python图片隐写_基于python的LSB隐写与分析

weixin_39988930 于 2020-12-09 19:46:10 发布 134 收藏
文章标签： python图片隐写

隐写

效果

隐写前

隐写后

灰度值直方图差别

部分源码

```
def dec_to_bin(dec):

return '{:08b}'.format(dec)

def bin_to_dec(binary_code):

dec = 0

for i in range(len(binary_code) - 1):

dec = dec + int(binary_code[i]) * int(pow(2, 7 - i))

return dec

# 文件信息转二进制流

def read_data_file(path):

fp = open(path, "rb")

stream = ""

s = fp.read()

for i in range(len(s)):

tmp = bin(s[i]).zfill(8)

stream = stream + tmp.replace('0b', '')

fp.close()

return stream

def lsb(image, data_stream, random_index):

for i in range(len(stream)):
```

```
x = random_index[i] % image.shape[0]

y = int(random_index[i] / image.shape[0])

value = image[x, y]

if value % 2 != stream[i]:

if value % 2 == 1:

image[x, y] = value - 1

else:

image[x, y] = value + 1

return image
```

分析

效果

由于二次隐写的随机性，分析图片存在误差，但能够看出是否被隐写

原图

隐写后

部分源码

```
# 进行二次隐写

def random_steg(image, rate):

pixel_len = image.shape[0] * image.shape[1]

random_ls = random.sample(range(0, pixel_len), int(pixel_len * rate))

random_ls.sort()

for i in random_ls:

k = random.randint(0, 1)

x = i % image.shape[0]

y = int(i / image.shape[0])

value = image[x, y]

if not value % 2 == k:

if value % 2 == 1:

image[x, y] = value - 1

else:
```

```
image[x, y] = value + 1
return image
# 获取灰度值
def get_gary_value(my_img):
pixel_value = []
gary_index = []
for i in range(256):
pixel_value.append(0)
gary_index.append(i)
for i in range(my_img.shape[0]):
for j in range(my_img.shape[1]):
pixel_value[my_img[i][j]] = pixel_value[my_img[i][j]] + 1
return pixel_value, gary_index
# 计算F1,F2
def calculate_f1f2(values):
f1 = 0
f2 = 0
for i in range(128):
tmp = abs(values[2 * i + 1] - values[2 * i])
f1 += tmp
for j in range(127):
tmp = abs(values[2 * j + 2] - values[2 * j + 1])
f2 += tmp
f2 += abs(values[0] - values[255])
return f1, f2
# 分析函数
def analysis(path):
img = cv2.imread(path, 0)
# 二次随机隐写
F1 = []
F2 = []
```

```
index = []
for k in range(11):
rate = k / 10
index.append(rate)
new_img = random_steg(img, rate)
new_count, new_index = get_gary_value(new_img)
f_1, f_2 = calculate_f1f2(new_count)
F1.append(f_1)
F2.append(f_2)
draw(F1, F2, index)
```

相关链接

项目链接

参考文档