PENYISIPAN PESAN PADA GAMBAR MENGGUNAKAN ALGORITMA ARNOLD CAT MAP (ACM), LEAST SIGNIFICANT BIT(LSB), DAN SCALE INVARIANT FEATURE TRANSFORM (SIFT)

Cahya Nurani Indah, *Fakultas Ilmu Komputer, Universitas Dian Nuswantoro*

Abstract — Communication has becomes a crucial things in human's life. One of form is an image which presented visually. In cyber space , communication needs security so it can be spared from any kind of data crimes like data theft, data destruction, data falsification, and data interception. Arnold Cat Map (ACM) algorithm is one of cryptography algorithms which used for image encryption by rotating the image continuously until the form becomes unregularly. But ,if the number of iteration reaches a certain value, the image will be back to its normal form. Least Significant Bit (LSB) algorithm will change every LSB value of cover's pixel to binary number of the message sequentially. On this research, LSB algorithm will be used to cover ACM's weakness. But LSB also has its own weakness because the message insertion position is sequential so it can be easily known by other people. Researcher will combine LSB with Scale Invariant Feature Transform SIFT) algorithm to get random message insertion position. Testing with salt & pepper and scratch will be performed to know about the message's endurance. From the result of tests performed, the method proposed by the researcher can be used to conceal or to take message. The produced stego-image quality is quite high as well.

***Keywords* — Cryptography, Steganography, Arnold Cat Map, Least Significant Bit, Scale Invariant Feature Transform, Salt & Pepper, Scratch.**

I. PENDAHULUAN

Komunikasi sudah menjadi bagian dalam kehidupan manusia. Terutama pada era informasi seperti sekarang, komunikasi menjadi hal yang sangat krusial. Ada saat di mana informasi itu bersifat penting dan rahasia. Oleh karena itu metode komunikasi yang digunakan harus dibuat sedemikian rupa sehingga tidak ada pihak lain yang mengetahui tentang informasi tersebut[1]. Citra (*image*) merupakan salah satu bentuk data atau informasi yang disajikan secara visual. Citra memainkan peranan penting dalam industri multimedia saat ini. Citra juga merupakan unsur pembentuk video, sebab sebuah video pada dasarnya disusun oleh rangkaian frame citra yang ditampilkan dalam tempo yang cepat[2].

Dalam dunia sekarang ini keamanan teknologi menjadi pusat perhatian. Dengan meningkatnya kejahatan di dunia maya (*cyber crime*), menyediakan keamanan jaringan saja tidak cukup. Keamanan yang disediakan gambar seperti *blue print* dari perusahaan, gambar rahasia yang digunakan dalam militer atau kepentingan perusahaan [3]. Contoh kejahatan di dunia maya atau *cyber crime* yaitu mengambil atau memodifikasi informasi yang bukan haknya [4].

Karena hal tersebut munculah teknik pengamanan pesan yaitu kriptografi dan steganografi. Kriptografi diciptakan sebagai suatu teknik untuk mengamankan kerahasiaan komunikasi. Berbagai metode telah dikembangkan untuk mengenkripsi dan mendekripsi data untuk menjaga kerahasiaan pesan. Teknik kriptografi saja tidak cukup untuk menjaga kerahasiaan pesan, sehingga diperlukan teknik steganografi untuk menyembunyikan pesan dengan suatu cara sehingga tidak ada seorangpun yang mengetahui atau menyadari bahwa ada suatu pesan rahasia yang disisipkan [2].

Algoritma A*rnold Cat Map* (ACM) merupakan salah satu algoritma kriptografi yang biasa digunakan untuk melakukan enkripsi pada citra. Konsep dari algoritma ini adalah memutar citra secara terus menerus sehingga menjadi bentuk yang tidak terlihat. Namun, bila iterasi putaran citra tersebut telah mencapai jumlah tertentu, citra tersebut dapat kembali seperti semula lagi[5]. Hal ini menjadi kelemahan bagi algoritma ACM. Beberapa peneliti mencoba untuk

Footnote (boleh dikosongkan)

menggabungkan algoritma ACM dengan algoritma kriptografi lainnya. Seperti pada penelitian berjudul "*An Efficient Image Cryptographic Technique by Applying Chaotic Logistic Map and Arnold Cat Map*"[6], peneliti menerapkan algoritma *Logistic Map* dan ACM untuk melakukan enkripsi pada citra. Disebutkan bahwa hasil dari metode tersebut cukup baik dan waktu yang dibutuhkan untuk melakukan enkripsi cukup singkat sehingga dapat pula digunakan dalam komunikasi melalui video, aplikasi komersil, dan sebagainya. Selain menggabungkan algoritma ACM dengan algoritma kriptografi lainnya, algoritma ACM juga dapat digabungkan dengan teknik steganografi. Pada jurnal yang berjudul "*High Security Image Steganography with Modified Arnold's Cat Map*"[7], peneliti mencoba untuk memodifikasi algoritma ACM dan kemudian digabungkan dengan teknik steganografi. Hasil akhir dari penelitian tersebut tidak aman terhadap serangan *noise* dan teknik kompresi sehingga diperlukan pengembangan lebih lanjut.

Dalam steganografi banyak metode yang dikembangakan, salah satu metode yang umum digunakan adalah metode *Least Significant Bit* (LSB). LSB merupakan bit terendah pada urutan nomor biner. Tujuan dari metode LSB yaitu untuk menempatkan atau menyisipkan data ke bit terakhir dari setiap piksel pada *cover image.* Penelitian pada jurnal yang berjudul "*Improved RGB-LSB Steganography Using Secret Key*"[8] memodifikasi algoritma LSB dengan menggunakan *key* untuk membantu dalam memilih *channel* warna yang akan disisipkan pesan. Menurut peneliti, metode yang diusulkan tersebut akan susah dipecahkan oleh orang lain bila orang tersebut tidak mengetahui *key* yang digunakan.

Untuk menutupi kelemahan algortima LSB , maka pada penelitian ini dilakukan penggabungan dengan algoritma Scale *Invariant Feature Transform (SIFT)*. *SIFT* merupakan algoritma yang dapat diaplikasikan pada *image matching* yang memiliki ketahanan terhadap citra yang mengalami perubahan transformasi seperti rotasi , ditemukan oleh David G. Lowe pada tahun 1999, seorang peneliti dari University of British Columbia. Dalam penelitian ini algoritma SIFT suatu citra akan di ubah menjadi vektor fitur lokal yang kemudian akan digunakan sebagai pendekatan dalam mendeteksi objek yang dimaksud. Secara garis besar, algoritma yang digunakan pada metode *SIFT* terdiri dari empat tahap, yaitu mencari nilai ekstrim pada skala ruang, menentukan kandidat *keypoint* , penentuan orientasi, dan deskriptor *keypoint*[9].

Scale invariant feature transform merupakan salah satu algoritma yang bekerja cukup baik dalam mendeteksi ciri pada suatu citra, output dari algoritma ini berupa titik titik kunci yang berada di sekitar pola dari citra yang biasa disebut dengan *keypoint descriptor*, yang mana nantinya *keypoint descriptor* dari sebuah citra dapat dibandingkan dengan *keypoint descriptor* pada citra lain yang selanjutnya dapat ditentukan tingkat kemiripannya [10]. Titik-titik kunci inilah yang akan digunakan untuk menyisipkan pesan yang telah dienkripsi menggunakan algoritma ACM.

Berdasarkan penelitian dari jurnal[7] algoritma ACM dapat digabungkan dengan teknik steganografi. Namun karena tidak ada pengembangan terhadap teknik steganografi maka keamanan pesan pada steganografi tidak terlalu ada peningkatan. Sementara, pada jurnal [11] menyatakan bahwa teknik kriptografi dapat digabungkan dengan teknik steganografi dengan modifikasi menggunakan algoritma SIFT untuk dijadikan *key* dan hasilnya sangat aman untuk transaksi data dalam waktu dekat. Oleh karena itu, peneliti memiliki ide untuk menggabungkan teknik kriptografi, yaitu dengan algoritma ACM, dan teknik steganografi, yaitu dengan algoritma LSB. Algoritma LSB akan dimodifikasi dengan algoritma *Scale Invariant Feature Transform* (SIFT). Penggunaan algoritma ACM, LSB, dan SIFT diharapkan dapat meningkatkan keamanan pesan dan mengatasi kelemahan dari algoritma ACM dan LSB.

II. METODE YANG DIUSULKAN

A. *Arnold Cat Map*

Algoritma ACM pertama diciptakan oleh Vladimir Arnold pada tahun 1960. Algoritma ACM biasa digunakan untuk melakukan enkripsi pada file citra karena konsep dari algoritma ini adalah memutar citra secara terus menerus sehingga menjadi bentuk yang tidak beraturan. Namun, bila iterasi putaran citra tersebut telah mencapai jumlah tertentu, citra tersebut dapat kembali seperti semula lagi. Rumus dari algoritma ACM adalah:

$$\begin{bmatrix} x' \\ y' \end{bmatrix} \rightarrow \begin{bmatrix} 1 & p \\ q & pq+1 \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} \bmod n \qquad (1)$$

Dimana x dan y merupakan posisi piksel citra, misal posisi piksel (1,2) berarti $x = 1$ dan $y = 2$. Sedangkan $x`$ dan $y`$ adalah posisi piksel yang baru. Pada implementasinya, nilai $x`$ dan $y`$ ditambah dengan 1 untuk menghindari nilai 0 karena pada matriks, posisi awal / terkecil adalah (1,1). Nilai p dan q ditentukan sendiri dengan syarat yaitu angka yang dipilih haruslah bilangan bulat positif dan $\det\begin{bmatrix} 1 & p \\ q & pq+1 \end{bmatrix} = 1$. Nilai dari n adalah ordo citra. Bila ukuran citra adalah 124 x 124, maka n = 124. Selain inputan rumus diatas, diperlukan satu inputan lagi yaitu jumlah iterasi perputaran citra yang diinginkan.

B. *Least Significant Bit*

LSB adalah bit terkecil pada urutan bilangan biner [6]. Sedangkan bit terbesarnya disebut most significant bit (MSB). Bila terdapat bilangan biner 10000000, maka bit LSBnya adalah angka 0 yang terletak di sebelah kanan dan bit MSBnya adalah angka 1 yang terletak di sebelah kiri.

Algoritma LSB merupakan algoritma steganografi yang paling sederhana. Algoritma ini paling sering digunakan untuk steganografi citra, walaupun dapat pula dipakai untuk tipe steganografi lainnya. Seperti namanya, algoritma LSB memanfaatkan bit LSB sebagai tempat untuk menyisipkan pesan dengan cara mengubah nilai bit LSB pada cover menjadi nilai biner pesan secara berurutan.

C. *Scale Invariant Feature Transform*

Scale Invariant Feature Transform atau SIFT merupakan metode yang digunakan pada *computer vision* untuk mendeteksi dan mendeskripsikan fitur lokal dari sebuah gambar. Metode ini diperkenalkan oleh David Lowe pada tahun 1999. Metode SIFT ini banyak digunakan pada pengenalan objek, *mapping* dan navigasi robotik, pemodelan 3D, pengenalan gestur, *video tracking*, dan masih banyak lagi.
Metode SIFT ini banyak digunakan karena invariant terhadap skala dan rotasi gambar, dan juga terhadap perubahan iluminasi/pencahayaan.Algoritma SIFT terdiri dari empat tahap, yaitu :

1. ***Scale Space Extrema Detection***

   Pada tahapan ini, keypoint pada gambar akan dideteksi secara berurut dari atas ke bawah menggunakan pendekatan *filtering* yang menggunakan algoritma yang efisien untuk mendeteksi kandidat-kandidat lokasi dan skala yang akan diulang beberapa kali dengan *view* yang berbeda-beda. Untuk mendeteksi lokasi yang invariant terhadap perubahan skala dari gambar bisa dilakukan dengan mencari fitur yang stabil pada seluruh kemungkinan perubahan skala gambar, menggunakan fungsi kontinu dari skala yang dikenal dengan scale space yang menggunakan kernel fungsi Gaussian.

   Untuk mendeteksi lokasi *keypoint* yang stabil pada *scale space*, David Lowe menggunakan *scale-space extrema* dari perubahan fungsi gausian yang dikonvolusikan dengan gambar, yang mana bias dihitung dari perbedaan dua skala yang berdekatan dibagi dengan faktor pengali yang konstan.

$$D(x, y, \sigma) = (G(x, y, k\sigma) - G(x, y, \sigma)) * I(x, y) = L(x, y, k\sigma) - L(x, y, \sigma) \quad (2)$$

2. **Keypoint Localization**

   Setelah kandidat keypoint ditemukan, langkah selanjutnya adalah melakukan deteksi detail terhadap kandidat keypoint tersebut, untuk lokasi, skala, dan rasio *principal curvature*. Info detail ini memungkinkan untuk suatu titik di hapuskan dari kandidat keypoint apabila memiliki kontras yang rendah (dan sensitif terhadap noise) atau berada di sebuah tepi.

3. **Orientation Assignment**

   Orientasi ditetapkan pada setiap *keypoint* yang terpilih, untuk menghasilkan titik yang *invariant* terhadap rotasi. Orientasi ditentukan berdasarkan arah gradien gambar.

4. **Keypoint Descriptor**

   Keypoint descriptor pertama-tama akan menghitung arah gradien dan orientasi pada setiap titik sample dalam area di sekitar lokasi keypoint. Sample-sample tersebut kemudian akan diakumulasikan kedalam orientasi histrogram yang membagi ke dalam 4x4 *subregion*, dengan panjang setiap panah merupakan penjumlahan dari perubahan gradien di area tersebut. 4x4 array ini mengandung masing-masing 8 orientasi, sehingga fitur vektor yang dihasilkan untuk satu keypoint sebanyak 4x4x8=128 elemen vektor.

D. *Mean Square Error (MSE) dan Peak Signal to Noise Ratio (PSNR)*

Penilaian kualitas citra dilakukan dengan menggunakan besaran Mean Square Error (MSE) dan Peak Signal to Noise Ratio (PSNR) [9]. MSE adalah rata-rata kuadrat nilai kesalahan antara citra asli dengan citra hasil pengolahan. Semakin rendah nilai MSE maka akan semakin baik [8]. Rumus untuk menghitung MSE adalah:

$$MSE = \frac{1}{MN}\sum_{y=1}^{M}\sum_{x=1}^{N}[I(x,y) - I'(x,y)]^2 \quad (3)$$

Dimana MSE = nilai Mean Square Error citra hasil pengolahan, M = panjang citra hasil pengolahan (dalam piksel), N = lebar citra hasil pengolahan (dalam piksel), $I(x,y)$ = nilai piksel dari citra asli, dan $I'(x,y)$ = nilai piksel pada citra hasil pengolahan.

Peak Signal to Noise Ratio (PSNR) adalah perbandingan antara nilai maksimum dari sinyal yang diukur dengan besarnya derau yang berpengaruh pada sinyal tersebut. PSNR diukur dalam satuan desibel (dB). Untuk melakukan penghitungan nilai PSNR, terlebih dahulu harus dicari nilai MSE-nya. Semakin besar nilai PSNR maka semakin baik kualitas citra. Rumus untuk menghitung PSNR adalah:

$$PSNR = 10.\log_{10}\left(\frac{MAXi^2}{MSE}\right) \quad (4)$$

Dimana PSNR = nilai PSNR citra, MAX_i = nilai maksimum piksel, dan MSE = nilai MSE citra.

E. *Teknik Penyerangan pada Citra (Attack)*

Attack merupakan teknik untuk merusak maupun memodifikasi sebuah citra yang bertujuan untuk merusak maupun menghilangkan watermark dari sebuah citra. Teknik ini biasa dilakukan untuk mengetahui kualitas sebuah citra apabila dilakukan sebuah modifikasi pada citra tersebut. Dalam penelitian ini, digunakan dua teknik attack, yaitu salt & pepper dan scratch.

a. Salt & pepper

   Salt & pepper merupakan salah satu jenis noise. Bentukdari metode attack ini seperti butir-butir garam dan merica, sehingga dinamakan salt & pepper.

b. Scratch

Scratch merupakan salah satu metode attack pada citra digital. Metode ini dilakukan untuk merusak citra dengan cara membuat garis-garis pada citra.

III. IMPLEMENTASI

A. *Implementasi Proses Merahasiakan Pesan*

Pada proses untuk merahasiakan pesan, input yang akan digunakan adalah pesan, cover, dan kunci rahasia. Pesan dan cover yang digunakan berupa file citra dengan kriteria file citra 12-bit dengan tipe warna grayscale, format *.bmp, dan ukuran citra yang sama baik lebar maupun panjangnya (berbentuk persegi). Ada beberapa kunci rahasia yang digunakan yang disimbolkan sebagai p, q, r,dan n. Kunci p, q, r, dan n digunakan untuk algoritma ACM.

Kunci p dan q digunakan untuk melakukan penghitungan pada rumus ACM, kunci r merupakan jumlah iterasi perputaran citra dengan ACM, dan kunci n merupakan ordo citra pesan. Dan input citra pembanding pada proses SIFT.

Ada tiga tahap dalam proses untuk merahasiakan pesan ini, yaitu tahap enkripsi pesan, tahap pembuatan map penyisipan, dan tahap penyisipan pesan. Pesan akan dienkripsi terlebih dahulu dengan menggunakan algoritma ACM. Sebelum disisipkan ke dalam cover, akan dibuat map penyisipan terlebih dahulu dengan menggunakan algoritma SIFT. Kemudian pesan akan disisipkan ke dalam cover menggunakan algoritma LSB dengan posisi penyisipan sesuai dengan map penyisipan yang telah dibuat.

B. *Implementasi Proses Pengambilan Pesan*

Input yang dibutuhkan pada proses pengambilan pesan adalah file stego-image dan kunci rahasia. Kunci rahasia yang digunakan disimbolkan sebagai p, q, r, dan n. Kunci p, q, dan r digunakan untuk penghitungan pada rumus ACM, dan kunci n adalah ordo citra pesan. Selain digunakan untuk penghitungan ACM, kunci n juga akan digunakan dalam ekstraksi pesan.

Ada tiga tahap pada proses pengambilan pesan, yaitu tahap pembuatan map ekstraksi, tahap ekstraksi pesan, dan tahap dekripsi pesan. Sebelum melakukan ekstraksi pesan, akan dibuat map ekstraksi terlebih dahulu menggunakan algoritma SIFT. Setelah itu dilakukan ekstraksi pesan dari dalam stego-image menggunakan algoritma LSB dengan posisi pengambilan sesuai dengan map ekstraksi. Karena pesan yang diekstrak masih berupa ciphertext, sehingga harus didekripsi menggunakan algoritma ACM agar didapatkan pesan yang sesungguhnya.

C. *Implementasi Pengujian Metode*

Ada dua jenis pengujian, yaitu evaluasi non-attack dan evaluasi attack. Evaluasi non-attack akan menghitung nilai MSE dan PSNR dari stego-image yang belum diserang. Sementara evaluasi attack akan melakukan penyerangan terlebih dahulu pada stego-image dengan teknik Salt & Pepper ataupun Scratch. Berdasarkan gambar di atas, stego-imagea merupakan stego-image yang diserang dengan teknik Salt & Pepper. Sedangkan stego-imageb merupakan stego-image yang diserang dengan teknik Scratch. Kemudian stego-imagea dan stego-imageb akan dihitung nilai MSE dan PSNR-nya masing-masing sehingga dapat dibandingkan dengan nilai MSE dan PSNR dari stego-image yang belum diserang. Semakin rendah nilai MSE-nya, semakin baik kualitas stego-image. Semakin tinggi nilai PSNR-nya, semakin baik kualitas stego-image. Selain menggunakan nilai MSE dan PSNR untuk menguji metode, akan dilakukan pula pengambilan pesan dari stego-imagea dan stego-imageb agar dapat dilihat dampak penyerangan dengan teknik Salt & Pepper ataupun Scratch terhadap pesan.

Dengan demikian, berdasarkan hasil pengujian nantinya akan dapat disimpulkan apakah metode yang diusulkan oleh penulis dapat merahasiakan pesan dengan baik atau tidak.

IV. HASIL & PEMBAHASAN

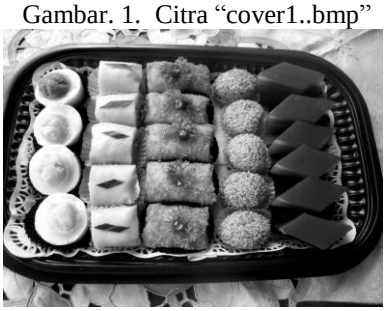
A. *Kebutuhan Data Citra*

Data citra yang digunakan adalah 8 buah citra 8-bit dengan tipe warna grayscale, format *.bmp, dan ukuran citra yang sama baik lebar maupun panjangnya (berbentuk persegi) yang akan digunakan sebagai pesan dan cover. Satu buah citra cover dan satu buah citra pesan akan digunakan sebagai contoh untuk memberikan gambaran dari metode yang diusulkan. Sementara sisanya digunakan sebagai objek eksperimen dalam menerapkan metode yang diusulkan.

Dalam pemilihannya, jumlah piksel pada citra cover haruslah lebih banyak daripada jumlah bit biner pada pesan sehingga semua bit biner pesan dapat disisipkan ke dalam citra cover.

Citra cover yang akan digunakan untuk eksperimen adalah:

Gambar. 1. Citra “cover1..bmp”



Gambar. 2. Citra “cover2.bmp”

Gambar. 3. Citra “cover3.bmp”

Sedangkan citra pembanding yang akan digunakan untuk eksperimen adalah:

Gambar. 4. Citra "key1.bmp"

Gambar 5. Citra "key2.bmp"

Gambar 6. Citra "key3.bmp"

Sedangkan citra pesan yang akan disisipkan adalah sebagai berikut:

Gambar 7. Citra "pesan1.bmp"

Gambar 8. Citra "pesan2.bmp"

Gambar 9. Citra "pesan3.bmp"

B. Hasil Implementasi Motode

Pada implementasinya, citra "mansion124.bmp" akan disisipkan ke dalam citra "lena512.bmp" dan hasilnya adalah citra "stego1.bmp". Citra "fruit100.bmp" akan disisipkan ke dalam citra "boat500.bmp" dan hasilnya adalah citra "stego2.bmp". Sedangkan citra "lake100.bmp" akan disisipkan ke dalam citra "flowers480.bmp" dan hasilnya adalah citra "stego3.bmp". Stego-image yang dihasilkan bentuknya tidak berubah dari cover aslinya, namun terjadi penurunan kualitas citra karena adanya perubahan nilai pada beberapa piksel. Berikut adalah hasil pengukuran MSE dan PSNRnya.

TABLE
I
NILAI MSE DAN PSNR STEGO-IMAGE DAN PESAN YANG DIAMBIL

Citra	MSE (dB)	PSNR (dB)
"stego1.bmp"	0,00099695	78.1441
"pesan1.bmp"	0	Inf
"stego2.bmp"	0.0012	77.4459
"pesan2.bmp"	0	Inf
"stego3.bmp"	0.0018	75.4935
"pesan3.bmp"	0	Inf

C. Hasil Pengujian Attack

Pengujian attack bertujuan untuk mengetahui dampak serangan (attack) terhadap pesan yang terdapat dalam stego- image. Teknik attack yang digunakan adalah salt & pepper dan scratch. Stego-image yang sudah diserang akan dicoba untuk diambil kembali pesan di dalamnya dan dilakukan penghitungan MSE dan PSNR baik pada stego-image maupun pesan yang sudah diambil agar dapat diketahui kualitasnya. Semakin rendah nilai MSE dan semakin tinggi nilai PSNR berarti semakin bagus kualitas citra. Berikut adalah hasil dari pengujian attack.

TABLE
II
NILAI MSE DAN PSNR STEGO-IMAGE DAN PESAN
PADA
PENGUJIAN ATTACK

Citra	MSE (dB)	PSNR (dB)
"stego1-saltpepper.bmp"	217.9390	24.7475
"pesan1-saltpepper.bmp"	11.0500	37.6970
"stego1-scratch.bmp"	16.0605	36.0732
"pesan1-scratch.bmp"	0.1600	56.0896
"stego2-saltpepper.bmp"	229.6697	24.5198
"pesan2-saltpepper.bmp"	114.3333	27.5491
"stego2-scratch.bmp"	38.2513	32.3043
"pesan2-scratch.bmp"	0	Inf
"stego3-saltpepper.bmp"	21.6255	34.7811
"pesan3-saltpepper.bmp"	211.7156	24.8733
"stego3-scratch.bmp"	54.3255	30.7808
"pesan3-scratch.bmp"	0.0178	65.6320

D. Pembahasan Hasil Implementasi

Menurut [18], kualitas citra dianggap rendah bila nilai PSNR-nya kurang dari 30dB dan kualitas citra dianggap tinggi bila nilai PSNR-nya di atas 40dB. Sementara bila nilai PSNR-nya berada di antara 30dB dan 40dB, maka kualitas citra tersebut masih dapat diterima. Berdasarkan hasil pengujian non-attack pada tabel di atas, nilai MSE pada masing-masing stego-image cukup rendah, yaitu di bawah 0dB, dan nilai PSNR-nya cukup tinggi, yaitu sekitar 75dB – 78dB. Nilai MSE pada "pesan1.bmp", "pesan2.bmp", dan "pesan3.bmp" adalah 0dB dan nilai PSNR-nya adalah Inf (tidak terhingga) yang berarti tidak ada perubahan nilai piksel pada citra hasil proses pengambilan pesan dengan citra pesan yang asli sehingga dapat dikatakan bahwa kualitas stego-image yang dihasilkan dengan menggunakan metode yang diusulkan pada penelitian ini cukup tinggi dan metode yang diusulkan dapat mengambil kembali pesan pada stego-image dengan baik.

Sedangkan pada pengujian attack, dapat dilihat bahwa nilai PSNR untuk pengujian salt & pepper pada stego-image adalah sekitar 24-34 dB dan nilai PSNR pada pesan adalah sekitar 27dB – 37dB. Posisi penyebaran titik hitam dan putih yang acak memberi dampak yang cukup besar terhadap kualitas citra pesan dan stego-image karena posisi penyisipan yang juga dilakukan secara acak memungkinkan seringnya titik dari salt & pepper menempati posisi piksel dimana terdapat bit biner pesan di dalamnya. Dengan demikian metode yang diusulkan pada penelitian ini masih rentan terhadap serangan noise seperti salt & pepper karena kualitas stego-image yang diserang dengan salt & pepper dan pesannya rendah.

Sementara pada pengujian scratch, nilai MSE dan PSNR stego-image maupun pesan lebih baik daripada pengujian salt & pepper. Nilai PSNR pada "stego1-scratch", "stego2-scratch", dan "stego3-scratch" adalah sekitar 30dB – 36dB. Sedangkan nilai PSNR pada "pesan1-scratch", "pesan2-scratch", dan "pesan3-scratch" adalah sekitar 0 dB -0,178 dB. Baik kualitas stego-image maupun pesannya tersebut dapat dikategorikan rendah atau masih dapat diterima. Posisi penyisipan yang acak pada metode yang diusulkan di penelitian ini dapat membingungkan si penyerang. Bila si penyerang tidak mengetahui bahwa posisi penyisipan dilakukan secara acak, maka si penyerang hanya akan melakukan scratch pada bagian atas stego-image seperti yang tampak pada citra "stego3-scratch.bmp". Walaupun kualitas citra pesan menurun, namun kerusakan yang ditimbulkan tidak terlalu besar bila dibandingkan dengan penyisipan yang dilakukan secara berurutan. Dari kedua pengujian attack tersebut, dapat disimpulkan bahwa kualitas citra stego-image maupun pesan menurun drastis namun pesan tersebut masih dapat diketahui bentuknya.

V. PENUTUP

Penggabungan teknik kriptografi dan steganografi menggunakan algoritma ACM, LSB, dan SIFT dapat digunakan untuk merahasiakan pesan dengan baik. Orang yang tidak mengetahui kunci rahasia yang digunakan akan kesulitan untuk mendapatkan pesan pada *stego-image.* Kualitas *stego-image* yang dihasilkan cukup tinggi yaitu lebih dari 70 dB, tetapi masih rentan terhadap serangan citra seperti *salt&pepper* dan *scratch* sehingga dalam penelitian selanjutnya perlu pengembangan terhadap algoritma steganografi yang digunakan.

REFERENCES

[1] R. F.-W. Suadi, "IMPLEMENTASI KRIPTOGRAFI DAN STEGANOGRAFI PADA MEDIA GAMBAR DENGAN MENGGUNAKAN METODE DES DAN REGION-EMBED DATA DENSITY.," pp. 1–7, 2011.

[2] V. L. Reddy and K. S. R. M. C. Engg, "Implementation of LSB Steganography and its Evaluation for Various File Formats," vol. 872, pp. 868–872, 2011.

[3] J. Kapur and A. J. Baregar, "Security Using Image Processing," *Int. J. Manag. Inf. Technol.*, vol. 5, no. 2, pp. 13–21, May 2013.

[4] R. Dean, Septian Ari Moesriami , Barmawi Ema, "MODIFIKASI METODE STEGANOGRAFI DYNAMIC CELL SPREADING (DCS) PADA CITRA DIGITAL," pp. 1–5, 2012.

[5] K. Struss, "A Chaotic Image Encryption," pp. 1–19, 2009.

[6] S. V. Kumari and G. Neelima, "An Efficient Image Cryptographic Technique by Applying Chaotic Logistic Map and Arnold Cat Map," vol. 3, no. 9, pp. 1210–1215, 2013.

[7] M. Mishra, "High Security Image Steganography with Modified Arnold ' s Cat Map," vol. 37, no. 9, pp. 16–20, 2012.

[8] A. Gangwar, “Improved RGB -LSB Steganography Using Secret,” vol. 4, pp. 85–89, 2013.

[9] D. G. Lowe, “Distinctive Image Features from Scale-Invariant Keypoints,” *Int. J. Comput. Vis.*, vol. 60, no. 2, pp. 91–110, Nov. 2004.

[10] S. Jatmiko, “Analisis Dan Implementasi Penggunaan Scale Invariant Feature Transform (SIFT) Pada Sistem Verifikasi Tanda Tangan,” 2013.

[11] N. Hamid, A. Yahya, R. B. Ahmad, and O. M. Al-qershi, “A Comparison between Using SIFT and SURF for Characteristic Region Based Image Steganography,” vol. 9, no. 3, pp. 110–116, 2012.

[12] P. Bateman, “Image Steganography and Steganalysis,” *Fac. Eng. Phys. Sci. Univ. Surrey*, no. August, 2008.

[13] R. Candra, N. Santi, S. Pd, and M. Kom, “Mengubah Citra Berwarna Menjadi Gray Scale dan Citra biner,” vol. 16, no. 1, pp. 14–19, 2011.

[14] K. J. Devi, “A Sesure Image Steganography Using LSB Technique and Pseudo Random Encoding Technique by,” no. May, 2013.

[15] R. R. Arief, “ANALISIS PENGGUNAAN SCALE INVARIANT FEATURE TRANSFORM SEBAGAI METODE EKSTRAKSI FITUR PADA PENGENALAN JENIS KENDARAAN,” 2010.

[16] P. O. Bergerak, S. E. Agustina, and I. Mukhlash, “Implementasi Metode Scale Invariant Feature Transform (SIFT) Dan Metode Continuosly Adaptive Mean-Shift (Camshift) Pada,” vol. 1, no. 1, pp. 1–6, 2012.

[17] D. K. Budiarsyah, “Pengujian Beberapa Teknik Proteksi Watermark Terhadap Penyerangan,” 2013.

[18] A. Cheddad, J. Condell, K. Curran and P. M. Kevitt, "Digital Image Steganography: Survey and Analysis of Current Methods," Signal Processing , vol. XC, no. 3, pp. 727-752, 2010.