مجله علمی- پژوهشی "پدافند الکترونیکی و سایبری"
سال ششم، شماره ۱، بهار ۱۳۹۷، ص ۴۶-۳۷

یک الگوریتم پیشنهادی برای رمزنگاری تصاویر خاکستری با الگوبرداری از شبکه‌های میان ارتباطی بنس و نگاشت آشوب Logistic map

مهدی احمدی پری[۱]، میثم مرادی[۲*]

۱ و ۲- دانشجوی کارشناسی ارشد، گروه کامپیوتر، دانشکده فنی و مهندسی، دانشگاه ملایر

(دریافت: ۹۵/۰۶/۰۲، پذیرش: ۹۶/۰۱/۲۲)

چکیده

در دو دهه اخیر با گسترش اطلاعات و ارتباطات، استفاده از شبکه‌های رایانه‌ای در سطح‌های مختلف، لزوم انتقال تصاویر دیجیتال در این شبکه‌ها بیش از گذشته مورد توجه قرار گرفته است. جهت حفاظت از تصاویر در مقابل دسترسی‌های غیرمجاز، از رمزنگاری تصویر استفاده می‌شود. الگوریتم‌های زیادی جهت رمزنگاری تصویر ارائه شده است. در این تحقیق، یک الگوریتم جدید جهت رمزنگاری تصویر طراحی و پیاده‌سازی شده است. در روش پیشنهادی با ترکیب نگاشت آشوب Logistic map ۱ و عمل Xor و الهام‌گرفتن از شبکه میان ارتباطی Beness، یک الگوریتم جدید و قدرتمند جهت رمزنگاری تصویر ارائه شده است. نتایج آزمایشات نیز نشان می‌دهد که الگوریتم ارائه‌شده از کارایی مناسب برخوردار است.

واژه‌های کلیدی: رمزنگاری تصاویر، نگاشت آشوب Logistic map ، شبکه میان ارتباطی Beness، تحلیل تصاویر

۱- مقدمه

امروزه با گسترش استفاده از شیکه‌های رایانه‌ای، اجتماعی و مخابراتی انتقال تصاویر دیجیتال نیز در این شبکه‌ها جهت کاربردهای مختلف افزایش یافته است. در برخی از کاربردها مانند کاربردهای پزشکی، نظامی، تجاری و ... لزوم حفاظت از تصاویر درمقابل دسترسی‌های غیرمجاز اهمیت زیادی پیدا کرده است. یکی از روش‌های مرسوم برای محافظت از داده‌ها رمزنگاری می‌باشد [۲-۱].

از آن‌جاکه انتقال تصاویر بیشتر در کاربردهای بدون تاخیر[۱] می‌باشد و هر تصویر حجم زیادی از داده‌ها را در خود جای داده است لذا استحکام در رمزنگاری تصاویر اهمیت زیادی دارد. الگوریتم‌های مرسوم رمزنگاری مانند DES، AES، IDEA و ... نمی‌توانند این خواسته را برآورده کنند [۴-۳].

در سال‌های اخیر، الگوریتم‌های زیادی جهت رمزنگاری تصاویر ارائه شده است. در برخی از این الگوریتم‌ها از جایگزینی مقادیر پیکسل‌های تصویر با مقدارهای جدید استفاده شده است [۸-۵]. در برخی دیگر از جابه‌جایی مکان پیکسل‌های تصویر استفاده شده است [۱۲-۹]. در برخی دیگر نیز از نگاشت آشوب برای رمزنگاری تصاویر استفاده شده است [۱۶-۱۳].

در الگوریتم‌های جایگشتی روش‌های مختلفی برای جایگشت[۲] استفاده می‌شود. در برخی از روش‌ها، از جایگشتی بیتی استفاده می‌شود [۱۷]. در برخی دیگر، مکان بیت‌های یک بایت با هم جابه‌جا می‌شوند. در روش‌های دیگر از جایگشت بایتی استفاده می‌شود [۱۸]. بدین‌صورت‌که مکان بایت‌های یک تصویر با هم جابه‌جا می‌شوند. در موارد دیگر، تصویر به بلوک‌هایی تقسیم‌شده و این بلوک‌ها با هم جابه‌جا می‌شوند [۱۹] و در بعضی از الگوریتم‌های جایگشتی نیز از ترکیبی از این سه روش استفاده شده است. الگوریتم‌های که فقط از روش جایگشتی استفاده می‌کنند در مقابل حملات known /chosen-plaintext مقاوم نیستند [۲۰] و می‌توان با مقایسه تصویر اصلی و تصویر رمزشده کلید را یافت.الگوریتم‌هایی نیز که فقط از نگاشت آشوب استفاده می‌کنند در مقابل حملات مقاومت خوبی ندارند. [۲۳-۲۱]. در این تحقیق، یک روش ترکیبی جدید برای رمزنگاری تصاویر ارائه شده است که در مقابل انواع حملات مقاوم بوده و ارزیابی‌های انجام‌شده نشان می‌دهد که از کارایی مناسبی نیز برخوردار است. در روش پیشنهادی با بلوک‌بندی تصویر و اعمال جایگزینی و جایگشتی و عمل XOR تصویر رمز می‌شود. برای جایگزینی و جایگشتی از شبکه میان ارتباطی Beness الگوبرداری شده است. در الگوریتم پیشنهادی به

* رایانامه نویسنده مسئول: en.m.moradi.co@gmail.com

1- Real Time

2- Permutation

پنج زیر کلید نیاز داریم که برای تولید آن‌ها از نگاشت آشوب Logistic map استفاده شده است.

ساختار تحقیق بهصورت زیر سازماندهی شده است. در بخش دوم کارهای مرتبط توضیح داده می‌شود، در بخش سوم نگاشت آشوب Logistic Map و در بخش چهارم روش پیشنهادی ارائه‌شده است. در بخش پنجم به ارزیابی روش ارائه‌شده پرداخته می‌شود و در بخش ششم، نتیجه‌گیری با اشاره به پیشنهاداتی برای گسترش در آینده بررسی خواهد شد.

۲- کارهای مرتبط

در سال‌های اخیر، تحقیقات متعددی جهت رمزنگاری تصاویر انجام شده است که به تعدادی از آن‌ها اشاره می‌شود. Liu و همکاران [۵]، رمزنگاری تصویر را براساس فاز تصادفی- تکراری مقادیر پیکسل‌ها بررسی کردند. Guo و همکاران [۶] با استفاده از تبدیل تصادفی- گسسته و تبدیل آرنولد مقادیر پیکسل‌ها، تصاویر رنگی را رمزنگاری کردند. Liu و همکاران [۷]، رمزنگاری تصویر را با استفاده از تبدیل آرنولد و چرخشی مقادیر پیکسل انجام دادند. Tao و همکاران [۸]، با استفاده از تبدیل سری فوریه، تصاویر را رمزنگاری کردند. Zhang و همکاران [۱۰]، رمزنگاری تصویر را با استفاده از جایگشت مکانی پیکسل‌های تصویر انجام دادند. Zhao و همکاران [۱۱] با استفاده از ماتریس ارگودیکی و روش‌های جایگشتی، رمزنگاری تصویر را بررسی کردند. Zhu و همکاران [۱۲] با استفاده از جایگشت بیتی، تصاویر را رمزنگاری کردند. Huang و همکاران [۱۳] با استفاده از به‌هم‌ریختن پیکسل‌ها، تصاویر را رمز کردند. Chen و همکاران [۱۴]، رمزنگاری تصویر را با استفاده از نگاشت آشوب سه‌بعدی انجام دادند. در تحقیقات متعدد از نگاشت logistic Map به‌عنوان نگاشت آشوب استفاده شده است که در این تحقیق از نگاشت آشوب logistic Map استفاده شده است.

۳- نگاشت آشوب Logistic map

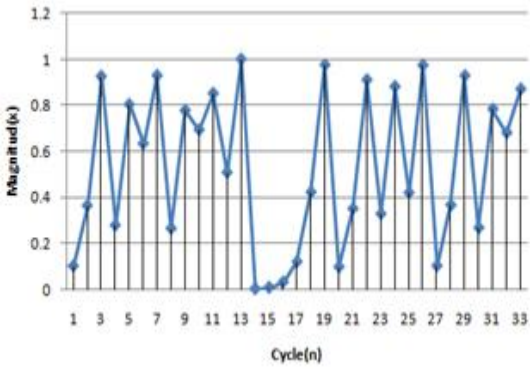
آشوب پدیده‌ای است که در سیستم‌های غیرخطی تعریف‌پذیر رخ می‌دهد که حساسیت زیاد به شرایط اولیه داشته و رفتار شبه تصادفی از خود نشان می‌دهند. رفتار این سیستم‌ها تحت تأثیر دو عامل مهم قرار دارند، مقدار اولیه سیستم و پارامترهای کنترلی سیستم که در واقع مقادیر موثر در تعیین حالت آشوب هستند و با تغییر کنترل‌شده آن‌ها می‌توان سیستم را در حالت آشوب قرار داد و یا از حالت آشوب خارج نمود.

ویژگی مهمی که باعث شده این پدیده برای رمزنگاری بسیار مورد توجه قرار بگیرد، تعریف‌پذیری سیستم در عین رفتار شبه تصادفی آن است که باعث می‌گردد خروجی سیستم از دید حمله‌گران، تصادفی به‌نظر برسد در حالی‌که از دید گشاینده رمز سیستمی، تعریف‌پذیر بوده و لذا قابل رمزگشایی است.

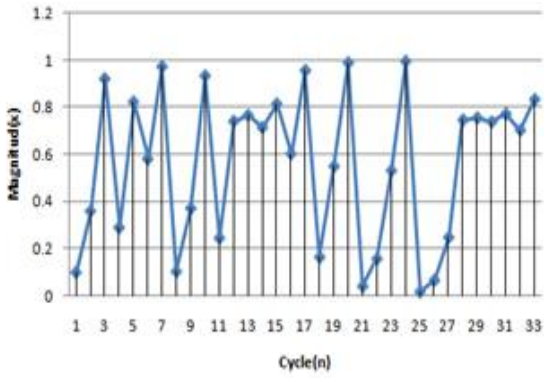
چندین نگاشت آشوب تاکنون ابداع شده‌اند که برخی از آن‌ها در رمزنگاری تصاویر مورد استفاده قرار گرفته‌اند [۲۳-۲۰]. نگاشت logistic Map یکی از نگاشت‌هایی است که در این زمینه استفاده شده است. این نگاشت نخستین بار در [۲۶] ارائه شدکه به‌صورت رابطه (۱) می‌باشد:

$$X_{n+1} = rX_n(1 - X_n) \qquad (۱)$$

X_nها مقادیری بین ۰ و ۱ هستند و r پارامتر کنترلی نگاشت می‌باشد. آزمایشات نشان داده است که مقدار r باید در بازه (3.57,4) باشد تا سیستم، آشوبی عمل کند. این نگاشت به مقدار اولیه X_0 و پارامتر r بسیار وابسته است و با تغییر جزئی در هرکدام، اعداد تولیدی توسط نگاشت کاملاً متفاوت است. شکل‌های (۲-۱) رفتار این نگاشت را برای مقادیر اولیه X_0=۰/۱۰۱ و X_0=۰/۱۰۰۰۰۰۱ تا ۲۰ تکرار اول نشان می‌دهد. در هر دو مورد، r=۳/۹۹۹ درنظر گرفته شده است.



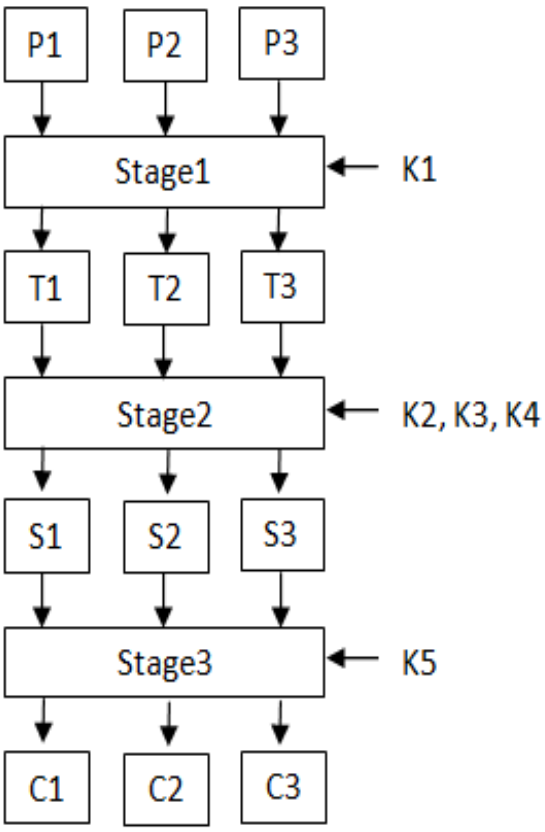
شکل (۱): رفتار نگاشت Logistic Map برای مقادیر اولیه X_0=۰/۱۰۱



شکل (۲): رفتار نگاشت Logistic Map برای مقادیر اولیه X_0=۰/۱۰۰۰۰۰۱

۴- الگوریتم پیشنهادی

در روش پیشنهادی، تصویری که قرار است رمزنگاری شود به بلوک‌های سه‌بایتی تقسیم شده است. هر بلوک به‌طور جداگانه و در سه مرحله رمزنگاری می‌شود. درصورتی‌که بلوک آخر کمتر از سه بایت شود، این بلوک رمزنگاری نمی‌شود. شکل (۳)، شمای کلی رمزنگاری هر بلوک را نمایش می‌دهد.



شکل (۳): شمای کلی الگوریتم رمزنگاری

در این شکل، P1، P2 و P3 بلوک سه‌بایتی هستند که قرار است رمزنگاری شود. C1، C2 و C3 حاصل رمزنگاری این بلوک سه بایتی هستند. Stage1، Stage2 و Stage3 مراحل سه‌گانه الگوریتم می‌باشند و K1 تا K5 نیز کلیدهایی هستند که در این مراحل مورد استفاده قرار می‌گیرند. اعمالی که در هر مرحله انجام می‌شود به صورت زیر می‌باشند:

مرحله اول (Stage1): در این مرحله، سه بایت P1، P2 و P3 به‌عنوان ورودی دریافت می‌شود، سپس به کمک یک کلید ۱۲ بیتی و طی فرآیندی برخی از بیت‌های این سه بایت با هم جابه‌جا شده و سه بایت جدید T1، T2 و T3 حاصل می‌شود.

فرآیند جابه‌جایی بیت‌ها را می‌توان به صورت شکل (۴) نمایش داد. در این شکل، همان‌طورکه مشاهده می‌شود از طبقه اول شبکه میان ارتباطی Beness الگوبرداری شده است، با این تفاوت که به‌جای سوئیچ‌های شبکه Beness از مالتی‌پلکسر استفاده شده است. همان‌طورکه در شکل دیده می‌شود، P_{10} تا P_{17} بیت‌های صفر تا هفت بایت P1 می‌باشند و همچنین P_{20} تا P_{27} بیت‌های بایت P2 و P_{30} تا P_{37} بیت‌های بایت P3 می‌باشند. برای بایت‌های T1، T2 و T3 نیز همین روال برقرار است. K_{10}، K_{12} ... K_{111} نیز دوازده بیت کلید مرحله اول (K1) هستند.

روال کار این مرحله بدین صورت است که هرکدام از بیت‌های کلید کنترل دو تا از مالتی‌پلکسرها را بر عهده دارند. ورودی هرکدام از این زوج مالتی‌پلکسرها[1] دو بیت از ورودی است. به‌عنوان مثال، K_{10} کنترل مالتی‌پلکسرهای اول (ازسمت چپ) و سیزدهم را بر عهده دارد. ورودی هر دوی این مالتی‌پلکسرها، بیت‌های P_{10} و P_{24} می‌باشد. با این تفاوت که در یکی، P_{10} را به ورودی اول مالتی‌پلکسر و در دیگری، P_{10} را به ورودی دوم آن وصل می‌کند. به این ترتیب، اگر مقدار K_{10} برابر ۱ باشد، جای دو بیت P_{10} و P_{24} با هم عوض می‌شود و اگر K_{10} برابر ۰ باشد، جای آن‌ها تغییری نمی‌کند. به عبارتی، می‌توان رابطه‌های ۳-۲ را نوشت:

$$T10 = (K10 \times P24) + (K10 \times P10) \quad (۲)$$

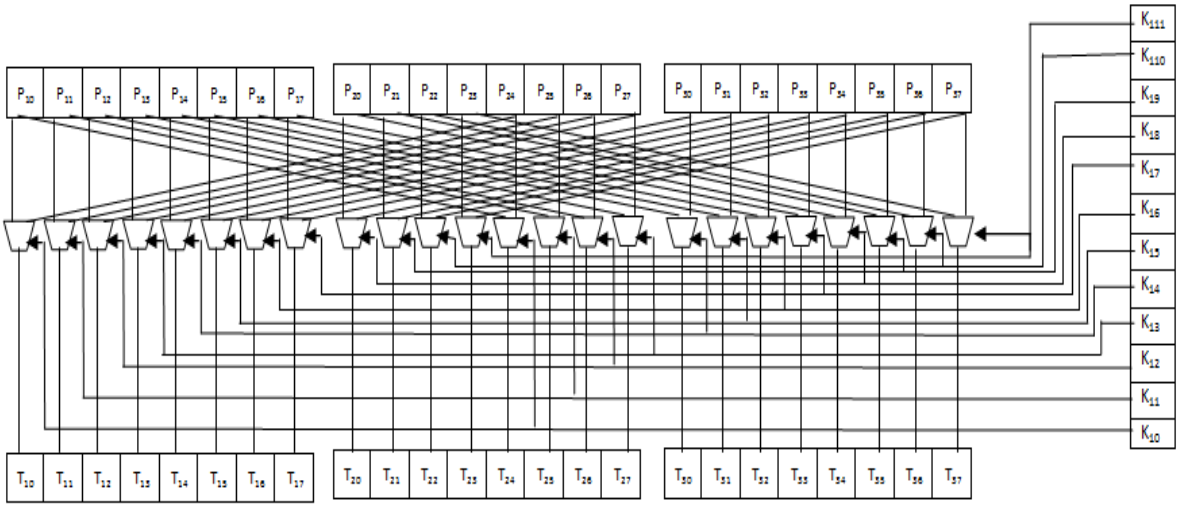
$$T24 = (K10 \times P10) + (K10 \times P24) \quad (۳)$$

در این رابطه‌ها، ضرب معادل AND و جمع معادل OR درنظر گرفته می‌شود. همان‌طورکه از روابط (۳-۲) مشخص است، مقدار K_{10} تعیین‌کننده جابه‌جایی بیت‌های P_{10} و P_{24} می‌باشد. برای سایر بیت‌های ورودی نیز همین رابطه‌ها وجود دارند:

$$T11 = (K11 \times P25) + (K11 \times P11)$$
$$T25 = (K11 \times P11) + (K11 \times P25)$$
$$T12 = (K12 \times P26) + (K12 \times P12)$$
$$T26 = (K12 \times P12) + (K12 \times P26)$$
$$T13 = (K13 \times P27) + (K13 \times P13)$$
$$T27 = (K13 \times P13) + (K13 \times P27)$$

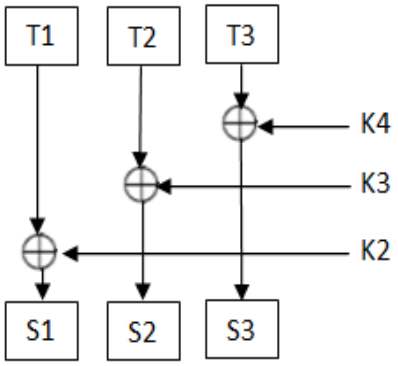
به‌عنوان مثالی دیگر، K11 تعیین‌کننده جابه‌جایی بیت‌های P11 و P25 می‌باشد، اگر K11 یک باشد جای این دو بیت عوض می‌شود وگرنه این دو بیت مستقیماً به خروجی منتقل می‌شوند.

1 - Multiplexer



شکل (۴): شبکه میان ارتباطی Beness

مرحله دوم (Stage2): در این مرحله، سه بایت T1، T2 و T3 به‌عنوان ورودی، دریافت شده و این سه بایت با سه کلید K2، K3 و K4، XOR شده و سه بایت S1، S2 و S3 به‌عنوان خروجی حاصل می‌شود (کلیدها هشت بیتی هستند). شکل (۵)، روند کار این مرحله را نمایش می‌دهد.



شکل (۵): کلیدهای هشت بیتی

مرحله سوم (Stage3): در این مرحله، سه بایت S1، S2 و S3 به‌عنوان ورودی دریافت‌شده و با استفاده از شیفت چرخشی به راست، مکان این سه بایت با هم عوض شده و سه بایت خروجی C1، C2 و C3 حاصل می‌شوند. این مسئله که بایت‌های ورودی چند بار شیفت داده شوند، کلید این مرحله (K5) تعیین می‌کند. اگر K5=۰ باشد، اصلاً شیفتی انجام نمی‌شود و ورودی‌ها مستقیماً به خروجی می‌روند. اگر K5=۱ باشد، آن‌گاه ورودی‌ها یک بایت به چپ شیفت پیدا می‌کنند و اگر K5=۲ باشد، ورودی‌ها دو بایت به چپ شیفت چرخشی پیدا می‌کنند.

فرآیند تولید زیرکلیدها: در روش پیشنهادی، از یک کلید N بایتی (KEY) برای رمزنگاری تصویر استفاده می‌شود. با استفاده از این کلید و به کمک نگاشت آشوب Logistic Map برای رمزنگاری هر بلوک از داده‌ها، پنج زیرکلید تولید می‌شود و این پنج کلید برای هر بلوک با بلوک دیگر متفاوت است.

روال کار به این صورت است که ابتدا به کمک کلید اصلی الگوریتم، پارامترهای اولیه نگاشت آشوب یعنی X_0 و r تولید می‌شوند. نحوه انجام کار در روابط ۶-۴ نشان داده شده است.

$$X_0 = \frac{KEY0 + KEY1\times 256 + KEY2\times 256^2 + KEY3\times 256^3 \ldots + KEY(N)\times 256^N}{2^{N\times 8}} \quad (۴)$$

$$y = \frac{KEY(N) + KEY(N-1)\times 256 + KEY(N-2)\times 256^{N-2} + \ldots + KEY0\times 256^N}{2^{N\times 8}} \quad (۵)$$

$$r = 0.41y + 3.58 \quad (۶)$$

در این روابط، KEY0، KEY1 ... KEY(N) بایت‌های کم ارزش تا پرارزش تشکیل‌دهنده کلید هستند. حال با استفاده از X_0 و r و به‌کمک نگاشت آشوب Logistic Map اعداد X_1، X_2، X_3 ... X_n تولید می‌شوند. تعداد اعدادی تولیدی برابر است با:

$$n = width \times height \quad (۷)$$

که در آن، width اندازه عرض تصویر و height اندازه ارتفاع تصویر است. هر پنج عدد از این اعداد تولیدی، برای تولید زیرکلیدهای یکی از بلوک‌ها استفاده می‌شوند. نحوه تولید زیرکلیدها در روابط ۱۲-۸ شرح داده شده است.

$$K1 = (X1 \times 10^6) \bmod 4096 \quad (۸)$$

$$K2 = (X2 \times 10^6) \bmod 256 \quad (۹)$$

$$K3 = (X3 \times 10^6) \bmod 256 \quad (۱۰)$$

$$K4 = (X4 \times 10^6) \bmod 256 \quad (۱۱)$$

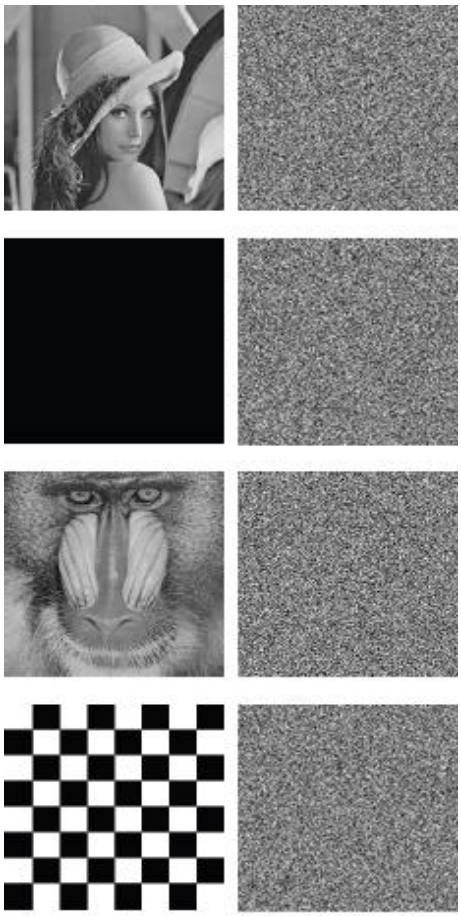
$$K5 = (X5 \times 10^6) \bmod 3 \quad (۱۲)$$

در این روابط، اعداد تولیدشده در رابطه آشوب اعـدادی بـین صفر تا یک هستند و ما آن‌ها را در $۱۰^۶$ انتخاب کرده تـا محـدوده آن‌ها به اعداد بزرگتر از یک تغییر یابـد. X_1 تـا X_5 پـنج عـددی هسـتند کـه بـرای هـر بلـوک اسـتفاده مـی‌شـود و K1 تـا K5 زیرکلیدهای لازم برای هر بلوک می‌باشد. Mod نیز عملگر پیمانـه می‌باشد.

۵- ارزیابی الگوریتم پیشنهادی

در این بخش با انجام چند آزمایش کـارایی الگـوریتم نشـان داده می‌شود. این ارزیابی در چند بخش انجام می‌شود.

۵-۱- ارزیـابی بصـری (Visual Testing): بـرای انجـام ایـن آزمایش، چهار تصویر خاکستری با اندازه‌های ۲۵۶×۲۵۶ پیکسـل انتخاب شده‌اند. شکل (۶)، تصاویر اصلی و رمزشده بـا اسـتفاده از الگوریتم پیشنهادی را نمایش می‌دهد.



شکل (۶): تصاویر اصلی و رمزشده آن

همان‌طورکه در شکل (۶) مشاهده می‌شود، هیچ شباهت قابل مشاهده‌ای بین تصاویر اصلی و رمزشده آن‌ها وجود نـدارد. بـرای ارزیابی دقیق‌تر بین دو تصویر، معیـار دیگـر UACI نـام دارد کـه برابر است با میانگین شدت تفاوت بین تصویر و معادل رمزشده آن از دو معیار استفاده می‌شود. معیـار اول، NPCR نـام دارد که درصد پیکسل‌های متفاوت پیکسل‌های دو تصویر را نشان می‌دهـد [۲۴]. روابـط ۱۴-۱۳ نحـوه محاسـبه ایـن دو معیـار را نمایش می‌دهد.

$$NPCR=\frac{\sum_{i=1}^{M}\sum_{j=1}^{N} D(i,j)}{M\times N}\times 100\% \quad (۱۳)$$

$$D(i,j)=۰ \text{ if } I_o(i,j)=I_{enc}(i,j) \text{ else } =۱$$

$$UACI=[\sum_{i=1}^{M}\sum_{j=2}^{N}\frac{Io(i,j)-Ienc(i,j)}{255}]\times\frac{100}{M\times N} \quad (۱۴)$$

که در این روابط، I_o تصویر اصلی و I_{enc} تصویر رمزنگاری‌شده است. M و N نیز عرض و ارتفاع تصویر هستند.

حالت مناسب برای این دو پارامتر، به این صـورت اسـت کـه NPCR باید تا جایی‌که ممکن است به مقدار ۱۰۰ ٪ نزدیک باشد و UACI باید نزدیک به مقدار ۳۳ ٪ باشد. در جـدول (۱)، مقـدار این پارامترها را برای چهار تصویر رمزنگاری‌شده در این تحقیـق نمایش می‌دهد.

جدول (۱): میزان تفاوت بین تصاویر اصلی و رمزشده آن‌ها

Image	NPCR(%)	UACI(%)
Lena	۹۹/۶۲	۳۳/۱۱
Babon	۹۹/۶۰	۳۳/۲۳
Black	۹۹/۵۹	۳۳/۳۳
Checkerboard	۹۹/۵۸	۳۲/۹۸

همان‌طورکه مشاهده می‌شود NPCR به حالت ایده‌آل خیلـی نزدیک است. UACI نیز برای دو تصویر اول خـوب اسـت و علـت این‌که دو تصویر دیگر کمی انحراف دارند این است که در ایـن دو تصویر، مقدار پیکسل‌ها یا ۰ است یا ۲۵۵ کـه همـین امـر باعـث شده است که تفاوت مقدار این پیکسل‌ها با تصویر رمزشده آن‌هـا زیاد شود.

۵-۲- فضای کلید: یک الگوریتم خوب باید فضای کلیـد نسـبتاً بزرگی داشته باشد تا در مقابل حمله قوی مقاومت داشـته باشـد. در الگوریتم پیشنهادی، کلید از N بایت تشکیل شـده اسـت و N می‌تواند متغیر باشد. در نتیجه، فضای کلید برابر $2^{N\times 8}$ مـی‌باشـد.

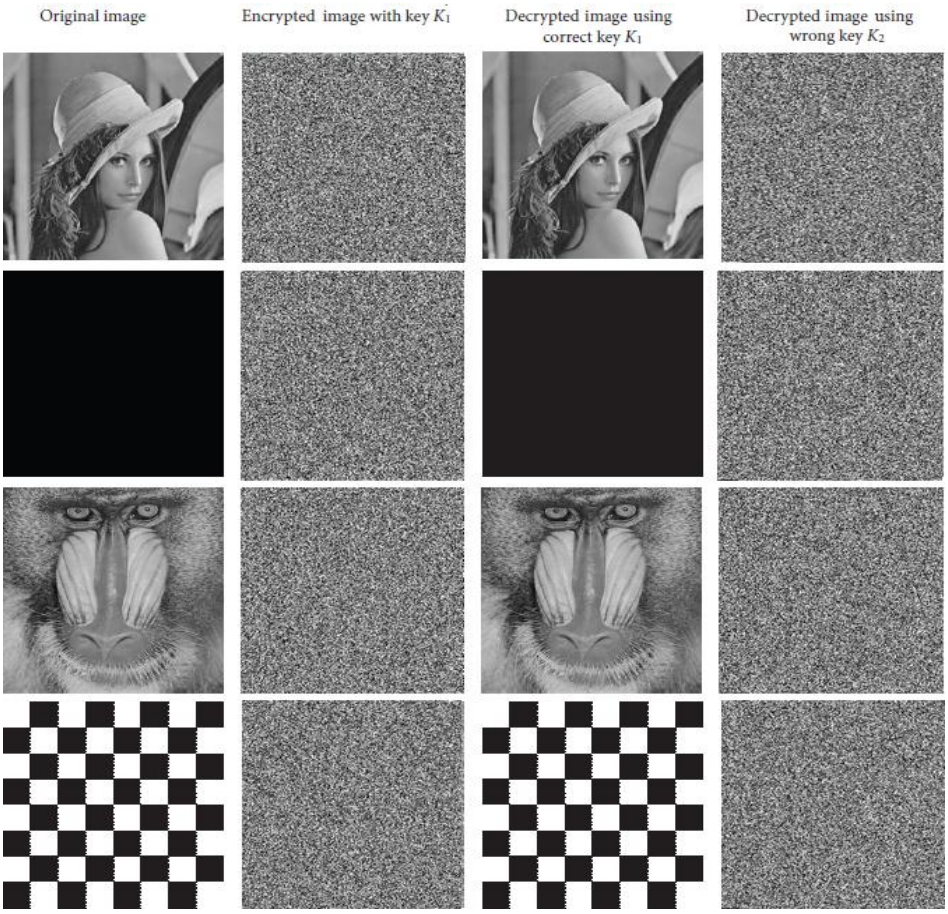
هرچه تعداد بایت‌های کلید بیشتر درنظر گرفته شود، فضای کلید نیز افزایش می‌یابد. کلیدی با طول ۱۰ بایت می‌تواند یک حالت خوب باشد، در این حالت، فضای کلید عددی معادل $۲^{۸۰}$ است که یک عدد مناسب می‌باشد.

۵-۳- حساسیت به کلید: برای نشان‌دادن حساسیت الگوریتم به مقدار کلید، هرکدام از چهار تصویر را با دو کلید K1 و K2 که در یک بیت با هم اختلاف دارند رمزنگاری کرده، سپس تفاوت تصاویر حاصل از لحاظ معیار NPCR و UACI ارزیابی می‌شود و این کار را برای هر تصویر با پنج زوج کلید تکرار کرده و سپس میانگین NPCRها و UACIها برای هر تصویر محاسبه می‌شود که در جدول (۲) نمایش داده شده است.

جدول (۲): تفاوت بین تصاویر رمزشده با دو کلید متفاوت

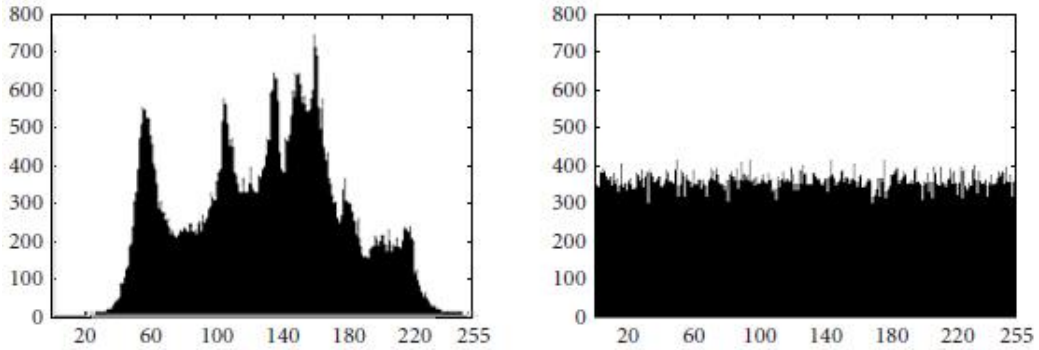
Image	Mean of NPCR(%)	Mean of UACI(%)
Lena	۹۹/۶۴	۲۸/۴۲
Babon	۹۹/۶۳	۲۷/۱۳
Black	۹۹/۶۰	۴۹/۸
Checkerboard	۹۹/۶۲	۴۶/۳۶

همان‌طورکه مشاهده می‌شود هر دو معیار به حالت ایده‌آل بسیار نزدیک می‌باشند و این نشان می‌دهد که تصاویر رمزنگاری‌شده با دو کلید که فقط در یک بیت تفاوت دارند کاملاً با هم متفاوتند و این همان خواسته الگوریتم رمزنگاری می‌باشد. در یک آزمایش دیگر، تصاویر را با کلید K1 رمزنگاری کرده، سپس آن‌ها را با دو کلید K1، K2 که K2 در یک بیت با K1 تفاوت دارد را رمزگشایی می‌کنیم. نتیجه آزمایش در شکل (۷) نمایش داده شده است.

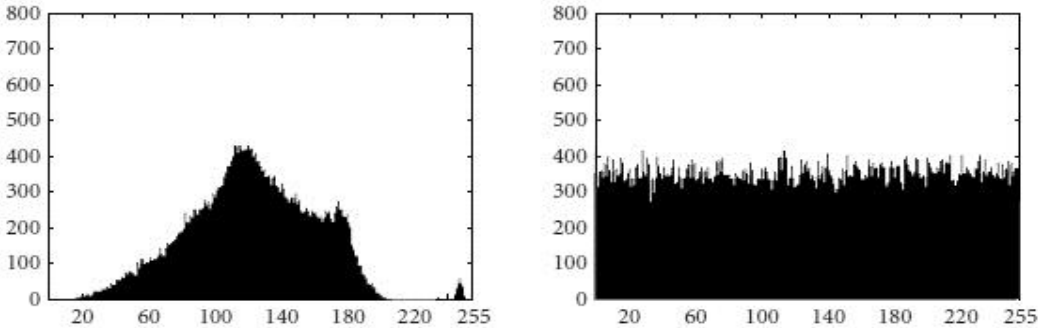


شکل (۷): رمزگشایی تصاویر مختلف با دو کلید متفاوت که فقط در یک بیت با هم تفاوت دارند.

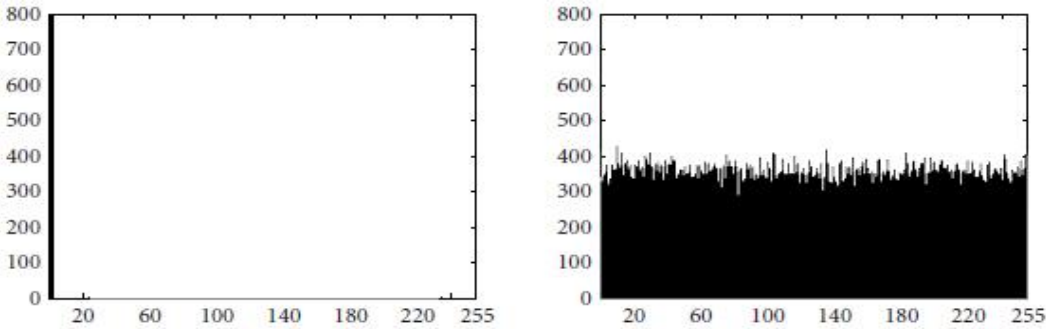
۵-۴- تحلیل آماری: یک الگوریتم رمزنگاری خوب باید الگوهای آماری تصویر را کاملاً به‌هم بریزد. برای ارزیابی این ویژگی معمولاً دو پارامتر نمودار هیستوگرام و تحلیل وابستگی مورد استفاده قرار می‌گیرد [۲۴]. در اشکال (۱۱-۸)، نمودار هیستوگرام تصاویر و تصویر رمزشده آن‌ها با هم مقایسه شده است.



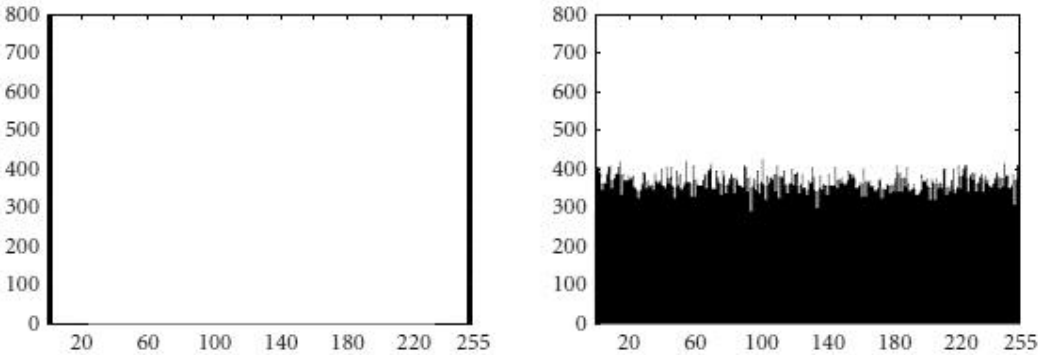
شکل (۸): هیستوگرام تصویر Lena و رمزشده آن



شکل (۹): هیستوگرام تصویر Babon و رمزشده آن



شکل (۱۰): هیستوگرام تصویر Black و رمزشده آن



شکل (۱۱): هیستوگرام تصویر صفحه شطرنج و رمزشده آن

همان‌طورکه از تصاویر قبل قابل مشاهده است، هیستوگرام تصاویر رمزشده کاملاً یک‌دست است. به‌عبارتی، الگوریتم همه ۲۵۵ رنگ را با تعدادی تقریباً برابر در تصویر توزیع کرده است و نمی‌توان از شمارش تعداد پیکسل‌های دارای یک رنگ خاص، اطلاعاتی از تصویر به‌دست آورد.

در یک تصویر، بین رنگ پیکسل‌های مجاور یک وابستگی وجود دارد و یک الگوریتم خوب باید این وابستگی‌ها را از بین ببرد یا به حداقل مقدار خود برساند. برای محاسبه این وابستگی، تعداد N زوج پیکسل مجاور (عمودی، افقی یا قطری) به‌صورت تصادفی از یک تصویر انتخاب شده، سپس به‌کمک روابط ۱۸-۱۶، میزان وابستگی بین آن‌ها محاسبه می‌شود.

$$E(x) = \frac{1}{N}\sum_{i=1}^{N} x_i \qquad (۱۶)$$

$$D(x) = \frac{1}{N}\sum_{i=1}^{N}(x_i - E(x))^2 \qquad (۱۷)$$

$$cov(x,y) = \frac{1}{N}\sum_{i=1}^{N}\big(x_i - E(x)\big)\big(y_i - E(y)\big) \qquad (۱۸)$$

$$\Upsilon_{xy} = \frac{cov(x,y)}{\sqrt{D(x)}\sqrt{D(y)}} \qquad (۱۹)$$

در این عبارات، x_i و y_i مقدار سطح خاکستری پیکسل‌های انتخاب‌شده است. یک الگوریتم خوب باید تا جایی‌که ممکن است این وابستگی‌ها را کم کند (نزدیک به صفر). جدول (۳) میزان محاسبه‌شده وابستگی بین پیکسل‌ها را برای تصاویر و رمزشده آن‌ها نشان می‌دهد. همان‌طورکه در این جدول قابل مشاهده است، در تصاویر رمزنگاری‌شده، مقدار وابستگی‌ها به مقدار ایده‌آل صفر بسیار نزدیک است که نشان‌دهنده قدرت الگوریتم پیشنهادی جهت از بین‌بردن وابستگی‌های بین پیکسل‌های تصویر می‌باشد.

جدول (۳): میزان وابستگی بین پیکسل‌های تصاویر و رمزشده آن‌ها

نام تصویر	وابستگی عمودی	وابستگی افقی	وابستگی قطری
Lena	۰/۹۵۹۱	۰/۹۲۶۸	۰/۸۸۳۰
Encrypted lena	-۰/۰۰۱۴	۰/۰۰۹۳	-۰/۰۰۴۶
Baboon	۰/۸۹۱۴	۰/۸۸۳۲	۰/۷۸۵۵
Encrypted Baboon	۰/۰۱۶۵	-۰/۰۰۰۵	۰/۰۰۲۰
Black	۱	۱	۱
Encrypted Black	-۰/۰۰۸۳	۰/۰۰۳۱	۰/۰۱۴۵
Checkerboard	۰/۹۵۵۲	۰/۹۵۴۲	۰/۹۳۲۳
Encrypted checkerboard	-۰/۰۰۳۳	۰/۰۰۷۲	-۰/۰۰۸۰

۵-۵- تحلیل آنتروپی: محاسبه آنتروپی برای الگوریتم‌های رمزنگاری تصاویر، نخستین بار در [۲۵] مطرح شد و معیاری برای محاسبه میزان آشفتگی سطوح خاکستری پیکسل‌های تصویر می‌باشد. برای محاسبه آنتروپی یک تصویر از رابطه (۲۰) استفاده می‌شود.

$$H(S)=\sum_{i=0}^{2N-1} P(Si)\log\left(\frac{1}{P(Si)}\right) \qquad (۲۰)$$

که در آن، N برابر با تعداد سطح خاکستری استفاده‌شده در تصویر (در تصاویر ۸ بیتی، برابر با ۲۵۶ خواهد بود) و P(Si) نشان‌دهنده احتمال وقوع سطح خاکستری iام در تصویر خواهند بود. در حالت ایده‌آل، مقدار آنتروپی برای یک تصویر رمزشده باید ۸ باشد و یک الگوریتم خوب باید بتواند آنتروپی تصویر را به این مقدار نزدیک کند. جدول (۴)، میزان آنتروپی را برای تصاویر استفاده‌شده در این تحقیق و رمزشده آن‌ها نشان می‌دهد.

جدول (۴): آنتروپی تصاویر و رمزشده آن‌ها

نام تصویر	آنتروپی تصویر اصلی	آنتروپی تصویر رمز شده
Lena	۷/۴۰۵	۷/۹۹۷
Baboon	۷/۱۶۵	۷/۹۹۷
Black	۰	۷/۹۹۶
checkerboard	۲/۲۶۴	۷/۹۹۷

همان‌طورکه به‌وضوح دیده می‌شود، مقدار آنتروپی تصاویر رمزشده توسط الگوریتم پیشنهادی به حالت ایده‌آل بسیار نزدیک است. در جدول (۵)، مقایسه‌ای بین مقدار آنتروپی به‌دست‌آمده برای تصویر Lena توسط الگوریتم پیشنهادی در این تحقیق و چند الگوریتم دیگر که همین تصویر را رمز کرده‌اند، ارائه شده است.

جدول (۵): مقایسه بین آنتروپی حاصل از الگوریتم پیشنهادی و چند الگوریتم دیگر

الگوریتم	مقدار آنتروپی
الگوریتم پیشنهادی در این تحقیق	۷/۹۹۷۴
الگوریتم پیشنهادی در[۲۰]	۷/۹۹۶۸
الگوریتم پیشنهادی در[۲۷]	۷/۹۲۶۰
الگوریتم پیشنهادی در[۲۸]	۷/۹۶۹۰
الگوریتم پیشنهادی در[۲۹]	۷/۹۹۵۰
الگوریتم پیشنهادی در[۳۰]	۷/۹۸۹۰

جدول (۵)، نشان می‌دهد که الگوریتم پیشنهادی در این تحقیق، نتیجه بهتری از نظر آنتروپی، نسبت به بقیه الگوریتم‌ها دارد.

۵-۶- زمان اجرای الگوریتم پیشنهادی

در این تحقیق، جهت محاسبه زمان اجرای عملیات رمزنگاری، الگوریتم پیشنهادی روی چهار تصویر با اندازه مختلف اجرا شد. سیستمی با پردازنده یک گیگا هرتز (Pentium IV) مورد استفاده قرار گرفت. جهت محاسبه زمان اجرا، ده مرحله این الگوریتم جهت فرایند رمزنگاری تصاویر اجرا شده است که میانگینی از همگراترین نتایج به‌عنوان زمان اجرا در جدول (۶) نشان داده شده است.

جدول(۶): مقایسه زمان اجرای تصاویر رمز شده با الگوریتم دیگر

اندازه تصویر برحسب تعداد پیکسل(N)	الگوریتم پیشنهادی در [۳۱] (زمان اجرا برحسب ثانیه (S))	الگوریتم پیشنهادی در این تحقیق (زمان اجرا برحسب ثانیه (S))
۲۵۶*۲۵۶	۰/۴	۰/۲۵
۵۱۲*۵۱۲	۱	۰۶۰
۱۰۲۴*۱۰۲۴	۳	۲/۱۵
۲۰۴۸*۲۰۴۸	۴	۲/۸۵

۶- نتیجه‌گیری

در این تحقیق، یک الگوریتم جدید برای رمزنگاری تصاویر خاکستری ارائه شد. در الگوریتم پیشنهادی، از شبکه میان ارتباطی Beness الگوبرداری شده است که تصاویر را به بلوک‌هایی تقسیم کرده و سپس هر بلوک را در سه مرحله رمزنگاری می‌کند. برای رمزنگاری هر بلوک، به پنج زیرکلید نیاز است که برای تولید این زیرکلیدها از نگاشت آشوب Logistic Map استفاده شده است. در این تحقیق، با آزمایشات مختلف، نشان داده شد که الگوریتم پیشنهادشده از کارایی مناسب برخوردار است. در آزمایش اول با رمزنگاری چهار تصویر مختلف قدرت الگوریتم از نظر بصری ارزیابی شد. حساسیت به کلید نیز با دو معیار NPCR و UACI بررسی شد که به مقدار ایده‌آل بسیار نزدیک بود. در تحلیل آنتروپی روش پیشنهادی با چند تحقیق مقایسه و مشاهده شد که روش پیشنهادی عملکرد بهتری دارد. در نهایت، کاهش زمان اجرای این الگوریتم روی تصاویری با اندازه‌های مختلف در مقایسه یا الگوریتم‌های دیگر نشان داده شد. با توجه به تنوع روش‌های رمزنگاری تصویر، در آینده روش‌های ترکیبی مختلفی در رمزنگاری تصویر بررسی خواهد شد.

۷- مراجع

[1] S. Sukalyan and S. Sayani, "A Non-adaptive Partial Encryption of Grayscale Images Based on Chaos," First International Conference on Computational Intelligence: Modeling Techniques and Applications (CIMTA), vol. 10, pp. 663-671, 2013.

[2] H. Khanzadi, M. Eshghi, Sh. Etemadi Borujeni, "Image Encryption Using Random Bit Sequence Based on Chaotic Maps," Springer, vol. 39, no. 2, pp. 1039–1047, February 2014.

[3] N. l. Pareek, V. Patidar, and K. Sud, "Image encryption using chaoticlogistic map," Image Vision Comput., vol. 24, no. 9, pp. 926–934, 2006.

[4] N. K. Pareek, V. Patidar, K. K. Sud, "Image encryption using chaotic logistic map," Image and Vision Computing, vol. 24, no. 9, pp. 926–934, September 2006.

[5] Z. Liu, L. Xu, C. Lin, J. Dai, and S. Liu, "Image encryptionscheme by using iterative random phase encoding in gyratortransform domains," Optics and Lasers in Engineering, vol. 49, no. 4, pp. 542–546, 2011.

[6] Q. Guo, Z. Liu, and S. Liu, "Color image encryption by using arnold and discrete fractional random transforms in IHSspace," Optics and Lasers in Engineering, vol. 48, no. 12, pp. 1174–1181, 2010.

[7] Z. Liu, H. Chen, T. Liu, et al., "Image encryption by using gyrator transform and arnold transform," Journal of Electronic Imaging, vol. 2, no. 4, pp. 345–351, 1993.

[8] R. Tao, X. Y. Meng, and Y. Wang, "Image encryptionwithmulti-orders of fractional fourier transforms," IEEE Transactions on Information Forensics and Security, vol. 5, no. 4, pp. 734–738, 2010.

[9] R. Zunino, "Fractal circuit layout for spatial decorrelation ofimages," Electronics Letters, vol. 34, no. 20, pp. 1929–1930, 1998.

[10] G. Zhang and Q. Liu, "A novel image encryption method based on total shuffling scheme," Optics Communications, vol. 284, no. 12, pp. 2775–2780, 2011.

[11] X.-Y. Zhao and G. Chen, "Ergodic matrix in image encryption," in Proceedings of the 2nd International Conference on Image and Graphics, vol. 4875, pp. 394–401, August 2002.

[12] Z.-L. Zhu, W. Zhang, K.-W. Wong, and H. Yu, "A chaosbasedsymmetric image encryption scheme using a bit-levelpermutation," Information Sciences, vol. 181, no. 6, pp. 1171–1186, 2011.

[13] C. K. Huang and H. H. Nien, "Multi chaotic systems basedpixel shuffle for image encryption," Optics Communications, vol. 282, no. 11, pp. 2123–2127, 2009.

[14] G. Chen, Y. Mao, and C. K. Chui, "A symmetric image encryptionscheme based on 3D chaotic cat maps," Chaos, Solitons and Fractals, vol. 21, no. 3, pp. 749–761, 2004.

[15] X. Y. Wang, L. Yang, R. Liu, and A. Kadir, "A chaotic imageencryption algorithm based on perceptron model," Nonlinear Dynamics, vol. 62, no. 3, pp. 615–621, 2010.

[16] Y. Wang, K. W. Wong, X. Liao, and G. Chen, "A newchaos-based fast image encryption algorithm," Applied Soft Computing Journal, vol. 11, no. 1, pp. 514–522, 2011.

[17] G. Dimitrakopoulos, Ch. Mavrokefalidis, K. Galanopoulos, and D. Nikolos, "Fast Bit Permutation Unit for Media Enhanced Microprocessors," IEEE, September 2006.

[18] K. D. Patel and S. Belani, "image encrption using differnt techniques: areview, "International Journal of Emerging Technology and Advanced Engineering (IJETAE), vol. 1, no.1, pp. 30-34, Nov. 2011.

[19] S. P. Indrakanti, P. S. Avadhani, "Permutation based Image Encryption Technique", International Journal of Computer Applications(IJCA), vol. 28, no.8, August 2011.

[20] K. Loukhaoukha & et al, "A Secure Image Encryption Algorithm Based on Rubik's Cube Principle", Journal of Electrical and Computer Engineering, vol. 2012, 2012.

[21] R. Rhouma, S. Belghith, "Cryptanalisis of a new image encryption algorithm based on hyper-chaos", physics letters , pp 5973-5978, 2008.

[22] L. Kocarev, "Chaos-based cryptography: a brief overview", IEEE, PP.6–21. 2001.

[23] Ponomarenko, VI., Prokhorov, MD., "Extracting information masked by the chaotic signal of a time-delay system", Phys Rev E2002; pp. 66:026215–21, 2002.

[24] G. Chen, Y. Mao, and C. K. Chui, "A symmetric image encryptionscheme based on 3D chaotic cat maps," Chaos, Solitons and Fractals, vol. 21, no. 3, pp. 749–761, 2004.

[25] O. Edward, "Chaos in Dynamical Systems", Cambridge UniversityPress, Cambridge, UK, 2nd edition, 2003.

[26] R. M. May, "Simple mathematical model with very complicateddynamics." Nature, vol. 261, pp. 459-467, 1976.

[27] M. S. Baptista, "Cryptography with chaos," Physics Letters, Section A, vol. 240, no. 1-2, pp. 50–54, 1998.

[28] K. W. Wong, S. W. Ho, and C. K. Yung, "A chaotic cryptographyscheme for generating short ciphertext," Physics Letters, Section A, vol. 310, no. 1, pp. 67–73, 2003.

[29] T. Xiang, X. Liao, G. Tang, Y. Chen, and K.W.Wong, "A novelblock cryptosystem based on iterating a chaotic map," Physics Letters, vol. 349, no. 1–4, pp. 109–115, 2006.

[30] Z. Lin and H. Wang, "Efficient image encryption using a chaos-based PWL memristor," IETE Technical Review, vol. 27, no. 4, pp. 318–325, 2010.

[31] C. Guanrong , M. Yaobin and k. Charles, "A symmetric image encryption scheme based on 3D chaotic cat maps," Chaos Solitons and Fractals, vol. 21, pp. 749–761, 2004.

A New Approach For Grayscale Image Encryption Using Beness Interconnection Networks And Logistic Map Chaos Function

M. Ahmadipari, M. Moradi*

*Malayer University

(Received: 23/08/2016 , Accepted: 11/04/2017)

ABSTRACT

With the development of information and communication in the last two decades, using computer networks at various levels and, as the result, the need of transferring digital images in this network have been considered important more than past. To protect images against unauthorized access, image encryption is used. Many algorithms have provided to encrypt the image. In this study, a new algorithm for image encryption is designed and implemented. In the proposed method, a new, and powerful algorithm is presented for encrypting images. This is done through combining the Logistic map functions chaos and Xor action and inspiriting from the Beness network. Test results also show that the proposed algorithm has a proper performance.

Keywords: Image Encryption, Logistic Map, Beness Interconnection Network, Image Analysis

* Corresponding Author Email: en.m.moradi.co@gmail.com