



## رمزنگاری تصاویر RGB با استفاده از تابع آشوب Logistic Map و عملگرهای برش و جهش

سهیل فاطری<sup>۱</sup>، رسول عنایتی فر<sup>۲</sup>، محمد تشنه لب<sup>۳</sup>

دانشگاه آزاد اسلامی، واحد بابل<sup>۱</sup>

fateri@gmail.com

دانشگاه آزاد اسلامی واحد فیروزکوه<sup>۲</sup>

r.enayatifar@iaufb.ac.ir

دانشگاه خواجه نصیرالدین طوسی<sup>۳</sup>

teshnehlabb@eetd.kntu.ac.ir

### چکیده

در این مقاله یک روش جدید برای رمزنگاری تصویر با استفاده از سیگنال های آشوب و عملگرهای ژنتیکی پیشنهاد شده است. در این روش از عملگرهای ژنتیکی برای پیچیده تر شدن الگوریتم رمزنگاری، افزایش امنیت الگوریتم رمزنگاری و تغییر مقدار سطح خاکستری هر پیکسل از تصویر اصلی استفاده می شود. نتایج تجربی نشان می دهد که روش پیشنهادی مقاومت بالایی در برابر حملات متداول از خود نشان می دهد. همچنین مقدار آنتروپی به دست آمده در این روش ۷٫۹۹۱۳ است که به مقدار ایده آل، یعنی ۸، بسیار نزدیک است.

### واژه های کلیدی

پنهان سازی تصویر، سیگنال آشوب، عملگرهای ژنتیکی، آنتروپی.

### ۱- مقدمه !

همچنین رفتار نویز گونه (شبهه به نویز) این سیگنالها در عین قطعی بودن اشاره کرد. در [5] یک روش برای پنهانی کردن تصویر با استفاده از جابجائی پیکسلها در حوزه مکان پیشنهاد شده است. در [6] نیز یک الگوریتم مبتنی بر کلید برای پنهانی کردن تصویر (CKBA<sup>2</sup>) پیشنهاد شده است که در این روش از یک سیگنال آشوب برای تغییر مقدار سطح خاکستری پیکسلها استفاده شده است، تحقیقات بعدی نشان داد که این روش (استفاده از سیگنال آشوب) به تنهایی دارای امنیت مناسبی نمی باشد [7].

با رشد سریع تولیدات چند رسانه ای و پخش گسترده محصولات دیجیتالی بر روی اینترنت محافظت از اطلاعات دیجیتالی در برابر کپی و توزیع غیر مجاز هر روز اهمیت بیشتری پیدا می کند. برای رسیدن به این هدف الگوریتم های گوناگونی برای پنهانی کردن تصویر<sup>۱</sup> پیشنهاد شده است [1-4]. اخیرا با توجه به توسعه زیاد استفاده از سیگنالهای آشوب برای کاربردهای مختلف، بسیاری از محققین بر روی استفاده از این سیگنالها برای پنهانی کردن تصویر متمرکز شده اند [5-9]. از مهمترین مزیت های سیگنال های آشوب می توان به حساسیت زیاد این سیگنالها به شرایط اولیه و

<sup>2</sup> Chaotic key-based Image Encryption

<sup>1</sup> Image Encryption

$$X_{n+1} = rX_n(1 - X_n) \quad (1)$$

در این فرمول  $X_n$  مقداری در بازه  $[0,1]$  را می‌پذیرد این سیگنال با توجه به تقسیم شدن پارامتر  $r$  در سه بازه مختلف، سه رفتار مختلف از خود نشان می‌دهد که رفتار این سیگنال با فرض  $X_0 = 0.3$  به صورت زیر قابل بررسی است:

(۱) اگر  $r \in [0,3]$ ، آنگاه رفتار سیگنال در ۱۰ تکرار اول تا حدی

آشوبگونه و بعد از تکرار دهم به ثبات می‌رسد (شکل a.۱).

(۲) اگر  $r \in [3,3.57]$ ، آنگاه رفتار سیگنال در ۲۰ تکرار اول تا

حدودی آشوبگونه و بعد از تکرار بیستم، بین دو مقدار ثابت

تغییر نوسان می‌کند. (شکل b.۱).

(۳) اگر  $r \in [3.57,4]$ ، آنگاه رفتار سیگنال به طور کلی آشوبگونه

می‌باشد (شکل c.۱).

با توجه توضیحات بیان شده و با توجه به نیاز مقاله به رفتار کاملا

آشوبگونه برای پنهانی سازی، از سیگنال آشوبگونه Logistic map

با مقادیر اولیه  $X_0 = 0.3$  و  $r \in [3.57,4]$  استفاده شده‌است.

در این مقاله یک الگوریتم جدید برای پنهانی کردن تصویر با استفاده از سیگنال آشوب و عملگرهای ژنتیکی<sup>۱</sup> برای پیچیده تر شدن الگوریتم رمزنگاری و افزایش امنیت الگوریتم رمزنگاری پیشنهاد شده است. در این روش از عملگرهای ژنتیکی برش<sup>۲</sup> و جهش<sup>۳</sup> برای تغییر سطح خاکستری هر پیکسل تصویر و از تابع آشوب برای ۳ عمل استفاده می‌شود: الف) انتخاب دو پیکسل از تصویر برای اعمال عملگرهای ژنتیکی. ب) تعیین تعدادی بیت در پیکسل های انتخاب شده، جهت عمل برش. ج) تعیین بیتی از پیکسل، جهت عمل جهش.

در ادامه مقاله ابتدا توضیح مختصری در مورد توابع آشوب آمده است، سپس به شرح روش پیشنهادی پرداخته خواهد شد و در پایان در بخش نتایج تجربی، کارایی این روش با استفاده از تصاویر مختلف و از نظر مقاومت در برابر حملات مختلف مورد ارزیابی قرار خواهد گرفت.

## ۲- سیگنال های آشوب

سیگنال آشوب ظاهری شبیه به نویز دارد ولی در عین حال کاملا قطعی است. یعنی با داشتن مقادیر اولیه و تابع نگاشت می‌توان دقیقا مقادیر سیگنال را دوباره تولید کرد. برخی مزایای این سیگنال عبارتند از:

الف) حساسیت نسبت به شرایط اولیه: سیگنال آشوب به شرایط اولیه بسیار حساس است به گونه‌ای که هر تغییر جزئی در مقادیر اولیه، اختلاف فاحشی در مقادیر بعدی تابع ایجاد خواهد نمود. لذا اگر مقادیر اولیه سیگنال اندکی تغییر کنند، سیگنال حاصل تفاوت بسیار زیادی با سیگنال اولیه خواهد داشت.

ب) رفتار ظاهر تصادفی: در قیاس با تولید کننده‌های اعداد تصادفی طبیعی که در آنها، رشته اعداد تصادفی تولید شده را نمی‌توان باز تولید نمود، روشهای مورد استفاده برای تولید اعداد تصادفی در الگوریتم های بر مبنای توابع آشوب، این امکان را به ما می‌دهند که در صورت داشتن مقدار اولیه و تابع نگاشت، همان اعداد تصادفی را باز تولید نمود.

ج) عملکرد قطعی: توابع آشوب در عین اینکه ظاهری تصادفی دارند اما کاملا قطعی هستند. یعنی همواره با داشتن تابع نگاشت و مقادیر اولیه می‌توان یک مجموعه از مقادیر را که به ظاهر هیچ نظم ظاهری در آنها وجود ندارد را تولید و دوباره به همان مقادیر دست یافت.

رابطه ۱، یکی از معروفترین سیگنالهایی که رفتار آشوب گونه دارد و به سیگنال Logistic Map معروف است را نشان می‌دهد.

<sup>1</sup> Genetic operator

<sup>2</sup> Cross over

<sup>3</sup> Mutation

مشخص می‌شود. مقدار این کلید ۸۰ بیتی را می‌توان به صورت زیر محاسبه نمود (رابطه ۲).

$$K = \sum_{i=0}^{79} (b_i \times 2^i) \quad (2)$$

در رابطه فوق  $K$  مقدار کلید و  $b_i$  بیت  $i$ -ام از کلید می‌باشد. مشخص است که این کلید ۸۰ بیتی را می‌توان توسط ۱۰ بایت نیز نشان داد که در این صورت می‌توان مقدار کلید را با توجه به رابطه (۳) محاسبه نمود. در این رابطه،  $B_i$  نشان دهنده مقدار بایت  $i$ -ام از کلید است. استفاده از ۸ بایت برای نشان دادن کلید دارای این مزیت است که با استفاده از جدول اسکی<sup>۴</sup> می‌توان کلید را بصورت رشته‌ای از حروف نیز نشان داد.

$$K = \sum_{i=0}^7 (B_i \times 256^i) \quad (3)$$

از کلید  $K$  برای تولید مقدار اولیه سیگنال آشوب استفاده خواهد شد. از آنجایی که مقدار اولیه تابع در بازه  $[0, 1]$  در نظر گرفته خواهد شد، کفایت برای محاسبه مقدار اولیه، مقدار کلید را به بازه  $[0, 1]$  تبدیل کرد. با توجه به اینکه مقدار کلید عددی در بازه  $[0, 2^{80} - 1]$  است، لذا می‌توان از رابطه ۴ برای این منظور استفاده کرد.

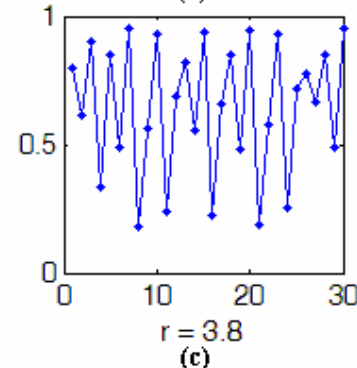
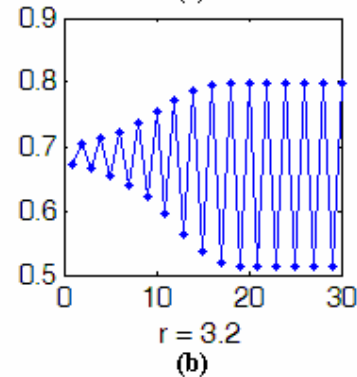
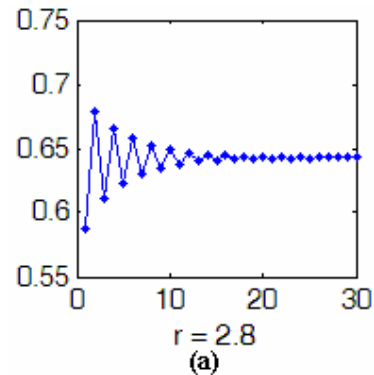
$$X_0 = \frac{K}{2^{80}} \quad (4)$$

گام ۲: انتخاب دو پیکسل از تصویر برای عمل برش: نحوه انتخاب این دو پیکسل به شرح زیر می‌باشد:  
پیکسل اول: اولین پیکسل از تصویر که در موقعیت سطر و ستون اول قرار دارد (در گام بعد دومین پیکسل از تصویر و همینطور در مراجعات بعدی پیکسل‌های متوالی تا پیکسل قرار گرفته در سطر و ستون آخر).

پیکسل دوم: برای انتخاب این پیکسل دوم از سیگنال آشوب استفاده می‌شود. حال می‌بایست با توجه به مقدار تولید شده توسط سیگنال آشوب شماره سطر و ستون پیکسل دوم را استخراج نمود. برای این منظور دو عدد از سیگنال آشوب تولید می‌شود، از عدد اول که آن را  $S_1$  می‌نامیم برای انتخاب سطر و از عدد دوم که آن را  $S_2$  می‌نامیم برای انتخاب شماره ستون استفاده می‌کنیم. اگر تعداد کل سطرها و ستون‌های تصویر به ترتیب  $R$  و  $C$  در نظر گرفته شوند، برای محاسبه شماره سطر و ستون پیکسل انتخابی می‌توان از رابطه (۵) کمک گرفت:

$$i = \lfloor (S_1 \times R) + 1 \rfloor \quad (5)$$

$$j = \lfloor (S_2 \times C) + 1 \rfloor$$



شکل ۱- رفتار آشوبگونه سیگنال logistic map با  $X_0 = 0.3$  با  $r \in [3.57, 4]$  c  $r \in [3, 3.57]$  b  $r \in [0, 3]$  a

### ۳- شرح روش پیشنهادی

در این روش از عملگرهای ژنتیکی برش و جهش برای تغییر سطح خاکستری هر پیکسل تصویر و از تابع آشوب برای سه کاربرد استفاده می‌شود: الف) انتخاب دو پیکسل از تصویر برای اعمال عملگرهای ژنتیکی. ب) تعیین تعداد بیت‌هایی از پیکسل‌های انتخاب شده در مرحله قبل، برای عمل برش. ج) تعیین بیتی از هر پیکسل که باید عمل جهش بر روی آن اعمال شود.

گام‌های انجام روش پیشنهادی به صورت زیر می‌باشد:

گام ۱: تعیین مقدار اولیه تابع Logistic map: برای امنیت بیشتر روش پیشنهادی، این مقدار از روی یک کلید ۸۰ بیتی

<sup>4</sup> Ascii Table

در الگوریتم پیشنهادی، عمل جهش بر روی بیت‌های پیکسل انتخابی اول (که در گام دوم مشخص شده است)، انجام می‌گیرد. بیتی که عمل جهش از آن محل انجام می‌گیرد، توسط تابع آشوب انتخاب می‌شود. برای اینکار کافیسیت عددی بعد را توسط تابع آشوب تولید کرد ( $S_4$ )، سپس با استفاده از رابطه (7) می‌توان مقدار تابع آشوب را به یک مقدار گسسته در بازه [1, 8] تبدیل نمود.

$$q = \lfloor (S_4 \times 8) \rfloor \quad (7)$$

با توجه به مقدار  $q$  با استفاده از الگوریتم زیر عمل جهش را تعریف می‌نماییم:

```

if q >= 4 then
    P[q] = not (P[q])
else
    for i = q to 8
        P[i] = not (P[i])
    end for
end if

```

در حقیقت  $q$  شماره بیتی است که عمل برش باید در آن انجام شود (بیت‌ها را از بیت کم ارزش به پر ارزش و از صفر تا هفت شماره گذاری می‌کنیم). از آنجایی که تغییر در بیت‌های کم‌ارزش تصویر، تاثیر چندانی در شدت رنگ پیکسل‌ها ندارد، لذا در الگوریتم جهش پیشنهادی، چنانچه یکی از چهار بیت کم ارزش برای عمل جهش انتخاب شوند، عمل جهش روی بیت انتخاب شده و تمامی بیت‌های پر ارزش‌تر از آن اعمال می‌کنیم، مانند آنچه در شکل ۳.b نشان داده شده است.

**گام ۵:** مراحل ۲ تا ۴ برای تمام پیکسل‌های تصویر انجام می‌شود.  
**گام ۶:** مراحل ۲ تا ۵ به تعداد ۲۰ بار بر روی تصویر انجام می‌شود.  
گام‌های فوق برای رمزنگاری تصویر پیشنهاد شده اند، بدیهی است برای خارج نمودن تصویر رمز شده گام‌های مذکور را می‌بایست بعکس انجام نمود.

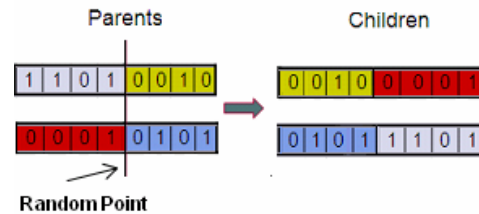
#### ۴- نتایج تجربی

یک رویه پنهانی سازی خوب باید در برابر انواع حملات از جمله حملات کشف رمز، حملات آماری و حملات افسارگسیخته<sup>۶</sup> پایدار باشد. در این بخش الگوریتم پیشنهادی از لحاظ تحلیل آماری، تحلیل حساسیت این روش نسبت به تغییرات کلید و تحلیل فضای کلید مورد بررسی قرار خواهد گرفت.

در رابطه فوق  $i$  و  $z$  شماره سطر و ستون پیکسل انتخاب شده می‌باشند.

**گام ۳:** انجام عمل برش بر روی پیکسل‌های انتخاب شده: دو پیکسل تعیین شده از گام قبلی، بعد از تبدیل سطح خاکستری به مبنای ۲، در این گام برای عمل برش در نظر گرفته می‌شوند. عمل برش بکار رفته در این مقاله یک عمل برش تک نقطه<sup>۷</sup> می‌باشد که به صورت زیر تعریف می‌شود (شکل ۲):

- یک نقطه تصادفی در طول رشته انتخاب می‌شود.
- والدین در این نقطه به دو قسمت می‌شوند.
- هر فرزند با انتخاب تکه اول از یکی از والدین و تکه دوم از والد دیگر بوجود می‌آید.



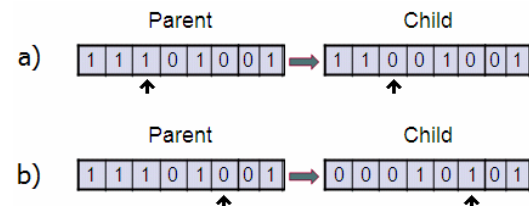
شکل ۲- برش تک نقطه

برای تعیین بیتی از پیکسل که به عنوان نقطه تصادفی در نظر گرفته شود، یک مقدار جدید توسط سیگنال آشوب (با توجه به آخرین مقدار تولید شده در مرحله قبل) تولید و آن را  $S_3$  می‌نامیم. مقدار بدست آمده به عنوان محل برش تصادفی در نظر گرفته می‌شود و طبق روش مطرح شده در فوق، عمل برش انجام می‌شود. فرزندان تولید شده به عنوان پیکسل‌های جدید جایگزین والدین خود می‌شوند. برای محاسبه محل برش با استفاده از مقدار سیگنال آشوب ( $S_3$ ) می‌توان از رابطه (۶) استفاده نمود.

$$p = \lfloor (S_3 \times 7) + 1 \rfloor \quad (6)$$

در رابطه فوق  $p$  محل برش می‌باشد.

**گام ۴:** انجام عمل جهش: جهش در الگوریتم‌های ژنتیک، بدین معناست که با استفاده از یک توزیع یکنواخت یک بیت بصورت تصادفی انتخاب و مقدار آن تغییر کند (شکل ۳.a).



شکل ۳- عملگر جهش

a. جهش در بیت ۵ b. جهش در بیت ۳

<sup>6</sup> Cryptanalytic

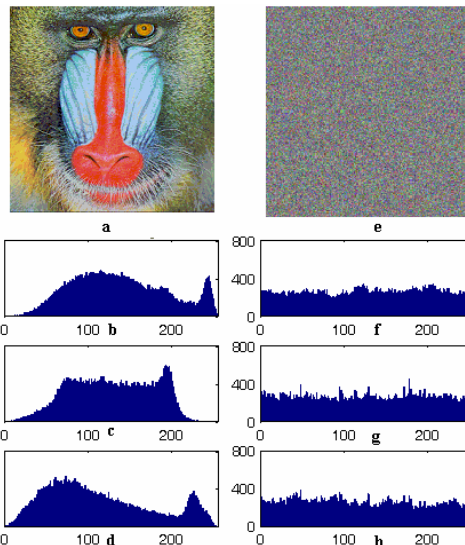
<sup>7</sup> brute-force

<sup>5</sup> Single-point Crossover

۴-۱. تحلیل هیستوگرام

هیستوگرام تعداد پیکسلها در هر سطح خاکستری را برای یک تصویر نشان می دهد. در شکل ۴ در فریم (a) می توان تصویر اصلی و در فریم های (b) و (c) و (d) به ترتیب هیستوگرام این تصویر را در سطح قرمز و سبز و آبی را مشاهده کرد. همچنین در فریم (e) می توان تصویر پنهان سازی شده (با کلید 'ABCDEF0123456789ABCD' در مبنای ۱۶) از روی تصویر اصلی (فریم (a)) و در فریم های (f) و (g) و (h) به ترتیب هیستوگرام تصویر پنهان سازی شده در سطح قرمز و سبز و آبی را مشاهده کرد.

همان طور که در شکل ۳ به وضوح قابل مشاهده است هیستوگرام تصویر پنهان سازی شده یک هیستوگرام یکنواخت است و این هیستوگرام با هیستوگرام تصویر اصلی کاملا متفاوت است که این مساله امکان حملات آماری را بسیار مشکل خواهد کرد.



شکل ۴- فریم (a) تصویر اصلی، فریم های (b) و (c) و (d) به ترتیب هیستوگرام تصویر بایون با سایز 256\*256 را در سطح قرمز و سبز و آبی، فریم (e) تصویر پنهان سازی شده (با کلید 'ABCDEF0123456789ABCD' در مبنای ۱۶)، فریم های (f) و (g) و (h) به ترتیب هیستوگرام تصویر پنهان سازی شده را در سطح قرمز و سبز و آبی، P=1024، N=8

۴-۲. تحلیل ضرایب همبستگی

در این بخش همبستگی افقی و عمودی و قطری بین پیکسلهای تصویر مورد بررسی قرار خواهد گرفت. برای این منظور به طور تصادفی ۴۰۹۶ جفت از پیکسلهای مجاور به صورت افقی و عمودی و قطری به عنوان نمونه در نظر گرفته می شود. در شکل ۵ می توان توزیع سطح خاکستری پیکسلهای مجاور برای تصویر اصلی و تصویر پنهان سازی شده را مشاهده کرد.

در شکل ۵ فریم های (a) و (b) و (c) به ترتیب توزیع سطح خاکستری را برای دو پیکسل مجاور افقی و عمودی و قطری از تصویر اصلی نشان می دهند. به طور مشابه فریم های (d) و (e) و (f) نیز به ترتیب توزیع سطح خاکستری را برای دو پیکسل مجاور افقی و عمودی و قطری در تصویر پنهان سازی شده نشان می دهند.

همچنین در این بخش با استفاده از رابطه (۸) ضریب همبستگی دو پیکسل مجاور محاسبه خواهد شد.

$$E(x) = \frac{1}{N} \sum_{i=1}^N x_i$$

$$D(x) = \frac{1}{N} \sum_{i=1}^N (x_i - E(x_i))^2 \quad (8)$$

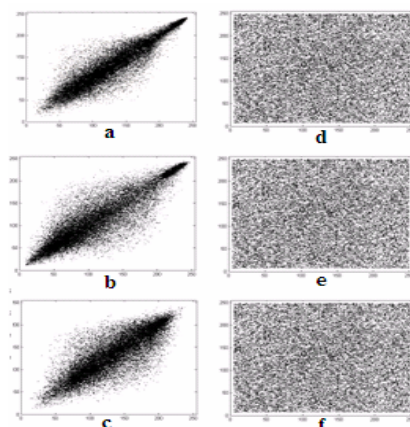
$$r_{xy} = \frac{cov(x, y)}{\sqrt{D(x)}\sqrt{D(y)}}$$

$$cov(x, y) = \frac{1}{N} \sum_{i=1}^N (x_i - E(x_i))(y_i - E(y_i))$$

نتایج به دست آمده در جدول ۱ نشان داده شده است.

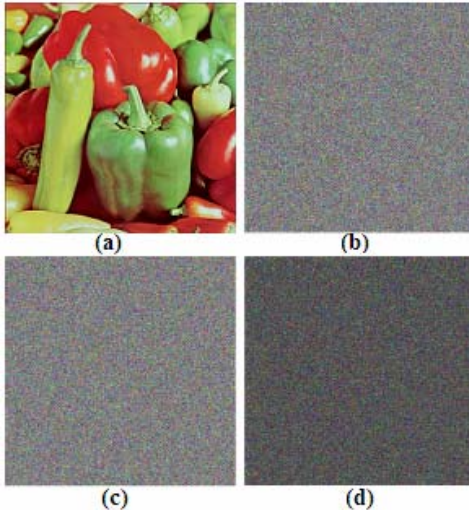
جدول ۱- ضریب همبستگی برای دو پیکسل مجاور در حالت (افقی، عمودی، قطری) برای تصویر اصلی و تصویر پنهان سازی

تصویر پنهان سازی شده	تصویر اصلی	مدل
۰,۰۳۲۷	۰,۹۲۳۱	افقی
۰,۰۲۱۸	۰,۸۸۳۷	عمودی
۰,۰۱۹۳	۰,۸۶۹۸	قطری



شکل ۵- فریم های (a) و (b) و (c) به ترتیب توزیع سطح خاکستری را برای دو پیکسل مجاور افقی و عمودی و قطری از تصویر اصلی. فریم های (d) و (e) و (f) نیز به ترتیب توزیع سطح خاکستری را برای دو پیکسل مجاور افقی و عمودی و قطری

روش نسبت به تغییراتی هر چند کوچک در کلید، حساسیت بالایی را نشان می‌دهد.



شکل ۶ - (a) تصویر فلفل (b) تصاویر پنهانی شده با روش پیشنهاد شده با کلید مشابه با یک بیت اختلاف (d) اختلاف بین دو تصویر (b) و (c) تصاویر پنهانی شده با روش

جدول ۲- اختلاف بین پیکسلها در تصاویر پنهانی شده با کلید مشابه با یک بیت اختلاف

تصاویر	درصد اختلاف بین دو تصویر پنهانی شده
Peper	۹۹,۵۷۸
Lena	۹۹,۵۰۲
Baboon	۹۹,۵۱۱

#### ۴-۵. تحلیل فضای کلید

در یک روش مناسب فضای کلید باید به حد کافی بزرگ باشد تا بتواند در برابر حملات افسار گسیخته از خود مقاومت نشان دهد. در روش پیشنهادی  $(1.20893 \times 10^{24} \approx 2^{80})$  ترکیب مختلف از کلید می‌تواند وجود داشته باشد که نتایج عملی نشان داده است که این تعداد ترکیب مختلف برای کلید جهت مقاومت در برابر انواع حملات افسارگسیخته کفایت می‌کند [12].

#### ۴-۶. آنتروپی اطلاعات<sup>۹</sup>

آنتروپی یکی از خصوصیات برجسته برای تصادفی بودن است. آنتروپی اطلاعات یک تئوری ریاضی برای ارتباط داده‌ای و ذخیره سازی است که در سال ۱۹۴۹ توسط Claude E Shannon معرفی شده است [13]. یکی از معروفترین فرمولها برای به دست آوردن آنتروپی به صورت زیر است:

#### ۴-۳. تاثیر تغییر یک پیکسل در تصویر اصلی بر روی تصویر پنهان سازی شده

تاثیر تغییر یک پیکسل در تصویر اصلی بر روی تصویر پنهان سازی شده را می‌توان با دو معیار اندازه گیری نمود: NPCR و UACI [10,11]. NPCR را می‌توان به صورت نرخ تغییر پیکسلها در تصویر پنهان سازی شده به ازای تغییر یک پیکسل در تصویر اصلی تعریف نمود. همچنین UACI را می‌توان به عنوان متوسط این تغییرات تعریف نمود. NPCR و UACI به صورت زیر تعریف می‌شوند (رابطه ۹).

$$NPCR = \frac{\sum_{i,j} D(i,j)}{W \times H} \times 100\% \quad (9)$$

$$UACI = \frac{1}{W \times H} \left[ \sum_{i,j} \frac{|C_1(i,j) - C_2(i,j)|}{255} \right] \times 100\%$$

که در آن  $H$  و  $W$  به ترتیب مشخص کننده طول و عرض تصاویر و  $C_1$  و  $C_2$  دو تصویر پنهان سازی شده هستند که از دو تصویر با یک پیکسل اختلاف گرفته شده اند و  $D$  به صورت رابطه ۱۰ تعریف می‌شود.

$$D(i,j) = \begin{cases} 1 & \text{if } C_1(i,j) = C_2(i,j) \\ 0 & \text{otherwise} \end{cases} \quad (10)$$

مقادیر به دست آمده برای یک تصویر با سایز  $256 \times 256$  به این صورت است: NPCR = 99.57%, UACI = 33.64%. مقادیر به دست آمده به وضوح نشان می‌دهد که این روش در برابر حملات تقاضی<sup>۸</sup> نیز مقاوم خواهد بود.

#### ۴-۴. تحلیل حساسیت به کلید

یک رویه پنهان سازی تصویر مناسب باید نسبت به تغییرات کوچک کلید حساس باشد بدین معنی که تغییر یک بیت در کلید باید سبب ایجاد یک نتیجه بسیار متفاوت شود. برای آزمایش عملکرد روش پیشنهادی در برابر تغییر یک بیت در کلید از تصویر ۶ استفاده می‌شود.

تصاویر ۶b و ۶c حاصل اعمال روش پیشنهادی بر روی تصویر ۶a با کلیدهای مشابه فقط با یک بیت متفاوت می‌باشند و تصویر ۶d اختلاف تصاویر ۶b و ۶c را نشان می‌دهد. این آزمایش برای چند تصویر منتخب از پردازش تصویر انجام شده است و نتایج حاصل در جدول ۲ آمده است. نتایج به دست آمده نشان می‌دهد که این

<sup>9</sup> Information Entropy

<sup>8</sup> Differential Attack

جدول ۳- مقایسه انتروپی بدست آمده در روش های مختلف

انتروپی	تصویر	روش
۷,۹۸۴۴	Paper	روش پیشنهادی این مقاله
۷,۹۹۰۱	Lena	
۷,۹۹۱۴	Baboon	
۷,۹۸۱۸	Paper	روش ارائه شده در [۸]
۷,۹۸۰۳	Lena	
۷,۹۸۴۸	Baboon	
۷,۹۷۳۱	Paper	روش ارائه شده در [۱۲]
۷,۹۶۷۰	Lena	
۷,۹۷۹۴	Baboon	

### ۵- نتیجه گیری

در این مقاله یک روش جدید برای پنهان سازی تصویر با استفاده از سیگنال های آشوب و عملگرهای ژنتیکی برای پیچیده تر شدن الگوریتم رمزنگاری پیشنهاد شده است. همان طور که در بخش نتایج تجربی نیز مشاهده شد، این روش در مقابل انواع حملات مختلف از جمله حملات کشف رمز، حملات آماری و حملات افسارگسیخته پایداری مناسبی از خود نشان می دهد. مقدار بالای انتروپی (۷,۹۹۱۳) در این روش کارایی بالای روش پیشنهادی را نشان می دهد.

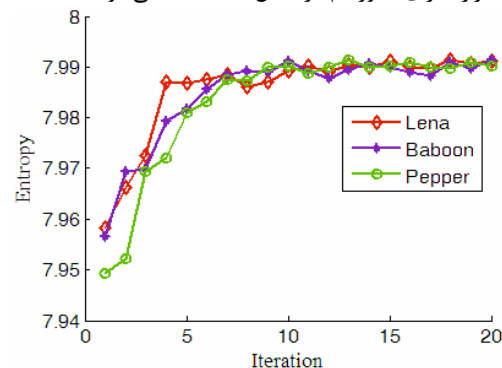
### مراجع

- [1] A. Mitra, Y. V. Subba Rao and S. R. M. Prasanna, "A New Image Encryption Approach using Combinational Permutation Techniques", International Journal of Computer Science Vol, 2006, pp: 1306-4428
- [2] Chin-Chen Chang , Tai-Xing Yu, "Cryptanalysis of an encryption scheme for binary images", Pattern Recognition Letters, 2002, pp: 1847-1852
- [3] Madhusudan Joshi, Chandrashakher, Kehar Singh, "Color image encryption and decryption using fractional Fourier transform", Optics Communications, 2007, pp:811-819
- [4] Yalon Roterman, Moshe Porat, "Color image coding using regional correlation of primary colors", Image and Vision Computing, 2007, pp: 637-651
- [5] Yas Abbas Alsultanny, "Random-bit sequence generation from image data", Image and Vision Computing", 2007, pp: 1178-1189
- [6] J.-C. Yen, J.-I. Guo, "A New Chaotic Key-Based Design for Image Encryption and Decryption", Proceedings IEEE International Conference on Circuits and Systems, vol.4, 2000, pp: 49-52.
- [7] S. Li, X. Zheng, "Cryptanalysis of a Chaotic Image Encryption Method", Scottsdale, AZ, USA, 2002, in:

$$H(S) = \sum_{i=0}^{2^N-1} P(s_i) \log\left(\frac{1}{P(s_i)}\right) \quad (13)$$

که در آن N برابر با تعداد سطح خاکستری استفاده شده در تصویر (در تصاویر ۸ بیتی برابر با ۲۵۶ خواهد بود) و  $P(s_i)$  نشان دهنده احتمال وقوع سطح خاکستری i-ام در تصویر خواهند بود. در تصاویری که به طور کامل تصادفی ایجاد شده است این مقدار برابر با ۸ خواهد بود که این مقدار به عنوان ایده آل در نظر گرفته می شود. هر چقدر مقدار به دست آمده برای انتروپی در یک روش به ۸ نزدیکتر باشد به این معنی خواهد بود که امکان پیش بینی پذیری این روش کمتر و در نتیجه امنیت این روش بالاتر خواهد بود.

برای تعیین میزان انتروپی روش پیشنهادی، انتروپی روش پیشنهادی برای سه تصویر منتخب پردازش تصویر و هر کدام در ۲۰ تکرار اجرای الگوریتم در شکل ۶ مشاهده می شود.



شکل ۷- انتروپی تصاویر Lena, Baboon, Pepper بعد از پنهانی شدن توسط روش پیشنهادی در ۲۰ تکرار

با توجه به شکل ۷ مشاهده می شود که انتروپی در روش پیشنهادی برای تصاویر مختلف در ۲۰ تکرار مختلف حداکثر برابر ۷,۹۹۱۳ به دست آمده است که بسیار نزدیک به مقدار ایده آل یعنی ۸ می باشد. که این موضوع امنیت این روش را در برابر با حملات موسوم به انتروپی نشان می دهد. انتروپی بدست آمده از روش های مشابهی که در [۸] و [۱۲] آمده است، بصورت میانگین ۰,۱۴٪ به عدد ۸ (مقدار ایده آل) نزدیک تر است. یادآوری می شود که تغییرات کوچک در راستای بهینه سازی انتروپی، تاثیرات بزرگی در کارایی الگوریتم دارد. در جدول ۳، مقایسه انتروپی بدست آمده توسط الگوریتم پیشنهادی و روش های ارائه شده در [۸] و [۱۲] را نشان می دهد، مقادیر جدول ۳ پس از ۵۰ بار تکرار الگوریتم بدست آمده اند.

- Proceedings IEEE International Symposium on Circuits and Systems, vol. 2, 2002, pp: 708–711.
- [8] H.S. Kwok, Wallace K.S. Tang, “A fast image encryption system based on chaotic maps with finite precision representation”, *Chaos, Solitons and Fractals*, 2007, pp: 1518–1529
- [9] S. Behnia , A. Akhshani , S. Ahadpour, H. Mahmodi , A. Akhavan, “A fast chaotic encryption scheme based on piecewise nonlinear chaotic maps”, *Physics Letters A* ,2007, pp: 391–396
- [10] Chen G, Mao YB, Chui CK, “A symmetric image encryption scheme based on 3D chaotic cat maps”, *Chaos, Solitons & Fractals*, 2004, pp:74-82.
- [11] Mao YB, Chen G, Lian SG, “A novel fast image encryption scheme based on the 3D chaotic baker map”, *Int Bifurcat Chaos*, 2004, pp:544-560
- [12] N.K. Pareek , Vinod Patidar , K.K. Sud, “Image encryption using chaotic logistic map , *Image and Vision Computing*” , 2006, pp: 926–934
- [13] C.E. Shannon, *Bell Syst. Tech. J.* 28 (1949) 656.