ICWR2020

کنفرانس بین المللی وب پژوهی
ایران، تهران، ۲۲ خرداد ۱۳۹۹

Introducing a High-Capacity Steganography Technique for Simultaneous Hide of Multiple Images

Zeinab Torabi [1], Seyyed Vahhab Shojaeddini [2]

[1] Master of Computer Engineering, Alzahra University, Tehran, Iran
ztorabi08@gmail.com

[2] Associate Professor in Biomedical Engineering, Iran Research Organization for Science and Technology (IROST), Tehran, Iran
shojadini@irost.ir

Abstract

The growth of information and communications technology and consequently need to secure data transfer has made steganography as an attractive and essential issue in recent years. Steganography techniques try to hide the original message in other forms of data, the most common of them are digital images. Although the methods based on Least Significant Bit (i.e. LSB) are widely used as the basis for many steganography algorithms in spatial domain, but some challenges such as limited capacity and efficiency may hamper the usability of these methods. This paper presents a modified version of the LSB based methods which hides the original message by utilizing the concept of transferring to even quantities in host image. Implementation and testing of the proposed method shows that this scheme may significantly increase the signal-to-noise ratio of the hiding procedure as well as the structural similarity compared to alternative methods.

Keywords: Steganography, Least Significant Bit Method, Peak Signal To Noise Ratio, Embedding Capacity.

JCWR2020
کنفرانس بین المللی وب پژوهی
ایران، تهران، ۲۲ خرداد ۱۳۹۹

ارایه یک روش پنهان نگاری با ظرفیت بالا به منظور مخفی سازی همزمان چندین تصویر

زینب ترابی [۱]، سید وهاب شجاع الدینی [۲]

[۱] دانشجوی کارشناسی ارشد مهندسی کامپیوتر، دانشگاه الزهرا، تهران، ایران
ztorabi08@gmail.com

[۲] دانشیار مهندسی برق، سازمان پژوهش‌های علمی و صنعتی ایران، تهران، ایران
shojadini@irost.ir

چکیده

رشد روزافزون فناوری اطلاعات و ارتباطات و لزوم انتقال امن داده‌ها، پنهان‌نگاری را به عنوان موضوعی جذاب و ضروری در سالیان اخیر مطرح نموده است. در فرآیند پنهان‌نگاری عمدتاً یک داده را در دیگر شکل‌های داده، مثل فایل‌های عکس یا متن مخفی می‌کنند که از معروف‌ترین آن‌ها بکارگیری تصاویر گرافیکی به عنوان مکان‌هایی برای انجام چنین اختفایی می‌باشد. اگر چه روش‌های مبتنی بر کم ارزش ترین بیت بعنوان مبنایی برای بسیاری از الگوریتم‌های پنهان‌نگاری در حوزه فضایی استفاده می‌شوند؛ ولیکن چالش‌هایی مانند ظرفیت و کارآیی عواملی بوده‌اند که عملکرد این روش‌ها را با محدودیت مواجه نموده‌اند. در این مقاله نسخه اصلاح شده‌ای از روش‌های مبتنی بر کم ارزش ترین بیت ارایه می شود که اساس کار آن بر زوج سازی میزبان به منظور انجام اختفا استوار است. پیاده سازی و آزمون روش پیشنهادی حاکی از آن است که این روش نسبت بیشینه سیگنال به نویز و همچنین شباهت ساختاری را نسبت به روش های رقیب خود به اندازه قابل توجهی افزایش می‌دهد.

کلمات کلیدی

پنهان‌نگاری، روش کم ارزش ترین بیت، نسبت بیشینه سیگنال به نویز، ظرفیت جاسازی

۱- مقدمه

دنیای امروز، عصر فن آوری اطلاعات می‌باشد و بر این اساس توسعه اینترنت و پیشرفت سیستم‌های مخابراتی و کامپیوتری و نرم افزارهای پردازش تصویر، صوت و ویدئو باعث شده است که اطلاعات براحتی و با هزینه ای کم بر روی شبکه انتقال داده شوند. یکی از موضوعات مهم در بحث انتقال اطلاعات دیجیتال، امنیت اطلاعات می‌باشد. برای برقراری یک ارتباط امن، دو روش کلی وجود دارد. روش اول موضوع علم رمزنگاری و روش دوم موضوع علم پنهان نگاری است که شاخه‌ای از علم اختفا اطلاعات می‌باشد [۱].

عملیات پنهان‌نگاری از دو بخش اساسی تشکیل شده است: بخش اول فشرده سازی یا جاسازی و بخش دوم استخراج داده می‌باشد. رسانه می‌تواند هر قالبی اعم از متن، صوت، تصویر و یا فیلم داشته باشد. پیام سری، در رسانه ی میزبان با الگوریتم مناسب گنجانده شده و فایل پنهان‌نگاری شده ی حاصل به فرستنده ارسال می‌گردد [۲]. در هر سیستم پنهان نگاری، سه عامل ظرفیت جاسازی، کشف ناشدنی بودن و مقاومت در برابر حملات از مهم ترین معیارهای سنجش کیفیت به شمار می‌آیند[۳]. ظرفیت جاسازی مقدار اطلاعات سری است که می‌تواند درون رسانه میزبان جاسازی شود. بدین ترتیب، الگوریتمی ظرفیت بالاتری دارد که توانایی بیشتری برای گنجاندن اطلاعات سری داشته باشد. از سوی دیگر، کشف ناشدنی بودن عبارت است از این که اطلاعات سری به گونه ای درون میزبان پنهان باشد که تصادفا توسط کسی دیده نشود والا پنهان نگاری با شکست مواجه می‌شود [۴]. با توجه به

نیاز روزافزون به محافظت از داده‌های ابری و اطلاعات موجود در شبکه اینترنت در برابر مهاجمان، هرگونه بهبودی در پارامترهای مذکور منجر به یک ارتقای قابل توجه در فرآیند پردازش داده‌های بزرگ و انتقال اطلاعات گردد.

روش‌های مبتنی بر کم ارزش ترین بیت[۱]، عملاً بعنوان مبنایی برای طیف وسیعی از الگوریتم‌های معمول پنهان نگاری حوزه فضایی[۲] استفاده می‌شوند [۵]. در تصاویر دیجیتال هر پیکسل تصویر از سه بایت تشکیل شده است، که میزان رنگ‌های قرمز، سبز و آبی آن تصویر را نشان می‌دهند. الگوریتم n-bit LSB، تعداد n بیت با اهمیت کم‌تر از تصویر میزبان را، با بااهمیت‌ترین n بیت پیام جایگزین می‌نماید. با استفاده از این روش تغییرات رنگ میزبان (پس از جاسازی پیام) به حداقل می‌رسد. مراحل استخراج پیام نیز با خواندن کم اهمیت ترین n بیت و انجام شیفت بیتی، به آسانی صورت می‌گیرد. با وجود آنکه اعوجاج ناشی از این کار قابل تشخیص با چشم نیست، این روش در برابر حملات پنهان شکنی بسیار آسیب پذیر می‌باشد[٦]، چراکه تحلیل‌های آماری براحتی می‌تواند الگوی پیکسل‌های تغییر یافته را تشخیص دهد.

بر این اساس تا کنون پژوهش‌های مختلفی به منظور افزایش ظرفیت در سیستم‌های پنهان نگار مبتنی بر ایده فوق، صورت پذیرفته است. در پاره‌ای از پژوهش‌ها، برای غلبه بر محدودیت‌های روش LSB، روش‌های HRVSS[۳] مختلف [۷] و [۸] استفاده شده‌اند که به نوعی از رفتار بیولوژیکی چشم‌های انسان برای پنهان نمودن یک تصویر سیاه و سفید در یک تصویر رنگی بهره می‌برند. در این روش‌ها ابتدا تصاویر میزبان و پیام به رمز در آورده شده، سپس تصویر رمزی میزبان از حالت نمایش رنگ RGB[۴] به HSV[۵] تبدیل می‌گردد. در نهایت تصویر رمزی پیام در شدت رنگ میزبان گنجانده می‌شود. بااین حال این روش‌ها در برابر فشرده‌سازی و اعوجاج بسیار حساس می‌باشند.

کارهای بسیاری از جداسازی پیچیدگی صفحه بیتی[۶] در هر یک از دو حوزه‌ی فضایی و تبدیل[۷] استفاده می‌کنند [۹،۱۰،۱۱]. در این روش‌ها ابتدا صفحه بیتی تصویر میزبان را با یک معیار مناسب به دو قسمت ساده و پیچیده تقسیم می‌کنند و سپس پیام را در قسمت پیچیده پنهان می‌نمایند. با این وجود کاهش کیفیت تصاویر پنهان‌نگاری شده با این روش‌ها قابل توجه است. سایر الگوریتم‌ها [۱۲،۱۳،۱٤،۱۵] پیام سری را در حوزه DCT[۸] و با تغییر مولفه‌های DCT، جاسازی می‌کنند. اما از آنجا که بسیاری از ضرایب DCT برابر با صفر می‌باشند، تبدیل مقادیر زیادی صفر به غیر صفر، موجب اعوجاج زیاد در میزبان می‌شود. بنابراین حوزه DCT نسبت به حوزه فضایی، ظرفیت بسیار کم‌تری دارد [۱٦]. استفاده از تبدیل موجک نیز ایده تعدادی از روش‌های پنهان نگاری در حوزه فرکانس می‌باشد. در [۱۷] با استفاده از تبدیل موجک میزبان، اطلاعات فرکانس بالا و پایین هر پیکسل جدا می‌شوند. سپس بیت‌های سرّی در سه مولفه‌ی فرکانس بالاتر پنهان می‌شوند. بنابراین تاثیرات بر روی میزبان به حداقل می‌رسند. اما این روش ظرفیت کمی دارد. در [۱۸] ابتدا تبدیل موجک بر روی میزبان و پیام انجام شده، سپس با استفاده از تبدیل‌های فوریه و SVD[۹]، عمل پنهان‌نگاری انجام می‌شود. این روش از لحاظ کشف نشدنی بودن وضعیت بهتری دارد، اما نسبت به حملات بسیار حساس می‌باشد. در [۱۹] نیز از تبدیل فوریه برای جاسازی داده سرّی استفاده شده است. در این روش با استفاده از فشرده سازی LZW[۱۰] و کدک بخش بندی در درختان سلسله مراتبی (SPIHT[۱۱]) و تکنیک فاز وفقی اصلاح (APM[۱۲]) به بهبود ویژگی کشف ناشدنی بودن کمک شده است.

در این مقاله روشی جدید به منظور به منظور افزایش ظرفیت در سیستم‌های پنهان نگار ارایه می‌شود؛ که اساس آن بر زوج سازی میزبان استوار است. در این روش پس از زوج سازی ماتریس میزبان با کاهش یک واحدی درایه‌های فرد، مقادیر باینری تصاویر سرّی را به این درایه‌ها اضافه می نماییم. بنابراین رخداد سه حالت محتمل می‌باشد؛ ممکن است یک درایه میزبان که قبل از زوج سازی فرد بوده است، با افزایش یافتن یک واحدی ناشی از یکِ باینری پیام سرّی، به حالت اولیه بازگردد، که موجب افزایش اندک PSNR[۱۳] تصویر پنهان نگاری شده می‌شود. امکان محتمل‌تر، زوج ماندن یک درایه‌ی فرد و یا فرد شدن یک درایه‌ی زوج می‌باشد؛ که این حالات موجب کاهش PSNR می‌شوند. بااین حال مقادیر این تغییر بسیار اندک و در بدترین حالت تغییر یک واحدی همه مقادیر ماتریس تصویر پنهان‌نگاری شده می‌باشد. این عمل سبب می‌شود PSNR تصویر پنهان‌نگاری شده نسبت به میزبان، با افزایش میزان بیت‌های پیام سری، تقریبا ثابت بماند. درحالی که در روش LSB با افزایش میزان بیت‌های پیام سری، میزان PSNR به شدت کاهش می‌یابد و به میزان داده بسیار حساس می‌باشد. در روش پیشنهادی با افزودن داده به میزبان، شباهت تصویر پنهان نگاری شده به میزبان، نه تنها کاهش محسوسی نمی‌یابد، بلکه ممکن است افزایش نیز پیدا کند. همچنین روش فوق، قابلیت پنهان نمودن چندین تصویر پیام در میزبان را نیز دارا می‌باشد.

ساختار این مقاله بدین صورت می‌باشد: در بخش دوم، روش مورد نظر برای نهان نگاری به همراه مراحل مختلف آن تشریح می‌گردند. در بخش سوم، روش مزبور بر روی تصاویر مختلف مرجع آزموده شده و نتایج به دست آمده از آن ارزیابی شده و بر این اساس عملکرد آن با روش‌های موجود مقایسه می‌گردد. در نهایت بخش پایانی مقاله نیز به نتیجه‌گیری اختصاص دارد.

۲- تشریح پیشنهادی

روش پیشنهادی این مقاله، تصویر میزبان و تصاویری که قرار است در آن جاسازی شوند را به عنوان ورودی گرفته و عملیاتی را در سطح پیکسل‌ها انجام می‌دهد تا تصویر جاسازی شده در میزبان را ایجاد نماید. برای جاسازی در حوزه فضایی از ایده ای مشابه با پنهان سازی LSB، اما هوشمندانه‌تر استفاده شده است. در این روش اطلاعات تعداد، ابعاد و محتوای پیکسل‌های سرّی، پس از مرتب و باینری شدن، با مقادیر زوج شده‌ی پیکسل‌های میزبان جمع می‌شوند. در این مدل ابعاد تصویر میزبان باید حتما بزرگتر از همه‌ی تصاویر پیام باشد؛ تا بتواند آن‌ها را در خود جای دهد. شکل (۱) مراحل انجام روش پیشنهادی را نشان می‌دهد. در فاز جاسازی، در صورتی که ابعاد تصاویر میزبان و پیام مناسب باشد و میزبان گنجایش تمامی تصاویر سری را داشته باشد، به ترتیب اطلاعات تعداد، ابعاد و مقادیر باینری شده‌ی پیکسل‌های پیام با روشی مشابه با روش رایج کم اهمیت ترین بیت، در میزبان زوج شده ذخیره می‌شود؛ در غیر این صورت کاربر بایستی تصاویر ورودی به الگوریتم را اصلاح نماید. این روش ذخیره (که همان جمع بیت‌های پیام با میزبان به روشی که در ادامه توضیح داده خواهد شد، می‌باشد)، سبب می‌شود که شباهت بین تصویر پنهان نگاری شده و میزبان حفظ شود و عملیات پنهان شکنی[۱۴] با چالش روبرو گردد. در مرحله استخراج تصویر نیز، اطلاعات ذخیره شده بر روی تصویر را به همان ترتیبی که در فاز جاسازی، ذخیره شده‌اند، استخراج گردیده و به صورت تصاویر جداگانه ذخیره می‌شوند.

۲-۱- جاسازی پیام در تصویر میزبان

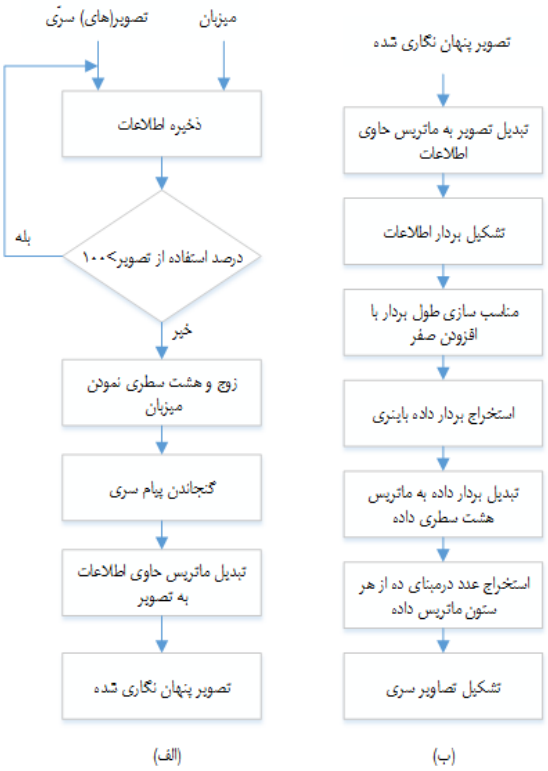
الگوریتم پنهان سازی تصویر در روش پیشنهادی به صورت زیر می‌باشد:

الف. ابتدا تصاویر میزبان و تمامی تصاویر پیام از ورودی دریافت گردیده، به صورت ماتریس‌های عددی نگه داری می‌شوند. سپس برای ذخیره اندازه یا طول، عرض و تعداد کانال‌های هر تصویر پیام به صورت جداگانه، از متغیری استفاده می‌کنیم که تعداد سطور آن برابر با تعداد تصاویر پیام، و برای هر سطر پیام، سه بعد طول، عرض و کانال تعریف می‌شوند.

ب. درصد استفاده شده ی تصویر را محاسبه می‌کنیم و در صورتی که بیشتر از حد مشخصی (در این جا۱۰۰) باشد، از برنامه خارج شده و درغیر این صورت به مرحله بعد می‌رویم. برای محاسبه ی درصد میزان استفاده از تصویر میزبان، ابتدا مجموع تعداد پیکسل‌های همه تصاویر سرّی را با اطلاعات تعداد و اندازه تصاویر تجمیع نموده و آن‌ها را هشت برابر می‌کنیم؛ چرا که برای گنجاندن هر یک از درایه‌های ماتریس پیام، به یک بایت از میزبان، نیاز داریم. سپس این مقدار را بر تعداد پیکسل‌های میزبان تقسیم نموده و درصد آن را برآورد می‌نماییم. در رابطه (۱) a، b، c و m، n و l، به ترتیب طول، عرض و تعداد کانال‌های میزبان و تصاویر پیام و k تعداد تصاویر پیام را نشان می‌دهند.

$$\text{Limit}=\frac{800*((\sum_{i=1}^{k} m_i*n_i*l_i)+4*k+1)}{a*b*c} \qquad (۱)$$

دلیل اضافه نمودن ترم 4k+1 به مجموع پیکسل‌های پیام، وجود اطلاعات افزونه‌ی تعداد و اندازه ی تصاویر پیام می‌باشد. از آنجا که الگوریتم پیشنهادی توانایی تعبیه نمودن همزمان چندین تصویر پیام در پوشه را داراست، باید تعداد این تصاویر را در میزبان تعبیه نمود. برای انجام این کار یک بایت از میزبان را استفاده خواهیم نمود. همچنین به دلیل آنکه هریک از تصاویر پیام می‌تواند اندازه ی متفاوتی را داشته باشد، برای هر تصویر پیام یک ماتریس با ابعاد هشت در چهار تعریف می‌کنیم که سه تا از ستون‌های آن به صورت مجزا، نمایانگر مقدار باینری طول، عرض و کانال پیام به میزان سه بایت و ستون آخر برای جدا نمودن پیام‌ها از یکدیگر و عدم تداخل آن‌ها به صورت تمام صفر در نظر گرفته شده است.



شکل (۱): طرح پیشنهادی. (الف) جاسازی داده. (ب) استخراج داده.

بنابراین جمعا برای هر تصویر سری، چهار بایت اطلاعات به صورت یک ماتریس باینری دارای هشت سطر و چهار ستون را به مقادیر قبلی اضافه می‌نماییم. از این ماتریس ابعاد پیام که تعداد ستون‌های آن چهار برابر تعداد تصاویر پیام می‌باشد، در مراحل بعد نیز استفاده خواهد شد.

ج. در این مرحله برای گنجاندن اطلاعات در میزبان، باید اندازه این ماتریس را متناسب با اطلاعات هشت بیتی خود تغییر داده و مقادیر آن را زوج نماییم. بدین منظور ماتریس میزبان را به ماتریسی باینری تبدیل می‌کنیم که ستون‌های ۸ بیتی (۸ سطر) داشته باشد، بنابراین اگر ابعاد این ماتریس بر ۸ بخش پذیر نباشد، آن را با اضافه نمودن تعداد مناسبی صفر، به ابعاد بخش پذیر بر ۸ می‌رسانیم. در رابطه (۲)، که تعداد صفرهای اضافه شده به میزبان را نشان می‌دهد، mod نمایانگر باقی مانده‌ی تعداد پیکسل‌های تصویر میزبان بر ۸ می‌باشد.

$$number\_of\_zeros=mod((a*b*c),8) \quad (۲)$$

برای راحتی و سادگی رابطه می‌توان در صورتی که تعداد پیکسل‌های تصویر میزبان مضربی از ۸ می‌باشد نیز، ۸ صفر به آن اضافه کرد. همه ی صفرهای اضافه شده به ماتریس برای انجام پذیر نمودن عملیات تعبیه به کار گرفته شده اند و باید پس از گنجاندن پیام در میزبان، حذف شوند.

درنهایت می‌بایست برای انجام عملیات بعدی، ماتریس میزبان را به صورت بخش پذیر بر دو تغییر داد. با انجام این کار، هنگامی‌که میزبان زوج شده با ماتریس باینری پیام جمع شود، در نقاط غیرصفر پیام، میزبان اصلاح شده، فرد می‌شود و براحتی قابل تشخیص می‌باشد. استفاده از این شیوه‌ی نوآورانه‌ی شبه LSB، به جای روش LSB، سبب می‌شود که علاوه بر مزایای یک پنهان سازی مناسب، بتوانیم از ظرفیت بالقوه و بالای میزبان برای تعبیه استفاده نماییم. بنابراین در این روش تنها پیام باینری می‌شود و اعداد ماتریس میزبان فقط در صورتی که فرد باشند، به صورت زوج در خواهند آمد. برای این کار کافی است درایه‌های ماتریس تحت عمل ساده ریاضی ذیل قرار گیرند. در رابطه (۳)، A، ماتریس میزبان و Floor تابع رند کننده به کف (نزدیک ترین عدد صحیح کوچکتر یا مساوی) می‌باشد.

$$B=2*Floor(\frac{A}{2}) \quad (۳)$$

د. در این مرحله عمل گنجاندن پیام سری در میزبان را انجام می‌دهیم. بدین منظور تنها کافی است ماتریس میزبان اصلاح شده را با ماتریس اطلاعات جمع کنیم. ستون اول ماتریس اطلاعات حاوی تعداد عکس‌ها، و ستون‌های بعدی آن به ترتیب شامل ماتریس ابعاد پیام و مقادیر شدت رنگ پیکسل‌های پیام می‌باشد.

برای اضافه نمودن مقادیر پیکسل‌های پیام، هر درایه ماتریس پیام را به یک ستون هشت بیتی –که مقدار باینری آن درایه را نشان می‌دهد– تبدیل کرده و با ستونی از ماتریس میزبان اصلاح شده جمع می‌کنیم. در رابطه (۴)، Secret، B و Stego به ترتیب ماتریس‌های اطلاعات سری، میزبان اصلاح شده و پنهان نگاری شده می‌باشند.

$$Stego = B \oplus Secret \quad (۴)$$

این جمع یک جمع ماتریسی ساده نمی‌باشد؛ چرا که تعداد ستون‌های اطلاعات با میزبان اصلاح یافته الزاما برابر نیست و عمل جمع به صورت ستون با ستون صورت می‌گیرد.

ه. در مرحله آخر صفرهای اضافه شده به انتهای ماتریس میزبان در مرحله سوم را حذف، و ماتریس پنهان نگاری شده را با ابعاد میزبان اولیه و در قالب مناسب، به صورت تصویر ذخیره می‌نماییم.

۲-۲- استخراج پیام

برای استخراج پیام‌های سرّی از تصویر پنهان‌نگاری شده‌ی ورودی، پس از تبدیل آن به ماتریس، اطلاعات تعداد تصاویر پنهان شده و سایز و مقادیر ماتریس‌های هر تصویر پیام به همان صورتی که جاسازی شده است، از تصویر پنهان‌نگاری شده استخراج شده و در قالب تصاویر جداگانه ذخیره می‌گردد. الگوریتم استخراج تصویر نیز به صورت زیر می‌باشد:

الف. ابتدا تصویر پنهان نگاری شده از ورودی دریافت، و به صورت یک ماتریس عددی ذخیره می‌شود.

ب. در این مرحله برای استخراج اطلاعات از تصویر، ماتریس تصویر به همان قالب حالت پنهان سازی بازگشته و با همان ترتیب پیام آشکار می‌شود. بدین منظور ابتدا تمامی درایه‌های ماتریس تصویر به صورت یک بردار اطلاعات در کنار هم قرار می‌گیرند.

ج. سپس طول این بردار با اضافه نمودن تعداد مناسبی صفر به انتهای بردار (همانند فاز پنهان سازی)، به برداری با طول بخش پذیر بر هشت تبدیل می‌نماییم.

د. در این مرحله باقی ماندهی تک تک درایه‌های بردار اطلاعات به دو را محاسبه می‌نماییم، تا یک بردار باینری حاصل شود. از آنجا که در فاز پنهان سازی، اطلاعات سرّی به شکل باینری به ماتریس زوج شده اضافه شده بود، این بردار باینری حاوی همان اطلاعات پیام (ها) می‌باشد.

ه. این بردار را به همان قالب هشت سطری تبدیل می‌نماییم تا اطلاعات را به همان ترتیب که گنجانده شده بودند، استخراج نماییم.

و. مطابق رابطه ذیل، تعداد تصاویر برابر با ستون اول این ماتریس در مبنای ده می‌باشد. در رابطه (۵)، برای به مبنای ده رساندن ماتریس اطلاعات، Num تعداد تصاویر و a_{i1} درایه‌های ستون اول ماتریس اطلاعات می‌باشند.

$$Num=a_{11}*1+a_{21}*2^1+a_{31}*2^2+a_{41}*2^3+...+a_{81}*2^7 \quad (۵)$$

ز. سپس در یک حلقه به تعداد تصاویر، برای هر تصویر چهار ستون بعدی ماتریس اطلاعات را در نظر می‌گیریم. سه ستون اول از چهار ستون ماتریس اطلاعات مربوط به ابعاد (طول، عرض و ارتفاع) تصویر مورد نظر و ستون آخر تمام صفر می‌باشد.

ح. پس از دستیابی به ابعاد تمامی تصاویر گنجانده شده، نوبت به استخراج مقادیر پیکسل‌های آن‌ها می‌رسد. بدین منظور در یک حلقه دیگر به تعداد تصاویر پنهان‌سازی شده، برای هر تصویر یک ماتریس صفر با ابعاد متناظر با آن تصویر ایجاد می‌کنیم. سپس برای پر کردن مقادیر شدت رنگ تصویر مربوطه به جای صفرها، در یک حلقه درونی به ابعاد هر تصویر، ستون‌های مناسب بعدی از ماتریس اطلاعات را مشابه قبل به مبنای ده برده و به عنوان شدت رنگ RGB، مکان‌های مناسب به جای صفرهای ماتریس تصویر قرار می‌گیرند. در نهایت از این حلقه خارج شده و ماتریس تصویر به دست آمده را با ابعاد متناظر آن و در قالب مناسب ذخیره نموده و از حلقه بیرونی نیز خارج می‌شویم.

۳- ارزیابی‌ها

برای ارزیابی کارایی روش پیشنهادی در مقایسه با روش‌های موجود شبیه سازی هایی بر بستر نرم افزار Matlab2018 و با بهره‌گیری از پردازشگر رایانه Intel core2 duo CPU 2.53GHz با رم 4GB انجام شد. مجموعه داده‌های مورد استفاده در این شبیه‌سازی‌ها تصاویر تصادفی وب بوده و محدودیتی برای تعداد کانال‌های (رنگی یا سیاه وسفید بودن) تصویر وجود ندارد. در اولین گونه از مقایسات به منظور آزمودن میزان کشف‌ناشدنی بودن، PSNR تصویر پنهان‌نگاری شده محاسبه شد. PSNR توسط خطای میانگین مربعات (MSE[۱۵]) شده و با فرض داشتن یک تصویر بدون نویز m*n به نام I و تقریب دارای اعوجاج آن با اسم K، خطای میانگین مربعات با رابطه (۶) تعریف می‌شود:

$$MSE = \frac{1}{mn}\sum_{i=0}^{m-1}\sum_{j=0}^{n-1}[I(i,j) - K(i,j)]^2 \quad (۶)$$

بنابراین PSNR (با واحد دسی بل) با رابطه (۷) توصیف می‌شود. در رابطه (۷)، MAX_I بزرگ ترین مقدار احتمالی پیکسل تصویر می‌باشد. هنگامی که تصویر هشت بیتی باشد، این مقدار ۲۵۵ می‌باشد. در حالت کلی زمانی که تصویر با B بیت در نمونه مشخص می‌شود، این مقدار برابر با $2^B - 1$ می‌باشد.

$$PSNR = 10.\log_{10}\left(\frac{MAX_I^2}{MSE}\right) = 20.\log_{10}\left(\frac{MAX_I}{\sqrt{MSE}}\right) =$$

$$= 20.\log_{10}(MAX_I) - 10.\log_{10}(MSE) \quad (۷)$$

شکل (۲) نمونه ای از عملکرد روش پیشنهادی این مقاله را نشان می‌دهد.

به همین منوال در جدول (۱) ملاحظه می‌شود، ۵ تصویر با اندازه‌های متفاوت در تصویر میزبان با اندازه ۱۶۰۰*۹۷۳ تعبیه شده‌اند، اما PSNR کاهش چندانی نداشته است. حتی در بعضی موارد افزایش ناچیزی نیز داشته است. بنابراین برخلاف روش‌های کم اهمیت ترین بیت که میزان PSNR در آن‌ها با افزایش مقدار داده‌ی پیام سری به شدت کاهش می‌یابد، در روش پیشنهادی از

(الف) (ب)

شکل (۲): تصاویر (الف) میزبان. (ب) پنهان نگاری شده

جدول (۱): اطلاعات تصاویر سری تعبیه شده در میزبان. ابعاد میزبان در تمامی آزمایشات ۱۶۰۰*۹۷۳ می باشد.

تعداد تصاویر سری تعبیه شده در میزبان	اطلاعات تصاویر سری و تصویر پنهان‌نگاری شده		
	ابعاد تصاویر سری	PSNR تصویر پنهان‌نگاری شده	درصد استفاده از تصویر پنهان‌نگاری شده
۱	۳۵۲*۲۸۸	۵۱.۲۲۲۱	۱۷.۳۶۵۷
۲	۳۵۲*۲۸۸ ۱۲۸*۱۲۸	۵۱.۲۲۲۹	۲۵.۷۸۵۷
۳	۳۵۲*۲۸۸ ۱۲۸*۱۲۸ ۱۵۹*۱۲۸	۵۱.۲۲۶۷	۳۶.۲۴۴۸
۴	۳۵۲*۲۸۸ ۱۲۸*۱۲۸ ۱۵۹*۱۲۸ ۲۶۷*۱۸۸	۵۱.۲۱۲۴	۶۲.۰۳۹۹
۵	۳۵۲*۲۸۸ ۱۲۸*۱۲۸ ۱۵۹*۱۲۸ ۲۶۷*۱۸۸ ۳۰۰*۱۶۸	۵۱.۱۷۶۷	۸۷.۹۳۹۹

این کاهش چشمگیر در PSNR خبری نیست و میزان آن تقریبا ثابت می‌ماند، یعنی می‌توان بدون نگرانی نسبت به کاهش PSNR، از بالاترین ظرفیت گنجایش داده در تصویر بهره برد.

جدول (۲) نیز مربوط به مقایسه جفت تصاویر میزبان–پنهان‌نگاری شده و تصاویر پیام قبل و بعد از استخراج می‌باشد.

در این جدول علاوه بر معیارPSNR ، از معیارهای SSIM[۱۶] و VIF[۱۷] نیز برای ارزیابی کیفیت تصاویر استفاده شده است. SSIM بر روی پنجره‌های مختلفی از یک تصویر اعمال می شود. مقیاس SSIM برای دو پنجره x و y با ابعاد NدرN به با رابطه (۸) محاسبه می‌شود. در رابطه (۸)، μ_x و μ_y به ترتیب، میانگین و σ_x^2 و σ_y^2، واریانس پنجره‌های x و y می‌باشند. σ_{xy} نیز کواریانس پنجره‌های x و y بوده و $c_1 = (k_1 L)^2$ و $c_2 = (k_2 L)^2$ برای مناسب سازی تقسیم با مقسوم‌علیه کوچک می‌باشند؛ که در آنها L پویایی مقادیر پیکسل‌ها بوده و با $2^b - 1$ مشخص می‌شود و b تعداد بیت‌های نمایش‌دهنده‌ی یک پیکسل و همچنین $k_1 = .01$ و $k_2 = .03$ می‌باشند.

$$SSIM(x,y)=\frac{(2\mu_x\mu_y+c_1)(2\sigma_{xy}+c_2)}{(\mu_x^2+\mu_y^2+c_1)(\sigma_x^2+\sigma_y^2+c_2)} \qquad (۸)$$

مقیاس VIF نیز افزایش واریانس در مقایسه با یک پایه متعامد را اندازه می گیرد. در حقیقت VIF روشی برای نمایش متغیر خاص x_K توسط یک مدل خطی مبتنی بر همه متغیرهای مستقل پایه متعامد می‌باشد. اگر مدل محاسبه شده تناسب بالایی را برای متغیر آزمون x_K نشان دهد، x_K احتمالا با یک یا چند متغیر از متغیرهای پایه‌ها هم‌خط[۱۸] می‌باشد. VIF برای k–امین متغیر توسط رابطه (۹) توصیف می‌گردد. در رابطه (۹)، r_k^2 معیار تناسب مدل خطی برای x_K برمبنای سایر متغیرها می‌باشد.

$$VIF_K = \frac{1}{(1-r_k^2)} \qquad (۹)$$

ملاحظه می شود که روش پیشنهادی از روش‌های مورد مقایسه، عملکرد بهتری داشته است. میزان PSNR این روش از بهترین رقیبش در جدول، به میزان حدودا ۱۲ دسی بل بیشتر بوده است و همچنین SSIM و VIF آن نیز با مقداری بیش از ۰/۲ از این رقیبان بالاتر بوده است.

جدول (۲): مقایسه سنجه‌های کیفیت بر روی جفت میزبان و تصویر پنهان نگاری شده با روش‌های گوناگون

روش	جفت میزبان و تصویر پنهان نگاری شده			
	PSNR	SSIM	VIF	RMSE
4bit-LSB	۳۱.۸۸	۰.۶۲۸۷	۰.۶۱۴۵	۶.۵۹
HRVSS [۷]	۳۹.۹۵	۰.۷۷۳۵	۰.۸۰۳۸	۲.۸۳
Baluja [۵]	۳۸.۹۷	۰.۷۷۹۶	۰.۷۷۵۹	۳.۰۵
پیشنهادی	۵۱.۲۲	۰.۹۹۰۱	۰.۹۶۸	۰.۱۶۸۴

نسبی میزان PSNR تصویر پنهان‌نگاری شده نسبت به میزبان، افزایش داده موجب آشفتگی بیشتر در تصویر پنهان‌نگاری شده نمی‌شود. با وجود ظرفیت بالا و ثبات روش پیشنهادی در مواجهه با افزایش داده‌های سرّی، این روشُ ضعف روش‌های فضایی را که مقاومت پایین دربرابر حملات گوناگون است را نیز دارا می‌باشد؛ و لذا اصلاحات آینده در این روش می‌تواند شامل استفاده از شبکه‌های عصبی عمیق و الگوریتم‌های تکاملی زیستی برای نیل به مقاومت بالاتر در آن، باشد.

٤- نتیجه گیری

در این مقاله، روشی جدید برای پنهان نمودن چندین تصویر در یک میزبان معرفی شد. روش پیشنهادی برمبنای اضافه نمودن مقادیر باینری پیام سرّی به میزبان زوج‌سازی شده استوار می‌باشد؛ که قابلیت مخفی‌سازی همزمان چندین تصویر در میزبان را داراست. این روش نسبت به روش‌های به‌روز حوزه فضایی و تبدیل رقیب مقایسه شده، از ظرفیت بالاتری برخوردار بوده و با توجه به ثبات

مراجع

[1] Mstafa, Ramadhan J., and Khaled M. Elleithy. "*Compressed and raw video steganography techniques: a comprehensive survey and analysis*." Multimedia Tools and Applications 76.20 (2017): 21749-21786.

[2] PrashantJohri, Amba Mishra, Sanjoy Das, Arun Kumar, "*Survey on Steganography Methods (Text, Image, Audio, Video, Protocol and Network Steganography*" 2016 International Conference on Computing for Sustainable Global Development (INDIACom).

[3] R. J. Mstafa and K. M. Elleithy, "*A highly secure video steganography using Hamming code (7, 4)*," in Systems, Applications and Technology Conference (LISAT), 2014 IEEE Long Island, 2014, pp. 1-6.

[4] Jenifer, J. & Ratna, S. & Loret, J.B. & Gethsy, D.. (2018). A Survey on Different Video Steganography Techniques.627-632.10.1109/ICOEI.2018.8553847.

[5] Shumeet Baluja. 2017. Hiding Images in Plain Sight: Deep Steganography. In NIPS.

[6] Daniel Lerch-Hostalot and David Megías. 2016. Unsupervised steganalysis based on artificial training sets. Eng. Appl. of AI 50 (2016), 45–59.

[7] Mohamed Elsadig Eltahir, Miss Laiha Mat Kiah, Bilal Bahaa, and A Zaidan. 2009. High Rate Video Streaming Steganography. In ICIME.

[8] Khan Muhammad, Muhammad Sajjad, Irfan Mehmood, Seungmin Rho, and Sung Wook Baik. 2018. Image steganography using uncorrelated color space and its application for security of visual contents in online social networks. Future Generation Comp. Syst. 86 (2018), 951–960.

[9] Eiji Kawaguchi and Richard O Eason. 1999. Principles and applications of BPCS steganography. In Multimedia Systems and Applications, Vol. 3528.

[10] GMK Ramani, EV Prasad, S Varadarajan, Tirupati SVUCE, and Kakinada JNTUCE. 2007. Steganography using BPCS to the integer wavelet transformed image. IJCSNS 7, 7 (2007), 293–302.

[11] Jeremiah Spaulding, Hideki Noda, Mahdad N Shirazi, and Eiji Kawaguchi. 2002. BPCS steganography using EZW lossy compressed images. Pattern Recognition Letters 23, 13 (2002), 1579–1587.

[12] J. J. Chae and B. S. Manjunath. 1999. Data hiding in video. In ICIP.

[13] Blossom Kaur, Amandeep Kaur, and Jasdeep Singh. 2011. Steganographic approach for hiding image in DCT domain. International Journal of Advances in Engineering & Technology 1, 3 (2011), 72.

[14] Amitava Nag, Sushanta Biswas, Debasree Sarkar, and Partha Pratim Sarkar. 2010. A novel technique for image steganography based on Block-DCT and Huffman Encoding. arXiv preprint arXiv:1006.1186 (2010).

[15] Shun Zhang, Liang Yang, Xihao Xu, and Tiegang Gao. 2018. Secure Steganography in JPEG Images Based on Histogram Modification and Hyper Chaotic System. IJDCF 10, 1 (2018), 40–53.

[16] Sahar A. El-Rahman. 2018. A comparative analysis of image steganography based on DCT algorithm and steganography tool to hide nuclear reactors confidential information. Computers & Electrical Engineering 70 (2018), 380–399.

[17] S. Kamila, R. Roy and S. Changder, "*A DWT based steganography scheme with image block partitioning*," 2015 2nd International Conference on Signal Processing and Integrated Networks (SPIN), Noida, 2015, pp. 471-476.

[18] S. K. Yadav and M. Dixit, "*An improved image steganography based on 2-DWT-FFT-SVD on YCBCR color space*," 2017 International Conference on Trends in Electronics and Informatics (ICEI), Tirunelveli, 2017, pp. 567-572.

[19] A. S. Khashandarag and N. Ebrahimian, "*A New Method for Color Image Steganography Using SPIHT and DFT, Sending with JPEG Format*," 2009 International Conference on Computer Technology and Development, Kota Kinabalu, 2009, pp. 581-586.

زیرنویس‌ها

[1] **L**east **S**ignificant **B**it (LSB)
[2] Spatial domain
[3] **H**igh **R**ate **V**ideo **S**treaming **S**teganography (HRVSS)
[4] **R**ed, **G**reen, and **B**lue (RGB)
[5] **H**ue, **S**aturation, and **V**alue (HSV)
[6] Bit plane complexity segmentation
[7] Transform domain
[8] **D**iscrete **C**osine **T**ransform (DCT)
[9] **S**ingular **V**alue **D**ecomposition (SVD)
[10] **L**empel–**Z**iv–**W**elch (LZW)
[11] **S**et **P**artitioning **I**n **H**ierarchical **T**rees (SPIHT)
[12] **A**daptive **P**hase **M**odulation (APM)
[13] **P**eak **S**ignal to **N**oise **R**atio (PSNR)
[14] Stegano Analysis
[15] **M**ean **S**quare **E**rror (MSE)
[16] **S**tructural **SIM**ilarity (SSIM)
[17] **V**ariance **I**nflation **F**actor (VIF)
[18] Colinear