Обратими растерни трансформации и тяхното приложение в стеганографията

Емануил Стоянов, Божидар Стоянов

Convertible Raster Transformations and Their Application in Steganography: *This article suggests an original method for hiding 8 bit raster images into 16 bit raster ones. The method is based on the distinctive features of the digital representation of images, as well as the reciprocally convertible raster transformations. In a way this technology approximates the classical method LSB (Least Significant Bit). However, the similarity is only ideological. In practice the propounded method combines two processes - hiding and encrypting with a key. It could be successfully applied to both achromatic (Grey Scale) and polychromatic (RGB) images.*

Key words: *LSB, Steganography, Secret Image, Stego Object, Cover Image, Convertible Operation.*

ВЪВЕДЕНИЕ

В края на XX век станахме свидетели на една нова революция - информационната. Информацията в съвременния свят [1] придобива все по-голямо значение. Може да се каже, че този, който владее информацията, всъщност контролира и света. Ето защо в днешно време скриването на информация, нейното съхранение и защитено предаване придобиват все по-голямо значение. В последните години все по-голяма популярност придобиха две науки - стеганографията и криптографията.

Стеганографията е изкуството [2] да се скрива информация по начин, който прави невъзможно разкриването на скритото послание. Тя включва голямо разнообразие от методи за тайно комуникиране, които скриват самото съществуване на посланието.

Стеганографията и криптографията са роднини в “шпионското” семейство. Криптографията разбърква посланието по начин, който не позволява то да бъде разбрано. Стеганографията скрива посланието[3] по начин, който не позволява то да бъде видяно.

През цялата история, хората са крили информация по множество удивителни начини. Днес особен интерес за нас представляват съвременните стеганографски методи, които тотално се различават от древните. Нещо повече, с развитието на компютърната техника, съвременте комуникации и Internet услугите се обособи едно ново направление - цифровата стеганография. Тя изучава възможностите за скриване на информация в друга, анализирайки особеностите в цифровото представяне на данни, както и слабостите на човешките възприятия. Един от най-популярните съвременни методи е LSB (*Least Significant Bit*) основаващ се на използването на най-малко значимия бит[4] за представяне на скрити данни.

Методът се състои в следното: При 8 битово RGB растерно изображение са нужни 24 бита за представянето на всеки пиксел (т.е. 3 цветни канала по 8 бита за всеки). Нека разгледаме групите от 9 байта:

(00100111 11101001 11001000)
(00100111 11001000 11101001)
(11001000 00100111 11101001)

Чрез тях може да бъде представена цветната информация на три пиксела. Ако във всеки байт бъде подменен най-младшия бит разликата в изображението би била незначителна [4] и не би се регистрирала от човешкото око. По този начин в три пиксела може да се скрие 9 битово двоично число, което е напълно достатъчно за представяне на пиксел от друго изображение или ASCII символ. Съвсем успешно могат да се използват и последните два [5] най-малко значими бита на всеки байт без това да се забележи от човешкото око.

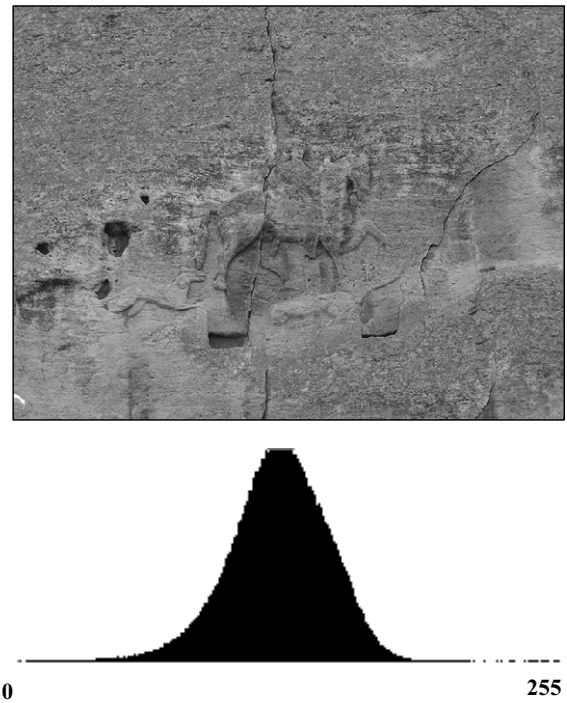
ИЗЛОЖЕНИЕ

1. Особености при преминаване от 8 битово в 16 битово изображение.

Разглеждат се 8 битови RGB растерни изображения. Тъй като те имат по 3 цветни канала (Chanel) за удобство се разглежда само единия. За останалите разсъжденията са по аналогия.

При 8 битовите RGB растерни изображения всеки пиксел може да приема до 2^8=256 отенъка на цветен канал. За онагледяване на тяхното присъствие се използват хистограми.

Хистограмата на отделен цветен канал показва степента в която присъстват в изображението всички нюанси в диапазона 0-255. Най-тъмните се изобразяват по оста x плътно в ляво, а най-светлите – плътно в дясно. Между тях са разпределени всички останали нива.



Фиг.1 *Хистограма на черно-бяло(Gray) изображение*

От хистограмата на снимката (Фиг.1) се вижда, че преобладават най-вече средните нива. Чистото черно и бяло присъстват незначително.

При 16 битовите растерни изображения възможните нива ще бъдат вече 2^{16}=65536 на цветен канал.

Ако 8 битово изображение се презапише като 16 битово, то ще запази своите 256 нива/Chanel, но те ще бъдат разпределени равномерно по целият диапазон 0-65535 със стъпка през 256 (Таблица 1). Получава се разреждане на 256 нива в 16 битов диапазон.

Нека с **k** е означено конкретно ниво от 8 bits/Chanel изображение (**k**∈**N**, 0≤**k**≤255), а с **L** неговите съответни в 16 битовото представяне. Тогава за разглежданата трансформация е в сила формула (1).

$$\mathbf{L = k.256 + p} \quad \mathbf{(L, p, k \in N; 0 \le k \le 255; 0 \le L \le 65535; 0 \le p < 256)} \qquad (1)$$

Параметърът **p** определя дължината на интервалите, които се отнасят до едно и също ниво в 8 битовото представяне.

Таблица 1

Дълбочина	**Съответствие на нивата на растерно изображение при преминаването от 8 в 16 бита**					
8 bits Chanel	**0**	**1**	**2**	**3**	**…**	**255**
16 bits Chanel	**0**…255	**256**…511	**512**…767	**768**…1023	…	**65280**…65535

Както се вижда от таблицата, варирането на **p** в посочените граници генерира интервали от по 256 нива (Level/Chanel), които съответстват на едно и също ниво в 8 битовото представяне (т.е. за всяко **k**-то ниво в 8 битовото представяне има запазен интервал от по 256 нива в 16 битовия вариант). От тях обаче значимо е само първото в интервала, или с други думи кратните на 256 (при **p**=0). Останалите не се използват и са празни.

Таблица 2

Нива (8 bits/Chanel)	**Интервали на съответствие в режим на 16 bits/ Chanel**			
	Значими нива	Неизползвани нива		
0	**0** | **00000000** | **00000000** |	1 | 00000000 | 00000001 |	…	255 | 00000000 | 11111111 |
1	**256** | **00000001** | **00000000** |	257 | 00000001 | 00000001 |	…	511 | 00000001 | 11111111 |
2	**512** | **00000010** | **00000000** |	513 | 00000010 | 00000001 |	…	767 | 00000010 | 11111111 |
...	**…**			
255	**65280** | **11111111** | **00000000** |	65281 | 11111111 | 00000001 |	…	65535 | 11111111 | 11111111 |

Разглеждайки този процес на ниско ниво установяваме, че варирането на **p** се отразява само върху младшите байтове на пикселите. Това показва, че след трансформирането на изображението от 8 в 16 бита/Chanel значими за отделните нива ще бъдат само старшите байтове. Младшите остават практически неизползвани (Таблица 2).

Направените разсъждения и заключения ни позволяват да се възползваме от свободните байтове в следващия етап на разглеждания метод.

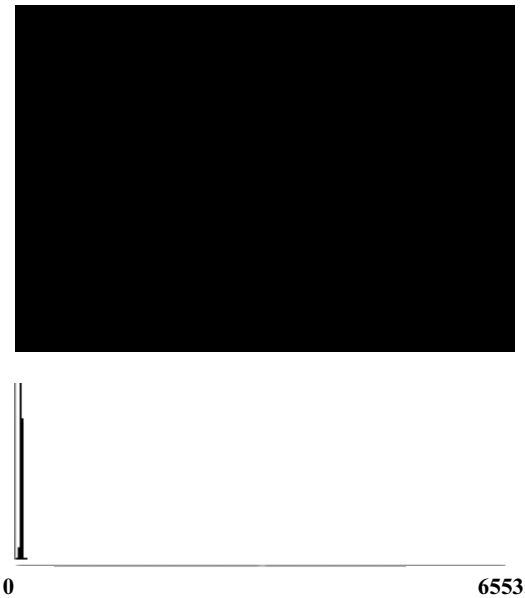
2. Редуциране (уплътняване) диапазона на използваните нива.

Разглеждаме изображения, които са трансформирани от 8 в 16 bits/Chanel. Както стана ясно в т. 2 от всички възможни нива значими ще бъдат само 256. Ще редуцираме всичките 65536 нива до значително по-малък брой, например до 512 (т.е. от 0 до 511). Това свиване може да се реализира чрез деление на 128

```
0000000|0  00000000  -   0
0000000|1  11111111  - 511
```

Фиг.2 *Използвани битове след редукция от 128 пъти*

Редукцията ще се отрази на битовото представяне така, че само младшите 9 бита (Фиг. 2) ще бъдат достатъчни за да поемат диапазона [0-511]. Останалите 7 бита от старшият байт ще бъдат свободни.



Фиг.3 *Хистограма след редукция от 128*

При подобно свиване нивата в хистограмата ще се „смачкат" до тънка черна ивица, намираща се в левия край, а изображението ще почернее (Фиг.3). Въпреки това в него ще бъде съхранена пълната 8 битова информация. Наистина, при разделяне двете страни на формула (1) с 128 се получават ограниченията показани във формула (2):

$$\mathbf{L = k.2 + p \quad (L, p, k \in N; 0 \le k \le 255; 0 \le L \le 511; 0 \le p < 2)} \qquad \mathbf{(2)}$$

Тогава при **k**=2, **L**={4,5}, а при **k**=255, **L**={510,511}

С други думи на всяко ниво от 8 битовото представяне ще съответстват по две нива в 16 битовото.

Както беше споменато по-горе изменението на **p** се отразява само в младшите битове на пикселите. От друга страна се вижда, че колкото по-голяма е компресията на нивата, толкова по-малък става интервалът за изменение на **p**. Това означава, че теоретично може да се свиват нивата до 256 пъти, без да се загуби 8 битовата информация. При такова уплътняване ще бъде в сила формула (3).

$$\mathbf{L = k \quad (L, p, k \in N; 0 \le k \le 255; 0 \le L \le 255; 0 \le p < 1} \ (\text{т.е. } \mathbf{p} = 0)) \qquad \mathbf{(3)}$$

На (Фиг.4) е показан начинът, по който ще се отрази подобно свиване в битовото представяне. Само младшият байт ще е достатъчен за да поеме този диапазон. Останалите 8 бита от старшия байт ще бъдат свободни.

```
00000000 |00000000| -   0
00000000 |11111111| - 255
```

Фиг.4 *Използвани битове след редукция от 256 пъти*

3. Метод за скриване на 8 битово растерно изображение в 16 битово.

Авторският метод използва една елементарна обратима математическа операция – сумиране. Известно е, че сумата на две естествени числа напълно съдържа техните стойности.

Да разгледаме израза:

c=a+b (**a,b**$\in$**N**)

Тогава, за да се възстанови едната стойност е необходима обратната математическа операция - изваждане, както и стойността на другата величина. Например **b=c-a** или **a=c-b.**

Ако **a** и **b** се разглеждат като две еднакви по размер 16 битови растерни изображения, тогава **c** ще бъде резултатът от тяхната пискелна сума. Това означава сумиране стойностите на нивата в съответните пиксели на двете изображения. Поради това трябва да се предвиди механизъм, който да гарантира, че в **c** няма да стане препълване на максимално допустимата стойност на нивата. За целта се извършва подготвителна дейност, която се изразява в две стъпки:

1.) Изображението, което ще скриваме (***Secret Image***) се свива значително по нива, както това беше обяснено по-горе в точка 3.

```
0000000|0  00000000 -   0
0000000|1  11111111 - 511
```

Фиг.5 *Диапазон на значимите нива в **тайното изображение** (Secret Image) след свиване*

В резултат остават да се използват само първите **m** от младшите битове (Фиг.5) Този брой зависи от степента на компресията.

2.) Изображението, което ще прикрива (***Cover Image***) също се свива отдясно, но незначително – толкова, че да се освободят поне **m** на брой от младшите битове (Фиг.6). Това е един задължителен минимум на свиване, който трябва да се спази. Нека разгледаме следващия пример:

```
00000000  00000000 -     0
11111101  11111111 - 65023
```

Фиг.6 *Диапазон на значимите нива в **прикриващото изображение** (Cover Image)*

Ако се приеме, че максиманият брой на използвани нива след прилагането на стъпка 1) към **тайното изображение**(*Secret Image**)*** е примерно 512 (т.е. свиването е 128 пъти), то в **прикриващото изображение**(*Cover Image**)*** ще редуцираме броя им само с 512 (т.е. 65535-512= 65023 използвани нива). Това ще освободи всички нива от 65024 до 65535.

По този начин след сумата на двете редуцирани изображения никъде няма да се превиши максимално допустимата стойност.

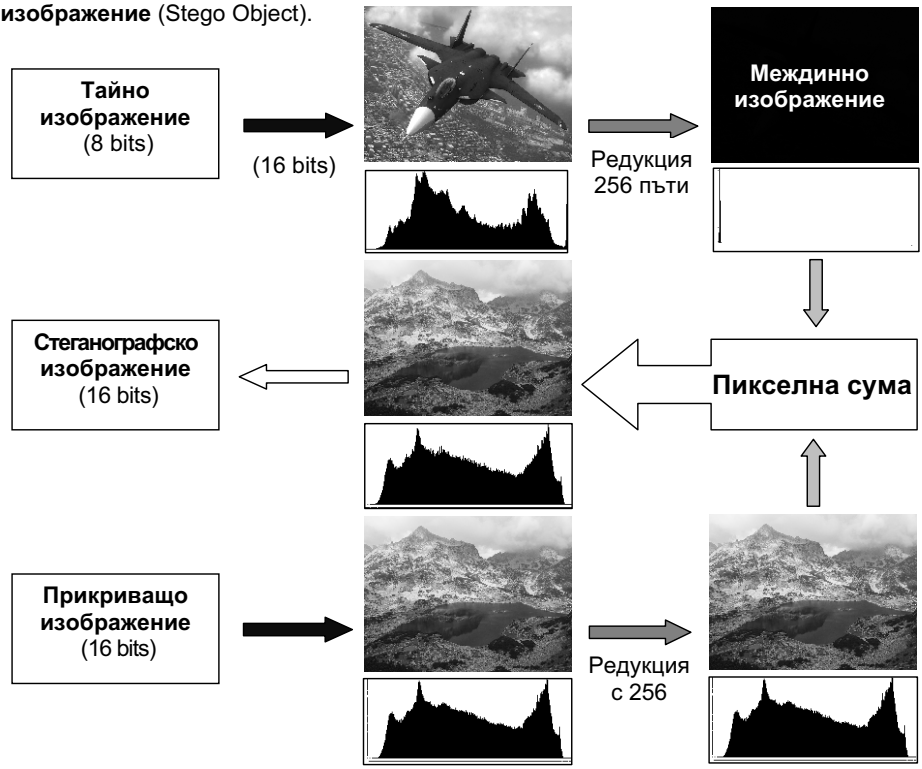
1111111|0 00000000 - 65024
1111111|1 11111111 - 65535

Фиг.7 *Диапазон на свободните нива в **прикриващото изображение** (Cover Image) след редукцията с 512*

Тази редукция, разгледана на битово ниво означава, че най-десните 9 бита на ***Cover Image*** могат да поемат сумата с нивата от ***Secret Image***. (Фиг. 7) показва това.

В най-лошия случай може да се извърши сумиране на двете най-светли нива на **Secret Image** и **Cover Image**. Това са стойностите 511 и 65023, в резултат на което ще се получи 65534.

По този начин чрез стъпки 1) и 2) се избягва възможността в най-неблагоприятния случай да стане байтово препълване в **стеганографското изображение** (Stego Object).

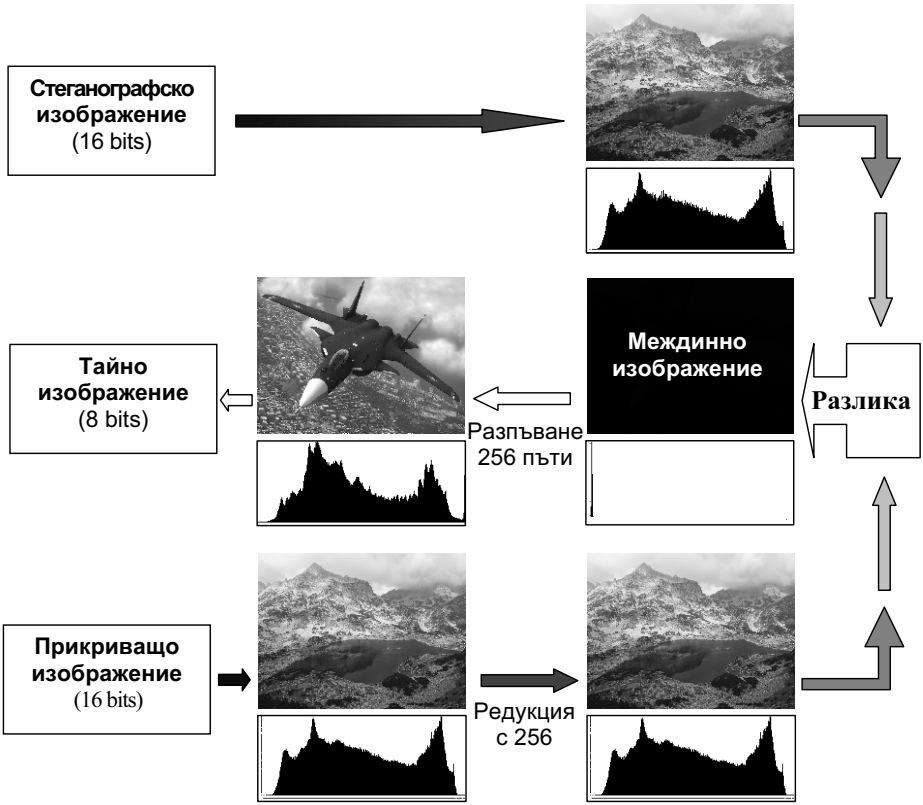


Фиг. 8 *Принципна схема за скриване на 8 битово **тайно изображение** (Secret Image) в 16 битово **прикриващо изображение** (Cover Image)*

На Фиг.8 е показана принципната схема на авторския метод, за скриване на 8 в 16 битово изображение който включва следните стъпки:

1. Превръщане на **тайното изображение** (Secret Image) в 16 битово.
2. Свиване диапазона на нивата в **тайното изображение** (Secret Image) до **n** пъти чрез попикселно разделяне на **n** (n≤256). Резултатът е **междинно изображение** (Intermediate Image), значително ограничено по брой нива.
3. Редуциране нивата на **прикриващото изображение** (Cover Image) с 65536/**n**.

4. Пикселно сумиране на **междинното изображение**(Intermediate Image) с **прикриващото изображение** (Cover Image).



Фиг. 9 *Принципна схема за извличане на 8 битово **тайно изображение** (Secret Image) от 16 битов **стеганографски обект** (Stego Object)*

На Фиг.9 е показана принципната схема за извличане на **скрито изображение** (Secret Image) от **стеганографски обект** (Stego Object), която включва:

1. Редуциране нивата на **прикриващото изображение** (Cover Image) с 65536/**n**.
2. Пикселна разлика между **стеганографския обект** (Stego Object) и **прикриващото изображение** (Cover Image). Резултатът е **междинно изображение** (Intermediate Image).
3. Разпъване на нивата на **междинното изображение** (Intermediate Image) чрез попикселно умножение с **n**.
4. Превръщане на резултатното изображение в 8 битово. Резултатът е точно **скритото изображение** (Secret Image).

ЗАКЛЮЧЕНИЕ

Анализирането на описаната технология позволява да се направят следните изводи:

1. Предложеният метод комбинира в себе си два процеса – скриване и криптиране с ключ. За да бъде възстановено скритото изображение е необходимо да се притежава освен стеганографско изображение(Stego Object) и оригинала на прикриващото изображение(Cover Image). В този смисъл Cover Image играе ролята на ключ. Това обаче не е достатъчно. Трябва да е известен и точният коефициент на редукцията, на която той предварително трябва да бъде подложен.
2. Този метод позволява да се скриват качествени пълноцветни изображения.
3. Резолюцията на скритото изображение съвпада с тази на прикриващото.
4. При необходимост могат да се скрият до три 8 битови черно-бели(Gray Scale) изображения разпределени в трите (RGB) канала.

ЛИТЕРАТУРА

[1]. Е. С. Стоянов. „Модели на взаимодействие в общество от независими агенти". Научна конференция „Научно технологичен трансфер", Институт за космически изследвания – БАН, Шуменски университет „Еп. К. Преславски", Висше военно училище за артилерия и противовъздушна отбрана „П. Волов", 2000г. , стр. 85-92.

[2]. Neil F. Johnson, Sushil Jajodia. "Exploring Steganography: Seeing the Unseen". Computing Practices. IEEE Press, February 1998, pp.26-34.

[3]. T. Aura, "Invisible Communication," *EET 1995*, technical report, Helsinki Univ. of Technology, Finland, Nov. 1995; http://deadlock.hut.fi/ste/ste_html.html.

[4]. W. Brown and B.J. Shepherd, *Graphics File Formats: Reference and Guide*, Manning Publications, Greenwich, Conn., 1995.

[5]. W. Bender et al., "Techniques for Data Hiding," *IBM Systems J.,* Vol. 35, Nos. 3 and 4, 1996, pp. 313-336.

За контакти:

ст. ас. Емануил Стоянов Стоянов, Катедра „Компютърна Информатика", Шуменскси Университет „Еп. Константин Преславски",
E-mail: emanuil_stoyanov@yahoo.com

ст. ас. Божидар Стоянов Стоянов, Катедра „Компютърна Информатика", Шуменскси Университет „Еп. Константин Преславски",
E-mail: bojidar_stoyanow@yahoo.com

Докладът е рецензиран.