**ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ**

**САНКТ-ПЕТЕРБУРГСКИЙ НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ, МЕХАНИКИ И ОПТИКИ**

**Факультет программной инженерии и компьютерной техники**

**Кафедра вычислительной техники**

**ЛАБОРАТОРНАЯ РАБОТА № 7**

**«Сокрытие данных в изображении-стегоконтейнере»**

**ПО ДИСЦИПЛИНЕ**

**«ЦИФРОВАЯ ОБРАБОТКА СИГНАЛОВ »**

Выполнили:

Третьяков Константин

Тищук Богдана

Группа: P3401

Преподаватель: Тропченко Андрей Александрович

Санкт-Петербург

2019

Задание №1.

1. Используя соответствующие функции пакета Matlab, напишите программу, реализующую алгоритм сокрытия данных в младшем разрядном срезе изображения. 2. Проведите эксперименты по сокрытию данных и их извлечению для различных изображений и типов скрываемых данных (бинарных изображений, данных типа .txt, .dat). 3. Определите количественные характеристики, определяющие соответствие между пустым и заполненным стегоконтйнером.

Задание №2

1. Самостоятельно реализуйте алгоритм сокрытия данных (текста) в псевдобелых и псевдо-черных пикселах изображения и с использованием средств

пакета Matlab напишите программу, реализующую данный метод.

2. Выполните п.п. 2 и 3 задания №1.

3. Сравните емкость стегоконтейнеров для обоих методов.

4. Самостоятельно провести подобные манипуляции с несколькими

различными изображениями. Привести вид исходного стегоконтейнера и

гистограмму яркости пикселов подлежащего сокрытию бинарного

изображения, стегоконтейнера после помещения в него скраваемой

информации (с гистограммой) и вид восстановленного бинарного

изображения (для каждого из изображений). Проанализировать недостатки

и достоинства методов и вариант модификации для помещения скрываемой

текстовой (символьной) информации

Алгоритм сокрытия данных:

function nnimg=fileto(im,fil,newimgname)

fid=fopen(fil); % Открываем дескриптор файла который хотим спрятать

file=fread(fid); % Присваиваем его сожержимое переменной file

status=fclose('all'); % Закрываем дескриптор

img=imread(im); % Содержимое файла присваиваем переменной img

nfile=dec2bin(file); % Переводим в двоичную систему содержимое файла

addnum=size(file,1); % Узнаем вес файла в байтах

sizze=size(dec2bin(addnum),2);

maxbytes=size(img,1)\*size(img,2)\*3/8; % Узнаем сколько байт информации можно спрятать в изображении

maxnum=ceil(log2(maxbytes));% Вычисляем длину максимального объема файла

a(1:maxnum)=dec2bin(0);

% Проверяем хватит ли нам заданного изображения

if addnum>maxbytes

nnimg='Недостаточно большое изображение'

else

a(maxnum-sizze+1:end)=dec2bin(addnum);

t=1;

nimg=dec2bin(img(:));% Переводим изображение в двоичную систему

for e=1:1:maxnum

nimg(e,8)=a(t); % Записываем в изображение длину скрываемого файла

t=t+1;

end

% Записываем в изображение содержимое файла

for i=1:1:size(nfile,1)

q=i\*8-7+maxnum

nimg(q,8)=nfile(i,1);

nimg(q+1,8)=nfile(i,2);

nimg(q+2,8)=nfile(i,3);

nimg(q+3,8)=nfile(i,4);

nimg(q+4,8)=nfile(i,5);

nimg(q+5,8)=nfile(i,6);

nimg(q+6,8)=nfile(i,7);

nimg(q+7,8)=nfile(i,8);

end

nimg=bin2dec(nimg); % Переводим код изображения в десятичное исчисление

nnimg=img;

nnimg(:)=nimg(:);

imwrite(nnimg,newimgname,'bmp'); % Запись изображения на диск

end

Алгоритм изъятия изображения

function nnfile=filefrom(im,filname)

img=imread(im); % Присваиваем содержимое изображения переменной

nimg=dec2bin(img(:));

maxbytes=size(img,1)\*size(img,2)\*3/8; % Вычисляем сколько байт информации можно спрятать в изображении

maxnum=ceil(log2(maxbytes)); % Вычисляем длину максимального объема файла

a(1:maxnum)=dec2bin(0);

t=1;

% Узнаем длину спрятаного файла

for e=1:1:maxnum

a(t)=nimg(e,8)

t=t+1

end

a=bin2dec(a);

nnfile=zeros(a,1);

nnfile(:)=255;

nnfile=uint8(nnfile);

nnfile=dec2bin(nnfile);

% Собираем файл из отдельных битов

for i=1:1:a

if i==1

q=maxnum+1;

else

q=i\*8-7+maxnum;

end

nnfile(i,1)=nimg(q,8);

nnfile(i,2)=nimg(q+1,8);

nnfile(i,3)=nimg(q+2,8);

nnfile(i,4)=nimg(q+3,8);

nnfile(i,5)=nimg(q+4,8);

nnfile(i,6)=nimg(q+5,8);

nnfile(i,7)=nimg(q+6,8);

nnfile(i,8)=nimg(q+7,8);

end

nnfile=bin2dec(nnfile);

fid=fopen(filname,'w'); % Открываем дескриптор нового файла

count=fwrite(fid,nnfile); % Записываем файл на диск

fclose(fid); %Закрываем дескриптор

% hidded FILE -> bit's row

 

**Рисунок 1 Пустой и заполненный контейнер**

Различие между изображениями состоит в битах яркости.

Алгоритм сокрытия данных (текста) в псевдобелых и псевдо-черных пикселах изображения

emb\_IMG\_name='./images/yard.bmp';

emb\_IMG = imread(emb\_IMG\_name);

emb\_IMG\_size\_y=size(emb\_IMG,1);

emb\_IMG\_size\_x=size(emb\_IMG,2);

emb\_num\_bits=ceil(log2(emb\_max\_Bytes));% bits

fid = fopen(emb\_FILE\_name);

emb\_FILE = fread(fid);

status =fclose(fid);

emb\_FILE\_size=size(emb\_FILE,1);% Bytes

if (emb\_FILE\_size\*8+emb\_num\_bits<=emb\_max\_Bytes\*8)

emb\_possible=true;

else

emb\_possible=false;

end;

emb\_DATA\_bits = uint8(im2bw([bitand(emb\_FILE,128),...

bitand(emb\_FILE,64), ...

bitand(emb\_FILE,32), ...

bitand(emb\_FILE,16), ...

bitand(emb\_FILE,8), ...

bitand(emb\_FILE,4), ...

bitand(emb\_FILE,2), ...

bitand(emb\_FILE,1)],0.6));

Встраиваем битовый ряд файла в стегоконтейнер.

emb\_new\_IMG(emb\_BITs\_ind)=bitshift(emb\_IMG(emb\_BITs\_ind),-1);

emb\_new\_IMG(emb\_BITs\_ind)=bitshift(emb\_new\_IMG(emb\_BITs\_ind),1)+emb\_BITs;

imwrite(emb\_new\_IMG,'stego\_murzik2.bmp','bmp');

imshow(emb\_new\_IMG);

 

**Вывод:**

Существует несколько способов зашифровать информацию в изображении. Для псевдобелых или псевдочерных ёмкость стегоконтейнера может быть меньше, за счет использования большего количества бит, особенно если изображение сильно затемнено или засветлено