**Лабораторная работа № 5**

По курсу «Цифровые технологии защиты информации»

**Встраивание невидимого ЦВЗ в пространственной области**

Метод замены младшего бита.

1. Сформировать контейнер-оригинал: изображение в формате BMP24 размером 320\*240 (Paint\Рисунок\Атрибуты) из рисунка с номером N, находящегося в папке \Lab5-2010\work task\Images 5.
2. Импортировать контейнер-оригинал в пакет "Mathematica" ( Import[] ) и определить размеры рисунка - ImageDimensions[].
3. Создать массив данных в формате RGB с помощью функции ImageData[\*\*\*,"Byte"].
4. Провести стирание младшего значащего бита во всех пикселях контейнера-оригинала используя функцию BitClear[\*\*\*,0].
5. Создать новое изображение из модифицированных данных - Image[\*\*\*,"Byte"] и сохранить его (правая кнопка мыши \Save Image as—файл bmp).
6. Сформировать серию изображений с 2,3,4,5,6-ю стертыми битами, провести визуальное сравнение изображений и определить порог различимости искажений – выделить и сохранить пример замеченных искажений контейнера – оригинала.
7. Для трех произвольных строк изображения-оригинала установить в "1" (BitSet[\*\*\*,k]) все биты в 7,6 и 5-ом разрядах. Сохранить модифицированное изображение.
8. В строке с номером N изображения-оригинала установить младшие биты зеленого цвета, кратные N в "0" ,а младшие биты синего цвета, кратные 2N – в "1". Сохранить модифицированное изображение.

Метод блочного сокрытия.

1. Сформировать контейнер-оригинал №2: изображение в формате BMP24 размером 360\*480 (Paint\Рисунок\Атрибуты) из рисунка с номером N, находящегося в папке \Lab5-2010\work task\Images 5.
2. Импортировать контейнер-оригинал №2 в пакет "Mathematica" и разбить его с помощью функции ImagePartition[\*\*\*, \*] на блоки размером 8х8 пикселей.
3. Определить размерность полученного массива - Dimensions[\*] .
4. Разделить изображение из блока с индексами 2N,2N на три составляющие в помощью функции ColorSeparate[].
5. Для каждого цвета получить массив значений интенсивностей цвета и выровнять его до уровня списка.
6. Определить сумму элементов списка и проверить эту сумму на четность - Divisible[\*\*\*,2].
7. С помощью функции RandomChoice[\*\*\*,\*\*] найти случайные индексы для элемента матрицы 8х8.
8. Изменить четность блока, прибавив по модулю 2 единицу (операция BitXor[\*\*,\*\*]) к значению интенсивности цвета случайного элемента блока.
9. В строке исходного массива блоков (п. 10) с номером N сформировать первые 20 блоков "четных", следующие 20 – "нечетные".
10. Создать новое изображение из модифицированных данных- ImageAssemble[]- и сохранить его.
11. Провести визуализацию встроенной информации (для всего рисунка):

четные блоки закрасить белым цветом,

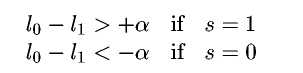
нечетные – черным.

Алгоритм "Langelaar".

1. Исходными элементами контейнера- оригинала служат блоки изображения, полученные в п.10. Для блока с индексами 60-N,45-N сформировать массив значений интенсивностей цвета RGB-цветовой модели - ImageData[\*\*\*,"Byte"].
2. Определить размерность полученного массива.
3. Используя функции SeedRandom[N] и RandomChoice[\*\*,\*\*] создать случайную маску (матрицу) с элементами "0" и "1", размером 8х8.
4. Сформировать вектор (список) коэффициентов для вычисления яркостной составляющей по RGB- компонентам:

***l*** = 0.299*R* + 0.587*G* + 0.114*B.*

1. Наложить (умножить) маску на блок с RGB- компонентами – таким образом будут выделены пиксели, принадлежащие к группе B1.
2. Выровнять полученную матрицу на первом уровне Flatten[\*\*\*,1].
3. Путем вычисления точечного (скалярного) произведения Dot[\*\*\*,\*\*\*] выровненной матрицы на вектор коэффициентов, получить вектор яркостной составляющей для группы B1.
4. Определить среднюю яркость ***l1***.
5. Сформировать инверсную маску - BitXor[\*\*\*,1] , с помощью которой будем выделять группу B0.
6. Повторить пункты п.22-25 с инверсной маской и получить значение средней яркости ***l0***.
7. Для величины порога встраивания α = 2, найти величину ***l1*** при которой будут выполняться условия встраивания как для "0", так и для "1":



* 1. С помощью вектора коэффициентов найти дельта-вектор {**R,G,B**}.
  2. Сформировать дельта-массив, содержащий 8х8 дельта-векторов ConstantArray[{R,G,B},{8,8}] и наложить на него маску группы B1.
  3. Провести встраивание бита в выбранный блок путем сложения исходного блока и дельта-массива, полученного в п.30.
  4. Для столбца исходной матрицы блоков контейнера-оригинала (см.10) провести встраивание последовательности из 30 нулей и 30 единиц.
  5. Создать новое изображение из модифицированных данных- ImageAssemble[]- и сохранить его.