بهبود امنیت روش پنهان‌نگاری LSBM با استفاده از الگوریتم ژنتیک، چندکلیدی و بلاک‌بندی

وجیهه ثابتی، سیده سپیده فیاضی و حدیثه شیرین‌خواه

چکیده: با افزایش دقت حملات پنهان‌شکنی در کشف روش‌های پنهان‌نگاری، نیاز به بهبود امنیت روش‌های پنهان‌نگاری بیشتر از گذشته احساس می‌شود. LSBM یکی از روش‌های ساده پنهان‌نگاری است که حملات نسبتاً موفقی برای کشف آن تا به حال ارائه شده است. هدف اصلی در این مقاله ارائه روشی برای بهبود LSBM است. انتخاب دنباله پیکسل‌ها برای جاسازی و چگونگی تغییر مقدار آنها در روش‌های مبتنی بر LSBM متفاوت هستند. در اغلب روش‌های موجود بعضی از این تصمیمات به صورت تصادفی گرفته می‌شود. در روش پیشنهادی در این مقاله، در مرحله اول از ایده چندکلیدی و در مرحله دوم از الگوریتم ژنتیک استفاده شده است تا تصمیمات بهتری اتخاذ شود. در روش پیشنهادی با عنوان MKGM، تصویر پوشش بلاک‌بندی شده و برای هر بلاک با چند کلید مختلف روش GLSBM اجرا می‌شود و در انتها بلاکی که کمترین تغییر هیستوگرام را نسبت به بلاک اولیه داشته باشد، در تصویر استگو قرار می‌گیرد. روش GLSBM، همان روش LSBM است با این تفاوت که برای تصمیم‌گیری در مورد افزایش یا کاهش پیکسل‌های غیر مطابق، از الگوریتم ژنتیک استفاده می‌شود. مقایسه معیارهای کیفیت تصویر و دقت حملات در کشف روش پیشنهادی، نشان‌دهنده بهبود این معیارها در مقایسه با روش LSBM اصلی است.

کلیدواژه: **الگوریتم ژنتیک، پنهان‌شکنی، پنهان‌نگاری، روش LSBM.**

۱- مقدمه

با پیشرفت فناوری ارتباطات دیجیتال، حفظ حریم خصوصی افراد به یک چالش بسیار مهم تبدیل شده است. حفظ محرمانگی و یکپارچگی پیام‌های ارسالی از طریق کانال‌های عمومی مانند اینترنت مهم‌ترین دغدغه کاربران می‌باشد. سال‌ها تلاش شده است روش‌های نوآورانه‌ای برای ارتباطات مخفی ایجاد شود. تا کنون روش‌های مختلفی برای برقراری یک ارتباط امن پیشنهاد شده که این روش‌ها در دو شاخه اصلی رمزنگاری و پنهان‌سازی اطلاعات دسته‌بندی می‌شوند.

آشکاربودن وجود یک ارتباط رمزشده و محرمانه بین طرفین ارتباط، مهم‌ترین عیب و کاستی رمزنگاری است و به همین دلیل در بعضی از کاربردها غیر قابل استفاده است. پنهان‌سازی اطلاعات، مجموعه‌ای از تکنیک‌های جاسازی اطلاعات در رسانه‌های دیجیتال است. این تکنیک‌ها می‌توانند در بسیاری از سناریوهای کاربردی مختلف مانند ارتباطات محرمانه، حفاظت از حق تکثیر یا احراز هویت محتوای دیجیتال مورد استفاده قرار گیرند. پنهان‌نگاری و ته‌نقش‌نگاری دو شاخه اصلی در این حوزه می‌باشند که علی‌رغم شباهت‌های مفهومی، از لحاظ کاربرد و معیارهای موفقیت با هم متفاوت هستند [۱].

هدف از پنهان‌نگاری به شکل مدرن و دیجیتال، درج و ارسال پیام محرمانه از طریق رسانه دیجیتال به گونه‌ای است که هیچ ظنی مبنی بر ارسال اطلاعات برانگیخته نشود. با توجه به محبوبیت تصویر در ارتباطات، پنهان‌نگاری در تصاویر، حوزه بسیار فعالی است. با این وجود در مقابل، برای کشف حضور داده پنهان در یک رسانه پوششی، دانشی به نام پنهان‌شکنی به وجود آمده است. فرستنده در یک ارتباط سرّی با استفاده از پنهان‌نگاری، فرایند جاسازی داده در تصویر پوشش را انجام می‌دهد و تصویر استگو را تولید و ارسال می‌کند. گیرنده، فرایند استخراج داده را از تصویر استگو انجام می‌دهد [۲].

در طراحی یک روش پنهان‌نگاری خوب سه الزام اصلی مورد نیاز است که عبارتند از شفافیت ادراکی، ظرفیت حمل و امنیت. شفافیت ادراکی یعنی رسانه پنهان‌نگاری شده و رسانه میزبان نباید از نظر ادراکی تفاوتی داشته باشند. معیارهایی از جمله PSNR، MSE و SSIM برای سنجش شفافیت ادراکی استفاده می‌شود. ظرفیت حمل به مفهوم حداکثر تعداد بیت‌هایی که می‌تواند در رسانه میزبان مخفی شود اشاره دارد. امنیت نیز یعنی حملات پنهان‌شکنی موجود قادر به کشف تصاویر پنهان‌نگاری شده تولیدشده به وسیله این روش نباشند [۳]. روش‌های مختلفی برای پنهان‌نگاری تصاویر بر اساس کاربرد و مراحل موجود در فرایند جاسازی، پیشنهاد شده است. این روش‌ها را می‌توان بر اساس معیارهای مختلف دسته‌بندی کرد. معیارهای استفاده‌شده در یکی از دسته‌بندی‌های اخیر عبارتند از [۴] نوع تصویر پوشش (دوبعدی یا سه‌بعدی)، فرایند بازیابی (برگشت‌پذیر یا غیر قابل برگشت)، ماهیت فرایند جاسازی (دامنه مکان یا تبدیل) و پنهان‌نگاری تطبیقی.

در روش‌های پنهان‌نگاری مکانی، از سطوح شدت پیکسل‌های تصویر پوشش به طور مستقیم یا غیر مستقیم برای جاسازی پیام محرمانه استفاده می‌شود. این روش‌ها از نظر پیچیدگی فرایندهای جاسازی و استخراج جزء ساده‌ترین تکنیک‌ها هستند. روش LSB (کم‌ارزش‌ترین بیت) از ساده‌ترین و معروف‌ترین رویکردهای حوزه مکانی است و LSBF[1] و LSBM[2] دو روش معروف با به کارگیری این رویکرد هستند. اعتقاد به کم‌اهمیت‌بودن بیت‌های کم‌ارزش پیکسل‌های تصویر و غیر محسوس بودن وجود تغییرات در این بیت‌ها توسط چشم انسان، دلیل اصلی شکل‌گیری این ایده بوده است. در روش LSBF پیام محرمانه در آخرین بیت (هشتمین بیت) از تمام یا برخی از پیکسل‌های تصویر، جاسازی می‌شود. روش‌های

این مقاله در تاریخ ۴ اردیبهشت ماه ۱۳۹۸ دریافت و در تاریخ ۶ آذر ماه ۱۳۹۸ بازنگری شد.

وجیهه ثابتی (نویسنده مسئول)، دانشکده فنی و مهندسی، دانشگاه الزهرا، تهران، ایران، (email: v.sabeti@alzahra.ac.ir).

سیده سپیده فیاضی، دانشکده فنی و مهندسی، دانشگاه الزهرا، تهران، ایران، (email: sepide.faiazi@gmail.com).

حدیثه شیرین‌خواه، دانشکده فنی و مهندسی، دانشگاه الزهرا، تهران، ایران، (email: hshirinkhah@gmail.com).

1. LSB Flipping
2. LSB Matching

پنهان‌شکنی بسیاری مانند حمله آماری Chi-square [۵]، تحلیل RS [۶]، تحلیل جفت نمونه [۷] و تحلیل استگو وزن داده شده [۸] در کشف LSBF موفق بوده‌اند.

در روش LSBM نیز پیام محرمانه در کم‌ارزش‌ترین بیت تصویر جاسازی می‌شود. با این تفاوت که اگر بیت پیام با بیت کم‌ارزش پیکسل برابر بود، پیکسل بدون تغییر باقی می‌ماند. اما در صورت عدم تطابق به مقدار پیکسل به صورت تصادفی یکی اضافه یا کم (+۱ یا −۱) می‌شود. این امر باعث شده که روش LSBM نسبت به LSBF با وجود ظرفیت یکسان امنیت بیشتری در مقابل حملات داشته باشد، اما با پیشرفت علم پنهان‌شکنی حملات نسبتاً موفق برای کشف LSBM نیز پیشنهاد شده است. در [۹] حمله‌ای بر اساس ادعای افزایش تعداد رنگ‌های همسایه در تصاویر رنگی بر اثر جاسازی داده به روش LSBM پیشنهاد شده است. Harmsen در [۱۰] با مدل‌سازی روش‌های پنهان‌نگاری به عنوان عامل اضافه‌کردن نویز به تصویر، موفق به کشف آنها شده است. اما Ker در [۱۱] و [۱۲]، این مدل‌سازی را برای روش LSB-M در تصاویر سطح خاکستری و رنگی انجام داده است. روش معرفی‌شده در [۱۳] بر اساس تأثیر LSBM بر روی کمینه‌ها و بیشینه‌های محلی هیستوگرام تصویر کار می‌کند. با هدف بهبود این روش، حمله دیگری در [۱۴] پیشنهاد شده است که از هیستوگرام تصویر و هیستوگرام همسایگی دوبعدی تصویر، ۱۰ پارامتر مختلف استخراج می‌کند. در [۱۵]، روشی پیشنهاد شده که برای کشف LSBM در تصاویر غیر فشرده مناسب است. به علاوه در یک حمله عام [۱۶]، از یک بردار ویژگی شامل ۲۷ خصیصه برای تشخیص تصویر استگو استفاده شده است. اما تمام این روش‌ها کارایی یکسانی ندارند و در اکثر موارد کارایی آنها بستگی به نوع تصویر پوشش مورد استفاده دارد [۱۷].

دلیل کشف روش‌های پنهان‌نگاری توسط حملات، تغییر ویژگی‌های آماری تصویر در اثر جاسازی داده است. یکی از نقاط ضعف روش LSBM که روش‌های پنهان‌شکنی از آن برای شکست LSBM استفاده می‌کنند، تغییر هیستوگرام تصویر استگو نسبت به هیستوگرام تصویر پوشش است. طراحان روش‌های پنهان‌نگاری سعی می‌کنند با استراتژی‌های مختلف این تغییرات را به حداقل برسانند. برای نمونه، روش OutGuess به گونه‌ای طراحی شده که سعی می‌کند با تغییر بیت‌های افزونه‌ای که داده در آنها جاسازی نشده است، تصویر استگویی تولید کند که ویژگی آماری مشابهی با تصاویر پوشش داشته باشد. این روش در حوزه تبدیل DCT و با حفظ هیستوگرام DCT عمل می‌کند [۱۸]. برای بهبود روش LSBM نیز روش‌هایی پیشنهاد شده که در حالت عدم تطابق انتخاب را به صورت تصادفی انجام نمی‌دهند، بلکه این انتخاب را به صورت هدف‌دار و برای رسیدن به یک هدف خاص انجام می‌دهند. بعضی از این اهداف عبارتند از:

۱) کمینه‌کردن تغییرات مقداری پیکسل‌ها (روش OPAP LSB [۱۹])
۲) کمینه‌کردن تغییرات هیستوگرام (روش A-LSB-M [۲۰])
۳) کمینه‌کردن نویز اضافه‌شده به تصویر (روش CAS-D و CAS-NE [۲۱])

هدف روش پیشنهادی در این مقاله نیز بهبود امنیت روش LSBM است، به عبارت دیگر هدف ارائه روش جدیدی بر مبنای LSBM به گونه‌ای است که در برابر حملات مقاومت بیشتری داشته و احتمال شکست آن کمتر باشد. با توجه به تغییر هیستوگرام تصویر استگوی حاصل از LSBM نسبت به هیستوگرام تصویر پوشش، اغلب حملات موفق ارائه‌شده برای شکست LSBM از ویژگی‌های مستخرج از هیستوگرام تصویر استفاده می‌کنند. بنابراین می‌توان امیدوار بود با کم‌کردن این تغییرات، احتمال موفقیت حملات آماری کمتر شود. در روش پیشنهادی برای رسیدن به این هدف از افزودن ایده‌های الگوریتم ژنتیک، چندکلیدی و بلاک‌بندی به LSBM استفاده شده است. ایده چندکلیدی برای انتخاب پیکسل‌های مناسب‌تر برای جاسازی، الگوریتم ژنتیک برای تصمیم‌گیری هدفدار میان دو انتخاب افزایش یا کاهش پیکسل و بلاک‌بندی نیز برای کوچک‌کردن محدوده‌های تصمیم‌گیری استفاده شده است.

در ادامه، در بخش دوم تعدادی از روش‌های پنهان‌نگاری که از الگوریتم‌های بهینه‌سازی استفاده کرده‌اند مرور گردیده و سپس در بخش سوم، الگوریتم پیشنهادی به صورت مفصل شرح داده شده است. در بخش چهارم، نتایج تست و مقایسه روش پیشنهادی و در انتها نتیجه‌گیری و پیشنهاداتی برای ادامه کار بیان شده است.

۲- مروری بر کارهای گذشته

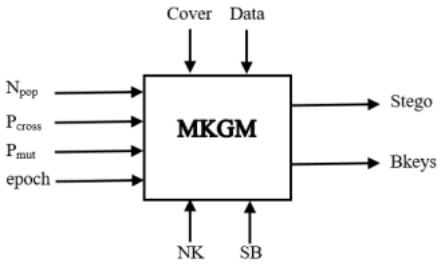
تا به حال روش‌های پنهان‌نگاری بسیاری از الگوریتم‌های بهینه‌سازی استفاده کرده‌اند. استفاده از الگوریتم‌های بهینه‌سازی در روش‌های پنهان‌نگاری معمولاً در سه زمان صورت می‌گیرد:

- پیش از مرحله جاسازی
- در حین جاسازی
- پس از جاسازی

در دسته پیش از مرحله جاسازی، معمولاً از الگوریتم بهینه‌سازی برای پیداکردن بهترین مکان برای انجام جاسازی یا تغییر بیت‌های پیام استفاده می‌شود. در دسته دوم، الگوریتم بهینه‌سازی برای تعیین چگونگی ذخیره‌سازی داده و تعیین مقدار پیکسل تصویر استگو استفاده می‌شود و در دسته سوم، الگوریتم بهینه‌سازی به کاهش تغییرات حاصل از جاسازی کمک می‌کند.

در [۲۲]، Roy و همکاران یک روش پنهان‌نگاری با استفاده از الگوریتم ژنتیک ارائه داده‌اند که در دسته سوم قرار می‌گیرد. در روش ارائه‌شده، ابتدا با روش LSB و جایگذاری در ۴ بیت کم‌ارزش هر پیکسل، تصویر استگو به دست آمده و سپس با اعمال الگوریتم ژنتیک روی تصویر استگو، تصویر استگوی بهینه ساخته می‌شود. برای این منظور ابتدا تصویر اصلی و استگو، هر کدام به صورت جداگانه در دو ماتریس ریخته شده و یک ماتریس با نام BSF (بهترین تا کنون) که دارای مقدار اولیه‌ای برابر با تصویر استگو است ایجاد می‌شود. سپس جمعیت تولیدشده و برازش هر عضو از جمعیت محاسبه می‌شود و گام‌های الگوریتم ژنتیک انجام می‌گیرد. این اعمال به تعداد مشخص تکرار شده و در نهایت نتیجه حاصل از کروموزوم با بالاترین مقدار برازش در BFS ذخیره می‌شود. نتایج حاصل از این مقاله نشان از بهبود شفافیت دارد.

Kanan و همکاران در [۲۳] روشی ارائه داده‌اند که هدفش تولید یک تصویر استگو با کیفیت بصری قابل اندازه‌گیری و بدون از دست دادن داده، بر اساس الگوریتم ژنتیک در حوزه مکان است. این روش در دسته اول قرار می‌گیرد. ایده اصلی، مدل‌سازی مسئله پنهان‌نگاری به عنوان یک مسئله جست‌وجو و بهینه‌سازی است. در این مقاله تلاش می‌شود تا بهترین مکان برای جاسازی بیت‌های اطلاعات محرمانه اصلاح‌شده، در تصویر میزبان، برای رسیدن به سطح بالایی از امنیت کشف شود. فرایند جاسازی در دو مرحله انجام می‌گیرد: مرحله اول اصلاح بیت‌های محرمانه و مرحله دوم جاسازی آنها در تصویر است. مکان‌های مختلفی در تصویر میزبان با خصوصیاتی مانند ترتیب اسکن، نقطه شروع اسکن، بهترین



شکل ۱: ورودی‌ها و خروجی‌های الگوریتم MKGM.

LSBهای هر پیکسل و ... وجود دارد. در این روش از الگوریتم ژنتیک برای پیداکردن بهترین مکان با بهترین خصوصیات در تصویر میزبان استفاده شده است تا این گونه به بالاترین PSNR دست پیدا کند. نتایج به دست آمده حاکی از بهبود PSNR به عنوان یک معیار ارزیابی شفافیت است و همچنین نشان می‌دهد که حتی زمانی که اندازه پیام افزایش پیدا می‌کند باز هم میزان شفافیت قابل قبول است.

در [۲۴]، Wang و همکاران برای جلوگیری از دسترسی غیر مجاز به اطلاعات محرمانه و مهم، از فرایند تصادفی‌سازی به همراه LSB بهینه به جای LSB ساده استفاده کرده‌اند. این روش در دسته اول قرار می‌گیرد. روش ارائه‌شده به این صورت است که اطلاعات محرمانه در k سمت راست‌ترین بیت‌های هر پیکسل قرار می‌گیرد. از آنجا که برای رسیدن به نتیجه بهینه، تمام جاسازی‌های ممکن باید بررسی و ارزیابی شود، از الگوریتم ژنتیک استفاده شده است. ابتدا بیت‌های کم‌ارزش پیکسل‌های تصویر پوشش (H) جدا شده و تصویر R به دست می‌آید. سپس تصویر E (پیام) که هدف، جاسازی آن در تصویر اصلی است، به تصویر باینری E' که بیت‌هایش با الگوریتم رمزنگاری جایگزینی تک‌مجرایی، تصادفی‌سازی شده است، درآمده و در R قرار می‌گیرد و سپس R با تصادفی‌سازی و LSB بهینه در تصویر اصلی جاسازی می‌شود و این گونه تصویر Z ساخته می‌شود. از آنجا که هرچه مقدار k بیشتر باشد تعداد حالات بیشتری به وجود می‌آید، زمانی که $k>۳$ می‌باشد برای بررسی تمام حالات از الگوریتم ژنتیک استفاده شده است.

Shah و همکاران در [۲۵] یک روش پنهان‌نگاری با استفاده از الگوریتم ژنتیک ارائه داده‌اند که در ادامه از آن با عنوان روش LCG یاد می‌شود. این روش نیز در دسته اول قرار می‌گیرد اما از ژنتیک برای پنهان‌سازی ماتریس ضرایب و نه پیام محرمانه، استفاده می‌شود. این روش به جای استفاده از روش LSB معمولی برای جایگذاری در تصویر از روش نگاشت داده استفاده می‌کند، بدین صورت که باید دو بیت از پیام در دو بیت از پیکسل تصویر پوشش ذخیره شود. با این شرایط، بیت‌ها به جای قرارگیری در LSBهای پیکسل در بهترین مکان قرار می‌گیرند (بهترین مکان جایی است که دو بیت پیام عیناً شبیه به بیت‌های پیکسل هستند و اگر دو بیت برابر پیدا نشد، جایگذاری مانند LSB معمولی انجام می‌شود). در هر پیکسل ۸ مکان (یا ۸ ضریب) برای جاسازی وجود دارد، بنابراین از یک ماتریس برای نگهداری ضرایب متناظر با هر پیکسل استفاده می‌شود. جایگذاری و ساخت ماتریس ضرایب با استفاده از الگوریتم ژنتیک انجام می‌شود. در این مقاله اندازه پیام به گونه‌ای انتخاب شده که فقط ۱/۴ از پیکسل‌های تصویر پوشش را اشغال می‌کند و الگوریتم ژنتیک از ۳/۴ دیگر برای ذخیره ماتریس ضرایب استفاده می‌کند. نتایج حاصل در مقایسه با جایگذاری LSB معمولی که ۲ بیت را در تصویر جایگذاری می‌کند، نشان می‌دهد که روش ارائه‌شده در این مقاله شفافیت بهتری دارد و نسبت به حملات آنالیز هیستوگرام مقاوم‌تر است.

جدول ۱: اختصارات استفاده‌شده در تشریح روش‌ها.

پارامتر	تعریف پارامتر	پارامتر	تعریف پارامتر
$Cover$	تصویر پوشش	$Cover_i$	بلاک i ام پوشش
$Stego$	تصویر استگو	$Stego_i$	بلاک i ام استگو
$Data$	داده محرمانه	$Data_i$	بلاک i ام داده
Key	مجموعه کلیدها	Key_i	کلید i ام
$BKeys$	مجموعه بهترین کلیدها	$BKeys_i$	بهترین کلید بلاک i ام
SB	اندازه بلاک	NK	تعداد کلید
N_{pop}	تعداد جمعیت ژنتیک	$epoch$	تعداد تکرار
P_{cross}	احتمال تقاطع	P_{mut}	احتمال جهش

با توجه به زیادبودن مقالات، امکان بررسی تمام آنها در اینجا وجود ندارد. بر اساس اطلاعات به دست آمده از این مقالات، الگوریتم ژنتیک رایج‌ترین الگوریتم بهینه‌سازی در میان الگوریتم‌های پنهان‌نگاری است و بر همین اساس در الگوریتم پیشنهادی در این مقاله نیز از این الگوریتم استفاده شده است. اما برخلاف اکثر روش‌های موجود که از PSNR به عنوان تابع برازندگی و LSBF به عنوان روش پایه برای جاسازی داده استفاده کرده‌اند، در روش پیشنهادی در این مقاله از اختلاف هیستوگرام قبل و بعد از جاسازی داده به عنوان تابع برازندگی و LSBM به عنوان روش پایه استفاده شده است.

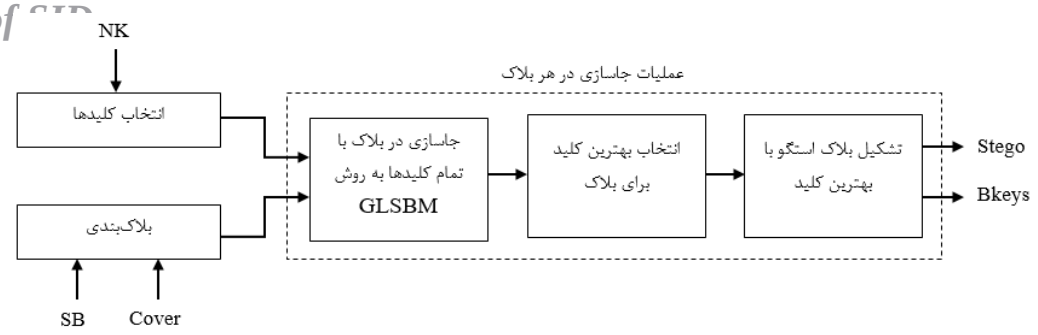
۳- روش پیشنهادی

امنیت بالاتر LSBM نسبت به LSBF با توجه به حملاتی که تا کنون ارائه شده و موفق به کشف LSBF شده است، قابل توجیه است. اما با وجود پیشرفت حملات پنهان‌شکنی، LSBM نیز قابلیت ایستادگی کامل در مقابل برخی حملات را ندارد. یکی از تکنیک‌های مناسب برای روش‌های پنهان‌نگاری جدید، ترکیب روش LSBM با ایده‌هایی است که بتواند به بهبود امنیت LSBM کمک کند. از طرف دیگر، با توجه به قابلیت الگوریتم‌های بهینه‌سازی در پیداکردن نقاط نزدیک به بهینه در مسایل مختلف، می‌توان امیدوار بود استفاده از الگوریتم‌های بهینه‌سازی در روش‌های پنهان‌نگاری به شرط در نظر گرفتن یک تابع برازندگی مناسب، بتواند به قابلیت‌های الگوریتم پنهان‌نگاری کمک چشم‌گیری کند. در ادامه یک روش جدید بر مبنای این ایده ارائه می‌شود.

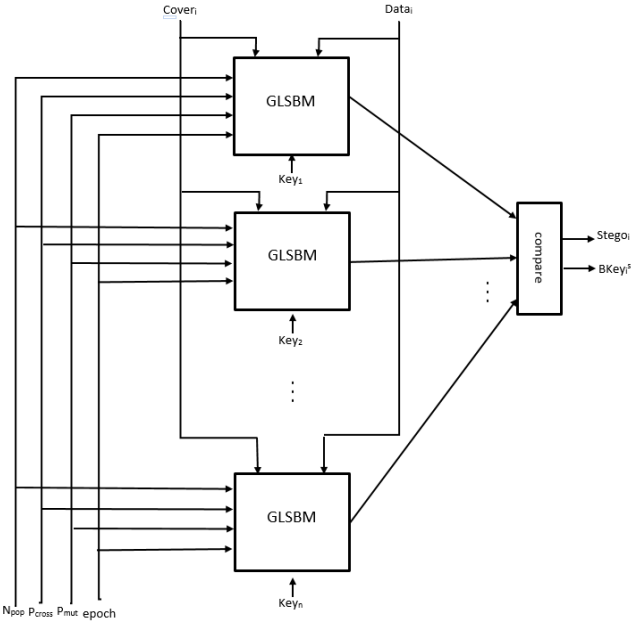
۳-۱ روش MKGM

روش پیشنهادی، MKGM (روش LSBM مبتنی بر ژنتیک چندکلیدی) نامیده شده است. در شکل ۱، ورودی‌های لازم و خروجی‌های حاصل از اجرای این روش آمده است. این روش مشابه تمام روش‌های پنهان‌نگاری دیگر، تصویر پوشش و دنباله داده محرمانه را از ورودی دریافت کرده و تصویر استگو را در خروجی تولید می‌کند. اما این روش ورودی‌ها و خروجی‌های دیگری نیز دارد که در تشریح الگوریتم، معرفی می‌شوند. شکل ۲، مراحل الگوریتم پیشنهادی را نشان می‌دهد. این مراحل با کمک اختصارات ذکرشده در جدول ۱ به صورت مبسوط در ادامه شرح داده می‌شود.

شکل ۲ نشان می‌دهد که این الگوریتم شامل ۵ مرحله اصلی است که دو مرحله ابتدایی یعنی بلاک‌بندی و انتخاب کلیدها یک بار اجرا می‌شود و مراحل بعدی برای هر بلاک از تصویر پوشش تکرار می‌شود. در دو مرحله ابتدایی، تصویر پوشش با توجه به پارامتر ورودی SB که نشان‌دهنده اندازه بلاک است، بلاک‌بندی می‌شود. سپس یک مجموعه Key شامل



شکل ۲: مراحل الگوریتم MKGM.

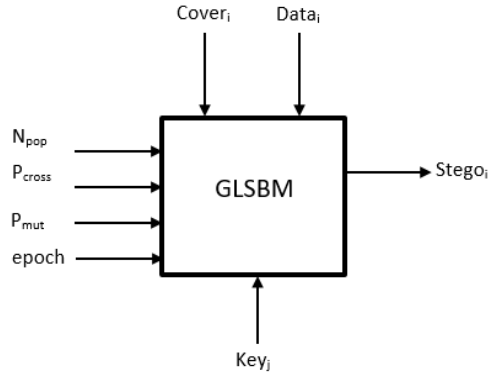


شکل ۳: مراحل عملیات جاسازی در هر بلاک در روش MKGM.

NK تا عدد انتخاب می‌شود که این اعداد با key_1، key_2، ... و key_{NK} نمایش داده می‌شوند. این اعداد به عنوان کلید در فرایند جاسازی استفاده می‌شوند.

هر الگوریتم پنهان‌نگاری به یک یا چند کلید نیاز دارد که از این کلیدها در مراحل مختلف جاسازی استفاده می‌کند. یکی از کاربردهای کلید، انتخاب ترتیب پیکسل‌ها برای جاسازی داده است. بسیاری از روش‌ها برای انجام این مرحله، از یک کلید که از قبل میان فرستنده و گیرنده توافق شده است استفاده می‌کنند، اما سؤالی که ممکن است در ذهن ایجاد شود، تأثیر مقدار کلید روی عملکرد روش پنهان‌نگاری است. به عبارت دیگر آیا با تغییر کلید می‌توان نتایج بهتری به دست آورد؟ با توجه به وجود این دغدغه، در روش MKGM استفاده از ایده چندکلیدی پیشنهاد شده است. ایده پیشنهادی بدین صورت است که عملیات جاسازی برای هر بلاک با کلیدهای مختلف تکرار می‌شود و بر اساس یک معیار مشخص‌شده، بهترین کلید انتخاب می‌شود و از این کلید برای ساخت بلاک استگو نهایی استفاده می‌شود. معیار انتخاب بهترین بلاک، میزان اختلاف هیستوگرام بلاک تصویر پوشش و استگو است که در ادامه با نام تابع $Dif(H_c, H_s)$ معرفی می‌شود. هرچه میزان این اختلاف کمتر باشد، بلاک استگوی تولیدشده مناسب‌تر است. روش جاسازی در هر بلاک، GLSBM نام دارد. شکل ۳ مراحل جاسازی در هر بلاک را نشان می‌دهد.

برای کامل‌شدن الگوریتم MKGM، باید نحوه جاسازی در هر بلاک شرح داده شود. روش جاسازی در بلاک، یک روش مبتنی بر LSBM



شکل ۴: ورودی‌های لازم و خروجی حاصل از اجرای الگوریتم GLSBM.

است و ایده این روش استفاده از الگوریتم‌های بهینه‌سازی برای بهبود عملکرد روش LSBM می‌باشد. الگوریتم LSBM شامل دو مرحله اصلی است:

۱) انتخاب تعدادی از پیکسل‌ها برای جاسازی داده (با توجه به طول داده): با استفاده از کلید

۲) جاسازی داده در پیکسل‌های انتخابی: اگر بیت داده مورد نظر با LSB پیکسل انتخابی مطابقت نداشته باشد، باید مقدار پیکسل به صورت تصادفی یک واحد افزایش یا کاهش یابد.

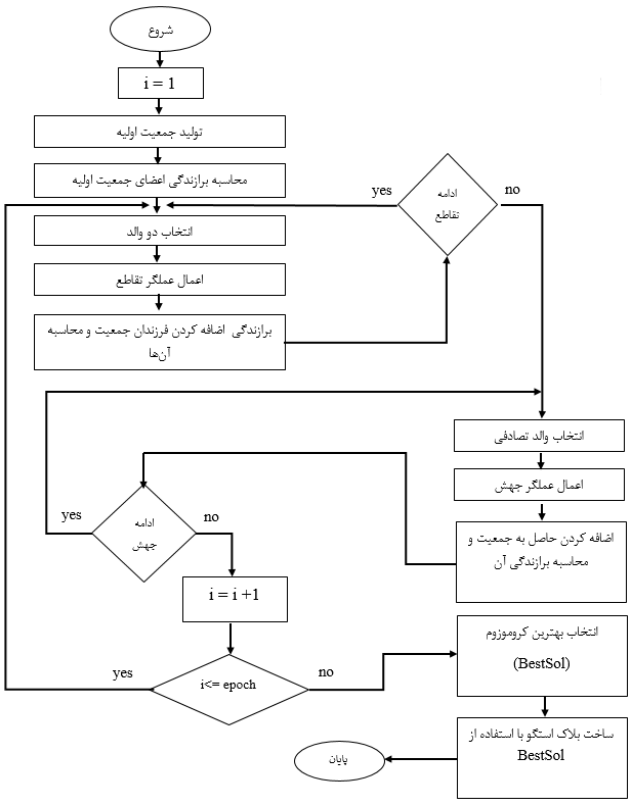
برای عملیات جاسازی در هر بلاک، روش GLSBM پیشنهاد شده است که مهم‌ترین تفاوت آن با LSBM، انجام هدفدار تصمیمات افزایش یا کاهش پیکسل‌ها در گام دوم با استفاده از الگوریتم‌های بهینه‌سازی است. در ادامه این مسئله با الگوریتم ژنتیک مدل‌سازی و حل شده است اما استفاده از هر الگوریتم بهینه‌سازی دیگری نیز برای حل این مسئله امکان‌پذیر است.

۳-۲ روش GLSBM (روش LSBM مبتنی بر ژنتیک)

روش پیشنهادی برای جاسازی در هر بلاک، روشی است که طبق دسته‌بندی ارائه‌شده در بخش مروری بر کارهای گذشته، در گروه دوم قرار می‌گیرد. به عبارت دیگر، قرار است در روش GLSBM از الگوریتم ژنتیک در مرحله جاسازی و برای تعیین مقدار پیکسل‌های حاوی پیام استفاده شود. شکل ۴، ورودی‌های لازم و خروجی حاصل از اجرای این الگوریتم و شکل ۵ مراحل الگوریتم GLSBM را نشان می‌دهد.

۳-۲-۱ تولید جمعیت اولیه

در الگوریتم ژنتیک یک راه حل با یک بردار جواب به عنوان یک فرد یا یک کروموزوم شناخته می‌شود و هر کروموزوم از بخش‌های مجزایی به نام ژن تشکیل یافته است. تعریف مقادیر ممکن برای ژن به مسئله مورد نظر بستگی دارد. اولین گام در الگوریتم ژنتیک تعیین ساختار کروموزوم و تولید جمعیت اولیه است. در روش GLSBM، طول کروموزوم برابر با



شکل ۵: مراحل الگوریتم GLSBM.

تعداد بیت‌های پیامی است که باید در بلاک تصویر پوشش جاسازی شود. اگر L نشان‌دهنده تعداد بیت پیام باشد، طول هر کروموزوم در جمعیت، L در نظر گرفته می‌شود.

برای تولید جمعیت اولیه از پارامتر key_j (کلید j ام توافق‌شده میان فرستنده و گیرنده) به عنوان مقدار هسته ابتدایی تابع تولید اعداد شبه‌تصادفی استفاده می‌شود. اگر ابعاد هر بلاک تصویر پوشش برابر M سطر و N ستون باشد، پس این بلاک $M \times N$ پیکسل دارد. با استفاده از تابع rand برداری شامل L عدد مختلف در بازه $[1, M \times N]$ انتخاب می‌شود که شماره پیکسل‌های انتخابی برای جاسازی داده را نشان می‌دهد (بردار LOC). برای تولید یک کروموزوم ابتدایی، طبق بردار LOC مقدار LSB هر پیکسل انتخابی با بیت پیام متناظر مقایسه می‌شود. در صورت تطابق، مقدار ژن متناظر در کروموزوم صفر در نظر گرفته می‌شود و این به معنای عدم نیاز به تغییر مقدار پیکسل در فرایند جاسازی است. در صورت عدم تطابق به صورت تصادفی مقدار ۱+ یا ۱− در مقدار ژن قرار می‌گیرد. مقدار ۱+ نشان‌دهنده افزایش یک واحدی مقدار پیکسل و ۱− نشان‌دهنده کاهش یک واحدی مقدار پیکسل در اثر جاسازی بیت پیام مورد نظر است. این فرایند در واقع تداعی‌کننده روش جاسازی LSBM است. شکل ۶ شبه‌کد چگونگی انجام اولین مرحله از روش GLSBM را بیان می‌کند.

۳-۲-۲ اعمال تابع برازندگی

هر الگوریتم بهینه‌سازی به دنبال حداقل‌کردن/ حداکثرکردن یک تابع برازندگی (تابع هدف) است که این تابع به مسئله مورد نظر بستگی دارد. با توجه به مقالات مرورشده در بخش دوم، در روش‌های پنهان‌نگاری توابع برازندگی مختلفی وجود دارد که در این مرحله قابل استفاده هستند. با مطالعه حملات خاص و مختلف ارائه‌شده برای کشف LSBM، این نکته قابل تأمل است که یکی از ویژگی‌هایی که از آن برای حمله به LSBM استفاده شده است، پارامترهای استخراج‌شده از هیستوگرام تصویر، قبل و

```
Population initialization
Input: cover_i, data_i, key_j, N_pop
Output: pop, LOC
L = size( data_i );
[M,N] = size( cover_i );
LOC = randperm(M*N);
For i=1 : N_pop
    For j = 1:L
        If mod( cover_i(j) , 2 ) = data_i(j)
            pop( I , j ) = 0;
        else
            r = rand();
            if  r > 0.5
                pop ( i , j ) = 1;
            else
                pop( i , j ) = -1;
```

شکل ۶: شبه‌کد چگونگی انجام اولین مرحله از روش GLSBM.

بعد از جاسازی با روش LSBM، مانند مرکز ثقل [۱۱] و نقاط اکسترمم محلی [۱۴] است. به همین دلیل به نظر می‌رسد با کم‌کردن تغییر هیستوگرام تصویر در اثر جاسازی می‌توان شکست‌پذیری LSBM را کمتر کرد. با توجه به این که هدف اصلی در روش پیشنهادی، افزایش امنیت است، بنابراین باید به دنبال کم‌کردن تغییرات هیستوگرام تصویر استگو نسبت به تصویر پوشش بود، زیرا هرچه این تغییرات بیشتر باشد، حملات پنهان‌شکنی راحت‌تر روش پنهان‌نگاری را کشف می‌کند.

اگر H_c و H_s به ترتیب نشان‌دهنده هیستوگرام یک بلاک تصویر پوشش و یک بلاک تصویر استگو و h_v و h'_v نیز به ترتیب فراوانی سطح روشنایی v در تصویر پوشش و تصویر استگو باشد، پارامترهای زیر در تصاویر سطح خاکستری قابل تعریف هستند

$$H_c = \{h_0, h_1, \ldots, h_{255}\}$$
$$h_v = \left|\{j \,|\, cover(j) = v\}\right| \quad (1)$$

$$H_s = \{h'_0, h'_1, \ldots, h'_{255}\}$$
$$h'_v = \left|\{j \,|\, stego(j) = v\}\right| \quad (2)$$

$$Dif(H_c, H_s) = \sum_{v=0}^{255} \left|h_v - h'_v\right| \quad (3)$$

$Dif(H_c, H_s)$ نشان‌دهنده اختلاف هیستوگرام بلاک تصویر پوشش و تصویر استگو است. با توجه به نقش مهم این معیار در کشف روش LSBM، روش GLSBM به دنبال حداقل‌کردن این معیار است. پس تابع برازندگی انتخابی به این شکل می‌باشد

$$Dif(H_c, H_s) \quad (4)$$

پس از تولید تصادفی جمعیت اولیه، باید میزان برازش برای هر عضو با استفاده از تابع برازندگی محاسبه شود.

۳-۲-۳ انتخاب

در الگوریتم ژنتیک برای تولید جواب‌های جدید از دو عملگر تقاطع و جهش استفاده می‌شود. برای اجرای عملگر تقاطع باید دو کروموزوم به

```
Stego Image Generation
input: Cover_i, LOC, BestSol
Output: Stego_i
    Stego_i = Cover_i;
    for i= 1:L
        If BestSol(i) != 0
            Stego_i(LOC(i)) = Stego_i(LOC(i)) + BestSol(i);
        endif
    endfor
```

شکل ۷: شبه‌کد مرحله تشکیل بلاک تصویر استگو بر اساس خروجی الگوریتم ژنتیک.

عنوان والد از بین جمعیت انتخاب شوند. برای اجرای عمل انتخاب روش‌های مختلفی وجود دارد. در روش GLSBM، کروموزوم‌هایی به عنوان والد انتخاب می‌شوند که کمترین مقدار تابع برازندگی را در میان جمعیت فعلی داشته باشند.

۳-۲-٤ عملگر تقاطع

یکی از پارامترهای الگوریتم ژنتیک P_{cross} است که نشان‌دهنده درصدی از جمعیت است که باید عملگر تقاطع روی آنها انجام شود. برای هر بار اجرای عملگر تقاطع به روش بیان‌شده در بخش انتخاب، دو والد از میان جمعیت انتخاب می‌شود. از میان انواع عملگرهای تقاطع موجود، در GLSBM از عملگر تقاطع تک‌نقطه‌ای استفاده شده است. در این روش، نقطه x به طور تصادفی از بازه $[1, L]$ انتخاب می‌شود و ژن‌های بعد از نقطه x در دو کروموزوم والد با هم جابه‌جا می‌شوند. مقدار تابع برازندگی برای دو فرزند محاسبه می‌شود و سپس این دو فرزند با دو کروموزومی که بدترین مقدار برازندگی در جمعیت را دارند، جایگزین می‌شوند. این عملیات باید به تعداد $P_{cross} \times N_{pop} / 2$ بار انجام شود تا مرحله اجرای عملگر تقاطع تکمیل شود.

۳-۲-۵ عملگر جهش

یکی دیگر از عملگرهای الگوریتم ژنتیک، P_{mut} است که نشان‌دهنده درصدی از جمعیت است که باید عملگر جهش روی آنها اعمال شود. این عملگر برای ایجاد تغییرات تصادفی در ژن‌ها استفاده می‌شود و روشی برای جست‌وجوی محلی است. این عملیات باید به تعداد $P_{mut} \times N_{pop}$ بار اجرا شود. در روش GLSBM برای هر بار اجرای عملگر جهش، ابتدا یک کروموزوم از جمعیت به صورت تصادفی انتخاب می‌شود و سپس یک نقطه y به صورت تصادفی در بازه $[1, L]$ انتخاب می‌شود. حال اگر ژن yام از کروموزوم انتخاب‌شده صفر باشد، تغییری روی کروموزوم انجام نمی‌شود، زیرا صفربودن مقدار ژن نشان‌دهنده تطابق بیت پیام با LSB پیکسل مورد نظر است و در این حالت داشتن مقدار صفر برای ژن بهترین انتخاب است. اما اگر ژن yام دارای مقداری به غیر از صفر باشد، عملیات جهش روی ژن انجام می‌گیرد. به این صورت که اگر مقدار ژن انتخاب‌شده برای عملیات جهش ۱+ باشد، مقدار جدید آن به ۱- تغییر می‌یابد و بالعکس. بعد از انجام جهش کروموزوم حاصل از آن به جمعیت اضافه شده و مقدار تابع برازندگی برای آن محاسبه می‌شود.

۳-۲-٦ پایان الگوریتم ژنتیک

معیارهای گوناگونی را می‌توان برای تشخیص پایان اجرای الگوریتم ژنتیک پیشنهاد داد که در GLSBM از تعداد تکرار استفاده شده است. پارامتر *epoch*، تعداد تکرار الگوریتم را نشان می‌دهد. با پایان‌یافتن الگوریتم، بهترین کروموزوم انتخاب می‌شود که در واقع بهترین راه حل یا خروجی الگوریتم ژنتیک است و *BestSol* نشان‌دهنده این کروموزوم می‌باشد.

۳-۲-۷ تشکیل بلاک تصویر استگو

آخرین گام، ساخت بلاک تصویر استگو با استفاده از خروجی الگوریتم ژنتیک است. برای این مرحله باید تمام پیکسل‌های بلاک تصویر پوشش که برای جاسازی انتخاب نشده‌اند یا در صورت انتخاب، ژن متناظر آنها در *BestSol* مقدار صفر دارد، بدون تغییر به بلاک تصویر استگو منتقل شوند. بقیه پیکسل‌های بلاک تصویر پوشش با توجه به مقدار ژن متناظر در *BestSol* افزایش یا کاهش یک واحدی یافته و به بلاک تصویر استگو منتقل می‌شوند. شکل ۷ نحوه انجام این مرحله را نشان می‌دهد.

٤- نتایج پیاده‌سازی

در بخش قبل با هدف بهبود LSBM، روش MKGM پیشنهاد شد. این روش در نرم‌افزار Matlab پیاده‌سازی گردید و برای مقایسه آنها و تعیین تأثیر پارامترهای موجود، تست‌های مختلفی انجام شد که ارائه نتایج تمام آنها امکان‌پذیر نیست. اما در ادامه نتایج مهم‌ترین تست‌های انجام‌شده ارائه می‌شود. با توجه به این که در روش پیشنهادی از یک الگوریتم بهینه‌سازی استفاده شده است، روش LCG [۲۵] که یک روش جدید مبتنی بر الگوریتم ژنتیک است برای تست انتخاب شده است. روش MKGM دو پارامتر ورودی بسیار مهم و تأثیرگذار (تعداد کلید و اندازه بلاک) دارد و بنابراین می‌توان این روش را با MKGM(NK,SB) نمایش داد. برای تعیین بهترین مقادیر این پارامترها، در تست‌های انجام‌شده سه حالت کلی در نظر گرفته شده است:

۱) MKGM(۱,*): این حالت ساده‌ترین روش است که در آن تعداد کلیدها یک فرض شده و به علاوه بلاک‌بندی روی تصویر نیز انجام نشده است. به عبارت دیگر، در این حالت روش GLSBM روی کل تصویر و فقط با یک کلید انجام می‌شود. بنابراین در این حالت، فقط ایده بهینه‌سازی وجود دارد.

۲) MKGM(n,*): در این حالت تعداد کلیدها n فرض شده و بلاک‌بندی روی تصویر انجام نشده است. به عبارت دیگر، در این حالت روش GLSBM روی کل تصویر و با n کلید متفاوت تکرار می‌شود. بنابراین در این حالت، دو ایده چندکلیدی و بهینه‌سازی در نظر گرفته شده است.

۳) MKGM(n,m): در این حالت آن تعداد کلیدها n و اندازه بلاک $m \times m$ فرض شده است. به عبارت دیگر، روش GLSBM برای هر بلاک تصویر با n کلید متفاوت تکرار می‌شود. این حالت شامل هر سه ایده چندکلیدی، بهینه‌سازی و بلاک‌بندی است.

مقدار پارامترهای انتخاب‌شده در این پیاده‌سازی برای الگوریتم ژنتیک عبارتند از

$$N_{pop} = ۲۰ \ , \ P_{cross} = ۰/۷ \ , \ P_{mut} = ۰/۱ \ , \ epoch = ۲۰$$

٤-۱ ارزیابی معیارهای سنجش

کیفیت تصویر استگو یکی از مهم‌ترین فاکتورها در پنهان‌نگاری است که معمولاً برای سنجش آن از معیار PSNR استفاده می‌شود. در اولین گام برای چند تصویر نمونه شامل Lena، Baboon و Peppers معیار PSNR برای روش‌های پیشنهادی و تعدادی از روش‌های قبلی، برای دو درصد جاسازی ۰/۳ bpp و ۰/۸ bpp در جدول ۲ نشان داده شده است. با

جدول ۲: معیار PSNR برای سه تصویر نمونه در روش‌های جاسازی مختلف با دو جاسازی ۰٫۳ BPP و ۰٫۸ BPP.

درصد جاسازی	روش جاسازی	Lena	Baboon	Peppers
۰٫۳ bpp	LSBM	۵۶٫۳۷	۵۶٫۳۵	۵۶٫۳۶
	PVD [۲۶]	۵۱٫۷۷	۴۹٫۷۷	۴۱٫۹۸
	CBL [۲۷]	۵۶٫۳۹	۵۶٫۳۷	۵۶٫۳۶
	LCG [۲۵]	۵۳٫۳۷	۵۳٫۳۷	۵۳٫۳۶
	MKGM(۱,*)	۵۵٫۶۸	۵۶٫۴۱	۵۵٫۶۵
	MKGM(۱۰,*)	۵۶٫۳۹	۵۶٫۳۷	۵۶٫۴۰
	MKGM(۴,۱۶)	**۵۶٫۵۷**	**۵۶٫۶۸**	**۵۶٫۶۱**
۰٫۸ bpp	LSBM	۵۲٫۱۰	۵۲٫۱۰	۵۲٫۱۱
	PVD [۲۶]	۴۵٫۵۷	۴۴٫۷۰	۳۳٫۹۳
	CBL [۲۷]	۵۲٫۱۱	۵۲٫۱۲	۵۲٫۱۱
	LCG [۲۵]	–	–	–
	MKGM(۱,*)	۵۱٫۶۱	۵۲٫۱۳	۵۱٫۴۵
	MKGM(۱۰,*)	۵۲٫۱۰	۵۲٫۱۲	۵۲٫۱۲
	MKGM(۴,۱۶)	**۵۲٫۲۶**	**۵۲٫۲۸**	**۵۲٫۲۵**

توجه به این که در روش‌های مبتنی بر LSBM در صورت تغییر پیکسل، تنها یک واحد مقدار پیکسل تغییر می‌یابد، مقدار PSNR این روش‌ها نسبت به روش‌هایی که تغییر بیشتری در مقدار پیکسل به وجود می‌آورند، مانند PVD [۲۶] و LCG [۲۵]، بسیار بهتر است. با وجود این، از آنجا که در روش‌هایی مانند LSBM اصلی، CBL (LSBM تطابقی) [۲۷] و MKGM(۱,*) هیچ تلاشی برای پیداکردن مکان‌هایی با بیشترین تطابق نسبت به بیت‌های پیام انجام نمی‌شود، مقدار PSNR حاصل از اجرای آنها تقریباً مشابه است و مقدار آن به کلید استفاده‌شده در الگوریتم بستگی دارد. اما در روش MKGM(۱۰,*) و MKGM(۴,۱۶) با توجه به این که با چند کلید فرایند جاسازی تکرار می‌شود و از میان آنها بهترین جواب انتخاب می‌شود، احتمال این که PSNR بهتری به دست بیاید، بیشتر است. بهترین مقدار PSNR و در واقع بهترین کیفیت تصویر استگو در جدول ۲ پررنگ شده است. لازم به ذکر است که روش LCG قابلیت جاسازی حداکثر ۰٫۵ bpp در تصویر را دارد و به همین دلیل در جاسازی ۰٫۸ bpp در جدول ۲، برای این روش علامت – گذاشته شده است.

برای بررسی کامل‌تر معیار شفافیت، در جدول ۳ میانگین معیار PSNR و SSIM برای ۲۰۰ تصویر تست پایگاه داده NRCS نشان داده شده است. اطلاعات این جدول نشان می‌دهد که روش پیشنهادی در بهبود PSNR کاملاً موفق است اما روش CBL که یک روش تطبیقی است و به ویژگی‌های مکانی تصویر بیشتر توجه می‌کند و جاسازی را در نواحی لبه تصویر انجام می‌دهد، از نظر معیار SSIM موفق‌تر است. روش پیشنهادی، SSIM بهتری نسبت به روش LCG دارد.

روش پیشنهادشده دارای دو پارامتر اندازه بلاک و تعداد کلید می‌باشد که مقدار آنها روی کارایی روش تأثیر می‌گذارد. در گام دوم تست‌هایی برای تعیین بهترین مقدار برای این پارامترها انجام شده است. با توجه به این که زیادکردن تعداد کلیدها باعث می‌شود طول کلید توافقی میان فرستنده و گیرنده افزایش یابد، سه حالت ۱، ۲ و ۴ کلید برای تست در نظر گرفته شده است. اندازه بلاک هم یک پارامتر تأثیرگذار است و سه حالت ۸×۸، ۱۶×۱۶ و ۳۲×۳۲ برای تست استفاده شده است. برای تست از ۵۰۰ تصویر پایگاه داده NRCS استفاده شده است. حملات استفاده‌شده شامل Ker۱ [۱۱]، Ker۲ [۱۱]، CNGL [۱۵] و ALE [۱۴] است که روش‌های پنهان‌شکنی مخصوص LSBM می‌باشند. یکی از پارامترهای

جدول ۳: مقایسه میانگین PSNR و SSIM برای ۲۰۰ تصویر تست پایگاه داده NRCS.

درصد جاسازی	روش جاسازی	میانگین PSNR	میانگین SSIM
۰٫۳ bpp	LCG [۲۵]	۵۳٫۳۷	۰٫۹۹۸۲
	CBL [۲۷]	۵۶٫۳۵	۰٫۹۹۹۶
	MKGM(۴,۱۶)	۵۶٫۳۸	۰٫۹۹۹۲
۰٫۵ bpp	LCG [۲۵]	۵۱٫۱۵	۰٫۹۹۷۰
	CBL [۲۷]	۵۴٫۱۴	۰٫۹۹۹۱
	MKGM(۴,۱۶)	۵۴٫۳۴	۰٫۹۹۸۶
۰٫۸ bpp	LCG [۲۵]	–	–
	CBL [۲۷]	۵۲٫۱۰	۰٫۹۹۷۹
	MKGM(۴,۱۶)	۵۲٫۱۳	۰٫۹۹۷۵

جدول ۴: دقت کشف روش پیشنهادی با تعداد کلید و اندازه بلاک‌های مختلف توسط چهار حمله.

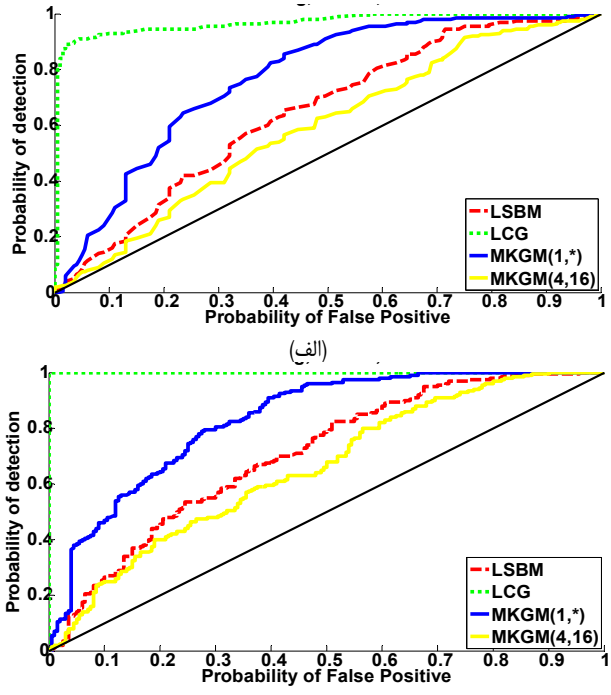
تعداد کلید	اندازه بلاک	Ker۱	Ker۲	ALE	CNGL
۱	۸×۸	۰٫۱۶۸۵	۰٫۰۷۸۴	۰٫۳۰۲۰	۰٫۲۰۰۳
	۱۶×۱۶	۰٫۱۸۲۳	۰٫۰۹۳۰	۰٫۳۱۳۹	۰٫۱۶۷۸
	۳۲×۳۲	۰٫۱۷۴۰	۰٫۰۹۰۵	۰٫۳۵۰۵	۰٫۲۳۰۶
۲	۸×۸	۰٫۱۶۸۶	۰٫۰۸۵۴	۰٫۳۲۷۳	۰٫۱۴۴۴
	۱۶×۱۶	۰٫۱۶۰۴	۰٫۰۸۵۴	۰٫۳۱۲۲	۰٫۱۶۰۱
	۳۲×۳۲	۰٫۱۷۴۲	۰٫۰۹۹۷	۰٫۳۳۰۰	۰٫۱۱۶۱
۴	۸×۸	۰٫۱۱۰۶	۰٫۰۹۳۱	۰٫۳۳۷۳	۰٫۱۱۲۶
	۱۶×۱۶	۰٫۱۵۸۷	۰٫۰۷۴۸	۰٫۲۹۹۹	۰٫۱۰۵۲
	۳۲×۳۲	۰٫۱۶۴۷	۰٫۰۹۱۵	۰٫۳۳۸۱	۰٫۱۵۲۷

عددی نشان‌دهنده دقت هر حمله، مقدار AUC است که این معیار عبارت است از مساحت میان نمودار ROC و قطر. این مساحت به گونه‌ای نرمالیزه می‌شود که مقدار آن برای یک روش کشف با موفقیت کامل، ۱ است. هرچه مقدار AUC به صفر نزدیک‌تر باشد، حمله مورد نظر ناموفق‌تر و در نتیجه روش جاسازی امن‌تر است. نتایج تست ارائه‌شده در جدول ۴، دقت کشف حملات مختلف برای درصد جاسازی ۰٫۵ bpp است.

نتایج جدول ۴ نشان می‌دهد که زیادشدن تعداد کلیدها تأثیر مستقیم روی امنیت روش دارد. زیادشدن تعداد کلیدها در واقع امکان بررسی دنباله‌های جاسازی بیشتر را فراهم می‌کند و بنابراین احتمال پیداشدن دنباله‌ای با تطابق بیشتر، بالاتر است. اما طبق توضیح قبل زیادکردن تعداد کلیدها، باعث افزایش طول توافقی میان فرستنده و گیرنده می‌شود.

بنابراین نمی‌توان خیلی تعداد کلیدها را افزایش داد. با کوچک‌ترشدن بلاک‌ها، تصمیمات محلی‌تر گرفته می‌شود و با توجه به پیوستگی نواحی موجود در تصویر، بلاک‌های کوچک‌تر برای تصمیم‌گیری مناسب‌تر هستند. اما کوچک‌ترشدن بیش از حد هم مناسب نیست. طبق نتایج جدول ۴، روش MKGM(۴,۱۶) در اکثر موارد امنیت بیشتری در برابر حملات دارد و حملات موجود کمتر قادر به کشف آن هستند.

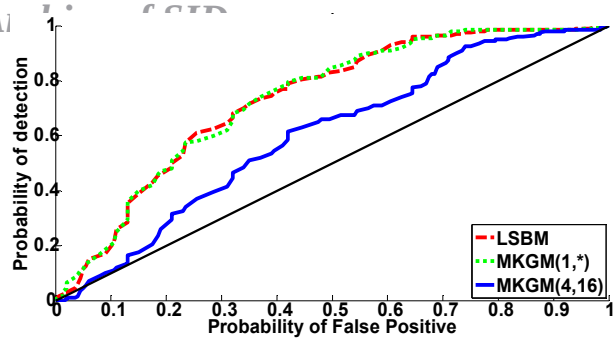
در گام آخر، امنیت روش‌های MKGM(۱,*) و MKGM(۴,۱۶) نسبت به روش LSBM اصلی و LCG مقایسه می‌شود تا مشخص شود روش پیشنهادی به چه اندازه در بهبود امنیت روش LSBM در برابر حملات موجود موفق بوده است. برای تست از ۵۰۰ تصویر پایگاه داده NRCS استفاده شده است. این مقایسه در سه درصد جاسازی ۰٫۳ bpp، ۰٫۵ bpp و ۰٫۸ bpp انجام شده و دقت کشف حملات مختلف به تفکیک درصد جاسازی در جدول ۵ ارائه شده‌اند. به علاوه برای درک بهتر، نمودار ROC حاصل از دو حمله Ker۱ و ALE در شکل‌های ۸ و ۹ نشان داده



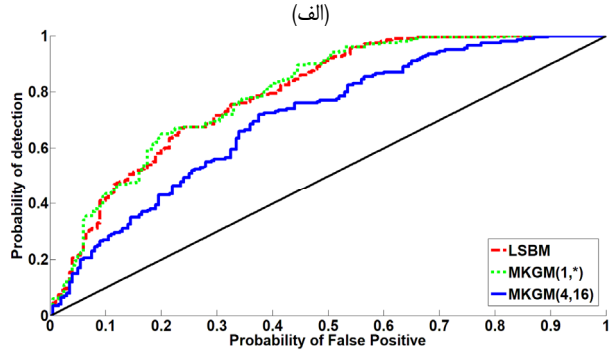
(الف)

(ب)

شکل ۸: نمودار ROC حمله (الف) Ker۱ و (ب) ALE برای چهار روش پنهان‌نگاری برای درصد جاسازی ۰٫۵ bpp



(الف)



(ب)

شکل ۹: نمودار ROC حمله (الف) Ker۱ و (ب) ALE برای سه روش پنهان‌نگاری برای درصد جاسازی ۰٫۸ bpp

جدول ۵: دقت کشف روش‌های مختلف توسط چهار حمله.

درصد جاسازی	روش جاسازی	Ker۱	Ker۲	CNGL	ALE
۰٫۳ bpp	LSBM	۰٫۱۵۲۲	۰٫۰۷۹۵	۰٫۰۹۹۵	۰٫۲۸۸۹
	LCG [۲۵]	۰٫۸۹۸۶	۰٫۶۸۸۱	۰٫۰۸۹۹	۱
	MKGM(۱,*)	۰٫۱۵۹۰	۰٫۰۶۲۲	۰٫۱۲۹۵	۰٫۲۸۴۰
	MKGM(۴,۱۶)	۰٫۰۸۶۷	۰٫۰۵۷۹	۰٫۰۸۹۰	۰٫۲۷۲۴
۰٫۵ bpp	LSBM	۰٫۲۹۲۳	۰٫۱۴۴۳	۰٫۱۴۴۵	۰٫۴۰۹۹
	LCG [۲۵]	۰٫۹۴۴۳	۰٫۸۲۲۸	۰٫۲۳۵۰	۱
	MKGM(۱,*)	۰٫۲۸۳۴	۰٫۱۳۴۹	۰٫۲۱۰۴	۰٫۴۰۳۸
	MKGM(۴,۱۶)	۰٫۱۵۸۷	۰٫۰۷۴۸	۰٫۱۰۵۲	۰٫۲۹۹۹
۰٫۸ bpp	LSBM	۰٫۴۶۵۱	۰٫۲۳۲۵	۰٫۲۶۰۹	۰٫۵۸۴۴
	LCG [۲۵]	–	–	–	–
	MKGM(۱,*)	۰٫۴۶۷۵	۰٫۲۵۴۷	۰٫۲۳۹۸	۰٫۵۹۵۳
	MKGM(۴,۱۶)	۰٫۲۰۷۱	۰٫۰۹۲۸	۰٫۱۷۹۶	۰٫۴۱۱۵

شده است. هر چه نمودار ROC یک روش به قطر نزدیک‌تر باشد، نشان‌دهنده امنیت بالاتر آن روش یا به عبارت دیگر موفقیت کمتر حمله مورد نظر در کشف روش است.

بررسی نمودارهای شکل ۸ و ۹ نشان می‌دهد اگرچه MKGM(۱,*) نتایجی مشابه و حتی در اندکی از موارد بدتر از LSBM دارد، اما روش MKGM(۴,۱۶) امنیت بالاتری نسبت به آن دو روش دارد. بنابراین می‌توان نتیجه گرفت بدون وجود ایده بلاک‌بندی و فقط استفاده از الگوریتم ژنتیک و چندکلیدی به تنهایی نمی‌توانست کمک زیادی به بهبود امنیت LSBM کند.

۴-۲ تحلیل عملکرد روش MKGM

بررسی نتایج حاصل از تست‌های مختلف نشان می‌دهد که روش پیشنهادی با استفاده از سه ایده مختلف موفق به بهبود عملکرد روش پایه LSBM شده است. در ادامه تأثیر هر کدام از این ایده‌ها بر عملکرد روش پیشنهادی بررسی می‌شود و به علاوه چالش‌ها و نحوه استفاده از هر کدام از آنها در روش‌های پنهان‌نگاری دیگر (با هدف بهبود عملکرد روش پنهان‌نگاری) نیز بحث می‌شود:

- **تأثیر ایده چندکلیدی:** هر الگوریتم پنهان‌نگاری تعدادی پارامتر ورودی دارد که یکی از آنها کلید است. معمولاً این کلیدها از قبل و بدون در نظر گرفتن معیار خاصی انتخاب می‌شوند. اما کلیدهای مختلف، تصاویر استگوی مختلف تولید می‌کنند که میزان تغییرات ویژگی‌های آماری این تصاویر استگو نسبت به تصویر پوشش متفاوت است. ایده چندکلیدی پیشنهاد می‌دهد که با تست کلیدهای مختلف، کلیدی انتخاب شود که بهترین خروجی را داشته باشد. تعیین معیار انتخاب بهترین خروجی، چالش بسیار مهم این ایده است. با توجه به حملات موجود برای LSBM، هیستوگرام تصویر به عنوان ویژگی آماری هدف در روش MKGM انتخاب شده است. نتایج نشان می‌دهد که برای تغییر کمتر در هیستوگرام، کلیدی انتخاب می‌شود که پیکسل‌های کمتری را تغییر دهد و در نتیجه معیار PSNR بهبود می‌یابد. اما ویژگی‌های آماری بسیار دیگری نیز وجود دارد که می‌تواند به عنوان معیار هدف انتخاب شود. یکی از مزایای این ایده، امکان استفاده از آن در روش‌های مختلف پنهان‌نگاری موجود بدون نیاز به تغییر الگوریتم جاسازی است.
- **تأثیر ایده الگوریتم بهینه‌سازی:** در روش MKGM، تبدیل روش LSBM به یک مسئله جستجو و حل آن با استفاده از الگوریتم بهینه‌سازی به بهبود روش LSBM کمک کرده است. انتخاب یک الگوریتم بهینه‌سازی از میان الگوریتم‌های موجود، انتخاب تابع برازندگی مناسب و تعیین پارامترهای الگوریتم بهینه‌سازی انتخاب‌شده از چالش‌های استفاده از ایده الگوریتم بهینه‌سازی است. با توجه به بررسی‌های انجام‌شده، در این حوزه استفاده از الگوریتم ژنتیک نسبت به روش‌های دیگر رایج‌تر است. به همین دلیل در روش MKGM نیز از الگوریتم ژنتیک استفاده شده است. برخلاف اغلب روش‌های موجود در حوزه پنهان‌نگاری که از PSNR یا

MSE به عنوان تابع برازندگی استفاده کرده‌اند، در روش MKGM از اختلاف هیستوگرام به عنوان تابع برازندگی استفاده شده و بهبود امنیت این روش در برابر حملات نتیجه این انتخاب است. اما از نظر نویسندگان، انتخاب الگوریتم بهینه‌سازی و تابع برازندگی یک مسئله باز در حوزه پنهان‌نگاری است و نیاز به تحقیقات بیشتر دارد و می‌توان به بهبود بیشتر عملکرد روش‌های پنهان‌نگاری با کمک الگوریتم‌های بهینه‌سازی امیدوار بود.

- **تأثیر ایده بلاک‌بندی:** اگرچه در ابتدا به نظر می‌رسید این ایده تأثیر چندانی در عملکرد روش MKGM نداشته باشد، اما نتایج تست نشان‌دهنده تفاوت محسوس عملکرد روش پیشنهادی بدون بلاک‌بندی ($MKGM(n,*)$) و با بلاک‌بندی ($MKGM(n,m)$) است. تأثیرگذاری این ایده به دلیل وجود وابستگی محلی در تصویر است. به عبارت دیگر، هنگام جاسازی بهتر است محدوده فضای جاسازی را کوچک‌تر کرد و برای تصمیم‌گیری در مورد نحوه جاسازی در هر محدوده از ویژگی‌های آن محدوده استفاده کرد. مزیت این ایده، سادگی و قابلیت استفاده در روش‌های پنهان‌نگاری مختلف است. اما تعیین اندازه بلاک‌ها، مهم‌ترین چالش این ایده است که می‌توان با تست مقادیر مختلف، چگونگی تأثیرگذاری این پارامتر را سنجید. اما ذکر این نکته لازم است که ایده بلاک‌بندی در هر دو حوزه پنهان‌نگاری و پنهان‌شکنی کارایی دارد. روش‌های پنهان‌شکنی وجود دارد که با استفاده از ایده بلاک‌بندی موفق به بهبود حملات خود شده‌اند [۲۸].

۵- نتیجه‌گیری

یکی از مهم‌ترین نقاط ضعف روش‌های پنهان‌نگاری که باعث کشف آنها می‌شود، تغییر هیستوگرام تصویر استگو نسبت به تصویر پوشش است. روش LSBM، یکی از ساده‌ترین روش‌های پنهان‌نگاری است که بهبود امنیت آن، هدف اصلی روش پیشنهادی ارائه‌شده است. روش GLSBM از ترکیب LSBM و الگوریتم ژنتیک ساخته شده و در این روش برای تصمیم‌گیری در مورد افزایش یا کاهش پیکسل‌های غیر مطابق، از الگوریتم ژنتیک استفاده گردیده است. در روش پیشنهادی به منظور تلاش برای انتخاب بهترین مکان‌ها برای جاسازی، روش GLSBM برای بلاک‌های مختلف تصویر با کلیدهای متفاوت اجرا می‌شود و برای هر بلاک بهترین مقدار کلید انتخاب می‌شود. نتایج تست برای تعیین پارامترهای روش پیشنهادی نشان می‌دهد که این روش در حالتی که از بلاک‌بندی ۱۶×۱۶ و ۴ کلید مختلف استفاده می‌شود، بهترین کارایی را دارد. به علاوه تست‌های انجام‌شده نشان می‌دهد اگرچه روش‌های مبتنی بر LSBM کیفیت تصویر استگوی بسیار خوبی دارند، اما برتری روش پیشنهادی نسبت به روش‌های قبلی در برابر حملات بسیار محسوس است. نتایج تست نشان می‌دهد هر کدام از ایده‌های استفاده‌شده به امنیت روش پیشنهادی کمک کرده‌اند.

مراجع

[1] Y. J. Chanu, T. Tuithung, and K. M. Singh, "A short survey on image steganography and steganalysis techniques," in *Proc. IEEE 3rd National Conf. on Emerging Trends and Applications in Computer Science,* pp. 52-55, Shillong, India, 30-31 Mar. 2012.

[2] V. Sabeti, S. Samavi, M. Mahdavi, and S. Shirani, "Steganalysis and payload estimation of embedding in pixel differences using neural networks," *Pattern Recognition*, vol. 43, no. 1, pp. 405-415, Jan. 2010.

[3] M. Hussain, A. W. A. Wahab, Y. I. B. Idris, A. T. Ho, and K. H. Jung, "Image steganography in spatial domain: a survey," *Signal Processing: Image Communication*, vol. 65, pp. 46-66, Jul. 2018.

[4] J. Kadhim, P. Premaratne, P. J. Vial, and B. Halloran, "Comprehensive survey of image steganography: techniques, evaluations, and trends in future research," *Neurocomputing*, vol. 335, pp. 299-326, Mar. 2019.

[5] A. Westfeld and A. Pfitzman, "Attacks on steganographic systems," *Lecture Notes in Computer Science,* vol. 1768, pp. 61-75, Springer, 2000.

[6] J. Fridrich, M. Goljan, and R. Du, "Detecting LSB steganography in color and gray-scale images," *Magazine of IEEE Multimedia*, vol. 4, no. 8, pp. 22-28, Oct-Dec. 2001.

[7] S. Dumitrescu, X. Wu, and Z. Wang, "Detection of LSB steganography via sample pair analysis," *IEEE Trans. on Signal Processing*, vol. 51, no. 7, pp. 1995-2007, Jul. 2003.

[8] A. Ker and R. Bohme, "Revisiting weighted stego-image steganalysis," *Proc. of SPIE Security, Forensics, Steganography, and Watermarking of Multimedia Contents X*, vol. 6819, pp. 1-17, 27-31 Jan. 2008.

[9] A. Westfeld, "Detecting low embedding rates," in *Proc. Int. Workshop on Information Hiding*, pp. 324-339, Oct. 2002.

[10] J. J. Harmsen and W. A. Pearlman, "Steganalysis of additive noise modelable information hiding," in *Proc. of SPIE Electronic Imaging, Security, Steganography, and Watermarking of Multimedia Contents*, vol. 5020, pp. 131-142, Santa Clara, CA, USA, 20-20 Jun. 2003.

[11] A. Ker, "Steganalysis of LSB matching in grayscale images," *IEEE Signal Processing Letters*, vol. 12, no. 6, pp. 441-444, Jun. 2005.

[12] A. Ker, "Resampling and the detection of LSB matching in color bitmaps," in *Proc. of SPIE Security, Steganography and Watermarking of Multimedia Contents VII,* pp. 1-15, 2005.

[13] J. Zhang, I. J. Cox, and G. Doerr, "Steganalysis for LSB matching in images with high-frequency noise," in *Proc. of the IEEE Workshop on Multimedia Signal Processing*, pp. 385-388, Crete, Greece, 1-3 Oct. 2007.

[14] G. Cancelli, I. J. Cox, and G. Doerr, "Improved LSB matching steganalysis based on the amplitude of local extrema," in *Proc. IEEE Int. Conf. on Image Processing*, *ICIP'08,* Oct. 2008.

[15] F. Huang, B. Li, and J. Huang, "Attack LSB matching steganography by counting alteration rate of the number of neighbourhood gray levels," in *Proc. IEEE Int. Conf. on Image Processing, ICIP'07*, vol. 1, pp. 401-404, San Antonio, TX, USA, 16-19 Sept. 2007.

[16] M. Goljan, J. Fridrich, and T. Holotyak, "New blind steganalysis and its implications," in *Proc. of SPIE Security, Steganography, and Watermarking of Multimedia Contents VIII,* vol. 6072, pp. 1-13, San Jose, CA, USA, 2006.

[17] G. Cancelli, G. Doerr, I. J. Cox, and M. Barni, "A comparative study of +1 steganalyzers," in *Proc. IEEE Int. Workshop on Multimedia Signal Processing,* pp. 791-96, Cairns, Australia, 8-10 Oct. 2008.

[18] N. Provos, "Defending against statistical steganalysis," in *Proc. 10th USENIX Security Symp.*, pp. 323-335, Washington, DC, USA, 13-17 Oct. 2001.

[19] C. K. Chan and L. M. Cheng, "Hiding data in images by simple LSB substitution," *Pattern Recognition*, vol. 37, pp. 469-474, 2004.

[۲۰] م. مهدوی، ش. سماوی، م. اخوت و ص. اکرمی، "روش پنهان‌نگاری تطبیقی بر اساس اغتشاش جمع‌شونده،" *پانزدهمین کنفرانس مهندسی برق ایران*، جلد مخابرات، ۹ صص.، تهران، ایران، اردیبهشت ۱۳۸۶.

[21] C. Liu, X. Li, X. Lu, and B. Yang, "A content-adaptive approach for reducing embedding impact in steganography," in *Proc. of the IEEE Int. Conf. on Acoustics, Speech and Signal Processing, ICASSP'10*, pp. 1762-1765, Dallas, TX, USA, 14-19 Mar. 2010.

[22] R. Rinita and S. Laha, "Optimization of stego image retaining secret information using Genetic Algorithm with 8-connected PSNR," *Procedia Computer Science*, vol. 60, pp. 468-477, 2015.

[23] H. R. Kanan and B. Nazeri, "A novel image steganography scheme with high embedding capacity and tunable visual image quality based on a genetic algorithm," *Expert Systems with Applications*, vol. 41, no. 14, pp. 6123-6130, Oct. 2014.

[24] R. Wang, C. Lin, and J. Lin, "Image hiding by optimal LSB substitution and genetic algorithm," *Pattern Recognition*, vol. 34, no. 3, pp. 671-683, Mar. 2001.

[25] P. D. Shah and R. S. Bichkar, "A secure spatial domain image steganography using genetic algorithm and linear congruential generator," in Dash S., Das S., Panigrahi B. (eds) *Int. Conf. on Intelligent Computing and Applications.* Advances in Intelligent Systems and Computing, vol 632. Springer, Singapore, 2018.

[26] D. C. Wu and W. H. Tsai, "A steganographic method for images by pixel-value differencing," *Pattern Recognition Letters*, vol. 24, no. 9-10, pp. 1613-1626, Jun. 2003.
[27] V. Sabeti, S. Samavi, and S. Shirani, "An adaptive LSB matching steganography based on octonary complexity measure," *Multimedia Tools and Applications*, vol. 64, no. 3, pp. 777-793, 2013.
[28] S. Cho, B. H. Cha, M. Gawecki, and C. C. J. Kuo, "Block-based image steganalysis: algorithm and performance evaluation," *J. of Visual Communication and Image Representation*, vol. 24, no. 7, pp. 846-856, Oct. 2013.

وجیهه ثابتی تحصیلات خود در هر سه مقطع کارشناسی، کارشناسی ارشد و دکترا را در رشته مهندسی کامپیوتر در دانشگاه صنعتی اصفهان و به‌ترتیب در سال‌های ۱۳۸۴، ۱۳۸۶ و ۱۳۹۱ به پایان رسانده است و هم‌اکنون به عنوان استاد گروه کامپیوتر در دانشکده مهندسی دانشگاه الزهرا مشغول به فعالیت می‌باشد. زمینه‌های تحقیقاتی مورد علاقه ایشان عبارتند از: پردازش تصویر، امنیت داده (پنهان‌نگاری، ته‌نقش نگاری و ...)، محاسبات نرم و الگوریتم‌های بهینه‌سازی.

سیده سپیده فیاضی تحصیلات خود را در مقطع کارشناسی مهندسی نرم‌افزار، در سال ۱۳۹۳، از دانشگاه خرم‌آباد به پایان رسانده است. او سپس مقطع کارشناسی ارشد مهندسی نرم‌افزار را در سال ۱۳۹۸ در دانشگاه الزهرا (س) تهران به پایان رساند. موضوعات تحقیقاتی مورد علاقه ایشان عبارتند از: پردازش تصویر، پنهان‌نگاری و الگوریتم های بهینه‌سازی تک هدفه و چند هدفه.

حدیثه شیرین‌خواه تحصیلات خود را در مقطع کارشناسی مهندسی نرم افزار، در سال ۱۳۹۱، از دانشگاه غیرانتفاعی ایوانکی به پایان رسانده است. او سپس مقطع کارشناسی ارشد مهندسی نرم‌افزار را در سال ۱۳۹۵ در دانشگاه الزهرا (س) تهران به پایان رساند. موضوعات تحقیقاتی مورد علاقه ایشان عبارتند از: پردازش تصویر و پنهان‌نگاری.