Perancangan Aplikasi Steganografi untuk Menyisipkan Pesan pada Citra dengan Metode Spread Spectrum

*Designing Steganographic Application to Insert Message in Image Using*

*Spread Spectrum Method*

I Made Arditha1, Imam Suharjo2

1,2Program Studi Informatika, Fakultas Teknologi Informasi, Universitas Mercu Buana Yogyakarta,

Jl. Wates Km. 10 Yogyakarta 55753, Indonesia

Email: 15111044.imadearditha@gmail.com1, imam@mercubuana-yogya.ac.id2

ABSTRAK

Perkembangan jaman yang semakin maju membuat lalu lintas pertukaran informasi didunia maya semakin mudah dan cepat. Dari semua bidang dan kalangan memanfaatkan teknologi untuk pertukaran informasi. Salah satu media yang dimiliki oleh banyak orang adalah perangkat mobile, seperti telepon genggam dan komputer tablet. Faktanya banyak orang sudah menggunakan perangkat mobile untuk fungsi pertukaran informasi, dan mengharapkan informasi yang dikirimkan dapat sampai dengan cepat, tepat, dan aman. Keamanan informasi yang dikirim menjadi sangat penting artinya apabila informasi tersebut bersifat rahasia. Salah satu cara yang dapat ditempuh untuk mengamankan informasi yang dikirim adalah dengan penyembunyian informasi ke dalam sebuah wadah (media) sehingga informasi sulit dikenali oleh indra manusia, atau biasa disebut dengan istilah steganografi. Pada penelitian ini dilakukan studi dan implementasi steganografi menggunakan metode Spread Spectrum pada perangkat mobile berbasis Android. Dari penelitian ini diperoleh hasil bahwa mutu berkas citra digital yang telah disisipkan pesan tidak mengalami perubahan berarti dengan rata-rata nilai MSE adalah 23.78 dan PSNR adalah 35.234019. Namun berkas citra digital yang telah disisipkan pesan tidak tahan terhadap proses editing (pemotongan dan perubahan ukuran piksel) yang dilakukan pada berkas citra digital tersebut.

**Kata kunci**: Android; Citra Digital; Spread Spectrum; Steganografi

ABSTRACT

Rapidly changing times make information exchange in the virtual world easier and faster. All fields and layers of society use technology for information exchange. One of the media owned by many people was mobile devices, e.g. cell phone and tablet computer. The fact is many people use mobile devices for information exchange with the expectation that the information they send will arrive quickly, accurately, and safely. The security of the sent information is very important if the information is confidential. One of the ways to secure the sent information is hiding the information is a vessel (media) so that the information was difficult to be recognized by human senses, or known as steganography. The present research studied and implemented steganography using Spread Spectrum method on Android-based mobile devices. The study showed that the quality of digital image files with inserted messages didn’t have any significant change with average MSE of 23.78 and PSNR of 35.234019. However, digital image files with inserted messages couldn’t withstand editing process (cropping and changing pixel size) on the digital image files.

**Keywords**: Android; Digital Image; Spread Spectrum; Steganography

**1. PENDAHULUAN**

Perkembangan teknologi informasi saat ini telah memberikan kemudahan dalam melakukan aktivitas manusia. Pengiriman data dan informasi menjadi lebih mudah dan cepat. Seiring dengan perkembangan teknologi informasi tersebut, semakin berkembang pula teknik kejahatan yang berupa perusakan maupun pencurian data oleh pihak yang tidak memiliki wewenang atas data tersebut. Dengan berbagai teknik pengambilan informasi secara ilegal yang berkembang, banyak yang mencoba untuk mengakses informasi yang bukan haknya. Oleh karena itu, pada saat ini telah dilakukan berbagai upaya untuk menjaga keamanan data dan informasi tersebut. Berbagai macam teknik digunakan dalam upaya mengamankan suatu data penting. Digital steganografi memerlukan suatu media sebagai tempat penyembunyian informasi. Secara teori penyisipan informasi pada data digital dengan menggunakan teknik steganografi dapat dilakukan pada semua format data digital yang ada dalam komputer sebagai media covernya seperti format teks, format gambar, bahkan untuk format audio dan sebagainya asalkan *file*-*file* tersebut mempunyai *bit*-*bit* data redundan yang dapat dimodifikasi.

Steganografi membuat data satu informasi yang dikirimkan menjadi lebih aman dengan berbagai cara. Salah satu metode steganografi adalah *Spread Spectrum*. Metode *Spread Spectrum* mentransmisikan sebuah sinyal pita informasi yang sempit kedalam sebuah kanal pita lebar dengan penyebaran frekuensi. Penyebaran frekuensi sendiri berfungsi menambah tingkat redudansi. Tujuan menambah redudansi adalah agar kode tidak mudah dipecahkan. Oleh karena gambar merupakan media yang paling sering digunakan pada internet dan merupakan media yang sulit dibedakan antara gambar yang sudah disisipi pesan dengan gambar yang asli, gambar merupakan media yang ideal untuk penyisipan pesan dengan steganografi.

Berdasarkan penjelasan diatas, maka peneliti tertarik untuk merancang sebuah aplikasi steganografi penyisipan pesan pada citra menggunakan metode *spread spectrum*.

**2. TINJAUAN PUSTAKA**

Steganografi penyisipan pesan sudah banyak diteliti berbagai kalangan. Anti, Kridalaksana, dan Khairina (2017) melakukan penelitian dengan judul Steganografi pada Video Menggunakan Metode *Least Significant Bit* (LSB) dan *End Of File* (EOF). Dalam penelitian ini membahas metode LSB dan EOF untuk menyisipkan pesan teks ke dalam *file* video. Metode LSB bekerja dengan mengganti *bit* terakhir kode biner citra dengan kode biner pesan, sedangkan metode EOF bekerja dengan menambahkan nilai desimal pesan ke dalam *pixel* citra terakhir.

Kuniadi, Puspitaninggrum, dan Coastera (2017) dalam penelitian dengan judul Perancangan dan Pembuatan Aplikasi Steganografi Pesan Teks pada Audio Digital dengan Metode *Least Significant Bit*. Penelitian ini mengimplementasikan steganografi menggunakan metode *Least Significant Bit* (LSB) yaitu dengan memodifikasi *bit* yang kurang signifikan atau *bit* terakhir dalam satu *byte* data dengan menggunakan audio WAV dan MP3 sebagai media penampung.

Darwis dan Kisworo (2017) dengan judul Teknik Steganografi untuk Penyembunyian Pesan Teks Menggunakan *Algoritma End Of File*. Penelitian ini akan menghasilkan *stego image* yang tidak berubah secara signifikan serta proses pengambilan pesan yang *relative* cepat sehingga menjadikan alternatif pengiriman pesan agar terhindar dari pencurian dan sabotase.

Darwis (2017) dengan judul Teknik Steganografi untuk Penyembunyian Pesan Teks Menggunakan Algoritma *Gifshuffle*. Dari hasil pengujian yang dilakukan yaitu gambar yang disisipkan pesan teks dapat menampung ukuran data yang relatif lebih besar yaitu mencapai 1,82 MB.

Abidin (2016) dalam Penerapan Metode Steganografi *One Domain* (OD) Untuk Keamanan Penyisipan Data Pada File Gambar. Dalam penelitian ini mengusulkan teknik steganografi OD sebagai teknik penyisipan data rahasia pada gambar digital.

**2.1 Steganografi**

Menurut Ariyus (2009) Steganografi (*steganography*) adalah ilmu dan seni komunikasi rahasia dengan menyembunyikan pesan pada objek yang tampaknya tidak berbahaya. Keberadaan pesan steganografi adalah rahasia. Steganografi membutuhkan dua properti, yaitu wadah penampung dan data rahasia yang akan disembunyikan. Steganografi digital menggunakan media digital sebagai wadah penampung, misalnya citra, suara (audio), teks dan video.

**2.2 Spread Spectrum**

*Spread spectrum* adalah sebuah metode komunikasi dimana semua sinyal komunikasi disebar di seluruh spektrum frekuensi yang tersedia. Pada awalnya dikembangkan untuk kepentingan militer dan intelejen. Ide dasarnya adalah untuk menyebarkan sinyal informasi melalui *bandwidth* yang lebih luas untuk mencegah dilakukannya pencegatan informasi dan gangguan-gangguan lainnya. Istilah *spread spectrum* digunakan karena pada sistem ini sinyal yang ditransmisikan memiliki *bandwidth* yang jauh lebih lebar dari *bandwidth* sinyal informasi (mencapai ribuan kali). Proses penebaran *bandwidth* sinyal informasi ini disebut *spreading*. Berikut ini merupakan penjelasan dari perhitungan manual dari masing-masing proses.

1. Perhitungan Manual Proses *Encode*

Perhitungan manual pada proses *encode* dapat digambarkan sebagai berikut. Dengan sebuah gambar yang berformat JPG, isi pesan “Hidup”, kata kunci "Indah". Langkah pertama, dilakukan pengecekkan ukuran pesan yang dimasukkan apakah lebih kecil dari ukuran gambar, yaitu dengan memasukkannya ke dalam rumus :

*Panjang Pesan=((Ukuran Pesan)+28)×4 ×8)*

(1)

Keterangan:

* Angka 28 adalah untuk tag pemberian tanda pada gambar yang sudah disisipkan
* Angka 4 adalah besar faktor pengali yang berguna untuk penyebaran *bit*
* Angka 8 adalah *bit* gambar

Setelah pengecekan ukuran pesan selesai, langkah selanjutnya yaitu mengubah pesan ke dalam bentuk biner. Hasil pengkonversian biner dari pesan "Hidup" adalah “01001000 01101001 01100100 01110101 01110000”. Kemudian biner dari pesan tersebeut disebar sebanyak 4 kali, sehingga akan menghasilkan segmen baru, yaitu :

00001111000000001111000000000000

00001111111100001111000000001111

00001111111100000000111100000000

00001111111111110000111100001111

00001111111111110000000000000000

Kemudian langkah selanjutnya adalah pembangkitan pseudonoise dengan bibit pembangkitan yang ditentukan berdasarkan kata kunci “Indah”.



74 (desimal)

Setelah mendapatkan nilai desimal dari kata kunci (74), nilai tersebut digunakan sebagai bibit awal pembangkitan bilangan acak. Langkah berikutnya, yaitu melakukan perhitungan pembangkitan bilangan acak sesuai dengan rumus pembangkitan bilangan acak LCG adalah:

 *mod m*

(2)

a = 17, b = 7, m = 84

Keterangan:

* Xn = bilangan acak ke-n dari deretnya
* Xn – 1 = bilangan acak sebelumnya
* a = faktor pengali
* b = increment
* m = modulus pembagi

Perhitungannya adalah sebagai berikut.

=(17×74+7) *mod* 84 hasilnya = 5

=(17×5+7) *mod* 84 hasilnya = 8

=(17×8+7) *mod* 84 hasilnya = 59

=(17×59+7) *mod* 84 hasilnya = 2

=(17×2+7) *mod* 84 hasilnya = 70

Demikian sterusnya untuk ,,,...

Sebagai contoh dilakukan lima kali penyebaran dan hasilnya hasilnya adalah “5 8 59 2 70”, selanjutnya diubah ke dalam bentuk biner menjadi “00000101 00001000 00111011 00000010 01000110”. Langkah berikutnya, yaitu segmen pesan akan dimodulasi dengan *pseudonoise signal* menggunkan fungsi XOR (*Exlusive* OR).

Segmen pesan:

00001111000000001111000000000000

00001111111100001111000000001111

00001111111100000000111100000000

00001111111111110000111100001111

00001111111111110000000000000000

*Pseudonoise signal*:

0000010100001000001110110000001001000110

Maka hasil proses modulasi antara segmen pesan dengan *pseudonoise signal* menggunakan fungsi XOR adalah:

00001010000010001100101100000010

01001001111100001111000000001111

00001111111100000000111100000000

00001111111111110000111100001111

00001111111111110000000000000000

Hasil dari proses modulasi inilah yang akan disisipkan ke dalam *bit*-*bit* dari *body* gambar. Sebagai contoh, misalkan mengambil sepuluh piksel dari gambar dan mengambil tiga puluh *bit* pertama dari hasil modulasi antara segmen pesan dan *pseudonoise signal*. Kemudian diubah menjadi biner dan disisipkan hasil proses modulasi antara segmen pesan dengan pseudonoise signal menjadi sebagai berikut.

Red Green Blue

0101110**0** 1011110**0** 1101001**0**

0101111**0** 1011101**1** 1101010**0**

0110000**1** 1011110**0** 1101001**0**

0101111**0** 1011101**0** 1101011**0**

0101110**1** 1011100**0** 1101010**0**

0101110**0** 1011100**1** 1101001**1**

0101111**0** 1011100**0** 1101001**1**

0101101**0** 1011110**1** 1101001**1**

0101110**0** 1011101**0** 1101010**0**

0101110**0** 1011100**0** 1101011**0**

Langkah tersebut berlanjut sampai hasil modulasi antara segmen pesan dan *pseudonoise signal* disisipkan semuanya.

1. Perhitungan Manual Proses *Decode*

Perhitungan manual pada proses *dencode* dapat digambarkan sebagai berikut. Pilih gambar yang akan diekstrak, serta gunakan kata kunci yang sama seperti saat proses *encode* yaitu “Indah”. Langkah awal, yaitu dilakukan proses penyaringan pada *body* gambar agar mendapatkan *bit*-*bit* hasil modulasi. Hasil dari proses penyaringan yang dilakukan akan mendapatkan *bit*-*bit* sebagai berikut :

00001010000010001100101100000010

01001001111100001111000000001111

00001111111100000000111100000000

00001111111111110000111100001111

00001111111111110000000000000000

Setelah semua *bit*-*bit* hasil modulasi diperoleh, kemudian langkah selanjutnya, yaitu melakukan proses demodulasi dengan *pseudonoise signal* dari kata kunci yang sama pada proses *encode* agar memperoleh *bit*-*bit* yang berkorelasi.

Hasil penyaringan :

00001010000010001100101100000010

01001001111100001111000000001111

00001111111100000000111100000000

00001111111111110000111100001111

00001111111111110000000000000000

*Pseudonoise signal*:

0000010100001000001110110000001001000110

Hasil demodulasi:

00001111000000001111000000000000

00001111111100001111000000001111

00001111111100000000111100000000

00001111111111110000111100001111

00001111111111110000000000000000

Langkah selanjutnya, yaitu membagi empat hasil demodulasi, yang berguna untuk menyusutkan hasil demodulasi menjadi isi pesan yang sebenarnya. Proses penyusutan (*de*-*spreading*) segmen tersebut menjadi:

01001000011010010110010001110101011110000

Hasil akhir “01001000 01101001 01100100 01110101 011110000” merupakan segmen pesan yang sama ketika disembunyikan pada proses *encode*. Hasil tersebut kemudian diubah ke bentuk karakter akan menjadi “Hidup”.

**2.3 Citra Digital**

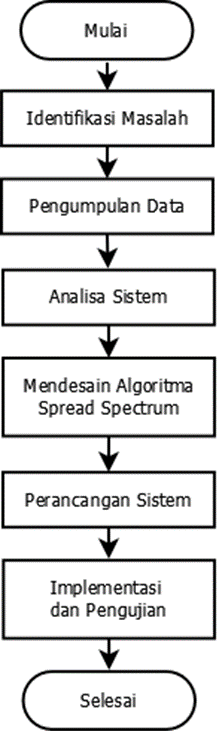
Menurut Andono (2012) Citra digital merupakan representatif dari citra yang diambil oleh mesin dengan bentuk pendekatan berdasarkan sampling dan kuantisasi. Sampling menyatakan besarnya kotak-kotak yang disusun dalam baris dan kolom. Dengan kata lain, sampling pada citra menyatakan besar kecilnya ukuran piksel (titik) pada citra, dan kuantisasi menyatakan besarnya nilai tingkat kecerahan yang dinyatakan dalam nilai tingkat keabuan (*grayscale*) sesuai dengan jurnlah *bit* biner yang digunakan oleh mesin, dengan kata lain kuantisasi pada citra menyatakan jumlah warna yang ada pada citra.

**2.4 Android**

Menurut Safaat (2015) *Android* adalah sebuah sistem operasi untuk perangkat *mobile* berbasis *Linux* yang mencangkup sistem operasi, *middleware* dan aplikasi. *Android* menyediakan *platform* terbuka bagi para pengembang untuk menciptakan aplikasi mereka.

**3. METODOLOGI PENELITIAN**

Langkah-langkah dalam perancangan aplikasi steganografi penyisipan pesan pada citra dalam penelitian ini ditunjukkan pada Gambar 1.

******

Gambar 1. Tahapan Metode Penelitian

**3.1 Identifikasi Masalah**

Pada penelitian ini, masalah penelitian secara umum bisa kita temukan lewat studi literatur atau lewat pengamatan lapangan, selanjutnya dilakukan pengidentifikasian masalah. Masalah yang akan diidentifikasi, yaitu bagaimana menerapkan metode *Spread Spectrum* dalam steganografi pada citra digital.

**3.2 Pengumpulan Data**

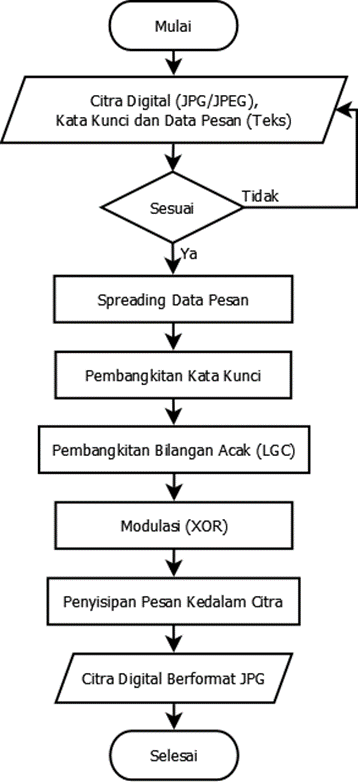
Tahapan ini merupakan langkah-langkah yang dilakukan untuk mendapatkan data yang dibutuhkan, yaitu dengan membaca buku-buku yang berhubungan dengan steganografi, metode *spread spectrum*, teori citra digital, serta mencari informasi dan literatur dari internet.

**3.3 Analisis Sistem**

Pada tahap ini dilakukan analisa data dan permasalahan yang telah dirumuskan, kemudian merancang sebuah sistem yang dapat menjawab permasalahan yang ada. Pada saat menganalisa data, ada beberapa tahap yang harus dilakukan, yaitu mengidentifikasi kebutuhan sistem serta merancang antar muka pengguna sistem yang akan dibangun.

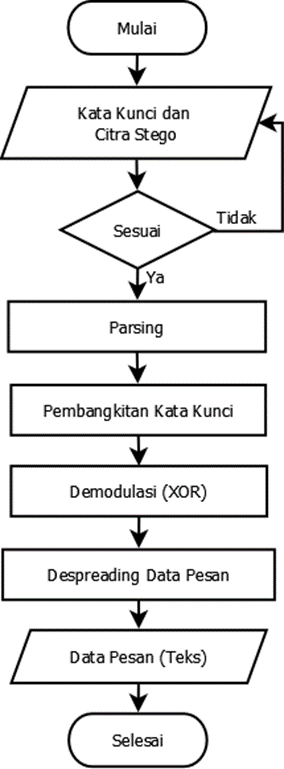
**3.4 Algoritma Spread Spectrum**

1. Algoritma penyisipan merupakan suatu cara menggambarkan proses dalam penyisipan data pesan berupa teks ke dalam citra digital dalam penelitian ini ditunjukkan pada Gambar 2.

**

Gambar 2. Algoritma Penyisipan

1. Algoritma ekstraksi merupakan suatu cara menggambarkan proses ekstraksi data sehingga diperoleh data pesan berupa teks yang telah disisipkan ke dalam citra digital dalam penelitian ini ditunjukkan pada Gambar 3.

**

Gambar 3. Algoritma Ekstraksi

**3.5 Perancangan Sistem**

1. Rancangan Antarmuka

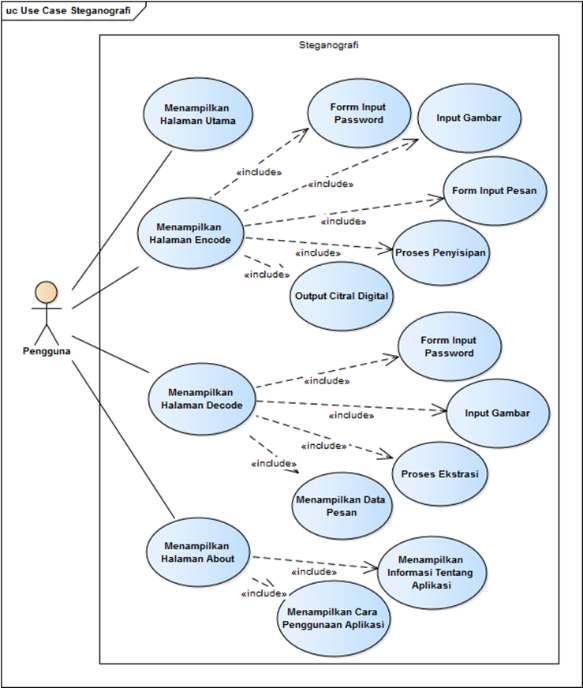
Perancangan antar muka diperlukan untuk mempermudah pengguna dalam menggunakan sistem steganografi ini. Sistem steganografi ini memiliki lima menu utama yang ditunjukkan pada Gambar 4.

**

Gambar 4. Tampilan Menu Utama

1. *Unified Modeling Language* (UML)

Perancangan-perancangan yang akan dijelaskan dalam analisa laporan ini meliputi perancangan model dalam bentuk UML (*Unified Modeling Language*) yang ditunjukkan pada Gambar 5.

**

Gambar 5. Use Case Diagram

**3.6 Implementasi dan Pengujian**

Tahapan implementasi merupakan tahap dimana sistem siap dioperasikan pada keadaan yang sebenarnya, sehingga akan diketahui apakah sistem yang dibuat benar-benar dapat menghasilkan tujuan yang ingin dicapai. Dalam implementasi dan pengujian ini akan digunakan:

1. Perangkat lunak dan perangkat keras.
2. Coding program Java.

**4. PEMBAHASAN**

**4.1 Hasil Penelitian**

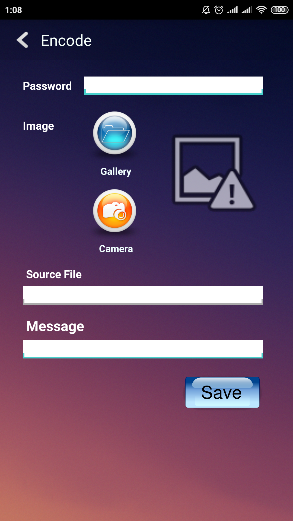
Hasil penelitian ini yaitu membuat sebuah aplikasi berbasis *android* yang mampu menyisipkan data pesan berupa teks ke dalam citra digital berformat JPG/JPEG, serta dapat melakukan ekstraksi data pesan yang telah disipkan ke dalam citra digital. Dengan hasil tampilan sebagai berikut :

1. Tampilan halaman *Home* muncul saat aplikasi pertama kali dibuka, pada halaman ini berisi empat tombol berupa gambar yang mengarah ke Halaman *Encode*, Halaman *Decode*, Halaman *About*, dan *Exit* yang ditunjukkan pada Gambar 6.

**

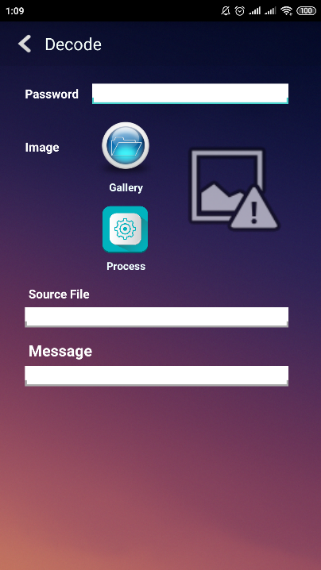
Gambar 6. Tampilan Halaman *Home*

1. Tampilan halaman *Encode* muncul saat menu *Encode* dipilih pada aplikasi. Pada halaman ini terdapat *form input password*, *input* pesan, serta dua tombol *input* gambar dari *Camera* atau *Gallery* dan juga terdapat tombol *Save* untuk memproses penyisipan data pesan berupa teks ke dalam citra digital yang ditunjukkan pada Gambar 7.



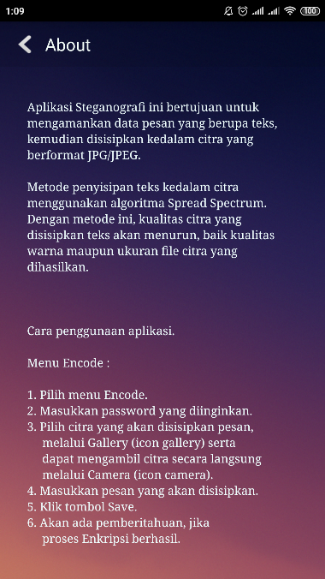
Gambar 7. Tampilan Halaman *Encode*

1. Tampilan halaman *Decode* muncul saat menu *Decode* dipilih pada aplikasi. Pada halaman ini terdapat *form input password*, tombol input gambar dari *Gallery*, serta tombol *Process* untuk proses ekstraksi data sehingga diperoleh data pesan berupa teks yang telah disisipkan ke dalam citra digital yang ditunjukkan pada Gambar 8.



Gambar 8. Tampilan Halaman *Decode*

1. Tampilan halaman *About* muncul saat menu *About* dipilih pada aplikasi. Pada halaman ini terdapat informasi mengenai aplikasi dan cara penggunaan aplikasi yang itujukkan pada Gambar 9.



Gambar 9. Tampilan Halaman *About*

**4.2 Hasil Perbandingan Citra Digital Sebelum dan Sesudah Disisipkan Data Pesan**

Program aplikasi dapat dijalankan menggunakan inputan teks yang diketikkan langsung pada aplikasi. Medium prantara adalah citra digital yang berformat JPG/JPEG, yang dapat diambil secara langsung melalui galeri dan kamera pada perangkat mobile. Berikut ini adalah hasil perbandingan citra digital sebelum dan sesudah disisipkan data pesan yang ditunjukkan pada Tabel 1.

1. Tabel 1. Hasil Perbandingan Citra Sebelum dan Citra Sesudah Disisipkan Data Pesan

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| No | Kunci | Pesan | | Citra Asli | Citra Stego | MSE | PSNR |
| 1 | Langit | Langit adalah bagian atas dari permukaan bumi | D:\15111044_Skripsi\Gambar Uji Coba\Citra Asli\Objek 16.jpg | | D:\15111044_Skripsi\Gambar Uji Coba\Citra Stego\1565352091198.jpg | 10.59 | 37.9149668 |
| 2 | 12345678910 | Belajar menghitung angka 1 sampai 10 | D:\15111044_Skripsi\Gambar Uji Coba\Citra Asli\Objek 17.jpg | | D:\15111044_Skripsi\Gambar Uji Coba\Citra Stego\1565352123049.jpg | 52.83 | 30.9356853 |
| 3 | (Penghijauan) | #selamatkanbumikita | D:\15111044_Skripsi\Gambar Uji Coba\Citra Asli\Objek 18.jpg | | D:\15111044_Skripsi\Gambar Uji Coba\Citra Stego\1565352163403.jpg | 42.44 | 31.8874048 |
| 4 | ard@gmail.com | Alamat e-mail untuk kepentingan pribadi | D:\15111044_Skripsi\Gambar Uji Coba\Citra Asli\Objek 19.jpg | | D:\15111044_Skripsi\Gambar Uji Coba\Citra Stego\1565352201453.jpg | 19.99 | 35.1558419 |
| 5 | (1\*2+(10-2)) | Hasilnya adalah 10 | D:\15111044_Skripsi\Gambar Uji Coba\Citra Asli\Objek 20.jpg | | D:\15111044_Skripsi\Gambar Uji Coba\Citra Stego\1565352239292.jpg | 17.08 | 35.8390581 |
| 6 | Bintang | Bintang marupakan banda langit yang mamancarakan cahaya | D:\15111044_Skripsi\Gambar Uji Coba\Citra Asli\Objek 21.jpg | | D:\15111044_Skripsi\Gambar Uji Coba\Citra Stego\1565368117260.jpg | 17.35 | 35.7714925 |

**4. KESIMPULAN**

Penelitian Perancangan Aplikasi Steganografi Penyisipan Pesan pada Citra dengan Metode *Spead Spectrum* menghasilkan kesimpulan sebagai berikut :

1. Dari hasil pengujian yang dilakukan dengan membandingkan kualitas citra digital sebelum dan sesudah disisipkan data pesan dengan menghitung nilai MSE dan PSNR, diperoleh hasil bahwa nilai rata-rata MSE adalah 23.78 dan nilai rata-rata PSNR adalah 35.234019.
2. Dari hasil validasi yang dilakukan dalam proses ekstraksi menggunakan kunci yang sama dan kunci yang berbeda, diketahui bahwa prosentase hasil ekstraksi data pesan mengunakan kunci yang sama dan kunci yang berbeda diperoleh kesesuain sebesar 100%.

**5. UCAPAN TERIMA KASIH**

Dengan segala kerendahan hati, penulis menyampaikan ucapan terimakasih yang mendalam kepada :

1. Bapak Supatman, S.T., M.T., selaku dekan Fakultas Teknologi Informasi Universitas Mercu Buana Yogyakarta.
2. Bapak A. Sidiq Purnomo, S.Kom., M.Eng., selaku kaprodi Informatika Universitas Mercu Buana Yogyakarta.
3. Bapak Imam Suharjo, S.T., M.Eng., selaku dosen pembimbing yang telah memberikan bimbingan, nasehat dan arahan dengan baik.

**DAFTAR PUSTAKA**

Abidin, R. Z. (2016). Penerapan Metode Steganografi One Domain (OD) Untuk Keamanan Penyisipan Data Pada File Gambar. Nusantara Journal of Computersand its Applications, Vol. 1, No.1, Juni 2016, ISSN 2528 0511, Hal. 1-5.

Adriansyah, R. (2017). Teknik Steganografi Citra Digital dengan Metode Arnold's Cat Map dan Least Significant Bit. Naskah Publikasi Prodi Teknologi Informasi.

Andono, P. N. (2017). Pengolahan Citra Digital. Yogyakarta: Penerbit ANDI.

Anti, U. A., Kridalaksana, A. H., & Khairina, D. M. (2017). Steganografi pada Video Menggunakan Metode Least Significant Bit (LSB) dan End Of File (EOF). Jurnal Informatika Mulawarman, Vol. 12, No. 2 September 2017, ISSN 1858-4853, Hal. 104-111.

Ariyus, D. (2009). Pengantar Ilmu Kriptografi. Yogyakarta: Andi.

Darwis, D., & Kisworo. (2017). Teknik Steganografi untuk Penyembunyian Pesan Teks Menggunakan Algoritma End Of File. Jurnal Sistem Informasi dan Telematika, Vol 8, No 2, Oktober 2017, ISSN 2087-2062, Hal. 98-108.

Darwis, Dedi. (2017). Teknik Steganografi untuk Penyembunyian Pesan Teks Menggunakan Algoritma Gifshuffle. Jurnal TEKNOINFO, Vol. 11, No. 1, 2017. ISSN 1693 0010, Hal. 1-6.

Hudaya, K. K. (2015). Cara Cepat Menguasai Java Desktop dengan Metode Pro-OOP. Jakarta: ANDI.

Kuniadi, B., Puspitaninggrum, D., & Coastera, F. F. (2017). Perancangan dan Pembuatan Aplikasi Steganografi Pesan Teks pada Audio Digital dengan Metode Least Significant Bit. Jurnal Rekursif, Vol. 5 No. 3 November 2017, ISSN 2303-0755, Hal. 285-297.

Kurniawan, A. (2017). Perancangan AplikasiI Kriptografi Data Teks Menggunakan Metode One Time Pad. Skripsi thesis, Universitas Mercu Buana Yogyakarta.

Nurul, A. (2016). Perancangan Steganografi pada Citra Digital Menggunakan Modified EOF. Naskah Publikasi Prodi Teknik Informatika.

Pramana, H. W. (2012). Aplikasi Inventory Berbasis Access 2003. Jakarta: PT. Elex Media Komputindo.

Sadikin, R. (2012). Kriptografi untuk Keamanan Jaringan. Yogyakarta: Andi Offset.

Safaat, N. (2015). ANDROID Pemrograman Aplikasi Mobile Smartphone dan Tablet PC Berbasis Android. Bandung: Informatika Bandung.

Singh, K. U. (2014). LSB Audio Steganograpy Approach. ISO 9001:2008 Certified Journal, Volume 4, Issue 4, April 2014, ISSN 2250-2459.