مقاله پژوهشی

مجله ماشین‌بینایی و پردازش‌تصویر
سال دهم، شماره دوم، تابستان ۱۴۰۲

طراحی روش جدید در تولید کلید رمزنگاری بیومتریک بر اساس تصویر قطعه‌بندی شده اثر انگشت

محمدرضا روزبهانی[۱]، ساناز سیدین[۲]، بهرام رشیدی[۳]

چکیده

هدف این مقاله استفاده از ویژگی‌های بیومتریک اثر انگشت برای دستیابی به کلیدهای رمزنگاری تصادفی می‌باشد. پیچیدگی الگوریتم تولید کلید، تعداد بیت بالا و تصادفی بودن سه فاکتور مهم برای کلیدهای رمزنگاری قوی می‌باشند. در روش پیشنهادی، ابتدا ویژگی‌های بیومتریک یعنی نقاط مینوشیا را با پردازش تصویر اثر انگشت استخراج می‌کنیم. سپس برای افزایش پیچیدگی روش تولید کلید و امنیت کلید تولیدی، هر تصویر را به ۴۴ قطعه تقسیم می‌کنیم تا با محاسبه فاصله اقلیدوسی و زاویه بین پیکسل‌های مرکزی هر ۴۴ قطعه با کل مینوشیاهای تصویر بتوانیم داده‌های تصادفی را افزایش دهیم. جهت افزایش بیشتر حالت تصادفی کلید، یک الگوریتم سه-گامه پیشنهاد می‌کنیم که شامل قرار دادن اعداد مربوط به زاویه و فاصله بصورت زوج و فرد در کنار یکدیگر، دو شکل جابجایی و جایگشت بیت‌ها و اعمال توزیع یکنواخت روی داده‌ها برای تولید کلید نهایی می‌باشد. به علت بالا بودن تعداد بیت کلید، می توان با استخراج زیر کلیدهای ۱۲۸، ۲۵۶ و ۵۱۲ بیتی از ماتریس کلید مذکور در رمزنگاری از آنها استفاده نمود. آنالیزهای آماری انجام شده همچون مجموعه تست‌های استاندارد NIST، تصادفی بودن و امنیت بالای کلید نهایی ۶۳۷۵۱ بیتی را اثبات می‌کند، و نشان دهنده عملکرد بهتر روش پیشنهادی در مقایسه با کارهای گذشته می‌باشد که تنها از فاصله یا زاویه بین نقاط مینوشیا برای تولید کلید رمزنگاری تصادفی با طول بیت بسیار کمتر استفاده کرده‌اند. الگوریتم پیشنهادی، با توجه به ۱۵ تست NIST، نسبت به کارهای جدید گذشته تا ۲۰٪ از نظر تصادفی بودن کلید تولید شده بهبود دارد.

کلید واژه‌ها

قطعه‌بندی تصویر اثر انگشت، کلید رمزنگاری تصادفی، مینوشیا، توزیع یکنواخت، فاصله اقلیدوسی، جایگشت.

۱- مقدمه

با توجه به پیشرفت سریع و گسترده نقل و انتقال داده‌ها و سیگنال‌های دیجیتالی در بسترهای مخابراتی، برقراری امنیت برای آنها بسیار ضروری می‌باشد. یکی از بهترین و مطمئن ترین روش‌ها برای حفظ امنیت داده‌های ارسالی استفاده از سیستم‌های رمزنگاری می‌باشد. کلیدهای رمزنگاری هسته اصلی این سیستم‌ها هستند. پیچیدگی کلید یکی از جنبه‌های مهم برای افزایش پیچیدگی الگوریتم رمزنگاری است که می‌تواند موجب بهبود عملکرد سیستم طراحی شده و حفاظت در برابر حمله‌های مولد اعداد تصادفی[۱]، متن اصلی[۲] و متن آشکار[۳] [1] شود. کلیدهای مورد استفاده در رمزنگاری باید از لحاظ تعداد بیت، طولانی و کاملا تصادفی باشند و از الگوی خاصی پیروی نکنند تا قابل کشف نباشند. تولید کلید رمزنگاری به چهار روش کلی انجام می‌شود. ۱- کلید رمزنگاری

این مقاله در شهریورماه ۱۴۰۱ دریافت، در آبان‌ماه بازنگری و سپس پذیرفته شد.

[۱] دانشجوی کارشناسی ارشد مهندسی برق، دانشگاه صنعتی امیرکبیر، تهران، ایران
رایانامه: ruzbahani@aut.ac.ir

[۲] دانشکده مهندسی برق، دانشگاه صنعتی امیرکبیر، تهران، ایران
رایانامه: sseyedin@aut.ac.ir

[۳] دانشکده فنی و مهندسی، گروه مهندسی برق، دانشگاه آیت‌الله بروجردی(ره)، بروجرد، ایران
رایانامه: b.rashidi@abru.ac.ir

نویسنده مسئول: ساناز سیدین

dorl.net/dor/20.1001.1.23831197.1402.10.2.1.1

[1] Random Number Generator attack
[۲] Cipher text-only attack
[۳] Known-plaintext attack

مبتنی بر تئوری آشوب، ۲- کلید رمزنگاری مبتنی بر الگوریتم‌های تکاملی، ۳- کلید رمزنگاری مبتنی بر خصوصیات بیومتریک، ٤- کلید رمزنگاری مبتنی بر مولدهای اعداد تصادفی [2,3]. سیستم‌های آشوب به دلیل حساسیت به شرایط اولیه و شباهت به رفتارهای تصادفی در بسیاری از زمینه‌ها مانند فشرده‌سازی، رمزنگاری و مدولاسیون مورد استفاده قرار می‌گیرند. امکان بازگشت از نوسانات آشفته و بی نظم به حالت اولیه، باعث شده از روش‌های آشوب به وفور در رمزگذاری داده‌ها، تولید کلید و رمزگشایی استفاده شود. یکی از روش‌های جدید تولید کلید، استفاده از مدل سازی فرآیندهای تکاملی ساده شده می‌باشد که به عنوان یک کلاس از الگوریتم‌های بهینه‌سازی در نظر گرفته می‌شود. دلیل معرفی این روش‌ها این است که تولید کلید رمزنگاری به روش‌های سنتی به علت پیچیدگی الگوریتم، زمان و توان مصرفی زیادی را می‌طلبد. کلید رمزنگاری مبتنی بر خصوصیات بیومتریک، روشی است که از اطلاعات بیومتریک جهت تولید کلیدهای رمزنگاری برای محافظت از امنیت داده‌ها استفاده می‌کند. سیستم‌های رمزنگاری بیومتریک یک زمینه تحقیقاتی در حوزه رمزنگاری می‌باشند که در آنها از بیومتریک یک فرد مانند صدا، چهره، اثرانگشت و عنبیه چشم برای تولید کلیدهای رمزنگاری استفاده می‌شود. در روش کلید رمزنگاری مبتنی بر مولدهای اعداد تصادفی، کلید رمزنگاری از مولدهای اعداد تصادفی مانند شیفت رجیستر بازخورد خطی[1] و شمارنده‌ها بدست می‌آید. علاوه بر چهار روش ذکر شده، شیوه‌هایی نیز برای توزیع امن کلید رمزنگاری بین رمزگذار و رمزگشا مبتنی بر اصول فیزیک کوانتوم وجود دارد. یکی از این روشها توزیع کلید کوانتوم (QKD)[2] و استفاده از پروتکل BB84 است. این پروتکل با استفاده از اصل عدم قطعیت در فیزیک کوانتومی، می‌تواند حضور استراق سمع کننده در طول ارتباط را آشکار کند [4].

براساس یک سیستم رمزنگاری قوی که شالوده‌ی اصلی آن کلید تصادفی با طول بیت زیاد می‌باشد می‌توان داده‌های دیجیتال را با امنیت بیشتری رمزگذاری کرد. در این مقاله، یک الگوریتم تولید کلید تصادفی براساس ویژگی‌های منحصر به فرد اثر انگشت پیشنهاد می‌کنیم. امنیت کلید تولید شده در گرو تصادفی بودن، تعداد بیت بالا و پیچیدگی الگوریتم مولد کلید می‌باشد. در این الگوریتم تلاش می‌کنیم هر سه نیاز مذکور را در نظر بگیریم و کلید مدنظر با بالاترین کیفیت از دید این معیارها را تولید کنیم. در بسیاری از تکنیک‌های رمزنگاری به کلیدهایی با طول بالای ۱۰۰۰۰ بیت نیاز است. از روش‌های موجود تولید کلید با استفاده از اثر انگشت که در [5-8] پیشنهاد شده است به علت تولید کلیدهایی با تعداد بیت محدود نمی‌توان در تکنیک‌های رمزنگاری مذکور استفاده کرد. اما یک مزیت مهم روش پیشنهادی این است که در این کار می‌توان کلیدهای تولید شده را به علت تعداد بیت و امنیت بالا در انواع تکنیک‌های رمزنگاری مورد استفاده قرار داد. در الگوریتم پیشنهادی تولید کلید رمزنگاری، ابتدا ویژگی‌های بیومتریک و منحصر به فرد از اثر انگشت که شامل نقاط مینوشیا هستند استخراج می‌شوند. سپس برای افزایش پیچیدگی الگوریتم و بالابردن امنیت کلید تولید شده، هر تصویر اثر انگشت را به ۴۴ قطعه مساوی تقسیم می‌کنیم و پس از شناسایی نقاط مرکزی هر قطعه، مختصات پیکسل‌های مرکزی هر قطعه در ماتریس ذخیره می‌شود. آنگاه فاصله اقلیدوسی و زاویه بین پیکسل‌های مرکزی هر ۴۴ قطعه با کل مینوشیاهای تصویر را محاسبه می‌کنیم. لازم به ذکر است که در روش پیشنهادی، برخلاف اکثر روش‌های قبلی که از فاصله بین خود نقاط مینوشیا استفاده می‌کردند، فاصله بین پیکسل‌های مرکزی هر قطعه با کل مینوشیاها را بدست می‌آوریم. به این ترتیب، داده‌های تصادفی افزایش خواهند یافت که یکی دیگر از مزایای روش پیشنهادی این مقاله می‌باشد، و تا جایی که اطلاع داریم این شیوه تولید کلید تصادفی با قطعه‌بندی تصویر و محاسبه فاصله و زاویه بین مراکز قطعات با نقاط مینوشیا تاکنون پیشنهاد نشده است. سپس یک الگوریتم ۳ گامه برای افزایش تصادفی کردن کلید نهایی پیشنهاد می‌کنیم. در گام اول، با قرار دادن اعداد مربوط به زاویه و فاصله بصورت زوج و فرد در کنار یکدیگر آنهارا ذخیره می‌کنیم. در گام بعدی پس از نرمالیزاسیون و تولید داده های ۸-بیتی طی عملیات جابجایی، ۴ بیت پر ارزش با ۴ بیت کم ارزش جابجا می‌شوند و هر بیت نیز بصورت دو به دو با بیت بعدی جابجا می‌شود. در گام آخر برای دستیابی به کلید نامنظم و تصادفی، یک توزیع یکنواخت روی داده‌های نهایی اعمال می‌کنیم تا کلید نهایی به دست آید. درنتیجه برخلاف برخی روش‌ها که از LFSR و S-box جهت تصادفی کردن کلید استفاده می‌کردند، عملیات پیشنهادی تولید کلید در این مقاله ساختار بسیار ساده‌ای دارد که از دیگر مزایای مهم آن به شمار می‌رود. به علت بالا بودن تعداد بیت کلید، می‌توان از ماتریس کلید مذکور، زیر کلیدهای ۱۲۸، ۲۵۶ و ۵۱۲ بیتی استخراج کرد و در رمزنگاری از آنها استفاده نمود. درنتیجه با روش پیشنهادی، کلید مورد استفاده در رمزنگاری برای هر فرد نیز تصادفی خواهد بود که یکی دیگر از مزایای مهم رویکرد پیشنهادی می‌باشد. به عبارت دیگر، به دلیل ثابت بودن اثر انگشت هر فرد و درنتیجه ویژگیهای استخراج شده یعنی زاویه و فاصله نقاط مینوشیا با مراکز قطعه‌های هر تصویر، کل کلید تولید شده در الگوریتم پیشنهادی که بطور متوسط ۶۳۷۵۱ بیت می‌باشد ثابت خواهد بود. اما نکته مهم این است که به علت تعداد بسیار زیاد زیرکلیدهای متفاوت تصادفی ۱۲۸، ۲۵۶ یا ۵۱۲ بیتی که می‌توان از این کلید بزرگ در هر بار الگوریتم رمزنگاری نهایی استخراج کرد، کلید نهایی حتی برای هر فرد نیز تصادفی خواهد بود و درنتیجه قابل کشف نمی‌باشد. عملکرد بالای الگوریتم پیشنهادی از نظر تصادفی بودن، تعداد بیت

[1] Linear Feedback Shift Register

[2] Quantum Key Distribution

بالا، پیچیدگی و امن بودن با معیارهایی از جمله NIST[1] [9] بررسی می‌شود.

ساختار مقاله حاضر به این شکل است که در بخش دوم کارهای پیشین معرفی می‌شوند. روش پیشنهادی تولید کلید رمزنگاری بیومتریک از اثر انگشت را در بخش سوم معرفی می‌کنیم. بخش چهارم حاوی نتایج آزمایشات و تحلیل نتایج روش پیشنهادی می‌باشد. سرانجام، مقاله در بخش پنجم جمع‌بندی می‌شود.

۲- کارهای پیشین

تاکنون تکنیک‌های متعددی برای تولید کلید رمزنگاری پیشنهاد شده‌است. روشهای تولید کلید رمزنگاری را می‌توان به ۴ دسته کلی کلید رمزنگاری مبتنی بر تئوری آشوب، کلید رمزنگاری مبتنی بر مولدهای اعداد تصادفی، کلید رمزنگاری مبتنی بر الگوریتم‌های تکاملی و کلید رمزنگاری مبتنی بر خصوصیات بیومتریک تقسیم کرد [2,3].

در دسته کلیدهای مبتنی بر تئوری آشوب می‌توان به کارهای [10] و [11] اشاره کرد. در [10] از کلیدی مبتنی بر الگوریتم آشوب برای رمزنگاری عکس‌های پزشکی استفاده شده است. در این مقاله برای غلبه بر مشکل پایین بودن امنیت در سیستم‌های رمزنگاری مبتنی بر الگوریتم آشوب تک بعدی، که بخاطر کوچک بودن سایز کلید اتفاق می‌افتد روشی پیشنهاد شده است که طبق آن از زوج کلید آشوب لجستیکی و نقشه های خیمه‌ای برای رمزگذاری استفاده می‌شود. در کار [11] نویسندگان برای تولید کلید از یک نقشه بهبود یافته سینوسی بهره می‌گیرند که عملکرد بهتری را نسبت به نقشه سینوسی در تولید کلید و رمزنگاری ارائه می‌دهد. استفاده از نقشه بهبود یافته سینوسی باعث ایجاد هرج و مرج پیوسته در خروجی می‌شود که آشفتگی و تصادفی شدن کلیدها را در پی دارد.

در دسته کلیدهای مبتنی بر مولدهای اعداد تصادفی می‌توان به کارهای [12] و [13] اشاره کرد. در کار [12] برای تولید اولیه کلید رمزنگاری مولدهای اعداد تصادفی به کار گرفته می‌شوند. سپس جهت افزایش پیچیدگی و خصوصی کردن کلید تولید شده، از تابع عمومی هش و تابع درهم سازی SH-1 استفاده می‌شود. در [13] به منظور تولید کلیدهای رمزنگاری قوی از شیفت رجیستر فیدبک-دار خطی استفاده شده است که باعث کاهش سخت افزار می‌شود ولی در عوض کلیدهایی با امنیت پایین تولید می‌کند. همچنین از این کلیدها فقط می‌توان در رمزنگاری کلید متقارن استفاده کرد.

در مجموعه روشهای مبتنی بر الگوریتم‌های تکاملی می‌توان به [14] اشاره کرد. در [14] از مکانیسم بازسازی کلید فازی برای استخراج یک الگوی محافظت شده استفاده شده است. سپس خطای تصحیح کد (ECC) برای تولید کلیدهای تصادفی بکار می‌رود.

دسته بعدی که جزو متداولترین شیوه‌های تولید کلید رمزنگاری محسوب می‌شود روشهای مبتنی بر خصوصیات بیومتریک می‌باشد. در این دسته از اطلاعات بیومتریک جهت تولید کلیدهای رمزنگاری برای محافظت از امنیت داده‌ها استفاده می‌شود و همین مساله مهمترین مزیت این روشها می‌باشد که در این مقاله نیز از آن بهره می‌گیریم. در [15] نویسندگان از کلیدی که از خصوصیات چهره انسان استفاده می‌کند برای رمزگذاری پیام‌های صوتی و نهان کردن آنها در داخل تصویر رنگی استفاده کرده‌اند. در [16] که جزو روشهای بیومتریک می‌باشد، پس از بررسی کاربرد الگوریتم ژنتیک در رمزنگاری یک روش ترکیبی جدید برای رمز داده ها بر اساس الگوریتم ژنتیک و مولد اعداد شبه تصادفی پیشنهاد می‌شود. از ویژگی‌های چنین رویکردی می‌توان به امنیت بالا و امکانپذیری انتقال برنامه‌های چند رسانه‌ای اشاره کرد. در [17] کلیدهای امن رمزنگاری براساس تابع درهم تولید می‌شوند. رویکرد تولید کلید رمزنگاری از ویژگی‌های بیومتریک اثر انگشت را می‌توان در کار [18] یافت که علاوه بر تولید کلید برای ذخیره کلیدهای رمزنگاری ۲۵۶ بیتی از الگوهای بیومتریک استفاده می‌کند. در [19] ابتدا مینوشیاهای اثر انگشت استخراج می‌شود. سپس با اعمال تبدیل قابل لغو روی نقاط مینوشیا، به قالب های قابل لغو ۲۵۶ بیتی تبدیل می‌شوند و کلید نهایی بدست می‌آید. کلیدهای مبتنی بر اثر انگشت در بخش فرستنده و گیرنده با استفاده از استگانوگرافی مبتنی بر کلید منتقل می‌شوند و در رمزنگاری متقارن مورد استفاده قرار می‌گیرند. کلیدهای تولیدشده در این کار ۲۵۶ بیتی بوده و در حالت کلی به دلیل طول بیت کم در الگوریتم‌های رمزنگاری محدودیت استفاده دارند، زیرا امکان تصادفی بودن کلید با طول بیت کم کاهش می‌یابد. علاوه براین هیچگونه آنالیز آماری روی کلیدها صورت نگرفته تا صحت تصادفی بودن و امنیت کلیدها را تایید کند. در کار [20] با استفاده از جهت و فاصله بین مینوشیاها اثر انگشت‌های مصنوعی به وجود می‌آید و این جهت‌ها و فواصل برای تولید کلیدهای رمزنگاری استفاده می‌شوند. نویسندگان در کار [21] علاوه بر تولید کلید رمزنگاری، با تعیین فاصله بین مینوشیاها، برای تضمین امنیت در طول انتقال داده از تصحیح خطای دو لایه استفاده می‌کنند. [22] یک رویکرد تولید کلید بر اساس اثر انگشت با استفاده از فواصل نسبی بین مینوشیاهای اثر انگشت کاربر برای ایجاد یک کلید بیومتریک منحصر به فرد را پیشنهاد داده است. علاوه براین، از یک تکنیک تصحیح خطای دو لایه برای افزایش قابلیت اطمینان سیستم در طول انتقال داده استفاده کرده است. [23] یک روش برای تولید کلید رمزنگاری بر اساس ویژگی های منحصر به فرد اثر انگشت و الگوریتم آستانه[2] پیشنهاد می‌کند. در کارهای [5, 6] از ویژگی‌های منحصر به فرد و

[1] National Institute of Standards and Technology

[2] Threshold

بیومتریک اثر انگشت به منظور تولید کلیدهای تصادفی رمزنگاری استفاده شده است. از محدودیت‌های این دو کار می‌توان به استخراج محدود داده از اثر انگشت نام برد. برای تولید کلید در [5] فاصله اقلیدوسی بین مینوشیاها همانند بسیاری از روشهای پیشین به کار رفته است. همچنین برای افزایش میزان تصادفی بودن کلیدها از S-box های ۸ بیتی در رمز قالبی AES استفاده شده است که حجم محاسبات را بالا می‌برد. سپس روی کلید تولید شده تنها یک عمل جابجایی بیتی اعمال می‌شود. کلید تولید شده با این روش همانند روش‌های دیگری که تعداد بیت پایین دارند، در رمزنگاری‌های کلید عمومی که به کلیدهایی با تعداد بیت بسیار بالا نیاز دارند، قابل استفاده نخواهد بود. در کار [6] علاوه بر فاصله، از زوایای بین نقاط مینوشیا هم بهره گرفته شده است که به افزایش طول بیت کلید کمک می‌کند. همچنین برای افزایش ویژگی‌های آماری و پیچیدگی الگوریتم از S-box های ۸-بیتی به کار برده شده در رمز CLEFIA به منظور جایگشت استفاده شده است. علاوه بر این نویسندگان، برای جابجایی از دو عدد LFSR ۱۰-بیتی نیز استفاده کرده‌اند که این امر موجب کند شدن عملیات تولید کلید و به تبع آن، کند شدن عملیات رمزنگاری می‌گردد. در [7] برای دستیابی به کلیدهای رمزنگاری از فاصله و زاویه بین مینوشیاها و جهت آنها استفاده شده است. سپس رشته بیت هایی با طول ۲۰۴۸ بیت استخراج می‌شوند و کلیدهای نهایی با طول ۲۵۶ بیت تولید می‌گردند. از محدودیت های این روش نیز می‌توان به طول بیت پایین کلیدهای تولید شده اشاره کرد. در [8] برای تولید کلید ابتدا تمام نقاطی که با احتمال بالا می‌توانند به عنوان نقاط مینوشیا شناخته شوند از تصویر اثر انگشت استخراج می‌شوند. سپس از بین همین نقاط، نقاط شبه مینوشیا نیز حذف می‌شوند تا بیومتریک بودن کلید تولید شده حفظ شود. در ادامه فاصله اقلیدوسی بین مینوشیاها محاسبه و ذخیره می‌شود. در انتها هم اعداد به دست آمده توسط کد خاکستری به اعداد باینری ۸-بیتی تبدیل می‌شوند و تمام این اعداد باینری در کنار یکدیگر قرار می‌گیرند و رشته بیت نهایی بدست می‌آید. در این کار به علت حذف شبه مینوشیاها ممکن است تعدادی از مینوشیاهای اصلی نیز حذف گردد، که این امر موجب کاهش نقاط مینوشیا، کاهش فواصل بین آنها و در نتیجه کاهش تعداد بیت های کلید می‌شود. همچنین میانگین تعداد بیت‌های کلید تولید شده در این روش ۱۶۸ بیت می‌باشد که به دلیل کوتاه بودن مطلوب نیست. علاوه براین کلید رمزنگاری باید تصادفی و دارای امنیت بالا باشد که در این کارهم تصادفی بودن کلید تولید شده مورد چالش قرار نگرفته و تایید نشده است.

همچنین از روشهای توزیع کلید کوانتومی نیز برای افزایش امنیت در توزیع کلیدهای رمزنگاری بین فرستنده و گیرنده استفاده شده است [24]. در [24] از پروتکل اصلاح شده و بهبود یافته BB84 بهره می‌گیرند. در این روش جدید توزیع کلید، رشته بیت‌های متوالی که کلید را تشکیل می‌دهند با استفاده از Symbol Legender به فوتون‌های قطبیده تبدیل می‌شوند. سپس برخلاف پروتکل BB84 که فوتون‌ها از طریق کانال کلاسیک جابجا می‌شوند در این روش فقط از کانال کوانتومی استفاده می‌شود که در نهایت منجر به صرف زمان و نرخ خطای کوانتومی کمتری می‌شود.

نویسندگان در کار [25] با بهره گیری از رمز قالبی و نقشه‌های آشوبناک به رمزگذاری و رمزگشایی تصاویر می‌پردازند. رمزگذاری داده‌هایی مانند تصاویر که دارای همبستگی بالا هستند، برای رمزهای بلوکی یک چالش است. زیرا ممکن است الگوهای تصویر اصلی پس از رمزگذاری باقی بمانند. در این کار برای غلبه بر این محدودیت از رمزهای بلوکی زنجیره‌ای به همراه نقشه‌های آشوبناک استفاده کرده‌اند. همچنین رمزهای بلوکی زنجیره‌ای مذکور، با الهام از معماری Rijndael و توسط نقشه‌های آشوبناک که به عنوان منبع آنتروپی می‌باشد، به عنوان الگوریتم اصلی رمزنگاری در نظر گرفته شده‌اند. این عملیات ترکیبی و الگوریتم پیشنهاد شده موجب بهبود نتایج رمزنگاری و شاخص‌های امنیتی نسبت به الگوریتم Rijndael شده است. در کار [26] برای رمزنگاری و انتقال امن و سریع فایل‌های متنی بین دستگاه‌ها و وسایل IOT از الگوریتم رمزنگاری متقارن کوچک (NTSA) استفاده کرده‌اند. یکی از امن‌ترین الگوریتم‌ها در ارتباط با انتقال اطلاعات بین وسایل IOT الگوریتم رمزنگاری کوچک است. یکی از محدودیت‌ها و ضعف‌های این الگوریتم استفاده از کلیدهای مشابه در تمام مراحل رمزنگاری است، که موجب کاهش امنیت می‌گردد. همچنین زمان رمزگذاری و رمزگشایی داده های متنی توسط این الگوریتم بسیار زیاد است. الگوریتم پیشنهاد شده با تغییر کلید و ایجاد درهم ریختگی در تک تک مراحل رمزنگاری باعث بهبود امنیت آن شده است.

در این مقاله یک روش تولید کلید تصادفی برای رمزنگاری تصاویر دیجیتال اثر انگشت با دقت و امنیت بالا مبتنی بر روشهای بیومتریک پیشنهاد شده است. از جمله نکات مهم در تولید این کلید تصادفی، افزایش طول بیت آن و پیچیدگی الگوریتم می‌باشد تا امنیت بالایی در سیستم رمزنگاری حاصل کند و در برابر حملات مقاوم باشد. در روشهای پیشین همچون [8] و [12] الگوریتم‌هایی برای تولید کلید تصادفی پیشنهاد شده‌اند، اما همچنان کلید استخراج شده مشکل طول بیت کم را دارد که موجب کاهش امنیت کلید تولید شده خواهد شد. این طول بیت کم خصوصا در زمان رمزنگاری اثر انگشت افراد یکسان مشکل خود را نشان می‌دهد، زیرا تعداد زیرکلیدی که می‌توان با سایزهای ۱۲۸، ۲۵۶ یا ۵۱۲ بیت در رمزنگاری از آن استخراج کرد محدود می‌شود. این محدود شدن تعداد زیرکلیدها باعث می‌شود که کلید ارسالی برای هر فرد دیگر تصادفی نباشد، یا میزان تصادفی بودن کاهش یابد و درنتیجه امکان کشف کلید وجود خواهد داشت. لذا در این مقاله با هدف استفاده از رویکردی ساده که حجم محاسباتی زیادی نداشته باشد و مبتنی بر استخراج نقاط مینوشیا باشد، الگوریتمی برای ساخت کلیدهای تصادفی بیومتریک پیشنهاد

می‌کنیم که مشابه روشهای قبلی مبتنی بر استخراج نقاط مینوشیا باشد، ولی برای افزایش سایز کلید پیشنهاد می‌کنیم که برخلاف روشهای قبلی، تصویر قطعه‌بندی شود و به جای فاصله/زاویه بین نقاط مینوشیا، فاصله و زاویه بین کل نقاط مینوشیا با مراکز قطعات تصویر محاسبه شود. به این ترتیب سایز کلید به میزان قابل توجهی افزایش می‌یابد و با الگوریتم‌های ساده جایگشت می‌توان کلید تصادفی نهایی را بدون افزایش حجم محاسباتی پیدا کرد.

۳- روش پیشنهادی تولید کلید رمزنگاری بیومتریک از اثر انگشت

هدف ما از این مقاله، استفاده از ویژگی‌های بیومتریک اثرانگشت در راستای تولید کلید تصادفی با طول بیت زیاد می‌باشد تا بعدا بتوان از آن در سیستم رمزنگاری با امنیت بالا استفاده کرد. در این رویکرد ما از مختصات پیکسل‌های مرکزی قطعات تصویر و مینوشیا بهره می‌گیریم. نمای کلی از مراحل روش پیشنهادی در شکل (۱) بصورت شماتیک نمایش داده شده است. در این مقاله ما از روش ارائه شده در کار [27] برای بهبود تصویر و استخراج مینوشیا استفاده می‌کنیم. فرایند کلی به این صورت است که ابتدا پیش پردازش روی تصویر به منظور حذف نویز و تغییرات ناشی از فشار انگشت روی سنسور صورت می‌گیرد. سپس حذف پیکسل‌های اضافی از تصویر موجب بهبود روشنایی و کنتراست می‌شود. همچنین برای تشخیص و استخراج بهتر نقاط مینوشیا، از باینری کردن و نازک‌سازی تصویر استفاده می‌شود. پس از استخراج نقاط مینوشیا از تصویر، مختصات x و y آنها بصورت جداگانه در دو ماتریس Fx و Fy ذخیره می شوند. در ادامه برای افزایش تعداد بیت کلید نهایی و در نتیجه تصادفی بودن آن، تصویر اصلی را به ۴۴ قطعه مساوی تقسیم می‌کنیم. سپس پیکسل‌های مرکزی هر قطعه را شناسایی و استخراج می‌کنیم و مختصات x و y آنها را در دو ماتریس Gx و Gy بصورت جداگانه ذخیره می‌نماییم. این قطعه‌بندی تصویر در عین سادگی الگوریتم، به دلیل افزایش زیاد بیت‌های تولیدی نهایی کلید یکی از پیشنهادات بسیار موثر پژوهش حاضر است که در کارهای پیشین تا آنجا که اطلاع داریم انجام نشده است. در ادامه به محاسبه فاصله اقلیدوسی بین نقاط مینوشیا و نقاط مرکزی هر قطعه و نرمال‌سازی آنها برای ایجاد اعداد ۸ بیتی پرداخته می‌شود و نتایج حاصل در یک ماتریس ذخیره می‌شود. بعد از این مرحله زاویه بین نقاط مینوشیا و نقاط مرکزی هر قطعه محاسبه شده و مقدار به دست آمده با انجام محاسباتی که در ادامه اشاره خواهد شد در بازه‌ی (0 , 360) درجه و در چهار ربع دستگاه مختصاتی قرار می‌گیرد. آنگاه نرمال‌سازی برای ایجاد اعداد ۸ بیتی روی زاویه‌های به دست آمده، صورت می‌گیرد. به عبارت دیگر، در این مقاله جهت افزایش داده‌های تصادفی، برخلاف روش‌های پیشین که فاصله یا زاویه بین مینوشیاها را پیدا می‌کردند و درنتیجه به دلیل ثابت بودن مینوشیاها به محدودیت می‌رسیدند، فاصله اقلیدوسی و زاویه بین پیکسل‌های

شکل (۱): نمای کلی از مراحل روش پیشنهادی برای تولید کلید تصادفی. اعداد داخل پرانتز در هر بلوک شماره روابط مرتبط می‌باشند.

مرکزی هر ۴۴ قطعه با کل مینوشیاهای تصویر را محاسبه می‌کنیم. در مرحله بعد داده‌های G و F بصورت زوج و فرد با یکدیگر ترکیب می‌شوند و ماتریس کلی کلید را می‌سازند. در مرحله آخر نیز به منظور بهبود کیفیت کلید به دست آمده، داده‌های ماتریس نهایی با استفاده از توزیع یکنواخت جابجا می‌شود و کلید نهایی با قابلیت اطمینان بالا بصورت تصادفی به دست می‌آید.

۳-۱- قطعه‌بندی تصویر

دراین بخش از کار، به منظور دستیابی به داده‌های بیشتر، هر تصویر مطابق شکل (۲) به ۴۴ قطعه (بر حسب طول و عرض تصویر اثر انگشت) تقسیم می‌کنیم. آنگاه مختصات x و y پیکسل‌های مرکزی هر قطعه در بردارهای جداگانه ذخیره می‌شود. فرض کنید مجموعه نقاط پیکسل‌های مرکزی با G نمایش داده شود در اینصورت $g(x_i , y_i)$ مختصات پیکسل مرکزی یک قطعه را نشان می‌دهد و n=44 خواهد بود.

$$G = \{g_1(x_1, y_1), g_2(x_2, y_2), g_3(x_3, y_3), \dots, g_n(x_n, y_n)\} \quad (۱)$$

در ادامه مختصات پیکسل‌های مرکزی هر قطعه را در دو بردار متفاوت (G_x, G_y) ذخیره می‌کنیم، که بردار G_x و G_y به ترتیب شامل تمام مقادیر مختصات x و y پیکسل‌های مرکزی هر قطعه می‌باشند:

$$G_x = [x_1, x_2, x_3, \dots, x_n]$$
$$G_y = [y_1, y_2, y_3, \dots, y_n] \quad (۲)$$

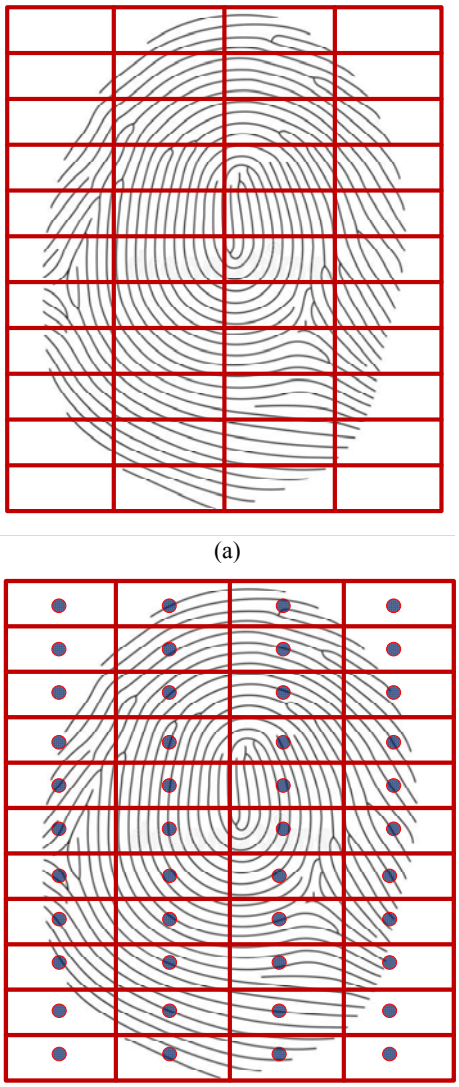
در این کار از پایگاه داده اثر انگشت DB3_B FVC2002 , DB1_B استفاده شده است. ابعاد هر تصویر در DB1_B برابر 374×388 و در DB3_B بصورت 300×300 می‌باشد [28]. برای یکپارچه کردن و به کارگیری تمام پایگاه‌های داده در روش پیشنهادی، ابعاد تصاویر پایگاه داده DB3-B با تکرار پیکسل‌های انتهایی برابر 374 × 388 قرار می‌گیرد. برای تقسیم تصویر به قطعات مساوی و دستیابی به حد کافی از داده‌ها، هر تصویر به ۴۴ قطعه مساوی تقسیم می‌شود. عدد ۴۴ با آزمایش تجربی و با توجه به ابعاد تصاویر موجود در پایگاه داده FVC2002 DB1_B و FVC2002 DB3_B انتخاب شده است.

این قطعه‌بندی منجر به تولید کلیدهایی با طول زیاد می‌شود. این کلیدها قابل استفاده در الگوریتم‌های رمزنگاری کلید عمومی و همچنین الگوریتم‌های کلید خصوصی می‌باشند. زیرا در رمزنگاری کلید عمومی به کلیدهایی با تعداد بیت بالا و در رمزنگاری کلید خصوصی به کلیدهایی با تعداد بیت کمتر نیاز است.

۳-۲-فاصله اقلیدوسی بین مینوشیا و پیکسل‌های مرکزی

در ادامه فاصله اقلیدوسی بین مینوشیاها و پیکسل‌های مرکزی قطعات محاسبه و دریک ماتریس ذخیره می‌شود. فاصله اقلیدوسی دو نقطه فرضی براساس بردارهای G_x , G_y با رابطه (۳) قابل محاسبه می‌باشد:

$$D_p = \sqrt{\left(G_x(i) - G_y(j)\right)^2 + \left(G_y(i) - G_x(j)\right)^2} \quad (۳)$$



(a)

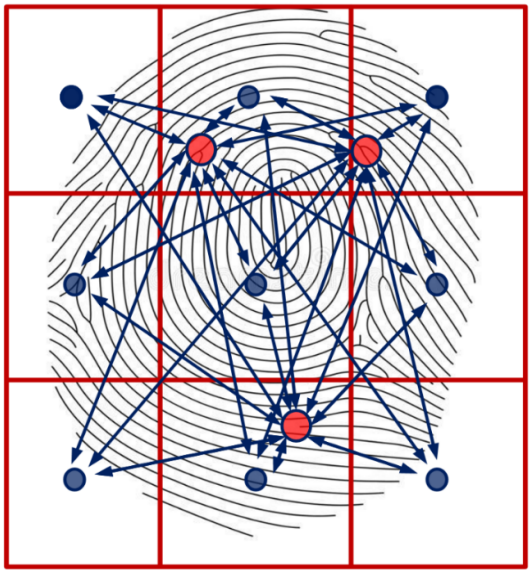
(b)

شکل (۲): (a): تقسیم کردن تصویر به ۴۴ قطعه مساوی (b): پیداکردن پیکسل های مرکزی هر ۴۴ قطعه.

تعداد فاصله‌های اقلیدوسی برابر حاصلضرب تعداد مینوشیاها در تعداد پیکسل‌های مرکزی قطعات خواهد بود که در این کار، چون تصویر به ۴۴ قطعه تقسیم شده است تعداد داده‌های مستخرج از فاصله اقلیدوسی از رابطه (۴) به دست می‌آید:

$$N_{Dp} = 44 \times \text{تعداد مینوشیاها} \quad (۴)$$

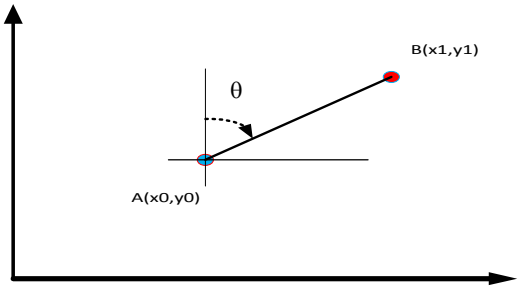
شکل (۳) فاصله اقلیدوسی بین پیکسل‌های مرکزی قطعات تصویر و چند مینوشیای فرضی را بطور نمادین نشان می‌دهد. در این شکل به علت جلوگیری از شلوغ شدن، تصویر به ۹ قسمت مساوی تقسیم شده و دارای ۳ مینوشیا می‌باشد. همچنین از رسم فاصله بین سایر پیکسل‌های قطعات و مینوشیاها پرهیز شده است.
از آنجا که تعداد نقاط مینوشیا در اثر انگشت محدود می‌باشد، در صورت عدم قطعه‌بندی به روش پیشنهادی، تصادفی بودن کلید به دست آمده کاهش می‌یابد.



شکل (۳): فاصله اقلیدوسی بین پیکسل‌های مرکزی قطعات و مینوشیاهای فرضی در یک تصویر سمبلیک.

۳-۳- زاویه بین مینوشیاها و پیکسل‌های مرکزی هر قطعه

در ادامه برای دستیابی به داده‌های بیشتر، زاویه بین پیکسل‌های مرکزی هر قطعه و مینوشیاها را پیدا می‌کنیم و در ماتریس جداگانه ذخیره می‌نماییم. فرض کنید دو نقطه با مختصات $A(x_0, y_0)$ و $B(x_1, y_1)$ وجود دارد، زاویه بین این دو نقطه با توجه به شکل (۴) بصورت روابط (۵) و (۶) محاسبه می‌شود. این زاویه نسبت به محور عمود سنجیده می‌شود.



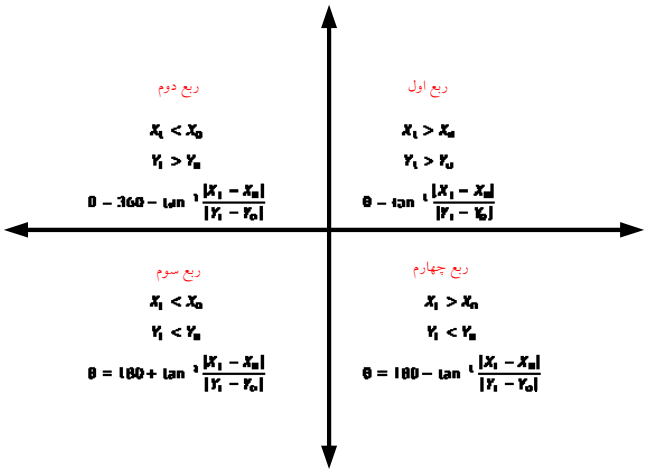
شکل (۴): محاسبه زاویه بین دو نقطه.

$$(x_1, y_1) = (x_0 + r\ sin\theta\ , y_0 + r\ cos\theta\ ) \qquad (۵)$$

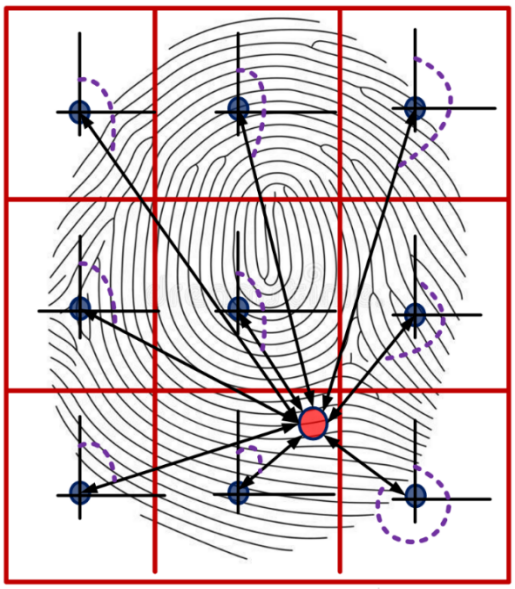
که r طول خط AB می باشد. در اینصوت θ بصورت رابطه (۶) محاسبه می شود.

$$\tan\theta = \frac{x_1 - x_0}{y_1 - y_0} \rightarrow \theta = tan^{-1}\frac{x_1 - x_0}{y_1 - y_0} \qquad (۶)$$

زاویه به دست آمده بین این دو نقطه با توجه به قوانین تانژانت در محدوده 90 و 90- درجه می‌باشد. برای اینکه زاویه به دست آمده در محدوده (360 , 0) درجه قرار گیرد باید چندین شرط هنگام محاسبه زاویه درنظر گرفته شود که در شکل (۵) شروط چهارگانه و روابط مربوط به محاسبه زاویه برای چهار ربع آورده شده است.
پس از قرار دادن زاویه‌ها در محدوده (360 , 0) آنهارا در ماتریس جداگانه ذخیره می‌کنیم. در شکل (۶) بصورت نمادین تصویر به ۹ قسمت مساوی تقسیم شده است و زاویه بین یک مینوشیای فرضی و نقاط مرکزی قطعات برای درک بهتر آورده شده است. در تصاویر اثر انگشت، محدود بودن نقاط مینوشیا باعث کاهش میزان تصادفی بودن و امنیت کلید تولید شده می‌شود. استفاده از فاصله اقلیدوسی و زوایای بین پیکسل‌های مرکزی قطعات و نقاط مینوشیا علاوه بر افزایش پیچیدگی و تصادفی بودن، باعث دستیابی به داده‌های بسیار زیاد می‌شود. برای مثال اگر یک تصویر دارای ۹۰ نقطه مینوشیا باشد، پس از اندازه‌گیری فاصله اقلیدوسی و زاویه بین پیکسل‌های مرکزی هر قطعه و نقاط مینوشیا، اعداد به دست آمده را تبدیل به اعداد باینری ۸-بیتی می‌کنیم و به یک داده باینری با طول ۶۳۳۶۰ بیت دست پیدا می‌کنیم.



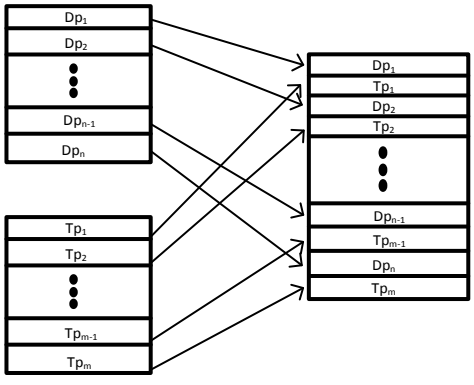
شکل (۵): شروط چهارگانه و روابط مربوط جهت محاسبه زاویه برای چهار ربع.



شکل (۶): زاویه بین پیکسل های مرکزی قطعات و مینوشیای فرضی در یک تصویر سمبلیک.

۳-۴- ترکیب دو ماتریس فاصله و زاویه بصورت زوج و فرد

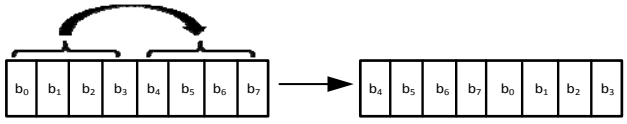
پس از محاسبه زاویه و فاصله اقلیدوسی بین نقاط مینوشیا و پیکسل‌های مرکزی هر قطعه داده‌های مربوط به فاصله و زاویه را بطور جداگانه نرمال‌سازی می‌کنیم و در ماتریس‌های D_p و T_p به ترتیب ذخیره می‌نماییم. نحوه نرمال‌سازی به این صورت است که برای قرار گرفتن داده‌ها در قالب اعداد ۸-بیتی، باقیمانده آنها نسبت به عدد 2^8 محاسبه و در ماتریس مربوط ذخیره می‌شود. بعد از نرمالیزاسیون، برای رسیدن به یک ماتریس واحد دو ماتریس فاصله و زاویه را بصورت زوج و فرد مطابق شکل (۷) با یکدیگر ترکیب می‌کنیم. اگر ماتریس Dp شامل n بردار مربوط به فاصله اقلیدوسی و ماتریس Tp شامل m بردار مربوط به زاویه باشد، ماتریس نهایی شامل Z=n+m داده خواهد بود. درایه‌های زوج ماتریس نهایی شامل داده‌های ماتریس Tp و درایه‌های فرد ماتریس مذکور شامل داده‌های ماتریس Dp می‌باشد. این جابجایی را جهت افزایش تصادفی بودن، بالا رفتن پیچیدگی الگوریتم و در نتیجه بهبود کلید تولید شده پیشنهاد کرده‌ایم.



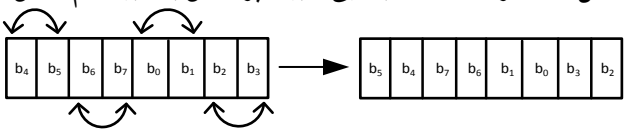
شکل (۷): ترکیب ماتریس فاصله (D_p) و زاویه (T_p) بصورت زوج و فرد.

۳-۵- جابجایی بیتی

در این مرحله درایه‌های ماتریس نهایی را که هر یک شامل یک عدد ۸-بیتی می‌باشند طی عملیات شیفت و چرخش در دو مرحله جابجا می‌کنیم. در مرحله اول ۴ بیت پرارزش با ۴ بیت کم ارزش جابجا می‌شوند. سپس در مرحله دوم هر بیت بصورت دو به دو با بیت بعدی جابجا می‌شود. این عملیات نیز یکی دیگر از پیشنهادات مقاله به منظور ایجاد داده‌های تصادفی‌تر روی تک تک داده های ۸-بیتی می‌باشد. در شکل های (۸) و (۹) یک درایه از ماتریس نهایی نمایش داده شده است که روند اعمال جابجایی‌های پیشنهادی روی آن در دو مرحله انجام شده است.



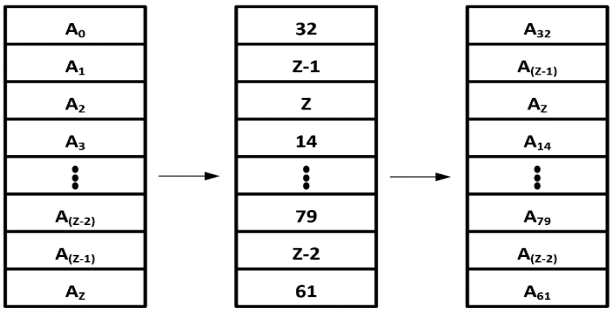
شکل (۸): مرحله اول: جابجایی ۴ بیت پر ارزش با ۴ بیت کم ارزش.



شکل (۹): مرحله دوم: جابجایی بیت های کناری بصورت دو به دو.

۳-۶- جایگشت

پیشنهاد ما در این بخش برای جابجایی درایه‌های ماتریس نهایی کلید تولید شده و افزایش تصادفی بودن کلید، استفاده از توزیع یکنواخت می‌باشد. به این صورت که ابتدا داده‌های تصادفی توسط تابع یکنواخت تولید می‌شوند. سپس از این اعداد به عنوان اندیس استفاده می‌کنیم و طبق آن روی داده‌های ماتریس نهایی جابجایی انجام می‌دهیم. این عملیات بمنظور افزایش پیچیدگی و تصادفی بودن کلید تولید شده صورت می‌گیرد. شکل (۱۰) روند کلی جایگشت را نشان می‌دهد.



شکل (۱۰): روند کلی جایگشت کلید. (a): ماتریس کلید قبل از اعمال توزیع یکنواخت. (b): تولید اعداد تصادفی توسط توزیع یکنواخت به تعداد درایه‌های کلید و اعمال آن روی ماتریس کلید. (c):تولید کلید نهایی پس ازاعمال توزیع یکنواخت وجایگشت درایه‌ها.

۴- آزمایشات و شبیه سازی

در این بخش جزئیات آزمایشات و نتایج و تحلیل آنها را می‌آوریم.

۴-۱- دادگان استفاده شده

در این کار از پایگاه داده اثر انگشت FVC2002 DB3_B , DB1_B [28] استفاده شده است. هر پایگاه داده شامل ۸۰ تصویر برای ۱۰ شخص متفاوت است، بصورتی که از اثر انگشت هر شخص ۸ تصویر مختلف وجود دارد. ابعاد هر تصویر در DB1_B 388×374 و در DB3_B بصورت 300 × 300 می‌باشد. رزولوشن تصاویر در هر دو پایگاه داده برابر با 500 dpi است. همچنین سنسور به کار رفته برای ثبت تصاویر در پایگاه داده DB1_B از نوع سنسور نوری Touch View II و برای پایگاه داده DB3_B از نوع سنسور خازنی FX2000 می‌باشد.

۴-۲-جزئیات شبیه‌سازی

در این روش ابتدا تمام تصاویر موجود در پایگاه داده FVC2002 DB1_B و FVC2002 DB3_B به تصاویر خاکستری و سپس به تصاویر باینری تبدیل می‌شوند. در ادامه با استفاده از مورفولوژی عمل نازک‌سازی تصاویر را انجام می‌دهیم. همچنین برای استخراج نقاط مینوشیا حول هر پیکسل پنجره‌هایی با ابعاد 3×3 اعمال می‌شود. الگوریتم استخراج نقاط مینوشیا طبق روش استاندارد در [27] انجام شده است. همچنین در روش پیشنهادی تصویر هر اثر انگشت را به ۴۴ قطعه مساوی تقسیم می‌کنیم. مطابق نتایج آزمایشات تجربی و با توجه به ابعاد تصاویر، دستیابی به داده‌های زیاد تصادفی و زیاد نشدن زمان تولید کلیدنهایی این عدد برابر ۴۴ انتخاب شده است. در این رویکرد، برای تصاویر موجود در پایگاه داده FVC2002 DB1_B مینیمم تعداد مینوشیا استخراج شده برابر ۲۶ و ماکزیمم آن ۶۴ می‌باشد. در حالی که برای پایگاه داده FVC2002 DB3_B این اعداد به ترتیب ۴۰ و ۱۰۵ بدست آمده است. سپس فاصله و زاویه بین مینوشیاها با پیکسل‌های مرکزی هر ۴۴ قطعه محاسبه می‌شود. در انتها هم دوعملیات جابجایی و جایگشت روی کلیدهای تولید شده صورت می‌گیرد و کلید نهایی به دست می آید. برای انجام این کار از ابزار شبیه سازی Matlab R2018b استفاده کرده‌ایم.

۴-۳- نتایج آزمایشات

آزمایش‌ها و تست‌های صورت گرفته در این کار عبارتند از: مجموعه تست‌های NIST، شمارش تعداد بیت‌های ۰و۱ کلیدهای تولید شده، نسبت تعداد یک‌های ماتریس کلید به کل بیت‌ها، طول بیت کلید تولید شده، زمان لازم جهت انجام مراحل الگوریتم پیشنهادی، تعداد کلیدهای ۱۲۸ بیتی قابل تولید از رشته بیت کلید نهایی، شمارش بیت‌های ۰و۱ به هم پیوسته با طول ۲ تا ۱۶ بیت، هیستوگرام و فاصله همینگ کلیدهای تولید شده.

مجموعه تست‌های NIST مجموعه‌ای از آنالیزهای آماری است که متشکل از ۱۵ تست است و برای بررسی تصادفی بودن دنباله های باینری ایجاد شده است [9]. دنباله‌های باینری ممکن است از مولدهای رمزنگاری تصادفی یا مولدهای رمزنگاری شبه تصادفی مبتنی بر سخت افزار یا نرم افزار تولید شده باشند. برای بکارگیری هرکدام از تست‌های NIST، باید تعداد بیت‌های دنباله مورد سنجش از مقدار آستانه تعریف شده برای آن تست بیشتر باشد.

۱۵ تست NIST عبارتند از: Monobit، Run، Binary Matrix Rank، Non Overlapping Template Match، Serial، Approximate Entropy، Cumulative Sums (forward)، Random Excursions، Frequency Test within a Block، Longest Run of Ones in a Block، Discrete Fourier Transform، Overlapping Template Matching، Linear Complexity، Cumulative Sums (Reverse) و Random Excursions Variant . این تست‌ها بر روی انواع مختلفی از بخش های غیرتصادفی که ممکن است در یک دنباله باینری وجود داشته باشد، تمرکز می‌کند. در ادامه به توضیح تعدادی از تست‌ها می-پردازیم:

Cumulative Sums (Cusum) test: در این تست هدف بررسی تصادفی بودن یا نبودن کل دنباله ورودی است. با تغییر بیت‌های (۰و۱) به (۱-و۱+) و جمع کردن بیت‌ها با یکدیگر هرچه به صفر نزدیکتر شویم، دنباله مدنظر تصادفی‌تر است.

Run test: این تست بررسی می‌کند که آیا تعداد صفرهای کنار هم و تعداد یک‌های کنار هم با طول‌های مختلف در کل دنباله، مطابق انتظار یک دنباله تصادفی می‌باشد یا خیر.

Binary Matrix Rank test: هدف از این آزمون بررسی وابستگی خطی بین رشته بیت‌های فرعی با طول ثابت در کل رشته بیت ورودی می‌باشد.

Linear Complexity test: تمرکز اصلی این تست روی طول شیفت رجیستر بازخورد خطی (LFSR) و میزان پیچیدگی دنباله ورودی می‌باشد. چنانچه طول LFSR زیاد شود پیچیدگی نیز بیشتر خواهد شد.

در جداول (۱) و (۲) نتایج ۱۵ تست NIST و مقادیر P-value برای تصاویر 3.tif_103، 1.tif_102، 1.tif_106، 6.tif_107، 7.tif_104 و 4.tif_101 از پایگاه داده اثر انگشت FVC2002 DB1_B آورده شده است. در جدول (۳) نتایج مجموعه تست‌های NIST برای تصویر 4.tif_105 با روش پیشنهادی و کارهای جدید دیگر آورده شده است. همچنین رشته بیت‌های تولیدی از نظر مقادیر P-value با هم مقایسه شده‌اند.

میانگین مقادیر P-value و میانگین تعداد تست‌هایی که تصادفی بودن کلیدهای تولید شده از پایگاه داده FVC2002 DB1_B را تایید می‌کنند، در جدول (۴) آمده است.

جدول (۵) مقادیر تعداد بیت‌های ۱ و ۰ و نسبت آنها به تعداد کل بیت‌های کلید در ماتریس کلید تولید شده از ۱۰ تصویر اثر انگشت را نشان می‌دهد. این مقادیر برای ماتریس‌های کلید تولید شده بر اساس ۱۰ تصویر اثر انگشت 101_1.tif تا 110_1.tif از پایگاه داده اثر انگشت FVC2002 DB1_B مورد بررسی قرار گرفته است.

جدول (۶) مقایسه نسبت یک‌های ماتریس کلید بر کل بیت‌ها را برای روش پیشنهادی و کارهای دیگر ارائه می‌دهد. در این مقایسه، ۱۰ تصویر از پایگاه داده FVC2002 DB1_B انتخاب شده است.

جدول (۷) مقایسه‌ی میانگین طول بیت کلیدهای تولید شده توسط روش پیشنهادی و روش‌های موجود را نشان می‌دهد. در این مقایسه برای ۸۰ تصویر پایگاه داده اثر انگشت FVC2002 DB3_B، ۸۰ کلید به دست آمده است و مقدار میانگین طول بیت آنها در جدول (۷) آمده است.

جدول (۸) تعداد کلیدهای تصادفی ۱۲۸ بیتی که می‌توان از میانگین رشته بیت‌کلیدهای نهایی استخراج کرد را در کار پیشنهادی و کارهای پیشین نمایش می‌دهد.

زمان لازم برای انجام تک تک مراحل الگوریتم پیشنهادی و مقایسه زمان تولید کلید نهایی با کارهای دیگر در جدول (۹) آورده شده است. این مراحل شامل: استخراج نقاط مینوشیا، قطعه‌بندی تصویر، محاسبه فاصله اقلیدوسی بین پیکسل‌های مرکزی قطعات و نقاط مینوشیا، محاسبه زوایای بین پیکسل‌های مرکزی قطعات و نقاط مینوشیا و قراردادن آنها در چهار ربع دستگاه مختصات، جابجایی بیتی و ترکیب دو ماتریس می‌باشد. میانگین زمانی برای هر تصویر براساس ۱۰ تصویر انتخابی از پایگاه داده اثرانگشت FVC2002 DB1_B به دست آمده است.

در جدول (۱۰) نتایج کلی حاصل از الگوریتم پیشنهادی و مقایسه آن با جدیدترین روش‌های این حوزه آورده شده است. این نتایج شامل: میانگین مقادیر P-value، میانگین تست‌های تایید شده NIST، طول بیت کلیدهای تولید شده، تعداد کلیدهای تصادفی ۱۲۸ بیتی قابل تولید از رشته بیت نهایی و زمان تولید کلید نهایی می‌باشد.

بخشهای (a) و (b) شکل (۱۱) به ترتیب شمارش تعداد بیت‌های صفر و یک به هم پیوسته با طول ۲ تا ۱۶ بیت در دو تصویر 106_1.tif و 108_1.tif از پایگاه داده اثر انگشت FVC2002 DB1_B را در روش پیشنهادی نشان می‌دهند. تعداد بیت به هم پیوسته در زیر هر ستون مشخص شده است.

در شکل (۱۲) هیستوگرام مقادیر کلید تصادفی برای دو تصویر 102_1.tif و 104_7.tif در روش پیشنهادی آورده شده است.

جدول (۱): نتایج ۱۵ تست NIST برای کلیدهای تولید شده به روش پیشنهادی روی ۳ تصویر انتخابی.

	نوع تست	مقدار P-value تصویر ۱-۱۰۲	مقدار P-value تصویر ۳-۱۰۳	مقدار P-value تصویر ۱-۱۰۶	نتیجه تست تصویر ۱-۱۰۲		نتیجه تست تصویر ۳-۱۰۳		نتیجه تست تصویر ۱-۱۰۶	
					تصادفی	غیرتصادفی	تصادفی	غیرتصادفی	تصادفی	غیرتصادفی
۱	Monobit	۰,۴۲۳۵۹	۰,۱۲۴۱۹	۰,۳۷۶۵۵	✓		✓		✓	
۲	Run	۰,۵۶۵۷۱	۰,۱۳۰۸۵	۰,۰۰۰۶۰	✓		✓			×
۳	Binary Matrix Rank	۰,۳۹۱۹۲	۰,۴۳۷۵۲	۰,۴۷۹۱۰	✓		✓		✓	
۴	Non Overlapping Template Match	۰,۴۴۰۱۸	۰,۰۶۰۱۶	۰,۶۹۹۸۲	✓		✓		✓	
۵	Serial	۰,۰۱۲۲۱	۰,۵۱۰۱۲	۰,۲۸۱۹۸	✓		✓		✓	
۶	Approximate Entropy	۰,۰۵۷۶۱	۰,۸۵۸۴۹	۰,۳۵۲۲۳	✓		✓		✓	
۷	Cumulative Sums(forward)	۰,۲۰۴۶۲	۰,۱۱۴۹۷	۰,۴۱۴۷۸	✓		✓		✓	
۸	Random Excursions	۰,۹۶۲۵۶	۰,۸۴۹۱۴	۰,۲۲۰۶۴	✓		✓		✓	
۹	Frequency Test within a Block	۰,۵۹۱۲۲	۰,۲۳۰۶۱	۰,۴۷۶۰۱	✓		✓		✓	
۱۰	Longest Run of Ones in a Block	۰,۱۲۷۳۲	۰,۰۵۸۵۴	۰,۶۴۸۱۳	✓		✓		✓	
۱۱	Discrete Fourier Transform	۰,۲۰۹۶۷	۰,۲۰۷۰۹	۰,۷۷۹۶۲	✓		✓		✓	
۱۲	Overlapping Template Matching	۰,۲۳۶۹۲	۰,۳۳۱۵۹	۰,۲۷۶۱۴	✓		✓		✓	
۱۳	Linear Complexity	۰,۸۸۸۴۷	۰,۳۰۱۸۱	۰,۸۰۱۷۸	✓		✓		✓	
۱۴	Cumulative Sums (Reverse)	۰,۷۸۴۳۴	۰,۲۴۴۷۸	۰,۲۵۸۷۱	✓		✓		✓	
۱۵	Random Excursions Variant	۰,۹۶۲۵۶	۰,۷۷۲۸۲	۰,۶۱۷۹۷	✓		✓		✓	

جدول (۲): نتایج ۱۵ تست NIST برای کلیدهای تولید شده به روش پیشنهادی روی ۳ تصویر انتخابی.

	نوع تست	مقدار-p value تصویر ۶-۱۰۷	مقدار-p value تصویر ۷-۱۰۴	مقدارp-value تصویر ۴-۱۰۱	نتیجه تست تصویر ۶-۱۰۷		نتیجه تست تصویر ۷-۱۰۴		نتیجه تست تصویر ۴-۱۰۱	
					تصادفی	غیرتصادفی	تصادفی	غیرتصادفی	تصادفی	غیرتصادفی
۱	Monobit	۰,۶۱۶۱۲	۰,۲۹۷۰۷	۰,۰۲۴۸۵۰	✓		✓		✓	
۲	Run	۰,۴۹۲۵۲	۰,۰۷۴۶۱	۰,۱۱۰۶۷	✓		✓		✓	
۳	Binary Matrix Rank	۰,۲۰۹۹۴	۰,۷۴۷۱۱	۰,۳۶۸۲۷	✓		✓		✓	
۴	Non Overlapping Template Match	۰,۶۱۳۰۶	۰,۴۸۶۵۵	۰,۰۶۴۴۰	✓		✓		✓	
۵	Serial	۰,۷۱۳۲۲	۰,۱۵۸۰۶	۰,۲۸۱۸۰	✓		✓		✓	
۶	Approximate Entropy	۰,۲۲۶۹۷	۰,۱۱۲۹۳	۰,۰۳۵۲۶	✓		✓		✓	
۷	Cumulative Sums(forward)	۰,۷۵۲۲۱	۰,۰۶۲۲۳	۰,۳۳۲۱۷	✓		✓		✓	
۸	Random Excursions	۰,۴۴۷۱۴	۰,۴۷۰۳۸	۰,۰۶۳۶۸	✓		✓		✓	
۹	Frequency Test within a Block	۰,۶۹۵۰۶	۰,۰۰۲۹۶	۰,۳۶۷۲۲	✓			×	✓	
۱۰	Longest Run of Ones in a Block	۰,۲۴۱۴۳	۰,۶۳۷۲۷	۰,۰۳۱۷۷	✓		✓		✓	
۱۱	Discrete Fourier Transform	۰,۳۷۵۳۶	۰,۶۱۲۴۴	۰,۴۲۰۹۲	✓		✓		✓	
۱۲	Overlapping Template Matching	۰,۷۵۱۷۰	۰,۹۹۸۹۵	۰,۷۱۹۷۳	✓		✓		✓	
۱۳	Linear Complexity	۰,۷۲۹۷۳	۰,۰۸۷۱۶	۰,۶۸۸۷۳	✓		✓		✓	
۱۴	Cumulative Sums (Reverse)	۰,۹۴۶۶۳	۰,۵۲۹۸۷	۰,۷۷۹۳۲	✓		✓		✓	
۱۵	Random Excursions Variant	۰,۲۱۵۹۲	۰,۷۰۵۴۵	۰,۳۳۷۳۵	✓		✓		✓	

جدول (۳): نتایج ۱۵ تست NIST برای یک کلید خاص تولید شده به روش پیشنهادی و روشهای دیگر

	نوع تست	کار [5]	کار [6]	روش پیشنهادی	کار [5]		کار [6]		روش پیشنهادی	
		مقدار-p value تصویر ۴-۱۰۵	مقدار-p value تصویر ۴-۱۰۵	مقدارp-value تصویر ۴-۱۰۵	نتیجه تست تصویر ۴-۱۰۵		نتیجه تست تصویر ۴-۱۰۵		نتیجه تست تصویر ۴-۱۰۵	
					تصادفی	غیرتصادفی	تصادفی	غیرتصادفی	تصادفی	غیرتصادفی
۱	Monobit	۰,۰۰۷۹۴	۰,۶۴۶۱۷	**۰,۶۵۵۰۴**		×	✓		✓	
۲	Run	۰,۰۸۱۷۷	۰,۰۳۸۵۰	**۰,۹۲۳۴۲**	✓		✓		✓	
۳	Binary Matrix Rank	۰,۰۰۳۹۷	۰,۰۲۴۸۶	**۰,۳۲۶۰۳**		×	✓		✓	
۴	Non Overlapping Template Match	۰,۰۴۶۱۰	۰,۵۴۳۲۸	**۰,۹۳۴۹۳**	✓		✓		✓	
۵	Serial	۰,۰۰۷۶۳	۰,۰۰۴۳۱	**۰,۴۰۱۷۶**		×		×	✓	
۶	Approximate Entropy	۰,۰۱۳۳	۰,۰۰۲۸۰	**۰,۰۸۲۶۷**	✓			×	✓	
۷	Cumulative Sums(forward)	۰,۰۰۵۰۶	۰,۲۸۸۹۸	**۰,۳۳۰۴۱**		×	✓		✓	
۸	Random Excursions	۰,۸۷۳۰۶	۰,۷۹۱۴۷	**۰,۹۰۲۰۳**	✓		✓		✓	
۹	Frequency Test within a Block	۰,۲۹۴۱۳	۰,۰۰۱۹۹	**۰,۳۴۷۹۸**	✓			×	✓	
۱۰	Longest Run of Ones in a Block	۰,۱۲۰۴۰	**۰,۶۳۰۸۳**	۰,۲۱۲۳۵	✓		✓		✓	
۱۱	Discrete Fourier Transform	**۰,۶۴۰۶۱**	۰,۲۹۲۲۵	۰,۱۶۴۰۹	✓		✓		✓	
۱۲	Overlapping Template Matching	۰,۰۰۷۵۳	**۰,۶۰۸۵۷**	۰,۴۶۵۱۹		×	✓		✓	
۱۳	Linear Complexity	۰,۳۲۴۹۵	۰,۰۰۴۹۱	**۰,۳۶۰۳۷**	✓			×	✓	
۱۴	Cumulative Sums (Reverse)	۰,۰۱۵۳۸	۰,۶۲۹۰۹	**۰,۶۸۳۹۸**	✓		✓		✓	
۱۵	Random Excursions Variant	۰,۴۶۳۱۴	**۰,۷۵۱۳۷**	۰,۲۱۰۱۵	✓		✓		✓	

جدول(۴): نتایج مجموعه تست‌های NIST و مقادیر p-value روی پایگاه داده FVC2002 DB1-B

میانگین مقادیر p-value	میانگین تست های تایید شده	کار
۰,۳۵۵	۱۱,۵	[5]
۰,۳۲۳	۱۲,۵	[6]
۰,۴۸۱	**۱۴,۵**	روش پیشنهادی

جدول (۵): مقادیر تعداد بیت های ۱ و ۰ در ماتریس کلید تولید شده از ۱۰ تصویر اثر انگشت.

شماره تصویر	تعداد صفرها	تعداد یک ها	تعداد کل یک و صفرها	نسبت صفرها به کل	نسبت یک ها به کل
۱-۱۰۱	۹۴۷۶	۸۸۲۸	۱۸۳۰۴	۰,۵۱۷	۰,۴۸۲
۲-۱۰۲	۱۶۷۳۱	۱۵۶۵۳	۳۲۳۸۴	۰,۵۱۶	۰,۴۸۳
۱-۱۰۳	۱۳۹۹۵	۱۲۷۵۷	۲۶۷۵۲	۰,۵۲۳	۰,۴۷۶
۱-۱۰۴	۲۳۳۱۱	۲۱۷۴۵	۴۵۰۵۶	۰,۵۱۷	۰,۴۸۲
۱-۱۰۵	۱۷۱۹۴	۱۵۸۹۴	۳۳۰۸۸	۰,۵۱۹	۰,۴۸۰
۱-۱۰۶	۲۰۰۱۲	۱۸۷۰۸	۳۸۷۲۰	۰,۵۱۶	۰,۴۸۳
۱-۱۰۷	۱۳۱۸۷	۱۲۱۵۷	۲۵۳۴۴	۰,۵۲۰	۰,۴۷۹
۱-۱۰۸	۲۵۳۶۷	۲۳۹۱۳	۴۹۲۸۰	۰,۵۱۴	۰,۴۸۵
۱-۱۰۹	۱۱۳۲۱	۱۰۵۰۳	۲۱۸۲۴	۰,۵۱۸	۰,۴۸۱
۱-۱۱۰	۱۴۷۴۹	۱۴۰۷۰	۲۸۸۱۹	۰,۵۱۱	**۰,۴۸۸**

جدول(۶): مقایسه معیار نسبت تعداد یک های ماتریس کلید بر تعداد کل بیت‌های کلید برای کار پیشنهادی و دیگر کارها.

میانگین معیار نسبت تعداد یک‌های ماتریس کلید بر تعداد کل بیت های کلید	کار
۰,۱۱۵۹	[11]
۰,۵۲۰۰	[13]
۰,۵۰۹۳	[6]
۰,۵۱۰۰	[5]
۰,۴۸۱۹	روش پیشنهادی

جدول (۷): مقایسه میانگین طول بیت کلیدهای تولید شده برای کار پیشنهادی و دیگر کارها.

طول بیت کلیدهای تولید شده	کار
۱۲۸	[8]
۲۵۶	[7]
۱۲۸	[12]
۲۰۴۸	[7]
۳۱۶۰	[5]
۶۳۲۰	[6]
۶۳۷۵۱	**روش پیشنهادی**

جدول(۸): تعداد کلیدهای تصادفی ۱۲۸ بیتی قابل تولید از متوسط رشته بیت‌های کلید نهایی

تعداد کلیدهای ۱۲۸ بیتی	کار
۱	[8]
۱	[12]
$۵,۷۷\times۱۰^{۷۵}$	[7]
$۱,۷۴\times۱۰^{۲۳۱}$	[5]
$۲,۲۰\times۱۰^{۲۷۰}$	[6]
$۴,۶۱\times۱۰^{۴۷۱}$	**روش پیشنهادی**

جدول (۹): میانگین زمان لازم جهت انجام تک تک مراحل الگوریتم تولید کلید رمزنگاری هر تصویر بر روی ۱۰ تصویر انتخابی برای کار پیشنهادی و دیگر کارها

نام مراحل	زمان(میلی‌ثانیه)
استخراج نقاط مینوشیا	۱۶
قطعه‌بندی	۲,۸
محاسبه فاصله اقلیدوسی	۱,۹
محاسبه زاویه	۶,۵
جابجایی بیتی	۳,۱
ترکیب ماتریس فاصله و زاویه	۰,۷
تولید کلید نهایی در روش پیشنهادی	**۳۱**
تولید کلید نهایی کار [5]	۴۶
تولید کلید نهایی کار [6]	۶۲

۳-۴- بحث و تحلیل نتایج

در این قسمت تمامی نتایجی که در بخش (۴-۲) آورده شده است مورد بررسی و تحلیل قرار می‌گیرد.

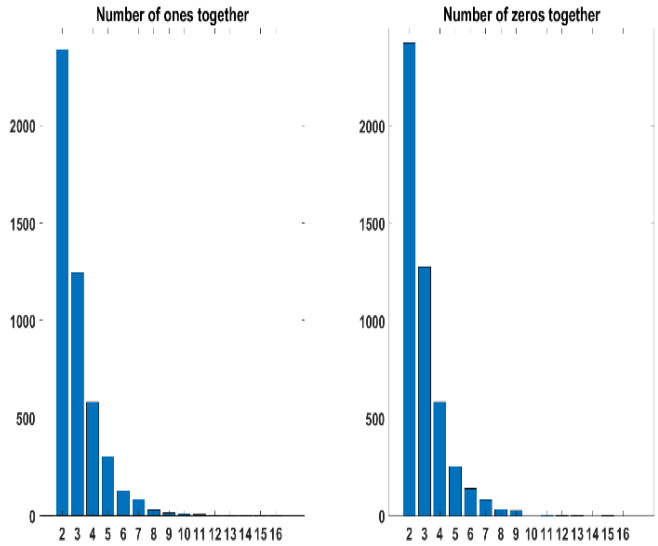
۱-۳-۴- بررسی مجموعه تست‌های NIST

برای تشخیص تصادفی بودن یا تصادفی نبودن یک دنباله در ۱۵ تست NIST از معیاری به اسم P-value استفاده می‌شود. اگر مقدار P-value ≥ 0.01 باشد، دنباله مورد نظر از دید آن تست تصادفی و اگر P-value < 0.01 باشد، آنگاه دنباله ورودی غیر تصادفی است. با توجه به نتایج NIST در جداول (۱) و (۲) مشاهده می‌شود که در اکثر تستها برای ۶ تصویر مورد تست، دنباله تصادفی تشخیص داده می‌شود. لذا الگوریتم پیشنهادی دارای قابلیت اطمینان بالایی است و می تواند دنباله‌های تصادفی ایجاد کند.

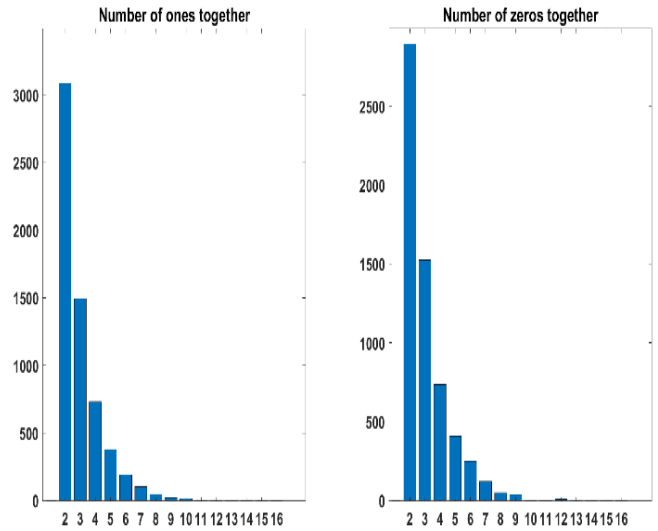
طبق جدول (۳)، تمامی تست‌های مجموعه NIST روی یک تصویر انتخاب شده، صحت تصادفی بودن کلید تولید شده توسط الگوریتم پیشنهادی را تایید می‌کنند. در حالی که در کارهای [5] و [6] نتایج برخی از تست‌های NIST تصادفی بودن کلید تولید شده نهایی را تایید نمی‌کنند.

جدول (۱۰): نتایج کلی حاصل از الگوریتم پیشنهادی و مقایسه با کارهای دیگران

تعداد کلیدهای تصادفی ۱۲۸ بیتی قابل تولید از رشته بیت‌های کلید نهایی	طول بیت کلیدهای تولید شده	میانگین زمان لازم برای تولید کلید نهایی (میلی‌ثانیه)	میانگین مقادیر p-value روی پایگاه داده FVC2002 DB1-B	میانگین تست‌های تایید شده از ۱۵ تست NIST	کار
$۱,۷۴\times۱۰^{۲۳۱}$	۳۱۶۰	۴۶	۰,۳۵۵	۱۱,۵	[5]
$۲,۲۰\times۱۰^{۲۷۰}$	۶۳۲۰	۶۲	۰,۳۲۳	۱۲,۵	[6]
$۴,۶۱\times۱۰^{۴۷۱}$	**۶۳۷۵۱**	**۳۱**	**۰,۴۸۱**	**۱۴,۵**	روش پیشنهادی

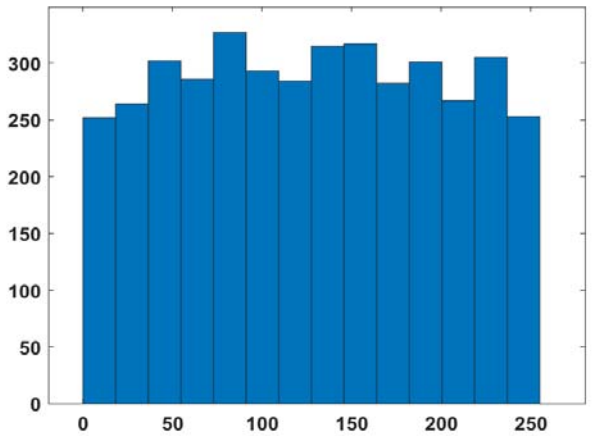


(a)

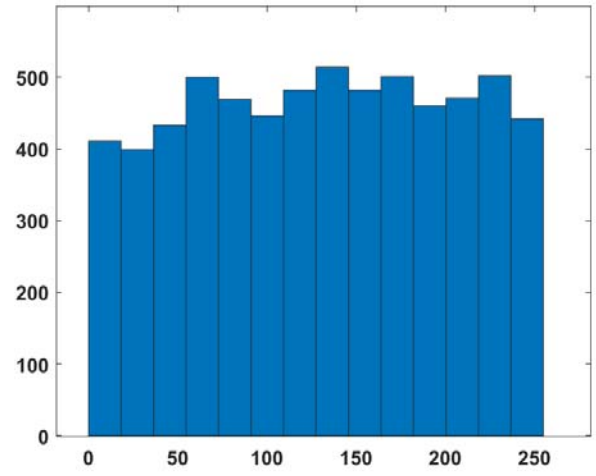


(b)

شکل (۱۱): نمودار تعداد شمارش بیت‌های ۰ و ۱ به هم پیوسته با طول ۲ تا ۱۶ بیت برای دو تصویر اثر انگشت (a) 106_1.tif و (b) تصویر 108_1.tif از پایگاه داده اثر انگشت FVC2002 DB1_B.



(a)



(b)

شکل (۱۲): هیستوگرام مقادیر ماتریس کلید تصادفی برای دو تصویر اثر انگشت (a) 102_1.tif و (b) 104_7.tif

تست mono bit به بررسی تصادفی بودن یا نبودن یک دنباله بیتی می‌پردازد. هرچه تعداد صفر و یک‌ها در دنباله به هم نزدیک‌تر باشد یعنی نسبت تعداد بیت‌های یک به تعداد کل نزدیکتر به 0.5 باشد، رشته بیت ورودی به حالت تصادفی نزدیکتر است. با توجه به جدول (۳) این تست، برای الگوریتم پیشنهادی تصادفی بودن و توزیع برابر صفر و یک‌ها را در کل دنباله نشان می‌دهد. همچنین توصیه شده است که مینیمم طول

بیت دنباله ورودی ۱۰۰ باشد. با توجه به این نکته نمی‌توان گفت که دلیل رد شدن این تست برای کار [5] کم بودن تعداد بیت است، بلکه دلیل آن عدم توازن و برابری صفر و یک و در نتیجه تصادفی نبودن کلید تولید شده است.

Serial test یکی دیگر از تست‌های NIST است. این تست به بررسی حضور الگوهای m-بیتی در کل دنباله می‌پردازد. با توجه به یکنواخت بودن دنباله‌های تصادفی، شانس وجود هر الگوی m بیتی باید با شانس بقیه الگوهای m بیتی برابر باشد. واضح است که دلیل تصادفی بودن کلید تولید شده توسط الگوریتم پیشنهادی، تعداد زیاد بیت‌های آن است، چرا که تعداد بیت زیاد موجب بیشتر شدن الگوها و در نتیجه بالارفتن شانس برابری الگوهای m-بیتی با یکدیگر می شود. دلیل رد شدن این تست توسط کار [5] و [6]، پایین بودن تعداد بیت‌های کلید تولید شده است.

تست Frequency Test within a Block رشته بیت ورودی را به بلوک های M-bit تقسیم می‌کند. سپس تصادفی بودن هر بلوک بطور جداگانه مورد بررسی قرار می‌گیرد. در این تست دیگر بالا بودن تعداد بیت مهم نیست بلکه آنچه مهم است توزیع یکسان ۰ و ۱ در بلوک هاست. همانطور که در جدول (۳) دیده می‌شود این تست تصادفی بودن کلیدهای تولید شده توسط روش پیشنهادی و در عین حال تصادفی نبودن کلیدهای تولید شده توسط کار [6] را نشان می‌دهد.

مطابق نتایج جدول (۴)، در پایگاه داده مذکور از ۱۵ تست NIST، بطور میانگین در ۱۴,۵ تست باتوجه به مقدار p-value تصادفی بودن کلیدها مورد تایید قرار گرفته است. همچنین طبق جدول (۴) پس از اعمال تستها روی کار [5] و [6] مشاهده شد که از ۱۵ تست مذکور به ترتیب ۱۱,۵ و ۱۲,۵ تست، تصادفی بودن رشته بیت‌های تولید شده را تایید کردند. درنتیجه نسبت به روش [5] ۲۰ درصد بهبودی داریم. نکته ای که لازم است به آن اشاره شود این است که هرچند مقدار میانگین p-value درکار [5] نسبت به [6] بیشتر است، اما تعداد تست‌های تایید شده که معیار به نسبت مهم‌تری است، کمتر است. پس الزاما بالا بودن مقدار میانگین p-value تعیین‌کننده بهتر بودن یک الگوریتم از نظر تصادفی بودن کلید نهایی نیست، مگر اینکه کل تست‌های NIST تایید شده باشند، یا تعداد تست‌هایی که تصادفی بودن را تایید می‌کنند بیشتر باشند. در آنصورت مانند روش پیشنهادی هرچه مقدار p-value بیشتر شود، درصد تصادفی بودن کلید هم بالاتر می‌رود. درکار [12] نیز تنها ۶ تا از تست‌های NIST روی کلیدها اعمال شده است. علت عدم استفاده از بقیه تست‌های NIST، پایین بودن تعداد بیت کلید می‌باشد. در ستون سوم جدول (۴)، برای تمام تست‌ها، مقادیر p-value محاسبه و میانگین آنها بدست آمد که برای روش پیشنهادی این عدد به ۰,۴۸۱ رسید که با توجه به بیشتر بودن تعداد تست‌های تصادفی نسبت به کارهای [5] و [6]، نتیجه می‌گیریم که رویکرد ما از نظر این معیار بهتر عمل می‌کند. علت مقایسه با روش‌های [5] و [6]، جدیدتر بودن این دو کار و بهتر بودن این روش‌ها در معیارهای دیگر که در بخش‌های بعدی بررسی می‌شوند از جمله نسبت تعداد یک‌های ماتریس کلید به تعداد کل بیت‌ها و همچنین طول بیت بیشتر کلید نهایی در مقایسه با روش‌های قبل‌تر است که خود معیاری از تصادفی بودن بیشتر کلید می‌باشد.

۲-۳-۴- بررسی شمارش ۰ و ۱ها در دنباله کلید

در این تست تعداد بیت های ۱ و ۰ در کل مقادیر ماتریس کلید محاسبه می‌شوند. اگر درصد تعداد بیت‌های ۱ و ۰ به هم نزدیک باشند، یعنی نزدیک ۵۰ درصد برای هر کدام باشد، نشان می‌دهد که توزیع بیت‌های ۱ و ۰ در کل ماتریس بصورت یکنواخت می‌باشد. درواقع، نسبت تعداد یک‌های ماتریس کلید تقسیم بر تعداد کل بیت‌های کلید به عنوان یک معیار مقایسه‌ای در نظر گرفته می‌شود. در حالت ایده‌آل مقدار این معیار باید برابر ۰,۵ باشد که نشان دهنده برابر بودن تعداد صفرها و یک‌های کلید و توزیع یکنواخت آنها می‌باشد. همان طور که در جدول (۵) دیده می‌شود، اعداد مربوط به تعداد بیت‌های ۱ و ۰ در ماتریس‌های کلید تولید شده به روش پیشنهادی نشان دهنده نزدیک بودن تعداد بیت‌های ۱ و ۰ در کلید می‌باشد. بنابراین کلید تولید شده دارای تعداد بیت‌های ۰ و ۱ متعادل است. از طرفی اختلاف نسبت صفرها به کل بیت‌ها با نسبت یک‌ها به کل بیت‌ها تقریبا ثابت بوده و برابر 0.03 می‌باشد. این امر نشان دهنده حساس نبودن روش پیشنهادی به نوع تصویر اثر انگشت می‌باشد. در جدول (۶) نیز نسبت تعداد یک‌های ماتریس کلید بر تعداد کل بیت‌های کلید برای مقایسه بین روش پیشنهادی و چند کار دیگر آورده شده است. مطابق جدول (۶)، میانگین این معیار در روش پیشنهادی در این مقاله برابر عدد ۰,۴۸۱۹ است. همان طور که در جدول (۶) مشاهده می‌شود، نتیجه بدست آمده برای روش پیشنهادی قابل قبول می‌باشد. توجه به این نکته لازم است که هر چند در کارهای [5] و [6]، مقدار این معیار کمی از روش پیشنهادی بهتر است، ولی مطابق نتایج جدول (۴)، نتایج معیار NIST در روش ما هم از لحاظ تعداد تستهای تصادفی تایید شده و هم مقدار P-value مقدار بهتری دارد که نشان دهنده میزان بیشتر تصادفی بودن کلید و در نتیجه امنیت بیشتر آن در روش پیشنهادی می‌باشد.

۳-۳-۴- بررسی تعداد ۱ و ۰های به هم پیوسته در دنباله کلید

بررسی میزان یک‌ها و صفرهای به هم پیوسته در ماتریس کلید نکته بسیار مهمی است. اگر تعداد بیت‌های ۱ یا ۰ به هم پیوسته اعداد بزرگی باشد نشان دهنده عدم توزیع یکنواخت بیت‌های ۱ و صفر در دنباله می‌باشد. ما برای بررسی بیشتر این موضوع به محاسبه تعداد بیت‌های صفر و یك به هم پیوسته با طول ۲ تا ۱۶ می‌پردازیم.

بخشهای (a) و (b) شکل (۱۱) به ترتیب شمارش تعداد بیت‌های صفر و یک به هم پیوسته با طول ۲ تا ۱۶ بیت در دو تصویر انتخابی از پایگاه داده اثر انگشت FVC2002 DB1_B را در روش پیشنهادی نشان می‌دهند. تعداد بیت به هم پیوسته در زیر هر ستون مشخص شده است. در روش پیشنهادی هر چقدر طول بیت های ۰ یا ۱ به هم پیوسته زیاد می‌شود اندازه ستون‌ها در شکل کاهش می‌یابد. در کار [6] تعداد یک‌های به هم پیوسته برای طول‌های ۱۴ بیت به بالا برابر صفر می‌شود، در حالی که در روش پیشنهادی این مقدار برای طول‌های ۱۱ بیت به بالا به صفر می‌رسد. علاوه براین در کار [6] تعداد صفرهای به هم پیوسته برای طول‌های ۱۳ بیت به بالا و در روش پیشنهادی برای طول‌های ۱۲ بیت به بالا صفر می‌شود. با توجه به این تست، کلیدهای تولید شده در کار ما، دارای توزیع یکنواخت بیت های ۰ و ۱ می‌باشند. در نتیجه ویژگی مهم غیر قابل کشف بودن کلید حاصل در روش پیشنهادی بیشتر است.

۴-۳-۴- هیستوگرام مقادیر کلید تصادفی

هیستوگرام نمایشي از توزیع داده‌های کمي پیوسته است که مي‌تواند تخمینی از توزیع احتمال باشد. اگر هیستوگرام صاف و مسطح نباشد می‌توان با حمله آماری به آن مقادیر قابل توجهی از اطلاعات را استخراج کرد، زیرا خاصیت عدم قطعیت در توزیع غیریکنواخت کمتر است. در نتیجه در رمزنگاری به دنبال یک هیستوگرام مسطح و یکنواخت هستیم. داده‌های ماتریس نهایی ۸-بیتی هستند و در محدوده ۰ تا ۲۵۵ قرار می‌گیرند. با توجه به شکل (۱۲) مقدار شمارش این داده‌ها در دو تصویر انتخابی نزدیک به یکدیگر است و در واقع از یک توزیع تقریبا یکنواخت پیروی می‌کنند. به همین دلیل احتمال قرار گرفتن اعداد در بازه‌ی ۰ تا ۲۵۵ در ماتریس نهایی کلید روش پیشنهادی تقریبا برابر می‌باشد و در نتیجه در برابر حمله آماری مقاوم خواهد بود.

۵-۳-۴- شمارش تعداد بیت کلیدهای تولید شده

طول بیت کلیدهای تولیدشده روش پیشنهادی بین ۱۹۰۰۸ و ۲۲۳۱۶۸ بیت متغیر است که مقدار میانگین بین ۸۰ کلید تولید شده ۶۳۷۵۱ بیت است. مطابق نتایج جدول (۷)، روش پیشنهادی بطور قابل ملاحظه‌ای منجر به کلیدهایی با بیشترین طول بیت در مقایسه با روشهای دیگر شده است که در تصادفی‌تر شدن کلید حاصل بسیار موثر می‌باشد. بطور مثال طول بیت متوسط روش پیشنهادی بیش از ۱۰ برابر روش [6] است که خود طول بیت بیشتری از روشهای پیشین مطابق جدول (۷) دارد.

۶-۳-۴- محاسبه فاصله همینگ

یک معیار دیگر برای اطمینان بیشتر از عدم تشابه کلیدهای تولید شده با یکدیگر و در نتیجه تصادفی بودن آنها، فاصله همینگ می‌باشد [29]. در این معیار دو کلید به عنوان ورودی در نظر گرفته می‌شود. سپس تک تک بیت‌های آنها باهم مقایسه می‌گردند. در صورت عدم تشابه، این فاصله افزایش و در صورت تشابه فاصله ثابت می‌ماند. برای دو کلید غیر مشابه هرچه فاصله همینگ بیشتر باشد جواب مطلوب‌تر است. در این بخش ابتدا به صورت تصادفی از پایگاه داده FVC2002 DB3_B برای هر شخص یک کلید استخراج می‌کنیم و به زیر کلیدهایی با طول ۲۵۶ بیت تقسیم می‌نماییم. سپس هر کلید بصورت تصادفی با ۲۰ کلید غیر مشابه مقایسه می‌شود. در نهایت مقدار میانگین ۲۰۰۰ فاصله همینگ را محاسبه می‌نماییم. میانگین فاصله همینگ در کار [7] برای کلیدهای غیر مشابه ۴۹,۹۴% و در کار پیشنهادی برابر ۵۰,۲۱% می‌باشد که این فاصله همینگ بیشتر در روش ما، عدم تشابه کلیدهای تولید شده در روش پیشنهادی و درنتیجه تصادفی بودن بیشتر آنها را تایید می‌کند. البته لازم به ذکر است که تستهای NIST جزو متداول‌ترین و دقیق‌ترین معیارها برای سنجش تصادفی بودن کلیدهاست، و طبق نتایج جدول (۴) هم روش پیشنهادی در این مقاله در این راستا بسیار موفق عمل کرده است.

۷-۳-۴- تعداد کلیدهای ۱۲۸بیتی قابل تولید از دنباله کلید پیشنهادی

به علت بالا بودن تعداد بیت کلید، معمولا در رمزنگاری از زیرکلیدهای ۱۲۸، ۲۵۶ و ۵۱۲ بیتی که از ماتریس کلید پیشنهادی استخراج می‌شوند استفاده می‌گردد. بطور مثال، در بسیاری از الگوریتم‌های رمزنگاری، طول بیت کلید مورد استفاده نهایی ۱۲۸ بیت می‌باشد. باتوجه به این موضوع و در نظرگرفتن میانگین طول بیت بسیار زیاد کلیدهای تولید شده توسط روش پیشنهادی (۶۳۷۵۱ بیت)، می‌توان برای هر شخص کلیدهای تصادفی ۱۲۸ بیتی مختلف از رشته بیت کلید نهایی استخراج کرد. تعداد کلیدهای تصادفی ۱۲۸ بیتی قابل استخراج از این دنباله کلید بزرگ پیشنهادی از فرمول ترکیب $C(63751,128)$ محاسبه می‌شود. طبق جدول (۸)، بطور تقریبی مقدار آن در روش پیشنهادی برابر است با $4{,}61\times10^{471}$ که یک عدد بسیار بزرگ با بیش از ۴۵۰ رقم می‌باشد. همچنین در جدول (۸) بین الگوریتم پیشنهادی و کارهای دیگر مقایسه‌ای انجام داده‌ایم. طبق نتایج جدول (۸)، روش پیشنهادی این مقاله نسبت به بقیه

روش‌های موجود در این زمینه بسیار بهتر عمل می‌کند. بطور مثال، این تعداد کلید ۱۲۸ بیتی قابل استخراج از روش ما حدود $۲,۱\times۱۰^{۲۰۱}$ برابر روش [6] می‌باشد که خود بیشترین تعداد ممکن را در جدول (۸) نسبت به روشهای پیشین دارد. در جدول (۸) سایز ۱۲۸ بیت بطور نمونه انتخاب شده است زیرا سایز کلید نهایی در برخی روشهای پیشین ۱۲۸ بوده است. اما با توجه به سایز بسیار زیاد کلید پیشنهادی نهایی (۶۳۷۵۱ بیت بطور متوسط)، روش ما در زیرکلیدهای بزرگتر همچون ۵۱۲ بیت نیز بسیار موفق خواهد بود و تعداد حالات ۵۱۲ بیتی قابل استخراج از آن نیز بسیار زیاد می‌باشد. درنتیجه با روش پیشنهادی، کلید مورد استفاده در رمزنگاری برای هر فرد نیز تصادفی خواهد بود که یکی دیگر از مزایای مهم رویکرد پیشنهادی می‌باشد. به عبارت دیگر، به دلیل ثابت بودن اثر انگشت هر فرد و درنتیجه ویژگیهای استخراج شده یعنی زاویه و فاصله نقاط مینوشیا با مراکز قطعه‌های هر تصویر، کل کلید تولید شده در الگوریتم پیشنهادی که بطور متوسط ۶۳۷۵۱ بیت می‌باشد ثابت خواهد بود. اما نکته مهم این است که به علت تعداد بسیار زیاد زیرکلیدهای متفاوت ۱۲۸، ۲۵۶ یا ۵۱۲ بیتی که می‌توان بصورت تصادفی از این کلید بزرگ در هر بار الگوریتم رمزنگاری نهایی استخراج کرد، کلید نهایی حتی برای هر فرد نیز تصادفی خواهد بود و درنتیجه قابل کشف نمی‌باشد.

۸-۳-۴- بررسی زمانی الگوریتم پیشنهادی

جدول (۹) میانگین زمان لازم جهت انجام تک تک مراحل الگوریتم تولید کلید رمزنگاری برای هر تصویر را با توجه به ۱۰ تصویر انتخابی برای کار پیشنهادی و دو کار [5] و [6] بعنوان نمونه‌ای از جدیدترین کارهای این زمینه نشان می‌دهد. مطابق نتایج نهایی جدول (۹)، میانگین زمان لازم برای استخراج کلید پیشنهادی در هر تصویر برابر ۳۱ میلی‌ثانیه است که ۳۳% نسبت به کار [5] و ۵۰% نسبت به کار [6] کمتر است. با توجه به ساده شدن مراحل جایگشت و جابجایی نسبت به کارهای [5] و [6]، زمان لازم جهت درهم‌ریختگی، پیچیده شدن و تصادفی‌تر کردن کلید تولید شده پیشنهادی کاهش یافته است. درنتیجه حجم محاسباتی الگوریتم این مقاله کمتر از روشهای پیشین می‌باشد که یکی دیگر از مزایای رویکرد پیشنهادی ما می‌باشد.

۹-۳-۴- مقایسه کلی الگوریتم پیشنهادی

مطابق نتایج جدول (۱۰)، الگوریتم پیشنهادی از نظر چند معیار مهم یعنی معیارهای مختلف تصادفی بودن تست‌های NIST، طول کلید تولید شده نهایی، زمان متوسط لازم برای استخراج کلید و تعداد زیرکلیدهای ۱۲۸ بیتی که می‌توان بطور متوسط از کلید نهایی برای رمزنگاری بعدی استفاده کرد از جدیدترین کارهای پیشین همچون [5] و [6] بهتر عمل می‌کند. بطور مثال از نظر تعداد تستهای تصادفی NIST، با توجه به ۱۵ تست، نسبت به روش [5] ۲۰ درصد و نسبت به [6] ۱۳,۳۴%بهبودی داریم. همچنین طول بیت متوسط روش پیشنهادی بیش از ۱۰ برابر روش [6] است که خود طول بیت بیشتری از روشهای پیشین مطابق جدول (۷) دارد.

۵- نتیجه‌گیری

هدف این مقاله تولید کلید تصادفی برای رمزنگاری تصاویر دیجیتال اثر انگشت با دقت و امنیت بالا بوده است، به گونه‌ای که کلید تولید شده طول بیت زیاد داشته باشد تا امنیت بالایی در سیستم رمزنگاری حاصل کند. همچنین پیچیدگی الگوریتم تولید کلید شامل مراحل متعدد نیز به این مهم کمک کرده است. در روش پیشنهادی در این مقاله با تولید کلیدهایی با میانگین طول ۶۳۷۵۱ بیت به این نیازها به خوبی پاسخ داده شده است. در این کار، پس از استخراج ویژگی‌های بیومتریک و منحصر به فرد از اثر انگشت که شامل نقاط مینوشیا هستند، برای افزایش پیچیدگی الگوریتم و بالابردن امنیت کلید تولید شده، هر تصویر اثر انگشت به ۴۴ قطعه مساوی تقسیم شد و مختصات پیکسل‌های مرکزی هر قطعه در ماتریسی ذخیره شدند. این قطعه‌بندی تصویر در عین سادگی الگوریتم، به دلیل افزایش زیاد بیت‌های تولیدی نهایی کلید یکی از پیشنهادات بسیار موثر پژوهش حاضر است که در کارهای پیشین تا آنجا که اطلاع داریم انجام نشده است. در این مقاله به منظور افزایش داده‌های تصادفی، برخلاف روش‌های پیشین که فاصله یا زاویه بین مینوشیاها را پیدا می‌کردند و درنتیجه به دلیل ثابت بودن مینوشیاها به محدودیت می‌رسیدند، فاصله اقلیدوسی و زاویه بین پیکسل های مرکزی هر ۴۴ قطعه با کل مینوشیاهای تصویر را محاسبه کردیم. آنگاه جهت افزایش بیشتر تصادفی کردن کلید، یک الگوریتم شامل ۳ گام قرار دادن اعداد مربوط به زاویه و فاصله بصورت زوج و فرد در کنار یکدیگر، دو شکل جابجایی بیت‌ها و در نهایت اعمال توزیع یکنواخت روی داده‌های نهایی را برای تولید کلید نهایی پیشنهاد کردیم. این گام آخر پیشنهادی یعنی اعمال توزیع یکنواخت روی داده‌ها نیز به دلیل سادگی زیاد در مقایسه با بسیاری از روش‌های اعمال شده در کارهای پیشین، درعین پیچیده کردن کلید نهایی به سرعت بیشتر الگوریتم تولید کلید کمک کرده است. هریک از گام‌های جابه‌جایی و جایگشت مذکور مراحل ساده‌ای هستند که در کنار یکدیگر باعث پیچیدگی الگوریتم تولید کلید و بالارفتن امنیت شدند. روش پیشنهادی توسط مجموعه تست‌های مختلف از جمله تستهای استاندارد NIST مورد آزمون قرار گرفت و تصادفی بودن کلیدهای نهایی تایید شد. به علت بالا بودن تعداد بیت کلید می‌توان زیر کلیدهای ۱۲۸، ۲۵۶ و ۵۱۲ بیتی از ماتریس کلید مذکور استخراج کرد و در رمزنگاری از آنها استفاده نمود. درنتیجه با روش پیشنهادی، کلید مورد استفاده در

رمزنگاری برای هر فرد نیز تصادفی می‌باشد که یکی دیگر از مزایای مهم رویکرد پیشنهادی است.

۶- مراجع

[1] Wu, W., Zhang, L., "LBlock: A lightweight block cipher," Int. Conf. on Applied Cryptography and Network Security, Vol 6715. pp. 327-344, 2011.

[2] Sadkhan, S. B., Al-Shukur, B. K., Mattar, A. K., "Survey of biometrie based key generation to enhance security of cryptosystems," Al-Sadeq Int. Conf. on Multidisciplinary in IT and Communication Science and Applications (AIC-MITCSA), 2016, pp. 1-6.

[3] Iswari, N. M. S., "Key generation algorithm design combination of RSA and ElGamal algorithm," 8th Int. Conf. on Information Technology and Electrical Engineering (ICITEE), 2016, pp. 1-5,

[4] Bennett, C., Brassard, G., "Quantum cryptography: public key distribution and coin tossing," IEEE Int. Conf. on Computers, 1984, pp. 175-179.

[۵]رشیدی، بهرام. "طراحی روشی برای تولید کلید رمزنگاری تصادفی بیومتریک بر اساس اثر انگشت"، مجله *ماشین بینایی و پردازش تصویر*، دوره ۹، شماره ۴، صفحات ۱ تا ۱۴، ۱۴۰۱.

[۶]رشیدی، بهرام، گودرزی، علی. "تولید کلید رمزنگاری تصادفی بیومتریک براساس ویژگی‌های منحصربه‌فرد اثر انگشت"، هوش محاسباتی در مهندسی برق، ۱۴۰۱.

[7] Dwivedi, R., Dey, S., Sharma, M. A., Geol, A., "A fingerprint based crypto-biometric system for secure communication," Journal of Ambient Intelligence and Humanized Computing, Vol. 11, pp. 1495–1509, 2020.

[8] Suresh, K., Rajarshi, P., Balasundaram, S. R., "Fingerprint based cryptographic key generation," Int. Conf. on Intelligent data communication technologies and internet of things, 2019, pp. 704-713.

[9] Rukhin, A., Soto, J., Nechvatal, J., Smid, M., et. al., "A statistical test suite for random and pseudorandom number generators for cryptographic applications," National Institute of Standards and Technology (NIST), 2010.

[10] Shankar, K., Elhoseny, M., Chelvi, E. D., Lakshmanaprabu, S. K., et. al,. "An efficient optimal key based chaos function for medical image security," IEEE Access, vol. 6, pp. 77145-77154, 2018.

[11] Ogras, H., Turk, M., "A secure chaos-based image cryptosystem with an improved sine key generator," American Journal of Signal Processing, Vol. 6, No. 3, pp. 67-76, 2016.

[12] Yuliana, M., Wirawan, Suwadi, "A simple secret key generation by using a combination of pre-processing method with a multilevel quantization," Entropy, Vol. 21, pp. 1-25. 2019.

[13] Tuncer, T., Avaroglu, E., "Random number generation with LFSR based stream cipher algorithms," 40th Int. Convention on Information and Communication Technology, Electronics and Microelectronics (MIPRO), 2017, pp. 171-175.

[14] Kanade, S., Petrovska-Delacretaz, D., Dorizzi, B., "Generating and sharing biometrics based session keys for secure cryptographic applications, 4th IEEE Int. Conf. on Biometrics: Theory, Applications and Systems, 2010, pp. 1-7.

[15] Majjed, I. A., Majeed, A. M., "Key generation based on facial biometrics," The 1st Int. Multi-Disciplinary Conf. Theme: Sustainable Development and Smart Planning, 2020, pp. 1-9.

[16] Anees, A., Chen, Y.P., "Discriminative binary feature learning and quantization in biometric key generation", Pattern Recognition, Vol. 77, pp. 289-305, 2018.

[17] Srinivas J., Mishra D., Mukhopadhyay S., Kumari S., "Provably secure biometric based authentication and key agreement protocol for wireless sensor networks," Journal of Ambient Intelligence and Humanized Computing, Vol. 9, No. 4, pp. 875-895. 2018.

[18] Jo, J. G., Seo, J. W., Lee, H. W., "Biometric digital signature key generation and cryptography communication based on fingerprint," Int. Workshop on Frontiers in Algorithmics, 2007, pp. 38-49.

[19] Barman, S., Samanta, D., Chattopadhyay, S., "Fingerprint-based crypto-biometric system for network security," EURASIP Journal on Information Security, Vol. 3, pp. 1-17. 2015.

[20] Li, S., Zhang, X., Qian, Z., Feng, G., Ren, Y., "Key based artificial fingerprint generation for privacy protection," IEEE Transactions on Dependable and Secure Computing, Vol. 17, Issue 4, pp. 828- 840. 2020.

[21] Wang, P., You, L., Hu, G., Hu, L., et. al., "Biometric key generation based on generated intervals and two-layer error correcting technique," Pattern Recognition, Vol. 111, pp. 1-36. 2020.

[22] Sarkar, A., Singh, B. K., "Cryptographic key generation from cancelable fingerprint templates," Int. Conf. on Recent Advances in Information Technology, 2018, pp. 1-6.

[23] You, L., Zhang, G., Zhang, F., "A fingerprint and threshold scheme-based key generation method," Int. Conf. on Computer Sciences and Convergence Information Technology, 2010, pp. 615- 619.

[24] Abdullah, A. A., Khalaf, R. Z., Habib, H. B., "Modified BB84 quantum key distribution protocol using Legendre symbol," 2nd Scientific Conf. of Computer Sciences, 2019.

[25] Artiles, J. A. P., Chaves, D. P. B., Pimentel, C., "Image encryption using block cipher and chaotic sequences,"

Signal Processing: Image Communication, Volume 79, pp. 24-31, 2019.

[26] Rajesh, S., Paul, V., Menon, V. G., Khosravi, M. R., "A secure and efficient lightweight symmetric encryption scheme for transfer of text files between embedded IoT devices," Symmetry, Vol. 11, Issue 2, 2019.

[27] Abraham, J., Kwan, P., Gao, J., "Fingerprint matching using a hybrid shape and orientation descriptor, "State of the Art in Biometrics," Intech Publisher, pp. 25-56, 2011.

[28] The Fingerprint Verification Competition. The Biometric System Laboratory, University of Bologna, Bologna, Italy. Accessed: Jan. 2016.

[29] Bookstein, A., Kulyukin, V. A., Raita, T., "Generalized hamming distance," Information Retrieval, Vol. 5, pp. 353–375, 2002.

محمدرضا روزبهانی مدرک کارشناسی خود را در مهندسی برق از دانشگاه آیت الله العظمی بروجردی (ره) در سال ۱۳۹۸دریافت کرد. ایشان هم‌اکنون دانشجوی مهندسی برق، الکترونیک-دیجیتال دانشگاه صنعتی امیرکبیر است. زمینه‌های تحقیقاتی ایشان شامل پردازش‌تصویر، یادگیری‌ماشین و رمزنگاری است.

ساناز سیدین مدرک کارشناسی و کارشناسی ارشد خود را در مهندسی برق، الکترونیک به ترتیب از دانشگاه‌های صنعتی امیرکبیر و علم و صنعت ایران در سالهای ۱۳۸۰ و ۱۳۸۴ دریافت کرد. او مدرک دکترای خود در مهندسی برق، الکترونیک را در سال ۱۳۸۹ در زمینه تخصصی پردازش مقاوم گفتار از دانشگاه صنعتی امیرکبیر اخذ نمود. ایشان هم‌اکنون استادیار دانشکده مهندسی برق دانشگاه صنعتی امیرکبیر و همچنین عضو ارشد انجمن IEEE می‌باشند. زمینه‌های تحقیقاتی ایشان شامل یادگیری ماشین و هوش‌مصنوعی، پردازش‌سیگنال (صوت، تصویر، سیگنال‌های بیولوژیکی)، حسگری فشرده و کدینگ تنک و جداسازی منابع است.

بهرام رشیدی مدرک کارشناسی مهندسی برق خود را در سال ۱۳۸۸ از دانشگاه لرستان و مدرک کارشناسی ارشد و دکتری خود را به ترتیب در سال های ۱۳۹۰ و ۱۳۹۵ از دانشگاه تبریز و دانشگاه صنعتی اصفهان اخذ کرد است. ایشان در حال حاضر استادیار گروه مهندسی برق دانشگاه آیت الله بروجردی (ره) است. علایق تحقیقاتی او شامل پیاده سازی سخت افزار برای محاسبات میدان های محدود، پردازش تصویر، سخت افزار رمزنگاری، اینترنت اشیاء (IoT)، رمزهای بلوکی و مدارهای VLSI برای سیستم‌های رمزنگاری منحنی بیضی است.