 6(n-1)\,. \quad (9)$$

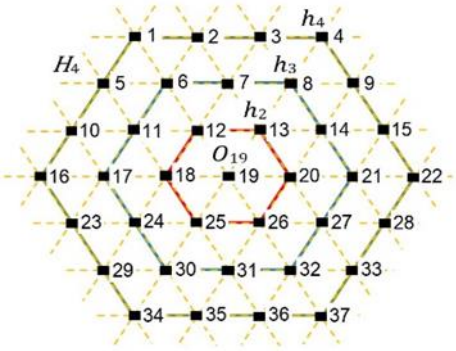

Рис. 5. Примеры контурных шестигранников h_4, h_3 и h_2
(Fig. 5. Example of the contour hexagons h_4, h_3 and h_2)

Минимальный контурный шестигранник – h_2. Шестигранник H_n можно представить как объединение контурных шестигранников:

$$H_n = \bigcup_{i=1}^{n} h_i = h_n \cup \ldots \cup h_1 , \tag{10}$$

где h_1 – центральная точка в H_n (вырожденный шестигранник).

Результирующая ПТИ шестигранника H_4 на рис. 5 будет иметь вид

$$\bar{S}_4 = \begin{cases} \overset{L_n}{\underset{i=1}{\Leftarrow}} \; l_i^4 = l_1^4 \parallel l_2^4 \parallel \cdots \parallel l_{L_4-1}^4 \parallel l_{L_4}^4 \\ \overset{L_n}{\underset{i=1}{\Rightarrow}} \; l_i^4 = l_{L_4}^4 \parallel l_{L_4-1}^4 \parallel \cdots \parallel l_2^4 \parallel l_1^4 , \\ \overset{L_n}{\underset{i=1}{\Leftrightarrow}} \; l_i^4 = l_5^4 \parallel l_{L_4-1}^4 \parallel \cdots \parallel l_1^4 \parallel l_3^4 \end{cases} \tag{11}$$

В соответствии с (11) проанализируем поворот контурного шестигранника h_3 на 60°, представленного на рис. 6. Записав ТИ в строку, получаем последовательность: 1 2 3 4 5 6 7 8 9 10 11 12, после поворота имеем последовательность 11 12 1 2 3 4 5 6 7 8 9 10 ($h_3^{60^\circ}$). Это соответствует циклическому сдвигу $(n-1)$ ТИ, где n – число ТИ в грани h_n. Таким образом, поворот шестигранника H_n представляет сдвиг на $(m-1)$ ТИ каждого из $(n-1)$ контурных шестигранников размерности m, где $m = \overline{2,n}$, m – целое число. Перестановка (поворот) шестигранника H_n может быть представлена как композиция перестановок (поворотов) контурных шестигранников, входящих в состав шестигранника H_n.

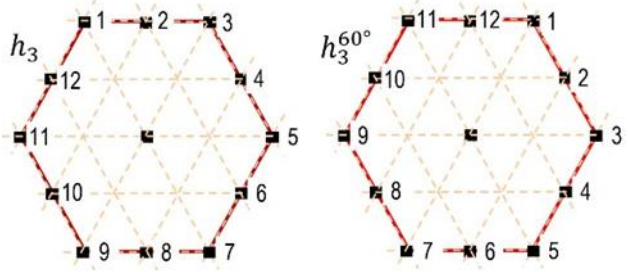

Рис. 6. Поворот контурного шестигранника h_3
(Fig. 6. Rotation of the contour hexagon h_3)

Декомпозиция любого шестигранника H_n на множество контурных шестигранников (10) позволяет увеличить количество вариантов перестановки элементов исходного шестигранника. Каждый из контурных шестигранников может быть повёрнут на угол φ:

$$H_n^{\Phi} = \bigcup_{i=1}^{n} h_i^{\varphi} \ , \tag{12}$$

при условии $\forall\varphi \in \Phi$ (1). На рис. 7 представлен пример разнонаправленной контурной ротации шестигранника H_4:

$$H_4^{\Phi} = \bigcup_{i=1}^{4} h_i^{\varphi} = h_4^{60^\circ} \cup h_3^{120^\circ} \cup h_2^{180^\circ} \cup h_1^{0^\circ} \ , \tag{13}$$

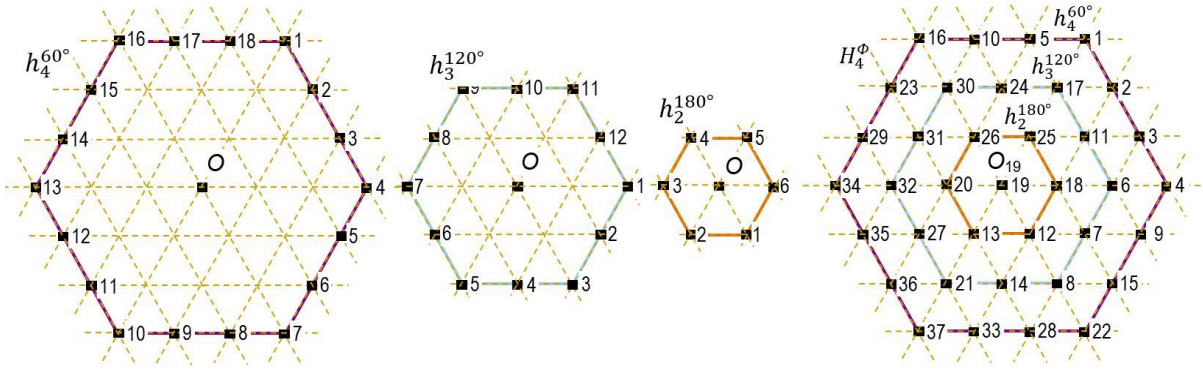

Рис. 7. Примеры контурных шестигранников h_4, h_3 и h_2, которые имеют разные углы поворота (60°, 120°, 180°; исходный H_4 представлен на рис. 5)
(Fig. 7. Example of the hexagons h_4, h_3 and h_2, with different angles of rotation (60°, 120°, 180°; original hexagon H_4 present on the fig. 5))

Итоговая ПТИ H_4, в которой сцепление строк с первой строкой будет осуществляться с правой стороны (показана на рис. 7):

$$H_4 = \overleftarrow{\bigparallel_{i=1}^{L_n}} \ l_i^4 \ = l_1^4 \parallel l_2^4 \parallel \cdots \parallel l_{L_4-1}^4 \parallel l_{L_4}^4. \tag{14}$$

Все контурные шестигранники h_4, h_3 и h_2, входящие в состав шестигранника H_4 (кроме вырожденного), имеют различные углы поворота: $h_4^{60^\circ}$, $h_3^{120^\circ}$ и $h_2^{180^\circ}$. Углы поворота каждого контурного шестигранника h_n могут выбираться произвольно из Φ (1).). В статье [9] стр. 22 на рис. 8 представлен шестигранник H_8, включающий семь контурных шестигранников и один вырожденный:

$$H_8^{\Phi} = \bigcup_{i=1}^{8} h_i^{\Phi} = h_8^{240^\circ} \cup h_7^{120^\circ} \cup h_6^{60^\circ} \cup h_5^{180^\circ} \cup h_4^{300^\circ} \cup h_3^{240^\circ} \cup h_2^{120^\circ} \cup h_1^{0^\circ} \ , \tag{15}$$

а также итоговая ПТИ H_8, в которой сцепление с первой строкой будет осуществляться с правой стороны (показана на рис. 8):

$$H_8 = \overleftarrow{\bigparallel_{i=1}^{L_n}} \ l_i^8 \ = l_1^8 \parallel l_2^8 \parallel \cdots \parallel l_{L_8-1}^8 \parallel l_{L_8}^8 \ . \tag{16}$$

1.4 Пирамидальная конструкция

С целью увеличения количества реализуемых перестановок ТИ усложним структуру элементарной перестановки, вместо шестигранника со структурой из п. 1.3 используем несколько шестигранников с убывающим значением стороны $(i = \overline{n,1})$, а сами шестигранники впишем в правильную шестиугольную пирамиду $\mathcal{P}_n$ (рис. 8). Очевидно, что количество шестигранников m будет равно длине стороны шестигранника H_n, лежащего в основании пирамиды, т.е. $m = n$. Тогда пирамидальное множество ТИ $\mathcal{P}_n$, состоящее из n шестигранников, будет:

$$\mathcal{P}_n = \bigcup_{i=1}^{n} H_i \ . \tag{17}$$

Например, на рис. 8 показано множество ТИ $\mathcal{P}_5$ в пирамиде из пяти шестигранников со сторонами $n = \overline{5,1}$:

$$\mathcal{P}_5 = \ H_5 \cup H_4 \cup H_3 \cup H_2 \cup H_1 \ , \tag{18}$$

где H_1 – это точка (вырожденный шестигранник).

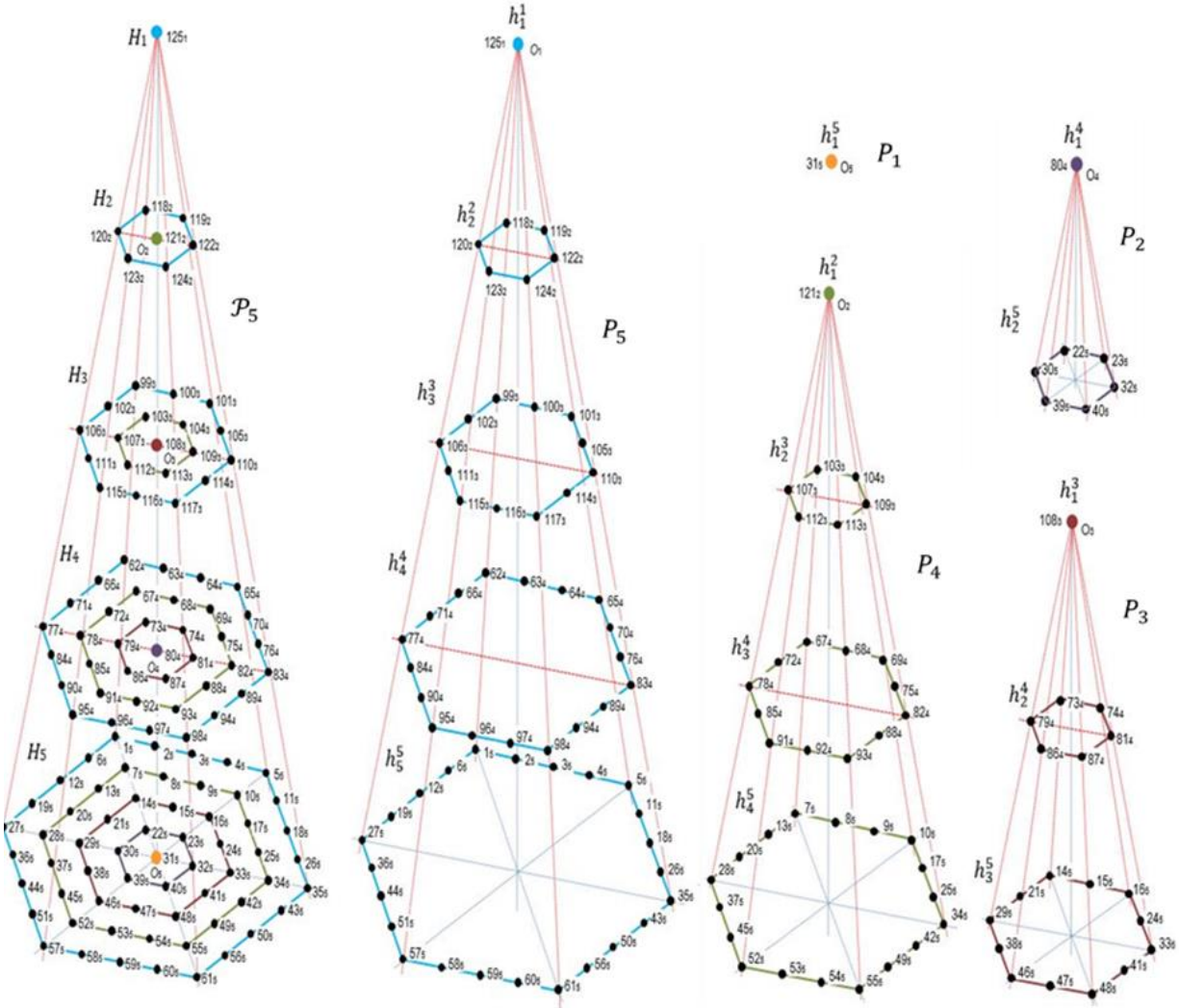

Рис. 8. Пирамидальное множество пиксель, состоящее из поверхностных пирамидальных подмножеств
(Fig. 8. Pyramidal set of pixels consists of surface pyramidal subsets)

Контурные шестигранники, образующие пирамидальное множество ТИ $\mathcal{P}_5$:

$$\mathcal{P}_5 = \bigcup_{i=1}^{5} H_i = \begin{cases} H_5 = h_5^5 \cup h_4^5 \cup h_3^5 \cup h_2^5 \cup h_1^5 \\ H_4 = h_4^4 \cup h_3^4 \cup h_2^4 \cup h_1^4 \\ H_3 = h_3^3 \cup h_2^3 \cup h_1^3 \\ H_2 = h_2^2 \cup h_1^2 \\ H_1 = h_1^1 \end{cases} . \quad (19)$$

Теперь представим пирамидальное множество $\mathcal{P}_n$ как множество ТИ, состоящее из поверхностных пирамидальных подмножеств ТИ P_i (рис. 8), т.е.:

$$\mathcal{P}_n = \bigcup_{i=1}^{n} P_i \ . \quad (20)$$

Пирамида в сечении (рис. 8), у которой контурные шестигранники со сторонами $n = \overline{1,5}$: $\mathcal{P}_5 = P_5 \cup P_4 \cup P_3 \cup P \cup P_1$, где P_1 – это точка (вырожденная пирамида),

$$\mathcal{P}_5 = \bigcup_{i=1}^{5} P_i = \begin{cases} P_5 = h_5^5 \cup h_4^4 \cup h_3^3 \cup h_2^2 \cup h_1^1 \\ P_4 = h_4^5 \cup h_3^4 \cup h_2^3 \cup h_1^2 \\ P_3 = h_3^5 \cup h_2^4 \cup h_1^3 \\ P_2 = h_2^5 \cup h_1^4 \\ P_1 = h_1^5 \end{cases} . \quad (21)$$

Очевидно, что:

$$\bigcup_{i=1}^{n} H_i = \bigcup_{i=1}^{n} P_i \ , \quad (22)$$

так как имеет место следующая матрица, объединяющая (18) и (22):

$$\begin{array}{ccccccccc} P_5 = & h_5^5 & \cup & h_4^4 & \cup & h_3^3 & \cup & h_2^2 & \cup & h_1^1 \\ & \cup & & \cup & & \cup & & \cup & & \\ P_4 = & h_4^5 & \cup & h_3^4 & \cup & h_2^3 & \cup & h_1^2 & & \\ & \cup & & \cup & & \cup & & & & \\ P_3 = & h_3^5 & \cup & h_2^4 & \cup & h_1^3 & & & & \\ & \cup & & \cup & & & & & & \\ P_2 = & h_2^5 & \cup & h_1^4 & & & & & & \\ & \cup & & & & & & & & \\ P_1 = & h_1^5 & & & & & & & & \\ & \| & & \| & & \| & & \| & & \| \\ & H_5 & & H_4 & & H_3 & & H_2 & & H_1 \end{array} . \quad (23)$$

Спроецируем поверхностные пирамиды на плоскость, т.е. сформируем шестигранники $\hat{P}_i^{\Phi}$, где $i = \overline{1,n}$ (рис. 9), шестигранник $\hat{P}_i$ состоит из контурных шестигранников $\hat{h}_j$ из сечения пирамидального множества. Контурный шестигранник $\hat{h}_j$ — это h_j контурный шестигранник шестигранника H_j в сечении пирамидального множества P_m (в случае с рис. 8 P_5), где $j \in [1, m]$ m – количество шестигранников (сечений) в пирамидальном множестве (в случае с рис. 8 m=5). На рис. 9 шестигранник $\hat{P}_5$ состоит из $h_5 \in H_5$, $h_4 \in H_4$, $h_3 \in H_3$, $h_2 \in H_2$, $h_1 \in H_1$.

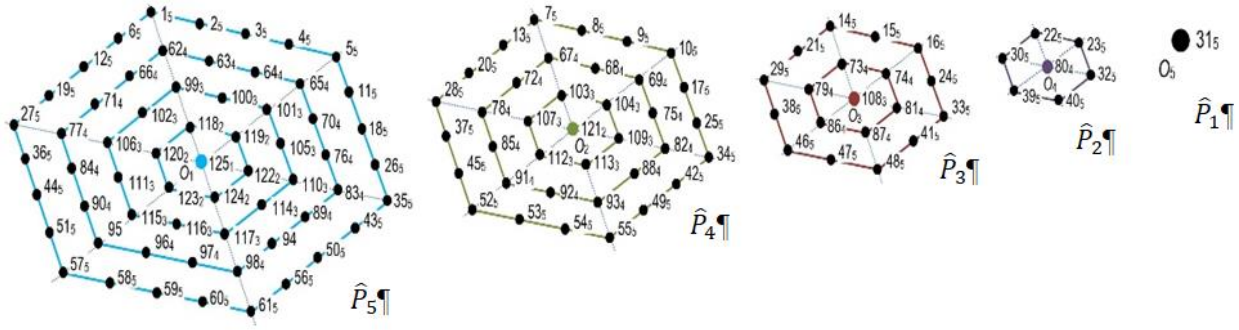

Рис. 9. Шестигранники, полученные из сечения пирамидального множества
(Fig. 9. Hexagons obtained from the pyramidal cross sections)

На следующем этапе проведём различные повороты контурных шестигранников в шестигранниках $\hat{P}_i^{\Phi}$, где $i = \overline{1,n}$:

$$\mathcal{P}_5 = \bigcup_{i=1}^{5} \hat{P}_i^{\Phi} = \begin{cases} \hat{P}_5^{\Phi} = h_5^{5(240^\circ)} \cup h_4^{4(120^\circ)} \cup h_3^{3(60^\circ)} \cup h_2^{2(60^\circ)} \cup h_1^{1(0^\circ)} \\ \hat{P}_4^{\Phi} = h_4^{5(300^\circ)} \cup h_3^{4(60^\circ)} \cup h_2^{3(120^\circ)} \cup h_1^{2(0^\circ)} \\ \hat{P}_3^{\Phi} = h_3^{5(240^\circ)} \cup h_2^{4(120^\circ)} \cup h_1^{3(0^\circ)} \\ \hat{P}_2^{\Phi} = h_2^{5(120^\circ)} \cup h_1^{2(0^\circ)} \\ \hat{P}_1^{\Phi} = h_1^{5(0^\circ)} \end{cases}. \quad (24)$$

Определим итоговые ПТИ в каждом из n шестигранников $\hat{P}_i^{\Phi}$, где $i = \overline{1,n}$, для случая, когда строки присоединяются к первой строке справа (рис. 10):

$$\begin{aligned} \bar{H}_5^{\hat{P}} &= \Big|\overset{L_5}{\Leftarrow}\Big|_{i=1} l_i^5 = l_1^5 \| l_2^5 \| \cdots \| l_{L_5-1}^5 \| l_{L_5}^5 \\ \bar{H}_4^{\hat{P}} &= \Big|\overset{L_4}{\Leftarrow}\Big|_{i=1} l_i^4 = l_1^4 \| l_2^4 \| \cdots \| l_{L_4-1}^4 \| l_{L_4}^4 \\ \bar{H}_3^{\hat{P}} &= \Big|\overset{L_3}{\Leftarrow}\Big|_{i=1} l_i^3 = l_1^3 \| l_2^3 \| \cdots \| l_{L_3-1}^3 \| l_{L_3}^3 \\ \bar{H}_2^{\hat{P}} &= \Big|\overset{L_2}{\Leftarrow}\Big|_{i=1} l_i^2 = l_1^2 \| \cdots \| l_{L_2}^2 \\ \bar{H}_1^{\hat{P}} &= l_1^1 \end{aligned}. \quad (25)$$

Аналогичным образом определим итоговые ПТИ в каждом из n шестигранников $\hat{P}_i^{\Phi}$, где $i = \overline{1,n}$, для случаев, когда строки присоединяются к первой строке слева и произвольным образом.

Определим итоговую ПТИ в пирамидальной конструкции:

$$\bar{\mathcal{P}}_n = \begin{cases} \Big|\overset{n}{\Leftarrow}\Big|_{i=1} \bar{S}_i^{\hat{P}} \\ \Big|\overset{n}{\Rightarrow}\Big|_{i=1} \bar{S}_i^{\hat{P}} \\ \Big|\overset{n}{\Leftrightarrow}\Big|_{i=1} \bar{S}_i^{\hat{P}} \end{cases}. \quad (26)$$

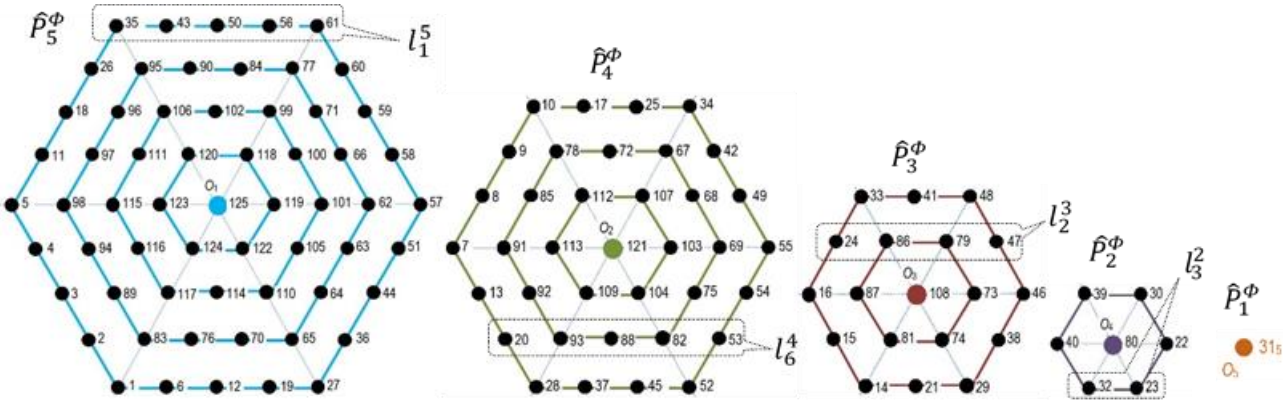

Рис. 10. Модифицированные в соответствии с (24) шестигранники
(Fig. 10. Pyramidal surface hexagons modified according to the formula (24))

Согласно (26) итоговая ПТИ $\bar{\mathcal{P}}_n$ может быть определена тремя способами сцепления итоговых ПТИ каждого модифицированного шестигранника $\overline{H}_i^{\hat{P}}$ $(i = \overline{1,n})$, т.е. последовательным присоединением к итоговой ПТИ максимального шестигранника итоговых ПТИ меньших шестигранников справа, слева или произвольным образом.

Другой вариант формирования итоговой ПТИ на основе пирамидальной конструкции заключается в произвольной перестановке строк шестигранников, входящих в пирамидальную конструкцию, т.е. в присоединении к первой строке максимального модифицированного шестигранника строк других шестигранников пирамидальной конструкции в произвольном порядке. Тогда ПТИ пирамидальной конструкции, сформированная в произвольном порядке, будет следующей:

$$\bar{\mathcal{P}}_n = \overset{\substack{i=n \\ j=L_n}}{\underset{\substack{i=1 \\ j=1}}{\Longleftrightarrow}} l_j^i \,. \tag{27}$$

Для примера, представленного на рис. 9, 10, $\bar{\mathcal{P}}_n$ (125 ТИ) будет иметь вид (порядок строк выбран случайным образом):

$$\bar{\mathcal{P}}_n = \overset{\substack{i=n \\ j=L_n}}{\underset{\substack{i=1 \\ j=1}}{\Longleftrightarrow}} l_j^i = l_1^3 \| l_1^1 \| l_2^5 \| l_4^3 \| l_4^4 \| l_5^5 \| l_3^3 \| l_7^4 \| l_6^5 \| \| l_1^4 \| l_3^4 \| \| l_5^3 \| l_2^4 \| l_2^3 \| l_4^5 \| l_1^2 \| l_3^5 \| l_2^2 \| l_9^5 \| l_5^4 \| l_7^5 \| l_6^4 \| l_8^5 \| l_3^2 \| l_1^5 \,. \tag{28}$$

2. Анализ свойств предлагаемого способа защиты

Предлагаемый способ защиты использует перестановку ТИ, поэтому требуется показать, что с помощью предлагаемого способа формирования перестановки ТИ возможно реализовать все $m!$ перестановок для ПТИ длиной m.

2.1 О перестановках ТИ

Перестановка может быть представлена в виде таблицы, где в первой строке записаны индексы элементов исходной ПТИ, а во второй строке новые индексы этих элементов. В случае с изображением формируется матрица перестановки, которая указывает, куда должен быть перемещен элемент исходной ПТИ. Если использовать в качестве элементарной перестановки (операция, которая осуществляется над исходной

ПТИ) – перестановку двух ТИ, то с её помощью можно реализовать все возможные перестановки над ПТИ. Предлагаемый способ генерации перестановки тоже сводится к матрице перестановки, однако элементарная перестановка – это поворот шестигранника. Процесс формирования перестановки *в общем случае* можно описать следующим образом:

1. Из исходной ПТИ выбирается подпоследовательность ТИ согласно ключевой информации.
2. Осуществляется элементарная перестановка подпоследовательности ТИ на основе ключевой информации.
3. ТИ из структуры после перестановки возвращаются на новые места в исходной ПТИ.
4. Процесс повторяется согласно ключевой информации.

Детально процесс формирования перестановки зависит от рассмотренных в [9] параметрах. Для пункта 1 подпоследовательность может быть выбрана различными способами, ключевая информация задает положение подпоследовательности в ПТИ и ее длину. В п. 2 требуется выбрать элементарную перестановку, как рассматривалось выше, это может быть шестигранник, шестигранник, состоящий их контурных шестигранников, и пирамидальное множество. Перестановка элементов в п. 2 осуществляется на основе ключевой информации, для шестигранника и контурных шестигранников – это угол поворота, для пирамидального множества – это еще и сдвиг по ребрам пирамиды.

Для исследуемого способа задания перестановки – поворота контурного шестигранника – элементарная перестановка эквивалентна циклическому сдвигу подпоследовательности ТИ. Циклический сдвиг последовательности $X = [x1, \dots, xn]$ на m вправо – это последовательность $\bar{X}_n = [x_{n-m+1}, \dots, x_n, x_1, \dots, x_{n-m}]$, сдвиг на m влево – $\bar{X}_n = [x_{m+1}, \dots, x_n, x_1, \dots, x_m]$. Требуется доказать, что рассматриваемый способ реализует все возможные перестановки исходной ПТИ. Для этого нужно либо явно это продемонстрировать для произвольного множества, либо показать, что с помощью последовательного применения элементарных перестановок (циклический сдвиг подпоследовательности) можно получить перестановку, которая может затронуть только два заранее заданных элемента множества. Преимущество предлагаемого способа заключается в том, что одна элементарная перестановка позволяет затронуть более одного элемента, следовательно, задать матрицу перестановки ПТИ можно меньшим количеством элементарных перестановок (в данном случае речь идет о перестановках, для которых количество неподвижных ТИ мало).

Лемма 1. *Если перестановка для* $\forall i, j \in [1, m]$ $i \neq j$, *где* m *– длина ПТИ* M, *позволяет поменять местами только ТИ с номерами* i *и* j, *то с помощью такой перестановки можно реализовать любую перестановку*.

Доказательство. Пусть для ПТИ X длиной m требуется реализовать перестановку, заданную таблицей:

x_1	…	x_m
x_{k1}	…	x_{km}

и $E_{i,j}$ – операция по перестановке ТИ, результатом которой является перестановка только ТИ с номерами i и j. Требуется, используя операцию $E_{i,j}$, поместить на позицию i, элемент с индексом k_i. Сделать это можно несколькими способами, продемонстрируем наиболее простой, который модифицирует таблицу перестановки в конце каждой итерации. Будем по порядку изменять исходную последовательность согласно таблице перестановки, для начала на место элемента с индексом 1 поставим элемент с индексом k_1, т.е. совершим операцию $E_{1,k1}$, тогда последовательность X имеет вид (x_{k1}, x_2, … x_{k1-1}, x_1, x_{k1+1}, … x_m), далее требуется изменить таблицу перестановки так, чтобы учесть, что элемент x_1 теперь имеет

индекс k_1, для этого найдем k_z такое, что $k_z = 1$, т.е. позицию z, на которую должен быть размещен элемент с индексом 1, и заменим k_z на k_1, чтобы таблица реализовывала ту же перестановку, что и исходная. Выставленный на свое место первый элемент можем не рассматривать, тогда таблица имеет вид:

x_2	…	x_z	…	x_m
x_{k2}	…	x_{k1}	…	x_{km}

Теперь, когда очередь дойдет до элемента с индексом z, на его место будет поставлен элемент с индексом k_1, который в исходной последовательности имел индекс 1, что и требуется сделать согласно исходной таблице перестановки. Для i-го элемента действуем аналогично, сначала осуществляем $E_{i,ki}$, а затем корректируем индексы в таблице перестановки, изменяем столбец с индексом z (z такой, что $k_z = i$):

x_z
x_{ki}

Повторив эту операцию m-1 раз, получим требуемую последовательность (x_{k1}, …, x_{km}).

Возвращаемся к задаче шестигранников, для того чтобы показать, что возможно осуществить все возможные перестановки элементов последовательности длины m, нужно найти такую операцию $E_{i,j}$, которая осуществляется с помощью последовательного применения поворотов шестигранников и позволяет поменять местами только два элемента исходной ПТИ.

Утверждение 1. *Для элементарной перестановки типа «поворот контурного шестигранника» (циклический сдвиг с длиной подпоследовательности, кратной шести) все возможные перестановки ПТИ не реализуются.*

Доказательство. Рассмотрим некоторые очевидные свойства перестановки типа «поворот контурного шестигранника»:

1. Два элемента нельзя обменять местами в рамках одного шестигранника.
2. Сдвиг можно осуществить только для подпоследовательности длины $6 \times (n-1)$, то есть длина подпоследовательности должна быть кратной 6, в таком случае минимальная длина подпоследовательности равна 6.

Определим, возможно ли с помощью последовательного применения перестановок типа «поворот контурного шестигранника» поменять местами только два элемента исходной последовательности.

Пусть дана последовательность ТИ $(x_1 \dots x_{i-1}\ x_i\ x_{i+1} \dots x_n)$ и нужно найти такое последовательное применение поворотов шестигранника, чтобы ТИ x_{i-1} и x_i поменялись местами и чтобы поменялись только эти элементы. Для осуществления поворота шестигранника (или, что эквивалентно, сдвига ПТИ) необходимо использовать ПТИ длиной, кратной 6. При перестановке элементов и формировании сдвига будем стремиться, чтобы его длина была минимальна.

1. Добавим r_1 элементов справа от x_{i-1} так, что $6 \mid (r_1 + 2)$, и сделаем сдвиг на 1. В результате имеем: $\left(x_1 \dots x_{i-2} \left\{x_i\ x_{i+1} \dots x_{i+r_1}, x_{i-1}\right\} \dots x_n\right)$, где между фигурными скобками «{}» заключены элементы, входящие в подпоследовательность, над которой осуществлялся сдвиг.
2. Чтобы получить $x_i\ x_{i-1}$, требуется выбрать регистр, начинающийся перед ТИ x_i и содержащий ТИ x_{i-1}, т.к. в подпоследовательности $(x_{i+1} \dots x_{i+r_1},\ x_{i-1})$ только $(r_1 + 1)$ ТИ, $6 \nmid (r_1 + 1)$ (где символ $\nmid$ – означает неделимость, т.е. 6 не делит $(r_1 + 1)$), то требуется добавить ТИ слева и осуществить сдвиг: $x_1 \dots x_{i-2}\ x_i \left\{X_{r_1}, x_{i-1}, X_{l_1}\right\} \dots x_n \mapsto$

$x_1 \dots x_{i-2}\ x_i\{x_{i-1}, X_{l_1}, X_{r_1}\} \dots x_n$, где «⟼» – символ сдвига. Длина подпоследовательности X_{l_1} равна l_1 и $6 \mid (r_1 + l_1 + 1)$.

3. Для того чтобы установить исходный порядок следования ТИ, требуется подпоследовательность X_{r_1} разместить слева от подпоследовательности X_{l_1}, т.е. эти подпоследовательности нужно поменять местами, но т.к. $6 \nmid r_1 + l_1$ (ввиду того, что $6 \mid r_1 + l_1 + 1$), то к подпоследовательности (X_{l_1}, X_{r_1}) требуется добавить справа еще l_2 ТИ так, чтобы $6 \mid r_1 + l_1 + l_2$, и осуществить сдвиг подпоследовательности $(X_{l_1}, X_{r_1}, X_{l_2})$, тогда имеем:

$$x_1 \dots x_{i-2}\ x_i, x_{i-1}, \left|X_{l_1}, X_{r_1}, X_{l_2}\right| \dots x_n \longmapsto x_1 \dots x_{i-2}\ x_i, x_{i-1}, |\ X_{r_1}, X_{l_1},\ X_{l_2}| \dots x_n.$$

4. Снова приходим к необходимости поменять местами две рядом стоящие подпоследовательности ТИ. В этом случае, из принципа минимизации расширяемой подпоследовательности, $l_1 = l_2 = 1$, то получаем задачу, эквивалентную исходной, когда требуется поменять местами две рядом стоящие ТИ, следовательно, шаги 1-3 будут выполняться, пока исходная последовательности не закончится и желаемое её состояние не будет достигнуто (количество носителей перестановки всегда > 2), но для последних элементов не возможно дополнить длину до кратности 6.

Используя только сдвиг ПТИ длиной, кратной 6, невозможно переставить два элемента местами и только их.

Теперь определим условия, при которых можно поменять местами две подпоследовательности X_1 и R_1, используя только циклический сдвиг подпоследовательности длины, кратной n.

Рассмотрим некоторые свойства делимости, которые потребуются далее:

1) если $n|(a + b)$ и $n|a$, то $n|b$ при условии, что $a > n$ и $b > n$;
2) если $n|(a + b)$ и $n \nmid a$, то $n \nmid b$.

Для произвольного размера регистра сдвига и двух произвольных подпоследовательности элементов. Рассмотрим часть последовательности $(Z_1\ X_1\ R_1\ X_2) \in X$, где $Z_1\ X_1\ R_1\ X_2$ – подпоследовательности последовательности X и следуют в исходной последовательности друг за другом. Требуется получить $(Z_1\ R_1\ X_1\ X_2)$. (Задача сближения подпоследовательности, используя сдвиг.)

Пусть $l(X_1)$ – длина последовательности X_1, $len(R_1)$ – длина последовательности R_1 и т.д., а n – кратность длины подпоследовательности, над которой можно осуществить сдвиг, тогда:

1) если $n \mid (len(X_1) + len(R_1))$, то задача решается за 1 операцию сдвига, т.к. над подпоследовательностью $(X_1\ R_1)$ может быть осуществлен сдвиг, и получим $(R_1\ X_1)$;

2) если $n \nmid len(X_1) + len(R_1)$, то требуется добавить слева подпоследовательность L_1, такую, что $n \mid len(X_1) + len(R_1) + len(L_1)$.

Рассмотрим два случая, когда $n \mid len(X_1)$ и когда $n \nmid len(X_1)$:

1) если $n \mid len(X_1)$, то, т.к. $n \nmid \big(len(X_1) + len(R_1)\big)$, следовательно, $n \nmid len(R_1)$, тогда т.к. $n \mid (len(X_1) + len(R_1) + len(Z_1))$, то $n \mid (len(R_1) + len(Z_1))$ при условии, что $(len(R_1) + len(Z_1)) > n$. Это достигается путем выбора Z_1 соответствующей длины, следовательно, можно одной перестановкой обменять местами подпоследовательности R_1 и L_1 и восстановить исходный порядок следования подпоследовательностей;

2) если $n \nmid len(X_1)$, тогда т.к. $n \mid (len(X_1) + len(R_1) + len(Z_1))$ и $n \nmid len(X_1)$, следовательно, $n \nmid \big(len(R_1) + len(Z_1)\big)$, следовательно, чтобы поменять местами подпоследовательности R_1 и Z_1 , требуется добавить подпоследовательность Z_2 такую, что $n \mid (len(R_1) + len(Z_1) + len(Z_2))$. Из исходной подпоследовательности $(Z_2\ R_1\ Z_1)$ после сдвига получаем: $(Z_1\ Z_2\ R_1)$. Для восстановления исходного порядка следования требуется

поменять местами блоки Z_1 и Z_2, но т. к. $n \nmid len(Z_2)$ (т.к. $n \nmid len(R_1) + len(Z_1)$) и $n \mid (len(R_1) + len(Z_1) + len(Z_2))$), и $n \nmid len(Z_1)$ (т.к. $n \nmid len(X_1) + len(R_1)$), и $n \mid (len(X_1) + len(R_1) + len(Z_1))$). Возможны 2 случая: $n \mid len(R_1)$ и $n \nmid len(R_1)$. Если $n \mid len(R_1)$, то $n \mid \big(len(Z_1) + len(Z_2)\big)$, и можно восстановить исходный порядок следования. Если $n \nmid len(R_1)$, то $n \nmid \big(len(Z_1) + len(Z_2)\big)$, и подпоследовательность $(Z_1\ Z_2)$ нужно расширять слева: $(Z_3\ Z_1\ Z_2)$. Причём $n \nmid (len(Z_1) + len(Z_2))$, $n \nmid len(Z_1)$, $n \nmid len(Z_2)$, $n \mid \big(len(Z_1) + len(Z_2) + len(Z_3)\big)$, следовательно, $n \nmid len(Z_3)$ и $n \nmid (len(Z_2) + len(Z_3))$. Осуществляем сдвиг подпоследовательности $(Z_3\ Z_1\ Z_2) \mapsto (Z_2\ Z_3\ Z_1)$. Приходим к необходимости поменять местами элементы $Z_2\ Z_3$, что аналогично задаче замены местами элементов $Z_1\ Z_2$. Произведя аналогичные операции, приходим к необходимости замены местами подпоследовательности $Z_3\ Z_4$ и т.д. В итоге не будет достигнута желаемая перестановка.

Следствие. Подпоследовательность $(Z_1\ X_1\ R_1\ X_2)$ может быть преобразована в подпоследовательность $(Z_1\ R_1\ X_1\ X_2)$ без влияния на другие элементы последовательности при использовании сдвига подпоследовательности длины, кратной n, только в следующих случаях:

1) $n \mid len(X_1) + len(R_1)$;
2) $n \nmid len(X_1) + len(R_1)$ и $n \mid len(X_1)$;
3) $n \nmid len(X_1) + len(R_1)$, $n \nmid len(X_1)$ и $n \mid len(R_1)$.

Рассмотрим случай с $n = 6$, $len(X_1) = 1$, в таком случае можно приблизить один элемент к другому, только если между ними есть 5 элементов, либо если расстояние между ними кратно 6. В случае с шестигранником не достигается ни одна из этих ситуаций.

Очевидно, что исследуемый способ не реализует все возможные перестановки, тогда оценим количество перестановок, которые способен реализовать данный способ. Сделаем это через формирование произвольной перестановки.

Пусть задана произвольная перестановка:

$$\begin{matrix} 1 & 2 & 3 & 4 & 5 & \dots & m \\ i_1 & i_2 & i_3 & i_4 & i_5 & \dots & i_m \end{matrix}$$

$\forall\, ij$, где $1 \le j \le m, 1 \le i_j \le m$; и $\forall\, j,k$, где $j \ne k$ и $1 \le j,k \le m, i_j \ne i_k$.

С помощью перестановки «поворот шестигранника» попытаемся получить заданную произвольную перестановку.

1. Для этого начнём собирать её, начиная с первой ТИ. Суть способа заключается в следующем: проходя по порядку по всем ТИ с помощью последовательного применения «поворота шестигранника», размещаем ТИ на требуемых позициях, начиная с первой. На позицию 1 передвигаем ТИ под номером i_1 (используя только сдвиги, это можно сделать большим количеством способов, например: используя только минимальный сдвиг подпоследовательности длины 6), т.к. теперь первый элемент целевой перестановки установлен на место, можно рассматривать исходную последовательность без первого установленного на место элемента и аналогичным образом установить второй элемент.

2. Получается, что аналогичным образом переставляя элементы, дойдем до подпоследовательности такого размера, что над ней невозможно осуществить преобразование (т.е. меньше 6). Таким образом, целевая перестановка не может быть достигнута, т.к. нельзя осуществить перестановку последних 5, элементов не затрагивая предыдущие.

Таким образом, способ реализует не менее чем $(m-5)!$ перестановок. Данный недостаток можно устранить, если ввести перестановку для частично заполненных

шестигранников либо для шестигранников, состоящих из произвольного количества элементов, или введя дополнительную степень свободы в шестигранник (см. п. 2.3).

2.2 Смещение ТИ от первоначальной позиции

Для исследуемого способа проведем оценку среднего расстояния, на которое переместится ТИ исходной последовательности в зависимости от параметров алгоритма. Пусть количество ТИ в контурном шестиграннике меняется в диапазоне от l до k и выбор каждого из них равновероятен, тогда среднее количество ТИ в контурном шестиграннике $(\boldsymbol{l+k})/\mathbf{2}$. В контурном шестиграннике размером l смещение ТИ от начального положения зависит от угла поворота и того, в какой грани контурного шестигранника находилась точка. Грань с индексом 0 – это грань начала строчного представления контурного шестигранника.

Найдем смещение для произвольной ТИ на шестиграннике. Выбор ТИ случаен и равновероятен, поэтому вероятность попасть в какую-либо из граней одинаковая и равна 1/6. Выбор угла поворота тоже случаен и равновероятен, соответственно вероятность каждого угла 1/6. Тогда среднее смещение выразим как математическое ожидание случайной величины (данные взяты из таблицы 2):

$$\boldsymbol{E} = 1/6\left(\frac{1}{6}\left(\frac{l}{6}+\frac{2l}{6}+\frac{3l}{6}+\frac{4l}{6}+\frac{5l}{6}\right)\right) + 1/6\left(\frac{1}{6}\left(\frac{l}{6}+\frac{2l}{6}+\frac{3l}{6}+\frac{4l}{6}+\frac{l}{6}\right)\right) + 1/6\left(\frac{1}{6}\left(\frac{l}{6}+\frac{2l}{6}+\frac{3l}{6}+\frac{2l}{6}+\frac{l}{6}\right)\right) + 1/6\left(\frac{1}{6}\left(\frac{l}{6}+\frac{2l}{6}+\frac{3l}{6}+\frac{2l}{6}+\frac{l}{6}\right)\right) + 1/6\left(\frac{1}{6}\left(\frac{l}{6}+\frac{4l}{6}+\frac{3l}{6}+\frac{2l}{6}+\frac{1l}{6}\right)\right) + 1/6\left(\frac{1}{6}\left(\frac{5l}{6}+\frac{4l}{6}+\frac{3l}{6}+\frac{2l}{6}+\frac{1l}{6}\right)\right) = 35l/108 \quad (29)$$

Таблица 2. Зависимость между индексом грани, углом поворота и смещением ТИ

Грань (edge index)	0	1	2	3	4	5
0° смещение (offset)	0	0	0	0	0	0
60° смещение (offset)	*l*/6	*l*/6	*l*/6	*l*/6	*l*/6	5*l*/6
120° смещение (offset)	2*l*/6	2*l*/6	2*l*/6	2*l*/6	4*l*/6	4*l*/6
180° смещение (offset)	3*l*/6	3*l*/6	3*l*/6	3*l*/6	3*l*/6	3*l*/6
240° смещение (offset)	4*l*/6	4*l*/6	2*l*/6	2*l*/6	2*l*/6	2*l*/6
300° смещение (offset)	5*l*/6	*l*/6	*l*/6	*l*/6	*l*/6	*l*/6

Согласно (29), одна ТИ смещается на величину $35l/108$, а в шестиграннике размером k – на $35k/108$. Тогда в среднем ТИ сместится на $35\,(l+k)/216$ (средний размер шестигранника $(l+k)/2$). При условии, что контурный шестигранник задается в каждой точке последовательности и может быть повернут на любой угол, в том числе нулевой, равновероятно, что ТИ в среднем поучаствует в $\left(\frac{l+k}{2}\right)$ шестигранниках и в каждом из них в среднем сместится на $35\,(l+k)/216$, тогда средний сдвиг точки относительно её исходного положения:

$$((l+k)/2\,) \times (35\,(\mathrm{l}+\mathrm{k})/216\,) = 35(l+k\,)^2/432\,. \quad (30)$$

Пусть m – длина текста, тогда $35(l+k\,)^2/432 = m/2$, выразим $l+k$ через m:

$$l+k = (216 * sqrt\,(m)/35)\,. \quad (31)$$

Пусть $l = 0$ – минимальный шестигранник, тогда $k = 216 * sqrt(m)/35$. Для изображения размером 1920×1080 k = 8892, что соответствует контурному шестиграннику h_{1482}.

Преодолеть ограничение числа перестановок возможно, если ввести дополнительную степень свободы в шестигранник, как это рассматривалось в п. 2.4. Идея заключается в том, чтобы, помимо поворота контурного шестигранника, добавить сдвиг элементов внутри половины диагонали шестигранника. Такая модификация позволяет для минимального шестигранника переставить местами только 2 соседних элемента, не затрагивая при этом все остальные элементы последовательности, а как было показано выше в **Лемме 1** это означает, что алгоритм может реализовать любую возможную перестановку на множестве элементов, что позволяет разрешить проблему реализации неполного числа перестановок.

3. Оценка стойкости способа защиты к вскрытию методом полного перебора

Особенность осуществления перестановок с помощью предлагаемого способа перестановки заключается в том, что число способов перестановки больше самих перестановок, а это означает, что для одной ПТИ существует множество ключей, которые реализуют одну и ту же перестановку. Как было показано в [9], в зависимости от выбранного подхода к реализации алгоритма, количество различных ключей может достигать $(6^m m!)^m$, что существенно больше возможного количества перестановок, равного $m!$. Кроме этого, длина ключа не постоянна, а зависит от размера последовательности [9].

3.1 Реализационные аспекты

Вначале рассмотрим блок-схему и алгоритм функционирования способа защиты, который кроме перестановки включает в себя сложение изображения с гаммой (псевдослучайной последовательностью) с целью сокрытия статистической информации о скремблированном изображении.

Способ защиты (рис. 11) включает следующие этапы:

1. Генератор гаммы формирует псевдослучайную последовательность длины, равной размеру исходного изображения.
2. Псевдослучайная последовательность подвергается перестановке с помощью предложенного способа перестановки.
3. Исходное изображение суммируется по модулю 2 с псевдослучайной последовательностью, подверженной перестановке элементов.
4. Результат сложения по модулю 2 подвергается перестановке с помощью предложенного способа перестановки, и в итоге получается закодированное изображение.

Подобная схема реализации защиты изображений достаточно очевидна, она предлагалась к использованию и в других способах защиты изображений, описанных в работах [12, 13, 14]. Но в них акцент был сделан на генерацию хаотических карт, а перестановка элементов могла и не использоваться [10]. Центральным элементом алгоритмов является генератор хаотических карт, представляющий собой гамму, с которой происходит сложение значений пикселей изображения по модулю. При таком подходе к генератору гаммы (генератору хаотических карт) предъявляются высокие требования. В предлагаемом алгоритме генератор гаммы дополнительно усиливается перестановкой, что позволяет использовать генератор с малым периодом и повторяющимися гаммами. Обзор работ по данной тематике был представлен в [11], общим вектором во всех работах было использование шифрования исходного изображения для представления его в

шумоподобном виде и дальнейшее встраивание его в другое изображение (стеганография) для сокрытия самого факта передачи изображения.

3.2 Атаки

Рассмотрим некоторые возможные примеры атак на данный способ защиты. Предположим, что злоумышленник завладел закодированным изображением и ему требуется найти соответствующее ему исходное изображение. В таком случае от него потребуется осуществить полный перебор ключей и сформировать критерий того, что полученный в результате перебора вариант является не шумом, а искомым изображением. Для этого нужно перебрать все начальные состояния генератора гаммы, все возможные перестановки сформированной гаммы и все возможные перестановки результата сложения гаммы и исходного изображения. При полном переборе перестановок не имеет смысла перебирать их с помощью ключей предлагаемого способа, так как их существенно больше, чем перестановок вообще, поэтому эффективнее перебирать все возможные матрицы перестановки, которых $m!$. В сформулированных условиях задача по полному перебору имеет сложность $\mathcal{O}(n) \times \mathcal{O}(m! \times m!)$, где $\mathcal{O}(n)$ – сложность перебора начального состояния генератора гаммы, а $\mathcal{O}(m! \times m!)$ – сложность перебора перестановок на этапах 2 и 4 (п. 3.1).

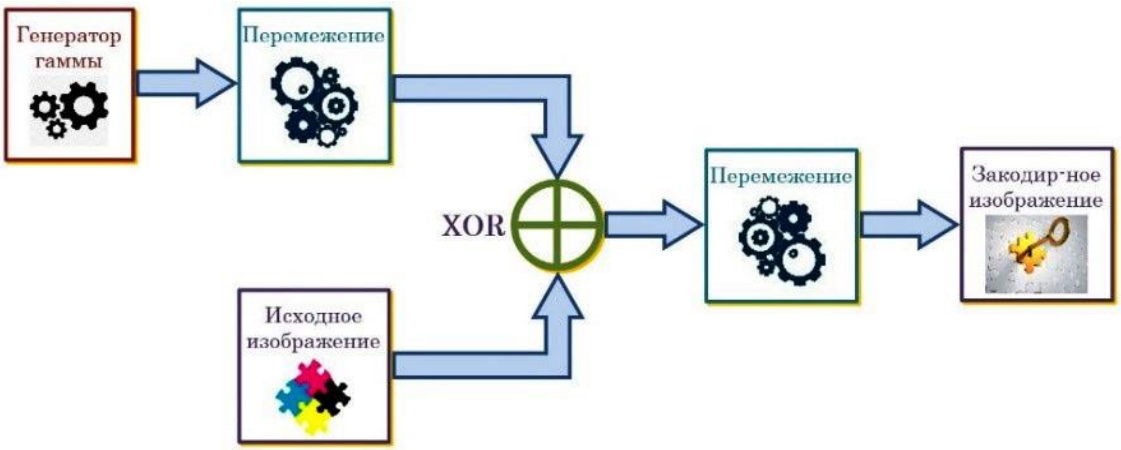

Рис. 11. Блок-схема предлагаемого способа защиты
(Fig. 11. The scheme of the proposed method of protection)

В случае, если злоумышленник располагает закодированным изображением и исходным и пытается обнаружить ключ, то задача несколько меняется. Злоумышленник может отдельно перебирать ключи для этапов 1, 2 и этапа 4 (п. 3.1), это связано с тем, что в результате перестановки ТИ на этапе 4 сохраняется статистическая информация, хранящаяся в изображении, поэтому злоумышленник может перебирать начальное заполнение гаммы и перестановку элементов сгенерированной последовательности гаммы до тех пор, пока не получит статистику значений ТИ, идентичную закодированному изображению. Возможны случаи, когда верная статистика будет получена на неверно подобранном ключе, причем установить этот факт не представляется возможным, и тогда восстановить точный ключ, который применялся в этом алгоритме, попросту невозможно. Если верная статистика была получена только на одном изображении, то далее для этого изображения нужно перебрать все возможные перестановки, чтобы получить исходное изображение. Таким образом, сложность перебора в конкретных условиях равна

$\mathcal{O}(n) * \mathcal{O}(m! + m!)$, где $\mathcal{O}(n)$ – сложность перебора начального состояния генератора гаммы, а $\mathcal{O}(m! + m!)$ – сложность независимого перебора перестановок на этапах 2 и 4 (п. 3.1). В случае, если злоумышленник располагает значением генератора гаммы и закодированным изображением и хочет восстановить исходное изображение, то задача эквивалентна предыдущей, за исключением необходимости перебора всех значений начального заполнения генератора гаммы, и имеет сложность $\mathcal{O}(m! + m!)$.

3.3 Предварительное искажение

В двух рассмотренных в п. 3.2 случаях возможно снижение вычислительной сложности перебора за счет использования статистической информации из закодированного изображения и гаммы. Однако стоит отметить, что алгоритм предназначается для изображений, то есть для информации, которая обладает сильной избыточностью (это используется в том числе и в способах сжатия изображений, например, формат *jpeg*), что позволяет вносить дополнительные, незначительные искажения в закодированное изображение, не нарушая его графической визуальной значимости, особенно, когда речь идет о фотографиях и кадрах видеосигнала, где изображение получено с помощью устройства, которое тоже вносит определенные искажения в изображение (два изображения сняты с использованием двух разных камер могут иметь некоторые незначительные отличия, которые никак не мешают воспринимать информацию на фотографии). В таком случае схема может выглядеть следующим образом (рис. 12):

Рис. 12. Блок-схема способа защиты с внесением случайных искажений
(Fig. 12. The scheme of the proposed method of protection with random distortion)

Как показано на рис. 12, способ защиты идентичен представленному на рис. 11, за исключением того, что на этапе 4 дополнительно происходит суммирование со случайным шумом. Главное требование к генератору случайного шума: его выходное значение должно быть очень трудно воспроизводимым, то есть перебор всех возможных значений случайного шума должен соответствовать по сложности перебору всех возможных значений ТИ. Случайный шум умножается на коэффициент искажения и подмешивается к исходному изображению и гамме, что позволяет полностью устранить возможность перебора с использованием статистической информации, но при этом раскодированное

изображение будет содержать определенный шум, величина которого определяется коэффициентом искажения. Анализ того, каким должно быть значение коэффициента искажения, и рекомендации по выбору генератора случайного шума планируются в будущей работе.

Область применения. Важным свойством предлагаемого способа защиты является его стойкость к локальным искажениям пикселей в закодированном изображении, это связано с тем фактом, что каждый пиксел кодируется вне зависимости от остальных, а значит искажение одного пикселя не приводит к искажению соседних. Такое свойство очень важно, когда в результате передачи могут возникнуть искажения в переданном сообщении, например, в стеганографии, когда закодированное изображение встраивается в другое изображение, или в алгоритмах сжатия с потерей информации. Предложенный способ может использоваться для кодирования изображения для визуально значимого кодирования [11, 15].

3.4 Оценка вычислительной стойкости

Проанализируем вычислительные ресурсы, которые необходимы, чтобы апробировать все гипотезы для простейшего случая со сложностью $\mathcal{O}(m! + m!)$. Предположим, имеется монохромное изображение в разрешении 1920×1080 пикселей. Пусть атомарная операция ЭВМ – это перебор одной перестановки (что является сильным допущением, т.к. для перебора одной перестановки требуется существенно больше операций), тогда потребуется $(1920 \times 1080)! \times 2$ операций, что составляет приблизительно 6×10^{12197811}. Вычислительная мощность самого быстрого компьютера на сегодняшний день составляет приблизительно $130\,PFLOPS$ или 130×10^{15} операций в секунду. Тогда для перебора одному такому суперкомпьютеру понадобится 4.6×10^{12197794} секунд или 1.5×10^{12197787} лет. Таким образом, перебор даже для достаточно простого случая – задача не тривиальная, а с учётом предложенных корректировок (например, см. п. 1.3, 3.3) она становится ещё более сложной. Если требуется решить задачу точного восстановления ключа для пары закодированного изображения и исходного, то в некоторых случаях такое решение вообще найти невозможно.

ЗАКЛЮЧЕНИЕ

В статье предложен усовершенствованный способ защиты неподвижных изображений [7], основанный на использовании трёхмерных геометрических перестановок ТИ, и схемы его реализации. В результате анализа различных вариантов способа формирования элементарной перестановки ТИ был сделан выбор в пользу применения пирамидальной конструкции (п. 1.4), состоящей из контурных шестигранников, что позволяет реализовать полный набор всех возможных перестановок ТИ. Предложено несколько вариантов способа защиты изображений, использующих гаммирование и перестановку. По результатам анализа стойкости способа защиты к атаке типа «полный перебор» был сделан вывод об эффективности способа защиты с использованием избыточности в графическом представлении данных, что делает этот способ специализированным для защиты графических данных и при этом обладает высокой стойкостью к методу полного перебора. Представленный анализ усовершенствованного способа показал высокую криптографическую стойкость и способность парировать современные и перспективные квантовые криптоаналитические атаки.

СПИСОК ЛИТЕРАТУРЫ:

1. Simon D.R. On the Power of Quantum Cryptography. 35th Annual Symposium on Foundations of Computer Science, Santa Fe, New Mexico, USA, 20-22 November 1994. P. 116–123. IEEE Computer Society, 1994.
2. Boneh D., Zhandry M. Secure Signatures and Chosen Ciphertext Security in a Quantum Computing World // Advances in Cryptology – CRYPTO 2013 – 33rd Annual Cryptology Conference, Santa Barbara, CA, USA, 18-22 August 2013. Proceedings, Part II. 2013. P. 361–379.
3. Roetteler M., Steinwandt R. A note on quantum related-key attacks // Information Processing Letters 115(1), P. 40–44, 2015.
4. Kaplan M., Leurent G., Leverrier A., Naya-Plasencia M. Breaking Symmetric Cryptosystems Using Quantum Period Finding // Matthew Robshaw and Jonathan Katz, editors, Advances in Cryptology – CRYPTO 2016 – 36th Annual International Cryptology Conference, Santa Barbara, CA, USA, August 14-18, 2016, Proceedings, Part II, Vol. 9815 of LNCS. P. 207–237. Springer, 2016.
5. National Institute of Standards and Technology, Internal Report 8105, «Report on Post-Quantum Cryptography», NISTIR 8105, April 2016. URL: https://doi.org/10.6028/NIST.IR.8105 (дата обращения: 25.02.2019).
6. National Institute of Standards and Technology. «Blockchain Technology Overview». NISTIR 8202, October 2018. https://doi.org/10.6028/NIST.IR.8202
7. Melnikov D., Jones A. Static Image Data Hiding and Encryption Method // Proceedings of the 3rd European Conference on Information Warfare and Security. Royal Holloway University of London, UK. 28-29 June 2004. P. 279–284.
8. Мельников Д.А. Метод защиты неподвижных изображений // Информационные технологии управления в социально-экономических системах. 2010. № 4. С. 130–139.
9. Мельников Д.А., Абрамов А.А., Горбатов В.С., Дураковский А.П., Махмутов Р.Д. Криптографический способ документирования покадровых изображений // Научная визуализация. 2016. Том 8, № 5. С. 13–25. URL: http://sv-journal.org/2016-5/02/en/index.php?lang=en (дата обращения: 25.02.2019).
10. Aboughalia R., Alkishriwo O. Color Image Encryption Based on Chaotic Block Permutation and XOR Operation // Libyan International Conference on Electrical Engineering and Technologies (LICEET2018), Tripoli, Libya, 3 – 7 March 2018. P. 492–497.
11. Murad S., Gody A., Barakat T., Enhanced Security of Symmetric Encryption Using Combination of Steganography with Visual Cryptography // International Journal of Engineering Trends and Technology (IJETT), Vol. 65 № 3, November 2018. P. 149–154.
12. Hoppena C., Kohayakawa Y., Moreira C. G., Sampaio R. M., Testing permutation properties through subpermutations // Theoretical Computer Science. ISSN 0304-3975 Vol. 412, № 29, 1 July 2011. P. 3555–3567.
13. Chena X., Huc C., Adaptive medical image encryption algorithm based on multiple chaotic mapping // Saudi Journal of Biological Sciences, ISSN 1319-562X, Vol. 24, № 8, December 2017. P. 1821–1827
14. Mandal M., Banik G., Chattopadhyay D., Nandi D., An Image Encryption Process based on Chaotic Logistic Map // IETE IETE Technical Review. ISSN 0256-4602, Vol. 29. № 5. P. 395–404. doi: 10.4103/0256-4602.103173.
15. Bao L., Zhou Y., Image encryption: Generating visually meaningful encrypted images // Information Sciences ISSN 0020-0255, Vol. 324, 10 December 2015. P. 197–207. doi: 10.1016/j.ins.2015.06.049

REFERENCES:

[1] Simon D.R. «On the Power of Quantum Cryptography». In: 35th Annual Symposium on Foundations of Computer Science, Santa Fe, New Mexico, USA, 20-22 November 1994. P. 116–123. IEEE Computer Society, 1994.
[2] Boneh D., Zhandry M. «Secure Signatures and Chosen Ciphertext Security in a Quantum Computing World». In: Advances in Cryptology - CRYPTO 2013 - 33rd Annual Cryptology Conference, Santa Barbara, CA, USA, August 18-22, 2013. Proceedings, Part II. P. 361–379, 2013.
[3] Roetteler M., Steinwandt R. «A note on quantum related-key attacks». Information Processing Letters 115(1), 40–44, 2015.
[4] Kaplan M., Leurent G., Leverrier A. and Naya-Plasencia M. «Breaking Symmetric Cryptosystems Using Quantum Period Finding». In Matthew Robshaw and Jonathan Katz, editors, Advances in Cryptology - CRYPTO 2016 - 36th Annual International Cryptology Conference, Santa Barbara, CA, USA, August 14-18, 2016, Proceedings, Part II, volume 9815 of LNCS. P. 207–237. Springer, 2016.
[5] Chen L., Jordan S., Liu Y.-K., Moody D., Peralta R., Perlner R. and Smith-Tone D., National Institute of Standards and Technology, Internal Report 8105, «Report on Post-Quantum Cryptography», April 2016. URL: https://doi.org/10.6028/NIST.IR.8105& (accessed: 25.02.2019).

[6] National Institute of Standards and Technology. «Blockchain Technology Overview». Draft NISTIR 8202, January 2018.
[7] Melnikov D., Jones A. «Static Image Data Hiding and Encryption Method». Proceedings of the 3rd European Conference on Information Warfare and Security. Royal Holloway University of London, UK. 28-29 June 2004. P. 279–284.
[8] Melnikov D.A. «Metod zhashchity nepodvizhnykh izobrazhenii». Infomatsionnye tekhologii upravleniya v socialno-ekonomicheskih sistemah. 2010. № 4. S. 130–139. (in Russian).
[9] Melnikov D.A., Abramov A.A., Gorbatov V.S., Durakovskiy A.P., Makhmutov R.D. «Cryptographic method of the pictures documenting». Scientific Visualization. 2016. Vol. № 8 № 5. (Open Access Electronic Journal), P. 13–25. ISSN 2079-3537. URL: http://sv-journal.org/2016-5/02/en/index.php?lang=en (accessed: 25.02.2019).
[10] Aboughalia R., Alkishriwo O. Color Image Encryption Based on Chaotic Block Permutation and XOR Operation. Libyan International Conference on Electrical Engineering and Technologies (LICEET2018), Tripoli, Libya, 3 – 7 March 2018. P. 492–497.
[11] Murad S., Gody A., Barakat T., Enhanced Security of Symmetric Encryption Using Combination of Steganography with Visual Cryptography. International Journal of Engineering Trends and Technology (IJETT), Vol. 65 № 3, November 2018. P. 149–154.
[12] Hoppena C., Kohayakawa Y., Moreira C. G., Sampaio R. M., Testing permutation properties through subpermutations. Theoretical Computer Science. ISSN 0304-3975 Vol. 412, № 29, 1 July 2011. P. 3555–3567.
[13] Chena X., Huc C., Adaptive medical image encryption algorithm based on multiple chaotic mapping. Saudi Journal of Biological Sciences, ISSN 1319-562X, Vol. 24, № 8, December 2017. P. 1821–1827
[14] Mandal M., Banik G., Chattopadhyay D., Nandi D., An Image Encryption Process based on Chaotic Logistic Map. IETE IETE Technical Review. ISSN 0256-4602, Vol. 29. № 5. P. 395–404. doi: 10.4103/0256-4602.103173.
[15] Bao L., Zhou Y., Image encryption: Generating visually meaningful encrypted images. Information Sciences ISSN 0020-0255, Vol. 324, 10 December 2015. P. 197–207. doi: 10.1016/j.ins.2015.06.049.

Поступила в редакцию – 25 февраля 2019 г. Окончательный вариант – 10 мая 2019 г.
Received – February 25, 2019. The final version – May 20, 2019.