




# 【图像加密】基于LSB算法实现图像隐写含Matlab源码

原创

Matlab科研工作室  于 2022-03-03 18:13:55 发布  206  收藏 2

分类专栏: [图像处理](#) 文章标签: [matlab](#) [算法](#) [开发语言](#)

版权声明: 本文为博主原创文章, 遵循 [CC 4.0 BY-SA](#) 版权协议, 转载请附上原文出处链接和本声明。

本文链接: [https://blog.csdn.net/qq\\_59747472/article/details/123260699](https://blog.csdn.net/qq_59747472/article/details/123260699)

版权



[图像处理](#) 专栏收录该内容

444 篇文章 76 订阅

订阅专栏

## 1 简介

随着多媒体技术的快速发展,信息安全变得越来越重要,如何保证信息不被窃取和篡改,已成为近年来信息安全领域的研究热点.介绍了信息隐藏技术的概念和发展,分析了最常用的LSB隐写算法原理,并且用MATLAB软件成功地实现了该隐写算法.最后指出了隐写术的发展趋势.

使用数学方程对 LSB 隐写算法进行抽象和建模。LSB 隐写算法既可以对图像中的像素做出  $\pm 1$  的改动,也可以不做任何改动,这依赖于待隐藏信息的特性和相应像素的 LSB 值。


的特性和相应像素的 LSB 值。在这里设  $I = \{x_i, i \in \Omega\}$ ,  $\Omega$  是一个索引集, 表示覆盖图像。集合  $\Omega$  划分为 3 个子集和  $\Lambda_1, \Lambda_2, \Lambda_3$ ,  $\Omega = \bigcup_{i=1}^3 \Lambda_i$  且  $\Lambda_i \cap \Lambda_j = \Phi (i \neq j)$ 。LSB 隐写图像的像素值集合是  $I_s = \{y_i, i \in \Omega\}$ , 其表达式为

$$y_i = \begin{cases} x_i + 1, & \text{if } i \in \Lambda_1, \\ x_i - 1, & \text{if } i \in \Lambda_2, \\ x_i, & \text{if } i \in \Lambda_3. \end{cases} \quad (1)$$

隐写分析的目标是估计  $I$  是否含有隐藏信息, 即判断  $\Lambda_1$  和  $\Lambda_2$  是否非空, 如果非空, 这两个集合又包含哪些元素。在达成目标之后, 就可能检测出覆盖图像中是否含有隐藏数据  $I$ 。当然, 在此过程中, 由于统计的不完整性和不确定性, 预先不知道数据隐藏键值, 隐写分析者会产生错误。错误的数量取决于  $\Lambda_i$  的集势。这又会引出另外一个问题: 在给定不可检测概率的情况下, 多少比特的信息可以隐写在  $I$  中(  $I$  的隐写容量)。

用以下符合实际的假设来计算隐写容量。  $x_i$  服从  $x_i \sim N(0, \sigma^2)$  的高斯分布。对每个  $i (i \in \Omega)$ , 隐写分析过程可以归结为一个多重假设检验问题。

$$H_j: y_i = x_i + d_i, i = 1, 2, 3, \quad (2)$$

$d_i = -1, 0, 1$  时可以表示为  $H_1: y_i \sim N(1, \sigma^2)$ ,  $H_2: y_i \sim N(-1, \sigma^2)$ ,  $H_3: y_i \sim N(0, \sigma^2)$ 。  天天Matlab

研究发现  $H_3$  的隐写图像和覆盖图像在统计上没有任何区别,因此只需检测  $H_1$  和  $H_2$  即可。如果  $H_1$  和  $H_2$  之一检验成功,就说明图像含有隐藏信息。如果覆盖图像中的像素比特和 LSB 中相应的像素比特相等,就可以肯定此图像没有包含任何隐藏信息。

再假设一个数据比特位值是 1 的概率是  $p^d$ ,  $0 < p^d < 1$ ; LSB 比特位值是 1 的概率是  $p^l$ ,  $0 < p^l < 1$ ; 隐藏数据的比特位和 LSB 比特位的取值是相互独立的,那么他们的联合概率为

$$P(d_i = 1, l_i = 1, \dots, d_i | \Omega, l_i | \Omega) = (p^d p^l)^{|\Omega|}, \quad (3)$$

$|\Omega|$  表示集的势。当  $|\Omega|$  无穷大时,联合概率趋近于 0。对每一个像素进行上述 3 种假设检验判断,并使用最小错误概率准则作为价值函数。最小错误概率检测也表现为最大后验概率(MAP)检测。真验概率表示为

$$H = \arg \max_j P(H_j) P(y_i | H_j). \quad (4)$$

因为  $y_i$  服从高斯分布,所以真验概率表达式

$$H = \arg \max_j P(H_j) \exp \frac{-(y_i - d_j)^2}{2\sigma^2} \quad (5)$$

与  $H_1, H_2, H_3$  相对应的  $d_j$  分别是 1、-1 和 0。这样就能够估计出含有隐藏信息的像素位置,也可以说像素位置是由  $H_1$  和  $H_2$  检测到的。当然,在此过程中也会产生错误,错误概率为  $p_{kj} = P(\text{decide } H_j | H_k \text{ true}), j, k = 1, 2, 3$ 。 $p_{kj}$  的值与图像的差异和分析方法有关。

天天Matlab

## 2 部分代码

```
function varargout = LSB(varargin)
% LSB MATLAB code for LSB.fig
%
%   LSB, by itself, creates a new LSB or raises the existing
%   singleton*.
%
%
%   H = LSB returns the handle to a new LSB or the handle to
%   the existing singleton*.
%
```

```

%     LSB('CALLBACK',hObject,eventData,handles,...) calls the local
%     function named CALLBACK in LSB.M with the given input arguments.
%
%     LSB('Property','Value',...) creates a new LSB or raises the
%     existing singleton*. Starting from the left, property value pairs are
%     applied to the GUI before LSB_OpeningFcn gets called. An
%     unrecognized property name or invalid value makes property application
%     stop. All inputs are passed to LSB_OpeningFcn via varargin.
%
%     *See GUI Options on GUIDE's Tools menu. Choose "GUI allows only one
%     instance to run (singleton)".
%
% See also: GUIDE, GUIDATA, GUIHANDLES
% Edit the above text to modify the response to help LSB
% Last Modified by GUIDE v2.5 22-Apr-2021 13:43:30
% Begin initialization code - DO NOT EDIT

gui_Singleton = 1;

gui_State = struct('gui_Name',       mfilename, ...
                  'gui_Singleton',  gui_Singleton, ...
                  'gui_OpeningFcn', @LSB_OpeningFcn, ...
                  'gui_OutputFcn',  @LSB_OutputFcn, ...
                  'gui_LayoutFcn',  [] , ...
                  'gui_Callback',   []);

if nargin && ischar(varargin{1})
    gui_State.gui_Callback = str2func(varargin{1});
end

if nargout
    [varargout{1:nargout}] = gui_mainfcn(gui_State, varargin{:});
else
    gui_mainfcn(gui_State, varargin{:});
end
% End initialization code - DO NOT EDIT

% --- Executes just before LSB is made visible.

```

```

function LSB_OpeningFcn(hObject, eventdata, handles, varargin)

% This function has no output args, see OutputFcn.

% hObject    handle to figure

% eventdata  reserved - to be defined in a future version of MATLAB

% handles    structure with handles and user data (see GUIDATA)

% varargin   command line arguments to LSB (see VARARGIN)

% Choose default command line output for LSB

handles.output = hObject;

% Update handles structure

guidata(hObject, handles);

% UIWAIT makes LSB wait for user response (see UIRESUME)

% uiwait(handles.figure1);

% --- Outputs from this function are returned to the command line.

function varargout = LSB_OutputFcn(hObject, eventdata, handles)

% varargout  cell array for returning output args (see VARARGOUT);

% hObject    handle to figure

% eventdata  reserved - to be defined in a future version of MATLAB

% handles    structure with handles and user data (see GUIDATA)

% Get default command line output from handles structure

varargout{1} = handles.output;

% --- Executes on button press in pushbutton1.

function pushbutton1_Callback(hObject, eventdata, handles)

% hObject    handle to pushbutton1 (see GCBO)

% eventdata  reserved - to be defined in a future version of MATLAB

% handles    structure with handles and user data (see GUIDATA)

[filename,filepath]=uigetfile('.bmp','选择原始bmp图片文件');

if(filename==0)

return;

end

global originalfile x

originalfile= strcat(filepath,filename);

x=imread(originalfile);

```

```

subplot(handles.axes1);
imshow(originalfile);
% --- Executes on button press in pushbutton2.
function pushbutton2_Callback(hObject, eventdata, handles)
% hObject    handle to pushbutton2 (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)
[filename,filepath]=uigetfile('.bmp','选择嵌入隐密文件');
if(filename==0)
return;
end
global secretfile y
secretfile= strcat(filepath,filename);
y=imread(secretfile);
subplot(handles.axes2);
imshow(secretfile);

% --- Executes on button press in pushbutton3.
function pushbutton3_Callback(hObject, eventdata, handles)
% hObject    handle to pushbutton3 (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)

```

### 3 仿真结果



## 4 参考文献

[1]李友, 张定会, 许赛赛,等. 基于图像的LSB隐写算法的实现[J]. 计算机安全, 2010(7):3.

博主简介: 擅长智能优化算法、神经网络预测、信号处理、元胞自动机、图像处理、路径规划、无人机等多种领域的Matlab仿真, 相关matlab代码问题可私信交流。

部分理论引用网络文献, 若有侵权联系博主删除。