Lecture 3. Real SG and SG-SS .

1. Real SG-LSB.
(Programs of these SG are available on Internet).

1.1 Jsteg. As CO is used RGB color image in Format JPEG.

Embedding is performed in the LSB of DCT coefficients (with exception for zero and one valued coefficients) following pseudorandom way that is determined by stegokey (password) .

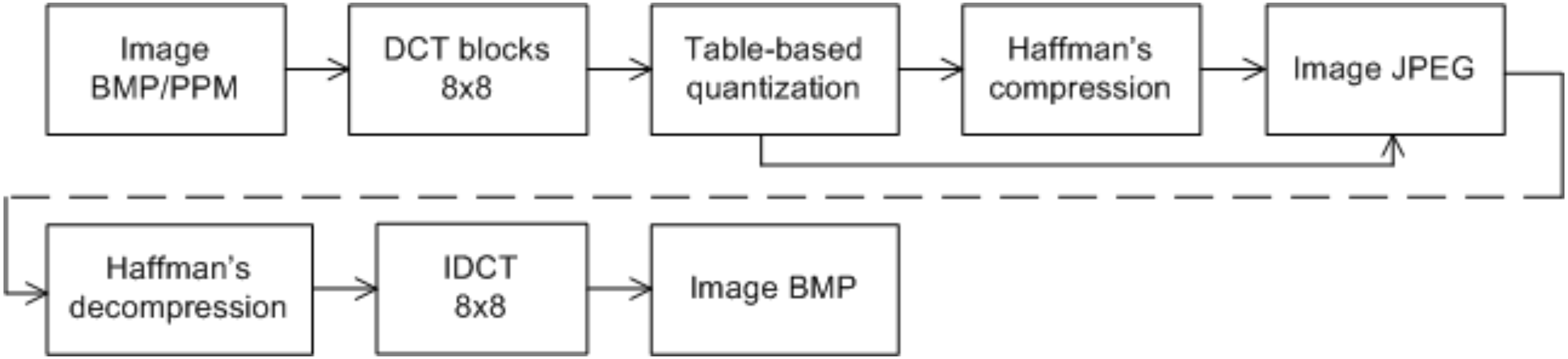


Fig 1. Short description of JPEG standard.

Jsteg cannot be detected by visual attack but it is detectable with the use of χ^2 – statistic and sample pair analysis.

1.2 Outguess. As CM is used RGB color image in Format JPEG . Algorithm executes operation system FreeBSD on C++. Algorithm is working with *command row* and requires passwords (stegokeys) for embedding and extraction of information.The embedding algorithm is design especially to be resistant against attack on statistic χ^2.

Embedding procedure is executed in two rounds : the first one is determined by pseudorandom stegokey (password) as it was done in Jsteg, while the second one changes only DCT coefficients which were unchanged in the first round for the purpose to make close to one another histograms of CM and SG providing in such a way a resistance to χ^2-attack.
However a detection of Outguess occurs still possible if the "nonhomogeneity" in blocks 8x8 of SG is found and compared with nonhomogeneity of CO estimation that in turn is found by special transform of SG (See detection of SG-SS and blind stegoanalysis in the sequel).

1.3 F5. As CO is used RGB color image in Format JPEG. However in comparison with Jsteg and Outguess, this is not “clear “ SG-LSB. The feature of F5 is to minimize the number of changes bit in CO given the number of information bits.

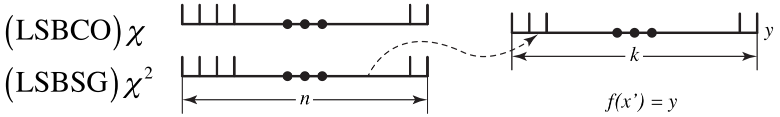
Example. $x_1, x_2 \in \{0,1\}$ are bits of secret information. Conventional LSB requires to change also 2 bits of CO.Modified LSB embedding (where a_1,a_2,a_3 –are bits of CO) will be the following:

$x_1 = a_1 \oplus a_3$, $x_2 = a_2 \oplus a_3$ => change nothing,
$x_1 \neq a_1 \oplus a_3$, $x_2 = a_2 \oplus a_3$ => change a_1,
$x_1 = a_1 \oplus a_3$, $x_2 \neq a_2 \oplus a_3$ => change a_2,
$x_1 \neq a_1 \oplus a_3$, $x_2 \neq a_2 \oplus a_3$ => change a_3.

We can see that in the all cases it is changed at most one bit. Given a_1,a_2,a_3 x_1, x_2. are recovered uniquely.

Generalization of LSB-changing minimization, that has been used in F5

Definition. A covering function $\operatorname{cov}(\rho, n, k)$ this is mapping f: $F_2^n \rightarrow F_2^k$, that, $\forall x \in F_2^n$, $y \in F_2^k, \exists\, x' \in F_2^n$, whereas $d(x, x') \leq \rho$ and $f(x') = y$, where $d(.)$ – is the Hamming distance.



$\frac{k}{n}$ - relative rate (minimum)

$\frac{\rho}{n}$ - relative radius of covering (minimum)

Using $\operatorname{cov}(\rho, n, k)$ there may be embedded *k* bits into *n* pixels producing at most ρ changing.
In order to design covering scheme linear codes can be used.
Particular case. Cov(1.2^k – 1.k) based on Hamming codes as follows:

- compute $z = x \cdot \mathbf{H}^T + y$, where **H** – is check matrix of Hamming code.
- find the i-th column of matrix H coinciding with z,
- change x_i in vector **x** = (x_1, …, x_i,…, x_k) to opposite.

Extension to arbitrary linear codes see in [56].
The maximum number of bits *m* that can be embedded into the binary block of the length *n* using at most *R* changes satisfied the following bound [56]: $m \leq n\, \mathbf{H}\left(\frac{R}{n}\right)$.

Sphere packing bound: $\sum_{i=0}^{\rho} \binom{n}{i} \geq 2^k$

The best known COV functions:
(3, 31, 12), (3, 127, 18), (3, 511, 24), (3, 22, 10); etc. (see [52])
These constructions are used as a rule with some nonlinear codes [56].

Algorithm F5 is implemented by JavaScript and uses an extension of approach presented before : matrix *(1,n,k)*-code, where n –is the number of pixels that can be changed , *k* – the number of message bits, that provides 1 as maximum number of changing pixels for embedding *k* message bits.

Parameters of F5: $n = 2^k - 1$, - the lengths of blocks, $1/2^k$ - density of embedding, $k/n = k/(2^k - 1)$ - embedding rate.

Embeddable bits are determined by pseudorandom sequence that is controlled in turn by stegokey (password) .Decreasing of density of embedding improves undetectability of SG.

However SG-F5 can be detected by comparison between histograms for chosen DCT coefficients of SG and estimation of CO:

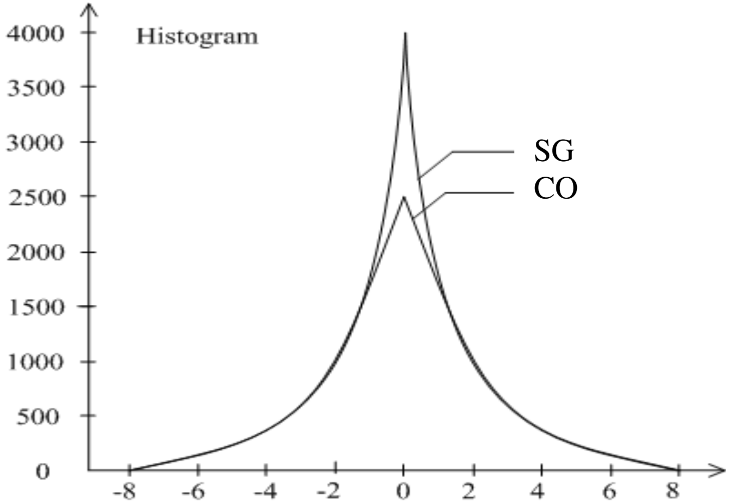


Fig. 2. Histograms of DCT-(2,1) coefficients for F5 and estimation of original CO.

We can see that SG can be detected.

Much better results can be obtained by blind steganalysis (see theme 7 in the sequel).

2. SG-SS.

All LSB-based SG are vulnerable to removal attack while keeping good quality of CO. This attack can be realized by a randomization of LSB either in time or in frequency domains.

In order to protect SG against this attack it is necessary to use *spread spectrum* (SS)- based SG:

$$C_W(n) = C(n) + \alpha(-1)^b \pi(n), n = 1,2...N, \tag{1}$$

where *α* – embedding coefficient , *π(n)* – pseudorandom (±1) or Gaussian sequence (PRS), that is generated with the use of secret stegokey, *N* – is the length of PRS, corresponding to embedding of one message bit (*b=1* or *0*).

Extraction of message bit (decoding) with unknown CO (*"blind" decoder*) is:

$$\sum_{n=1}^{N} (C'_W(n) - m_c)\pi(n) \left\langle \begin{matrix} > 0 \rightarrow b = 0 \\ , \\ < 0 \rightarrow b = 1 \end{matrix} \right. \tag{2}$$

where it is assumed that attack is executed by additive noise *ε(n)* as follows:

$$C'_W(n) = C_W(n) + \varepsilon(n), n = 1,2...N, \quad E\{\varepsilon(n)\} = 0 \tag{3}$$

and m_c = *E{C(n)}.*

Since *π(n)* is unknown for an attacker it results in an impossibility to remove the embedded message under the conditions of large enough *N* and small distortion of *C(n)*.

In fact, let us consider the probability of message bit error for legitimate user who knows ±1 sequence *π(n), n = 1, 2…N.*

$$p(1/0) = P\{\Lambda \le 0/b = 0\}, p(0/1) = P\{\Lambda > 0/b = 1\} \tag{4}$$

$$\Lambda = \sum_{n=1}^{N}((C_w^{'}(n) - m_c)\pi(n))$$

As *N → ∞, Λ ~ N(E(Λ),Var(Λ))* (See *Central Limit Theorem* (CLT) of probability theory)

$$E\{\Lambda\} = E\{\sum_{n=1}^{N}(C(n) - m_c + \alpha(-1)^b\pi(n) + \varepsilon(n))\pi(n)\} = \alpha(-1)^b N \tag{5}$$

$$Var\{\Lambda\} = \sum_{n=1}^{N} E\{((C(n) - m_c + \varepsilon(n))\pi(n))^2\} =$$

$$= NE\{(C(n) - m_c)^2 + 2\varepsilon(n)\pi(n)(C(n) - m_c) + \varepsilon^2(n)\}) = N(\sigma_c^2 + \sigma_\varepsilon^2), \tag{6}$$

where $\sigma_c^2 = Var\{C(n)\}, \sigma_\varepsilon^2 = Var\{\varepsilon(n)\}$.

If we let m_c = *0* in (3), then we get instead of (6):

$$Var\{\tilde{\Lambda}\} = N(E\{C^2(n)\} + \sigma_\varepsilon^2) = N(Var\{C(n)\} + m_c^2 + \sigma_\varepsilon^2) = N(\sigma_c^2 + m_c^2 + \sigma_\varepsilon^2) \tag{7}$$

$Var\{\tilde{\Lambda}\} \ge Var\{\Lambda\}$, that is much worse than decision rule (2).

Consider firstly the case *b = 0.* Ten we get :

$$p(1/0) = p\{\Lambda \le 0/b = 0\} = Q(E\{\Lambda\}/\sqrt{Var\{\Lambda\}}), \qquad (8)$$

$$Q(x) = \frac{1}{\sqrt{2\pi}}\int_x^\infty e^{-t^2/2}dt.$$

Substituting (5) and (6) into (8), gives:

$$p(1/0) = Q(\frac{\alpha}{\sqrt{N(\sigma_c^2 + \sigma_\varepsilon^2)}}) \qquad (9)$$

(It is easily to verify that similar relation be true also for the case *b = 1,* that results in the fact that *p(1/0) = p(0/1) = p*).

Introduce the notations :

$$\eta_w = \frac{\sigma_c^2}{\alpha^2}$$ - (*signal –to –noise ratio after WM embedding*), (10)

$$\eta_a = \frac{\sigma_c^2}{\alpha^2 + \sigma_\varepsilon^2}$$ - (*signal-to- noise ratio after embedding and attack*). (11)

Substituting (10) and (11) into (9), we get

$$p = Q(\sqrt{N\eta_a/(\eta_a\eta_w + \eta_w - \eta_a)}) \qquad (12)$$

Typical case is $\eta_w \ge \eta_a >> 1.$

Then we obtain for (12) the following approximation

$$p \approx Q(\sqrt{N/\eta_w}) \qquad (13)$$

Now we consider the case of *informed decoder* , when decoding rule is the following:

$$\Lambda' \left\langle \begin{array}{l} \geq 0 \rightarrow b = 0 \\ < 0 \rightarrow b = 1, \end{array} \right. \qquad (14)$$

$$\Lambda' = \sum_{n=1}^{N} (C_w^{'}(n) - C(n))\pi(n) \qquad (15)$$

By applying of CLT we get :

$$p' = P\{\Lambda' < 0 \,|\, b = 0\} = Q(\frac{E\{\Lambda'\}}{\sqrt{Var\{\Lambda'\}}}) \qquad (16)$$

$$E\{\Lambda'\} = E\{\sum_{n=1}^{N} (C(n) + \alpha\pi(n) + \varepsilon(n) - C(n))\pi(n)\} = \alpha N \qquad (17)$$

$$Var\{\Lambda'\} = Var\{\sum_{n=1}^{N} \varepsilon(n)\pi(n)\} = NVar\{\varepsilon(n)\}Var\{\pi(n)\} = N\sigma_\varepsilon^2 \qquad (18)$$

Substituting (17), (18) into (16) and using the notations (10), (11), we obtain

$$p' = Q(\frac{\alpha\sqrt{N}}{\sigma_\varepsilon}) = Q(\sqrt{\frac{N}{(\eta - 1)}}), \qquad (19)$$

where$\eta = \eta_w / \eta_a$.

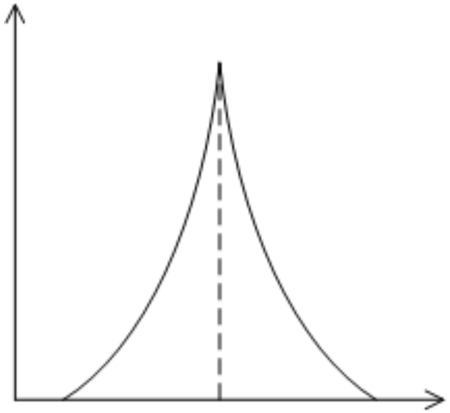
After comparison of *p* by (13) and *p’* by (19) we can see that $p \geq p'$.
In fact, given $p = p'$, but with different *N* (blind decoder) and *N*` (informed decoder) we get the relation :

$$N'/(\eta - 1) = N/\eta_w \Rightarrow \mathrm{N/N'} = \eta_w/(\eta - 1) \qquad (20)$$

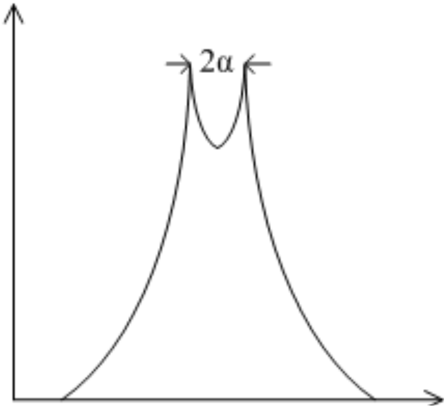
Example. Let $\eta_w = 120$, $\eta_a = 100$. Then $N/N' = 600$.
This means that for “blind” decoder the embedding rate is smaller than for informed decoder by factor 600 !
In order to decrease this difference (when CO is unknown at the decoder) it is used so called *informed encoder* that differs from encoder by (2). However it can results in a better detection of SS-based SG and therefore this version of encoder is commonly to use with WM system but not with SG (see next Themes).

Detecting of SG-SS

1. On histogram (first order statistic)[1]

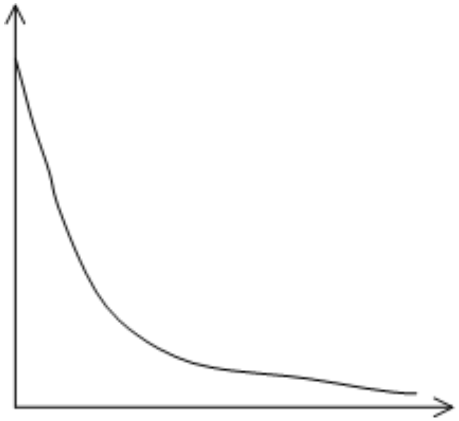


a) CO

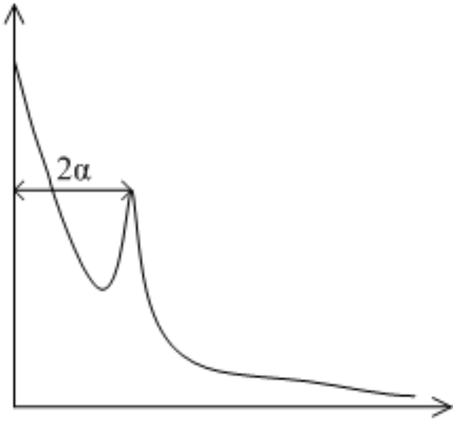


b) SG-SS, twin peak

2. On sample pair analysis (second order statistic)
(Using histograms of absolute values of luminance differences for adjacent pixels |C(n+1)-C(n)|)



a) CO



b) SG-SS

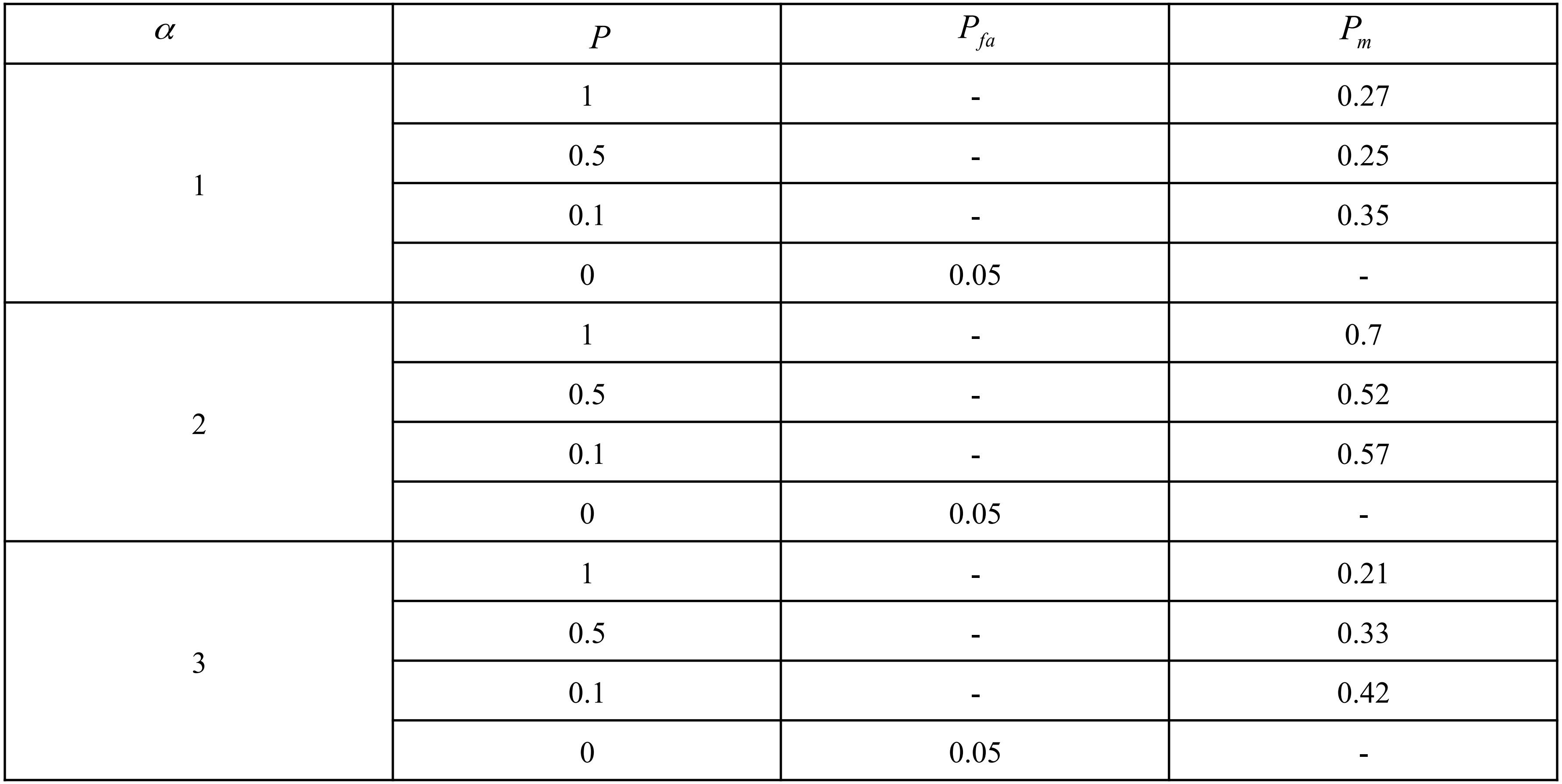
3. Using X^2 criterion (See Lecture 2)

α	P	P_{fa}	P_m
1	1	-	0.08
	0.5	-	0.02
	0.1	-	0.03
	0	0.15	-
2	1	-	0.17
	0.5	-	0.09
	0.1	-	0,2
	0	0.15	-
3	1	-	0.10
	0.5	-	0.07
	0.1	-	0.3
	0	0.15	-

We can conclude that this method works for the images of high quality (without digital noise).

The best results we get for the probability of embedding P = 0.5.

4. Sample pair analysis attack (see Lecture 2).



α	P	P_{fa}	P_m
1	1	-	0.27
	0.5	-	0.25
	0.1	-	0.35
	0	0.05	-
2	1	-	0.7
	0.5	-	0.52
	0.1	-	0.57
	0	0.05	-
3	1	-	0.21
	0.5	-	0.33
	0.1	-	0.42
	0	0.05	-

We can see that this method works not good but it can be used in a combination with other methods.

5. Method based on the calculation of zeros in histogram

The following reasonable claiming can be formulated:

The number of zeros is always less with SG than with CO.

α	P	P_{fa}	P_m
1	1	-	0.13
	0.5	-	0.5
	0.1	-	0.1
	0	0.1	-
2	1	-	0.12
	0.5	-	0.07
	0.1	-	0.1
	0	0.1	-
3	1	-	0.15
	0.5	-	0.06
	0.1	-	0.1
	0	0.1	-

We can see that this method works but not for all images and it is the best also if P=0.5.

6. Using statistic Γ *of square differences for adjacent pixels :*

$$\Gamma = \frac{1}{2N_0\sigma_c^2}\sum_{n=1}^{N_0}\left(C_W(n+1)-C_W(n)\right)^2, \qquad (21)$$

where $\sigma_c^2 = \frac{1}{N_0}\sum_{n=1}^{N_0} C_W^2(n) \approx E\{C_w^2(n)\} \approx E\{C^2(n)\}.$

N_0 – is the total number of image pixels .

Algorithm of SG-SS detecting:

$\Gamma > \gamma_0$ => SG is present ,

$\Gamma \le \gamma_0$ => SG is absent . (22)

In fact we have in the case of CO :

$$E\left\{\Gamma\right\} = \frac{N_0}{2N_0\sigma_c^2} E\left\{C^2(n+1)+C^2(n)-2C(n+1)C(n)\right\} = 1 - R_c(n,n+1), \qquad (23)$$

where $R_c(n,n+1)$ –relative correlation coefficients between luminances of adjacent pixels.

Remark. Embedding procedure by (1) does not provide undetectability of SG if $Var\{C(n)\} = \sigma_c^2$ is known because then:

$$Var\{C_w(n)\} = \sigma_c^2 + \alpha^2 > Var\{C(n)\}$$

In order to remove this attack it is necessary to execute the embedding by modified algorithm

$$C_w(n) = \beta C(n) + \alpha(-1)^b \pi(n), n = 1, 2...N, \qquad (24)$$

where $\beta = \sqrt{1 - \frac{\alpha^2}{\sigma_c^2}}$.

Then $Var\{C_w(n)\} = Var\{C(n)\} = \sigma_c^2$ (it can be verified easily).

$$E\left\{\Gamma'\right\} = \frac{N_0}{2N_0\sigma_c^2} E\left\{(C_w(n+1) - C_w(n))^2\right\} =$$

$$= \frac{N_0}{2N_0\sigma_c^2} E\left\{\left[\beta C(n+1) + \alpha(-1)^b \pi(n+1) - (\beta C(n) + \alpha(-1)^b \pi(n))\right]^2\right\} \qquad (25)$$

After conversion of (25) we get :

$$E\left\{\Gamma'\right\} = 1 - \beta^2 R_c(n, n+1).$$

Since *β < 1*, then $E\left\{\Gamma'\right\} \geq E\left\{\Gamma\right\}$ and this difference is the more the more is $R_c(n, n+1)$, that justifies an opportunity to detect SGS-SS.

Simulation of SG-SS detecting for 20 different images ~ 300x200 pixels and $\alpha = 1$.
Table 1

№	Γ	Γ`
1	0.004394	0.004433
2	0.039907	0.040459
3	0.021924	0.022316
4	0.021864	0.022040
5	0.105029	0.141615
6	0.042248	0.042728
7	0.033281	0.033327
8	0.013435	0.013484
9	0.004362	0.004406
10	0.058273	0.059046
11	0.001844	0.001979
12	0.018773	0.018774
13	0.080775	0.080947
14	0.023058	0.023409
15	0.004675	0.004879
16	0.070373	0.071065
17	0.029895	0.030077
18	0.058048	0.059319
19	0.032274	0.033378
20	0.014035	0.014167

We can see from Table 1 that CO and SG are distinguishable but the problem arises – how can be chosen a threshold?

Thus this approach is suitable in the case when it is necessary to distinguish which of two images (of the same cover object) is CO or SG?

There are more effective methods for such SG (see blind steganalysis in the sequel).