

【图像隐写】基于matlab GUI LSB图像隐写【含Matlab源码】 713期】

原创

海神之光 于 2021-04-06 15:54:50 发布 528 收藏 5

分类专栏: [Matlab完整代码](#) [代码事宜私信博主](#) [Matlab图像处理](#)

[代码事宜私信博主](#)

本文链接: <https://blog.csdn.net/TIQCmatlab/article/details/115462825>

版权

MATLAB
王者助手

紫极神光

[Matlab完整代码](#) 同时被 3 个专栏收录

1989 篇文章 2771 订阅

订阅专栏

MATLAB
王者助手

紫极神光

[代码事宜私信博主](#)

1971 篇文章 398 订阅

订阅专栏

MATLAB
王者助手

紫极神光

[Matlab图像处理](#)

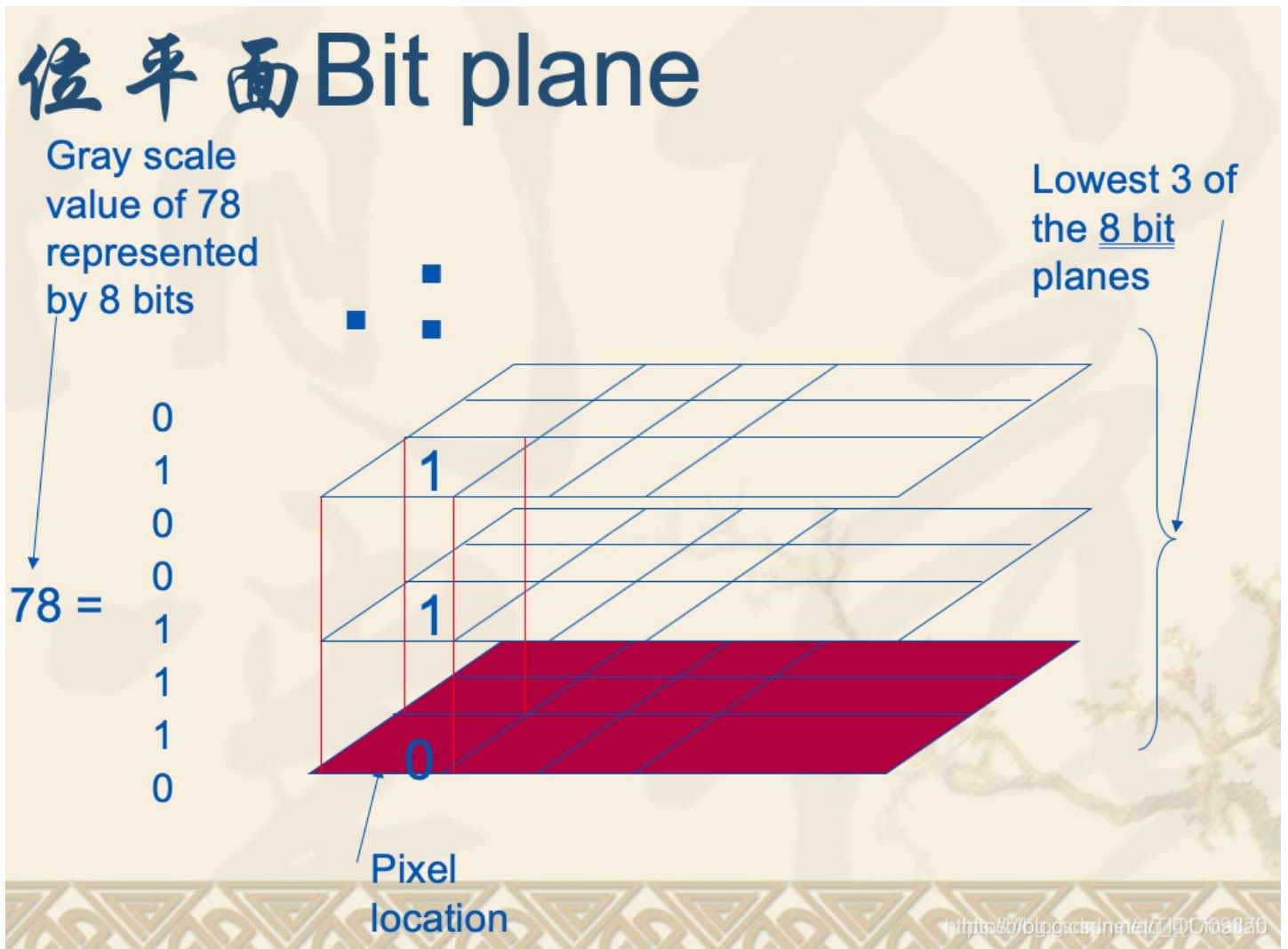
582 篇文章 123 订阅

订阅专栏

一、简介

1 LSB算法简介

LSB全称为 Least Significant Bit（最低有效位），是一种简单而有效的数据隐藏技术。LSB隐写的基本方法是用欲嵌入的秘密信息取代载体图像的最低比特位，原来的图像的高位平面与代表秘密信息的最低平面组成含隐蔽信息的新图像。



灰度化的图像为单通道格式存储像素，每个像素值在0~255内，而像素的位平面则是对应二进制的像素的各个位。以上图为例，某个像素的值为78，其二进制01001110，从左到右位权依次降低，最左边为最高有效位（MSB，其位权为 2^7 ）

），最右边位最低有效位（LSB，位权为 2^0 ）

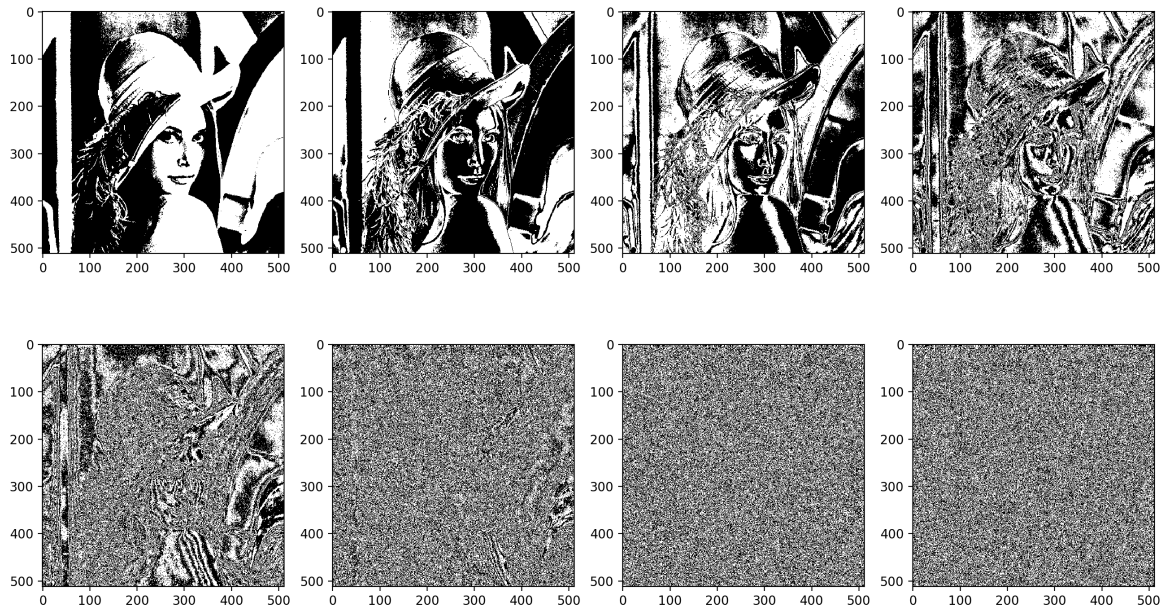
）。把每个像素的相同位抽取出来组成一个新的平面，就是所谓的图的位平面。而LSB隐写算法，如其名字，是在LSB也就是最低位平面进行信息嵌入/隐藏。

需要注意的一点是，LSB嵌入的时候，载体图像格式应该为灰度图格式

以著名的Lena图为例，一下是灰度图Lena原图：



下面是其各个位平面图，从左到右、从上到下位平面依次降低：



可以看到，位平面越高包含的原图像信息越多，对图像的灰度值贡献越大，并且相邻比特的相关性也越强，反之则相反。LSB最低位平面基本上不包含图像信息了，类似随机的噪点/噪声，因此，可以在此处填入水印/秘密信息。

嵌入示意图如下：

Original Image Bytes	Message to hide	Embedded Image Bytes
10010010	0	10010010
01010011	1	01010011
10011011	0	10011010
11010011	0	11010010
10001010	0	10001010
00000010	1	00000011
01110010	0	01110010
00101011	0	00101010

选取不同位平面嵌入时，LSB算法的保真度：

LSB算法的保真度

$$PSNR = -10 \log_{10} \left(\frac{1}{255^2 MN} \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} [d(m, n)]^2 \right)$$

- 最低1位

$$1/2, MN/2, PSNR = -10 \log_{10}(1/255/255/2) = 51.14 \text{dB}$$

- 最低2位

$$3MN/2, PSNR = 46.37$$

- 最低3位

$$7MN/2, PSNR = 42.69$$

- 最低4位, PSNR=39.38; 最低5位, PSNR=36.23.

2 算法原理

通俗来讲我们看到的图片都是由一个个小的像素点来构成的，所有像素点摆在一起，构成一个大方块，这个大方块就是我们所见的图像。灰度图像（也就是我们平时所说的黑白图像）是由一层像素点组成的，而彩色图像是由三层这样的灰度图像组成的。这里拿灰度图像举例，我们之所以能在图像上看到黑色和白色，是因为每个像素点的像素值不同。0表示纯黑，255表示纯白，灰色就是由这两个数字之间的值构成。越靠近0越黑，越靠近255越白。那为什么是0和255呢？因为计算机是二进制，它会用8个比特来表示一个像素点（也可以用更多的比特，这样图像的颜色分级就越多，同时图像也会占用更大的空间，但是普通人的眼睛并不能辨认这么多的颜色，除非你异于常人），所以最大值是255，最小是0。lsb就是基于2进制这一特点来隐藏信息的，因为人眼并不是很精密的颜色或亮度的感知器，所以把像素灰度上下微调1是不会被人眼察觉的，也就是修改8位二进制码中最小的一位。当我们把图片每个像素的最后一位按照我们的想法改变，使他表现为我们想要的信息，但用户却不能看出，也不会影响图片的内容。这就是lsb数字水印。

3 LSB算法的基本特点：

LSB是一种大容量的数据隐藏算法

LSB的鲁棒性相对较差（当stego图像遇到信号处理，比如：加噪声，有损压缩等，在提取嵌入信息时会丢失）

4 常见LSB算法的嵌入方法：

秘密信息在最低位平面连续嵌入至结束，余下部分不作任何处理（典型软件MandelSteg）

秘密信息在最低位平面连续嵌入至结束，余下部分随机化处理（也称沙化处理，典型软件PGMStealth）

秘密信息在最低位平面和次低位平面连续嵌入，并且是同时嵌入最低位平面和次低位平面

秘密信息在最低位平面嵌入，等最低位平面嵌入完全嵌入之后，再嵌入次低位平面

秘密信息在最低位平面随机嵌入

以上五种方式，当嵌入容量不同时，鲁棒性不同

二、源代码

```
function varargout = shuiyin(varargin)
% SHUIYIN MATLAB code for shuiyin.fig
%     SHUIYIN, by itself, creates a new SHUIYIN or raises the existing
%     singleton*.
%
%     H = SHUIYIN returns the handle to a new SHUIYIN or the handle to
%     the existing singleton*.
%
%     SHUIYIN('CALLBACK',hObject,eventData,handles,...) calls the local
%     function named CALLBACK in SHUIYIN.M with the given input arguments.
%
%     SHUIYIN('Property','Value',...) creates a new SHUIYIN or raises the
%     existing singleton*. Starting from the left, property value pairs are
%     applied to the GUI before shuiyin_OpeningFcn gets called. An
%     unrecognized property name or invalid value makes property application
%     stop. All inputs are passed to shuiyin_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 shuiyin

% Last Modified by GUIDE v2.5 12-Dec-2019 21:53:08

% Begin initialization code - DO NOT EDIT
gui_Singleton = 1;
gui_State = struct('gui_Name',       mfilename, ...
                  'gui_Singleton',  gui_Singleton, ...
                  'gui_OpeningFcn', @shuiyin_OpeningFcn, ...
                  'gui_OutputFcn',  @shuiyin_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 shuiyin is made visible.
function shuiyin_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 shuiyin (see VARARGIN)

% Choose default command line output for shuiyin
handles.output = hObject;

% Update handles structure
guidata(hObject, handles);

% UIWAIT makes shuiyin wait for user response (see UIRESUME)
% uiwait(handles.figure1);

% --- Outputs from this function are returned to the command line.
function varargout = shuiyin_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)
[filename1, pathname1] = uigetfile('*.jpg', '选择水印图片');
global pathfile1;
global pathfre1;
pathfile1=fullfile(pathname1, filename1);
pathfre1 = pathname1;
I0=rgb2gray(imread(pathfile1));
mrows=256;
ncols=256;
input=imresize(I0,[mrows ncols]);
axes(handles.axes1);axis off
imshow(input);
title('原始图像');

handles.input=input;
guidata(hObject, handles);
% --- 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)
[filename2, pathname2] = uigetfile('*.jpg', '选择水印图片');
global pathfile2;
global pathfre2;
pathfile2=fullfile(pathname2, filename2);
pathfre2 = pathname2;
M0=rgb2gray(imread(pathfile2));

```

```

no=rgb2g;uy=im2bw(palm112);
axes(handles.axes2);axis off
mrows=256;
ncols=256;
water=imresize(M0,[mrows ncols]);
imshow(water);
title('水印');
f=water;
I=handles.input;
for k=1:92 %迭代次数为T=192次
    for x=1:256 %图像的大小为256×256
        for y=1:256
            x1=x+y;
            y1=x+2*y;
            if x1>256
                x1=mod(x1,256);
            end
            if y1>256
                y1=mod(y1,256);
            end
            if x1==0
                x1=256;
            end
            if y1==0
                y1=256;
            end
            f_ar(x1,y1)=f(x,y);
        end
    end
    f=f_ar;
end

%*****嵌入水印*****
I_1_0=bitset(I,1,0);
J=bitset(I_1_0,1,f_ar);

%*****提取水印*****
f_get=bitget(J,1);

%*****解密*****
fg=mat2gray(double(f_get)); %double型必须用mat2gray进行显示

for k=1:100 %迭代次数为192次
    for x=1:256 %图像的大小为256×256
        for y=1:256
            x1=x+y;
            y1=x+2*y;
            if x1>256
                x1=mod(x1,256);
            end
            if y1>256
                y1=mod(y1,256);
            end
            if x1==0
                x1=256;
            end
            if y1==0
                y1=256;
            end
            f_re(x1,y1)=fg(x,y);
        end
    end
end

```

end
end

三、运行结果

shuiyin

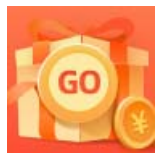
基于LSB的数字水印嵌入提取 相关系数NC=0.9898

<p>原始图像</p> 	<p>水印</p> 	<p>置乱后的水印图像</p> 
<p>选择任意图像 嵌入水印后图像</p> 	<p>选择水印图像 提取后的水印图像</p> 	<p>加密水印 解密后的水印图像</p> 
<p>含水印的图像</p>	<p>提取水印</p>	<p>解密后的水印图像</p>

 58%

四、备注

版本：2014a



[创作打卡挑战赛](#) >
[赢取流量/现金/CSDN周边激励大奖](#)