PROGRAM APLIKASI STEGANOGRAFI MENGGUNAKAN METODE SPREAD SPECTRUM PADA PERANGKAT MOBILE BERBASIS ANDROID

Rojali[1]; Afan Galih Salman[2]; Teddy Nugraha[3]

[1, 3] Mathematics & Statistics Department, School of Computer Science, Binus University
Jln. K.H. Syahdan No.9, Palmerah Jakarta Barat 11480
rojali@binus.edu

[2]Computer Science Department, School of Computer Science, Binus University
Jln. K.H. Syahdan No.9, Palmerah Jakarta Barat 11480, Indonesia
asalman@binus.edu

ABSTRACT

The exchange of traffic information in cyberspace grows fast. In all areas of life utilize technology to exchange information. One of the media owned by many people is mobile device such as mobile phone and tablet computer. In fact many people have been using mobile devices for information exchange function, and expect information to be transmitted quickly, accurately, and safely. The information security sent will be very important when the information is confidential. One way to secure information sent is the concealment of information into a media so that information hidden is beyond recognition by the human senses, which is commonly referred to steganography. This research studied and implemented steganography using spread spectrum Method on Android-based mobile devices. The results showed that the inserted image before and after the message was inserted is not different with PSNR value of about 75.

***Keywords**: steganography, Spread Spectrum, image, Android*

ABSTRAK

Jaman yang semakin maju membuat lalu lintas pertukaran informasi di dunia maya semakin berhembus kencang. Di semua bidang kehidupan dari semua 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. Hasil penelitian menunjukan citra sebelum disisipkan dan sesudah disisipkan pesan tidak berbeda dengan nilai PSNR sekitar 75.

***Kata kunci:** steganografi, metode Spread Spectrum, citra, aplikasi Android*

PENDAHULUAN

Perkembangan teknologi yang begitu pesat saat ini mempermudah manusia dalam melakukan berbagai hal karena teknologi dapat mempersingkat jarak dan waktu. Dalam bidang teknologi komputer dan internet, banyak sekali orang yang menggunakan dan memanfaatkan teknologi tersebut. Salah satu contoh nyata adalah banyaknya pengiriman informasi melalui jaringan internet. Tentunya pengiriman informasi melalui internet sangatlah menguntungkan karena selain cepat, biayanya pun murah. Namun di sisi lain juga memiliki kelemahan yaitu informasi yang dikirim dapat dengan mudah dan tanpa diketahui pemilik informasi, dicuri oleh oknum yang tidak bertanggung jawab.

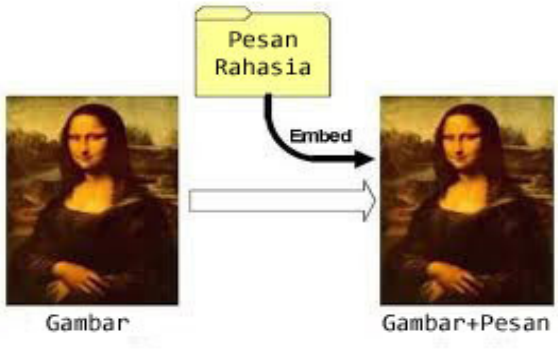
Perkembangan dunia internet pun kini telah menjamur ke media-media yang ada selain komputer, termasuk perangkat *mobile.* Kini, perangkat *mobile* tidak hanya dimiliki oleh golongan atas, tetapi hampir kebanyakan orang sudah memiliki perangkat tersebut. Terlebih lagi *mobile phone* tidak hanya digunakan untuk menelepon ataupun mengirimkan pesan singkat saja, tetapi digunakan juga untuk fungsi-fungsi lainnya yang berhubungan dengan dunia internet, seperti *chat, browsing, blogging, banking,* berinteraksi pada media sosial, ataupun aplikasi-aplikasi lainnya yang mendukung kemudahan memenuhi segala kebutuhan dan keinginan manusia.

Koneksi internet cepat dan murah yang didukung oleh perangkat yang menunjang serta aplikasi yang mudah dapat, menarik, dan murah, menjadi pilihan banyak orang di dunia yang menginginkan segalanya yang serba instan.Selain cepat dan murah, diperhitungkan juga segi lainnya yaitu faktor keamanan. Keamanan menjadi sangat menjadi penting apabila informasi yang dikirimkan merupakan informasi yang bersifat rahasia.

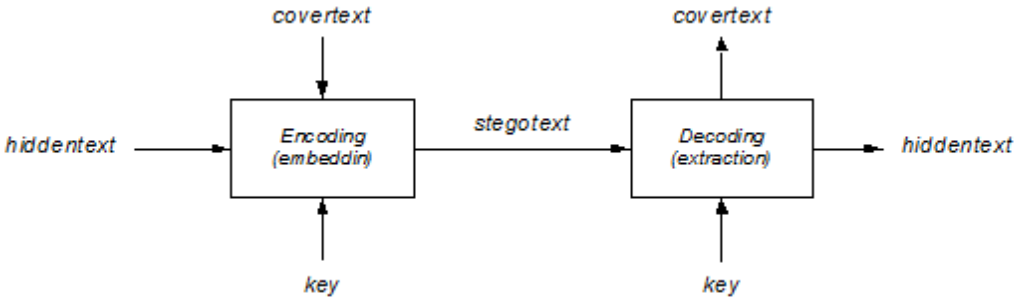
Steganografi adalah teknik penyembunyian informasi ke dalam sebuah wadah (media) sehingga data yang disembunyikan sulit dikenali oleh indra manusia. Teknik tersebut membuat orang lain tidak sadar bahwa ada informasi penting yang kita kirimkan tersembunyi di dalam media lain, seperti citra, audio, maupun video. Seandainya informasi yang telah disembunyikan pada suatu media tersebut pun dicuri, oknum pencuri tersebut belum tentu bisa mengetahui informasi yang ada di dalamnya, karena ada sandi (*key*) untuk bisa membuka informasi yang terkandung dalam media informasi tersebut. Sandi tersebut hanya diketahui oleh pengirim dan penerima. Salah satu metode steganografi adalah *Spread Spectrum.* Metode *spread spectrum* mentransmisikan sebuah sinyal pita informasi yang sempit ke dalam sebuah kanal pita lebar dengan penyebaran frekuensi. Penyebaran frekuensi sendiri berfungsi menambah tingkat redundansi (Winanti, 2008). Dengan menambah tingkat redundansi maka kode tidak mudah dipecahkan.

Menurut Baskara (2007), steganografi adalah ilmu dan seni menyembunyikan informasi dengan cara menyisipkan pesan di dalam pesan lain yang bertujuan menghindari kecurigaan pihak ketiga yang tidak berkepentigan terhadap informasi tertentu. Steganografi membutuhkan dua properti utama: wadah penampung dan informasi rahasia yang akan disembunyikan. Steganografi digital menggunakan media digital sebagai wadah penampung, misalnya citra, suara, teks, dan video. Informasi rahasia yang disembunyikan juga dapat berupa citra (Gambar 1), suara, teks, atau video.

Ada empat komponen utama steganografi, yaitu: (1) *embedded message* (*hiddentext*), yaitu pesan yang disembunyikan; (2) *cover-object* (*covertext*), yaitu pesan yang digunakan untuk menyembunyikan *embedded message*; (3) *stego-object* (*stegotext*), yaitu pesan yang sudah berisi pesan *embedded message*; (4) *stego-key*, yaitu kunci yang digunakan untuk menyisipan pesan dan mengekstraksi pesan dari *stegotext*. Berikut diagram penyisipan dan ekstraksi pesan (Gambar 2).



Gambar 1. Contoh gambar yang disisipi pesan.

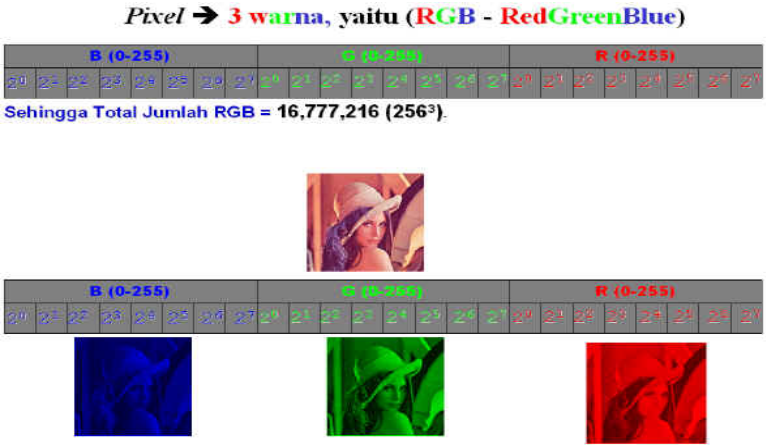


Gambar 2. Diagram penyisipan dan ekstraksi pesan.

Menurut Munir (2004), kriteria steganografi yang baik adalah: (1) *imperceptible*, yaitu keberadaan pesan tidak dapat dipersepsi oleh indra manusia. Jika pesan disisipkan ke dalam sebuah citra, citra yang telah disisipi pesan harus tidak dapat dibedakan dengan citra asli oleh mata. Begitu pula dengan suara, telinga haruslah mendapati perbedaan antara suara asli dan suara yang telah disisipi pesan; (2) *fidelity*, yaitu mutu media penampung tidak berubah banyak akibat penyisipan. Perubahan yang terjadi harus tidak dapat dipersepsi oleh indra manusia; (3) *recovery*, yaitu pesan yang disembunyikan harus dapat diungkap kembali. Karena tujuan steganografi adalah menyembunyikan informasi, sewaktu-waktu informasi yang disembunyikan ini harus dapat diambil kembali untuk dapat digunakan lebih lanjut sesuai keperluan.

Beberapa metode steganografi antara lain: *least significant bit* (*Lsb*), *low bit coding, phase coding, echo hiding,* dan *spread spectrum.* Metode *spread spectrum*, adalah metode yang akan dibahas dalam penelitian ini, yaitu sebuah teknik pentransmisian dengan menggunakan *pseudonoise code*, yang independen terhadap data informasi, sebagai modulator bentuk gelombang untuk menyebarkan energi sinyal dalam sebuah jalur komunikasi (*bandwidth*) yang lebih besar daripada sinyal jalur komunikasi informasi. Oleh penerima, sinyal dikumpulkan kembali menggunakan replika *pseudonoise code* tersinkronisasi. Berdasarkan definisi, dapat dikatakan bahwa steganografi menggunakan Metode *spread spectrum* memperlakukan *cover-object* baik sebagai derau (*noise*) ataupun sebagai usaha untuk menambahkan derau semu (*pseudonoise*) ke dalam *cover-object*. Proses penyisipan pesan menggunakan Metode *spread spectrum* ini terdiri dari tiga proses, yaitu *spreading*, modulasi, dan penyisipan pesan ke citra JPEG. Sedangkan Proses ekstraksi pesan menggunakan Metode *spread spectrum* ini terdiri dari tiga proses, yaitu pengambilan pesan dari matriks frekuensi, demodulasi, dan de-*spreading.*

Menurut Yusron (2011), citra digital merupakan fungsi intensitas cahaya f(x,y) pada bidang 2D, dimana harga x dan y merupakan koordinat spasial dan nilai fungsi tersebut pada setiap titik (x,y) merupakan tingkat kecemerlangan citra pada titik tersebut. Berikut adalah ilustrasi piksel (Gambar 3).



Gambar 3. Ilustrasi piksel.

Peak Signal to Noise Ratio (PSNR) adalah perbandingan antara nilai maksimum yang diukur dengan besarnya *error* yang berpengaruh pada sinyal tersebut. PSNR biasanya diukur dalam satuan desibel. PSNR digunakan untuk mengetahui kualitas citra hasil kompresi. Sementara, *Mean Square Error* (*MSE*), yaitu sigma dari jumlah *error* antara citra hasil kompresi dan citra asli.

$$MSE = \frac{1}{mn}\sum_{i}^{m}\sum_{j}^{n}||I(i,j) - K(i,j)||^2$$

Di mana:
MSE = nilai *Mean Square Error* citra
M = panjang citra (piksel)
N = lebar citra (piksel)
(x,y)= koordinat masing-masing piksel
I = nilai bit dari citra pada koordinat (x,y)
I'= nilai bit dari citra terkompresi pada koordinat (x,y)

Peak Signal to Noise Ratio (PSNR)

$$PSNR = 20log_{10}\left[\frac{255}{(MSE)^{1/2}}\right] = 20log_{10}\left(\frac{255}{RMSE}\right)$$

Nilai *MSE* yang rendah akan lebih baik, sedangkan nilai PSNR yang tinggi akan lebih baik.

METODE

Metode Penelitian dalama penelitian ini terbagi kedalam 3 bagian: (1) studi pustaka, yaitu dengan membaca, memahami, dan mempelajari dengan seksama dari buku-buku dan berbagai macam artikel yang berhubungan dengan topik ini, yaitu steganografi dan Metode *Spread Spectrum.* Menggali sumber-sumber pustaka lainnya yang juga mendukung, seperti jurnal, forum diskusi, pendapat ahli; (2) metode analisis. Metode analisis dalam penelitian ini dibagi menjadi beberapa tahap: mempelajari

lebih dalam karakteristik steganografi, mempelajari metode *spread spectrum*, dan mempelajari bahasa pemrograman *Android.*

Berikut gambaran mengenai perhitungan yang terjadi di dalam Metode *Spread Spectrum*. Pada proses *encode* dapat digambarkan sebagai berikut. Dengan sebuah gambar dengan format JPEG, isi pesan "*test*", kata kunci "sonny". Fungsi akan membaca pesan yang dimasukkan dan mengecek ukuran pesan yang dimasukkan apakah lebih kecil dari ukuran gambar pada yaitu memasukkan ke dalam rumus:

Panjang Pesan = ((ukuran Pesan) + 28) * 4 * 8

Angka 28 adalah untuk *tag* pemberian tanda pada gambar yang sudah disisipkan, angka 4 adalah besar faktor pengali yang berguna untuk penyebaran bit serta angka 8 adalah bit gambar. Setelah mengecek ukuran *file* selesai kemudian dilakukan pengecekan ukuran gambar, metode steganografi yang digunakan dan kata kunci, jika syarat semua sudah terpenuhi dilanjutkan ke dalam proses penyisipan. Sebelum penyisipan dilakukan, fungsi akan membaca gambar dan mengambil *header* dari gambar JPEG yang sudah disiapkan sebelumnya, kemudian gambar dari *body* ini nanti yang akan disisipi pesan. Sebelum proses penyebaran, yang dilakukan adalah mengubah pesan ke bentuk biner. Hasil pengkonversian biner dari pesan "test" adalah 01110100 01100101 01110011 01110100. Kemudian biner pesan disebar dengan besaran skalar pengalinya empat, sehingga akan menghasilkan segmen baru, yaitu:

00001111111111110000111100000000
00001111111100000000111100001111
00001111111111110000000011111111
00001111111111110000111100000000

Kemudian langkah selanjutnya adalah pembangkitan *pseudonoise* dengan bibit pembangkitan yang ditentukan berdasarkan kata kunci "sonny".

```
s =   01110011
o =   01101111
      --------
      00011100
n =   01101110
      --------
      01110010
n =   01101110
      --------
      00011100
y =   01111001
      --------
   01100101 ⟶ 101 (decimal)
```

Setelah mendapatkan nilai dari kata kunci (101), nilai tersebut digunakan sebagai bibit awal pembangkitan bilangan acak. Perhitungan pembangkitan bilangan acak sesuai dengan rumus pembangkitan bilangan acak LCG adalah sebagai berikut:

$$X_{n+1} = (aX_n + c) \bmod$$

a = 17, c = 7, m =84

X_n = bilangan bulat ke-n

Perhitungannya adalah sebagai berikut.

$X_1 = (17 * 101 + 7) \bmod 84$ hasilnya $X_1 = 44$

$X_2 = (17 * 44 + 7) \bmod 84$ hasilnya $X_2 = 83$

$X_3 = (17 * 83 + 7) \bmod 84$ hasilnya $X_3 = 74$

Demikian seterusnya untuk $X_4, X_5, X_6, X_7, X_8 \ldots X_n$

Sebagai contoh dilakukan lima kali penyebaran dan hasilnya hasilnya adalah "44 83 74 5 8" jika diubah dalam bentuk biner menjadi "00101100 01010011 01001010 00000101 00001000."

Untuk mendapatkan hasil modulasi, segmen pesan akan dimodulasi dengan *pseudonoise signal* menggunkan fungsi XOR (*Exlusive OR*).

Segmen pesan:

00001111111111111000011110000000
00001111111100000000111100001111
00001111111111111000000011111111
00001111111111111000011110000000

Pseudonoise signal:

00101100010100110100101000000101000010000

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

00100011101011000100010100000101
00000111111100000000111100001111
00001111111111111000000011111111
00001111111111111000011110000000

Hasil dari proses modulasi inilah yang akan disisipkan ke bit-bit gambar. Sebagai contoh, misalkan mengambil sepuluh piksel dari gambar dan mengambil tiga puluh bit pertama dari modulasi antara segmen pesan dan *pseudonoise signal.*

Red = 180 181 185 182 181 183 186 184 184 187
Green = 166 172 174 171 170 173 176 174 176 179
Blue = 163 169 172 169 168 171 175 173 174 177

Kemudian diubah menjadi biner dan disisipkan hasil proses modulasi antara segmen pesan dengan *pseudonoise signal* menjadi sebagai berikut.

Red	Green	Blue
10110100	10100110	10100011
10110100	10101100	10101000
10111001	10101111	10101101
10110110	10101011	10101000
10110101	10101011	10101000
10110110	10101100	10101011
10111010	10110000	10101110
10111001	10101110	10101101
10111000	10110000	10101110
10111010	10110010	10110001

Langkah tersebut berlanjut sampai modulasi antara segmen pesan dan *pseudonoise signal* disisipkan semua. Proses terakhir setelah proses terakhir penyisipan adalah mengembalikan *header* supaya gambar tidak mengalami kerusakan.

Pada proses ekstraksi prosesnya adalah kebalikan dari proses *encode*. Pilih gambar yang akan diekstrak, gunakan kata kunci yang sama seperti saat *encode* yaitu “sonny”. Langkah awal adalah membaca gambar apakah gambar tersebut sudah pernah disisipi gambar atau belum. Apabila belum kemudian suatu fungsi akan mengambil *header* gambar terlebih dahulu, selanjutnya pada *body* gambar dilakukan proses penyaringan agar mendapatkan bit-bit hasil modulasi. Hasil dari proses penyaringan yang dilakukan akan mendapatkan bit-bit sebagai berikut:

00100011101011000100010100000101
00000111111100000000111100001111
00001111111111111000000011111111
00001111111111111000011110000000

Setelah semua bit-bit hasil modulasi diperoleh, kemudian dilakukan proses demodulasi dengan *pseudonoise signal* dari kata kunci yang sama pada proses modulasi agar memperoleh bit-bit yang berkorelasi. Hasil penyaringan:

00100011101011000100010100000101
00000111111100000000111100001111

00001111111111111000000011111111
00001111111111111000011110000000

Pseudonoise signal:

00101100010100110100101000000101000010000

Hasil demodulasi:

00001111111111111000011110000000
00001111111100000000111100001111
00001111111111111000000011111111
00001111111111111000011110000000

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

01110100 01100101 01110011 01110100

Hasil akhir "01110100 01100101 01110011 01110100" merupakan segmen pesan yang sama ketika disembunyikan pada proses *encode*. Hasil tersebut kemudian diubah ke bentuk karakter akan menjadi "test".

Metode Perancangan

Metode perancangan program yang akan digunakan pada perancangan program aplikasi ini adalah aturan *linear sequential* (*waterfall*). Metode ini memiliki lima tahapan yaitu, *communication*, *planning*, *modelling*, *constraction,* dan *deployment* (Pressman, 2010, p.39).

HASIL DAN PEMBAHASAN

Sejalan dengan perkembangan teknologi pertukaran data dan informasi menjadi hal yang penting dan mendesak serta menuntut kemudahan dan kecepatan, dan juga yang pasti adalah keamanan. Steganografi adalah salah satu cara yang ada untuk membantu menyelesaikan masalah keamanan dalam mengirimkan informasi.

Pertukaran informasi pun sekarang sudah semakin mudah, bukan hanya menggunakan sarana komputer *desktop*, tetapi juga melalui perangkat *mobile.* Tentu saja fenomena ini tidak lepas dari keinginan manusia untuk mendapatkan sesuatu fasilitas yang mudah, cepat, dan simpel. Sekarang, dapat dikatakan hampir semua orang sudah memiliki perangkat *mobile,* terutama telepon genggam baik yang sederhana, yaitu yang digunakan hanya untuk telepon dan *SMS*, maupun yang canggih, yaitu telepon genggam yang dapat digunakan untuk banyak fitur selain telepon dan *SMS*. Fitur-fitur yang dimaksud adalah aplikasi – aplikasi yang dapat dimasukkan ke dalam perangkat sesuai dengan kebutuhan. Misalnya aplikasi untuk media sosial, *chat, email*, *GPS*, dan masih banyak lagi.

Setelah menganalisis masalah yang ada serta melihat kebutuhan dan keinginan tentang keamanan berkirim data berdasarkan hasil kuesioner, maka diputuskan untuk membuat suatu program aplikasi yang dapat meningkatkan keamanan dalam mengirimkan informasi (steganografi) yang akan diimplementasikan pada perangkat *mobile.* Hasil dari penyembunyian teks pada gambar menggunakan Metode *Spread Spectrum.* Sedangkan implementasinya adalah pada sistem operasi *Android*.

Beberapa tahun belakangan ini, *Android* sedang naik daun, dan dapat dikatakan merupakan sistem operasi yang sedang berkembang pesat di dunia. Perkembangan *Android* tidak terlepas dari berkembangnya pasar *mobile phone* dan tablet yang sedang *booming* belakangan ini. Hal tersebutlah

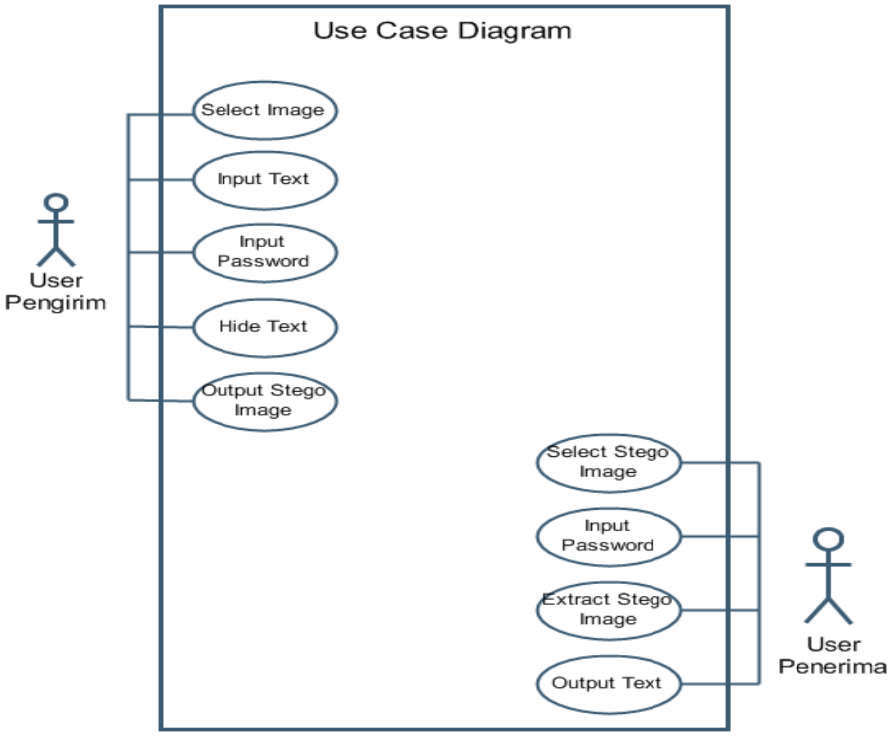
yang mengangkat pamor sistem operasi *Android* yang terkenal juga karena sifatnya *open source*, yang berarti berbagai *developer* perangkat *mobile phone* dan tablet dapat mengembangkan sistem operasi tersebut secara bebas.

Berbeda dengan sistem operasi lain, *Android* berkembang sangat cepat dan secara konsisten mengeluarkan versi-versi yang terus diperbarui dalam waktu yang cukup singkat. Nama-nama yang digunakan untuk versi *Android*-pun cukup unik karena menggunakan nama makanan dan huruf awalnya disusun berdasarkan abjad sehingga mudah untuk mengetahui versi mana yang lebih baru (*Cupcake, Donut, Eclair, Froyo, Gingerbread, Honeycomb, Ice Cream Sandwich,* dan seterusnya). Sistem operasi *Android* adalah sistem operasi yang *open source*, tetapi pada kenyataannya Google juga menerapkan lisensi terhadap pembuat *hardware* yang menggunakan *Android*, yang tentunya sedikit banyak akan mempengaruhi terhadap layanan yang akan didapatkan oleh *user*.

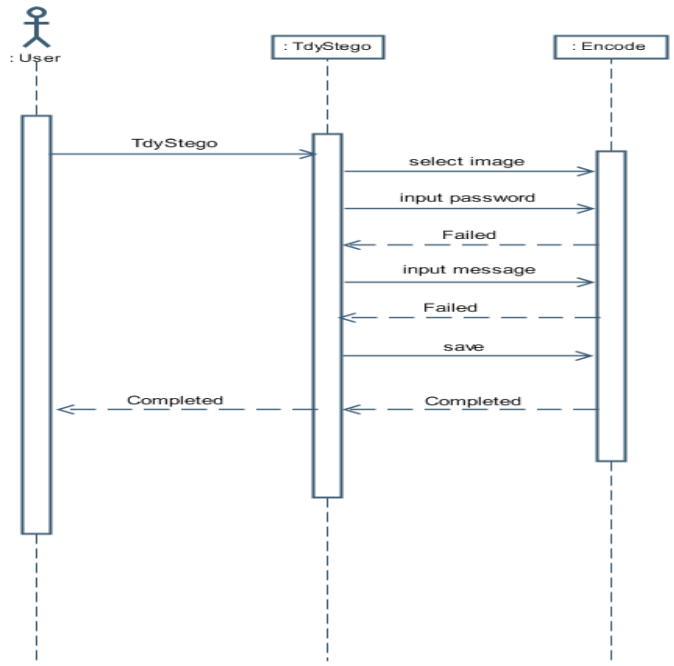
Perancangan

Pada *use case diagram* (Gambar 4), *user* pengirim memilih citra yang akan disisipi oleh teks. Kemudian *user* pengirim menuliskan pesan yang akan disembunyikan ke citra, memasukkan *password*, dan kemudian menekan tombol *save* maka teks akan disembunyikan di dalam citra tersebut. Hasilnya adalah berupa citra yang telah disisipi teks (*stego image*).

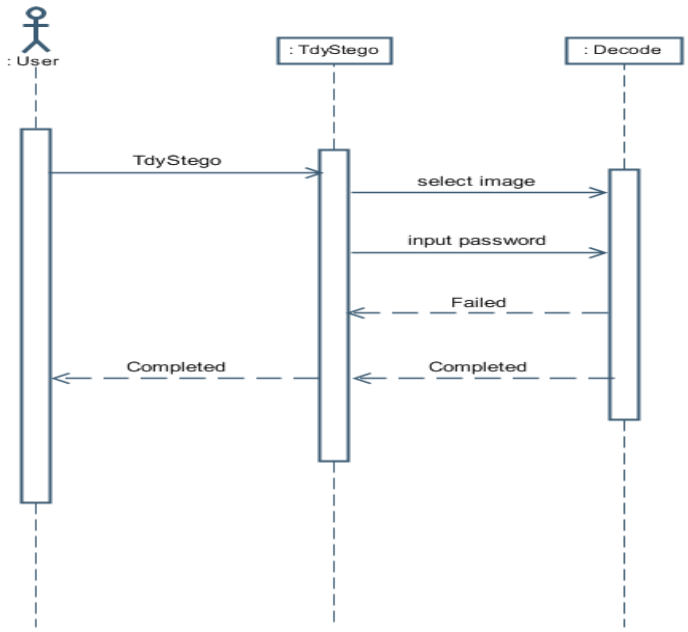
Sedangkan *user* penerima adalah menerima *stego image*, kemudian memasukkan *password*, dan melakukan ekstraksi, sehingga pesan yang disisipkan dan disembunyikan akan muncul. Gambar 5 dan 6 berikut adalah *sequence diagram encode* dan *sequence diagram decode.*



Gambar 4. Use case diagram.



Gambar 5. Sequence diagram encode.

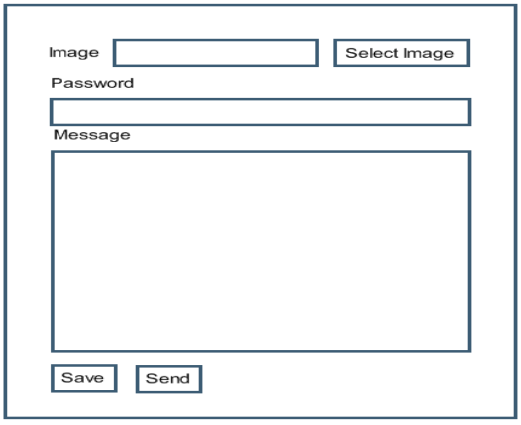


Gambar 6. Sequence diagram decode.

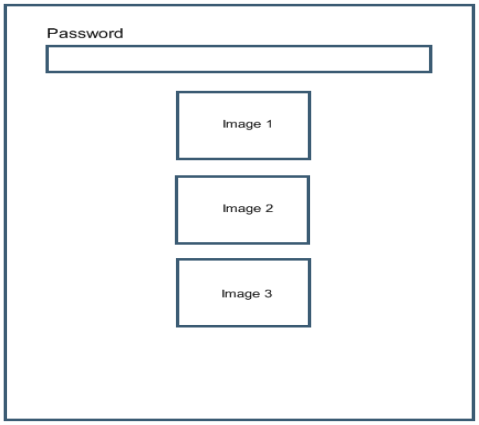
Rancangan Layar Encode dan Decode

Pada halaman ini *user* dapat memilih gambar yang akan menjadi media yang akan disisipkan teks, memasukkan teks yang menjadi pesan rahasia, memilih kata kunci (*password*) yang merupakan salah satu bagian pengamanan pesan yang dibutuhkan oleh metode yang digunakan dalam pemrograman. Setelah itu, *user* dapat melakukan penyisipan (*encode*) (Gambar 7), dan melakukan

pengiriman gambar ke tujuan, atau kembali ke layar utama. Pada rancangan layar *decode* (Gambar 8), *user* dapat memilih gambar berisi pesan, kemudian akan diekstrak sehingga pesan dapat diketahui.



Gambar 7. Rancangan layar *encode.*



Gambar 8. Rancangan layar *decode.*

Evaluasi Program

Program aplikasi dapat dijalankan menggunakan inputan teks yang diketikkan langsung pada aplikasi. Medium perantara adalah citra yang mempunyai format JPEG, dan hasil *encode* citra berformat JPEG (*Stego-object*), di mana ukuran *Stego-object* selalu lebih kecil dari gambar asli sebelum disisipi oleh pesan. Namun kualitas gambar sendiri tidak turun terlalu jauh sehingga tidak kasat mata. Beberapa contoh gambar yang digunakan dalam pengujian program aplikasi, dengan kata kunci “operasi ninja”, dengan pesan “segera amankan Presiden, karena akan ada kudeta besok pagi di istana”.tabel 1 berikut memuat perbandingan citra asli dan citra berisi pesan.

Tabel 1
Perbandingan Citra Asli dan Citra Berisi Pesan

No.	Citra Asli	Citra Setelah Disisipi Pesan	PSNR
1			78.77074
2			78.37334

3			81.03603
4			81.73806
5			70.45341

PENUTUP

Berikut adalah kesimpulan dari penelitian ini. Pertama, program aplikasi yang dibuat dapat menyisipkan, menyembunyikan, ekstraksi informasi berupa teks ke dalam citra berformat JPEG. Kedua, metode *Spread Spectrum,* yang merupakan salah satu metode steganografi, berhasil diimplementasikan pada perangkat *mobile* berbasis *Android.* Ketiga, secara kuantitatif nilai PSNR menunjukkan bahwa *error* yang dihasilkan baik yaitu diatas 30.

Untuk pengembangan lebih lanjut disarankan untuk menggunakan metode steganografi lainnya agar dapat mengoptimalkan pesan yang disembunyikan dan kompleksitas penyembunyian serta ketahanan terhadap berbagai serangan. Data yang kirimkan tidak hanya berupa teks yang dapat disembunyikan, tetapi juga audio dan video. Disarankan supaya dibuat konten aplikasi supaya dapat mengirimkan langsung produk aplikasi ke *user* lainnya.

DAFTAR PUSTAKA

Baskara, Tara. (2007). Studi dan Implementasi Steganografi pada MP3 dengan Teknik Spread Spectrum. Diakses dari http://rasta-shared.blogspot.com/p/dan-lain2.html.

Munir, R. (2004). Steganografi dan Watermarking. Departemen Teknik Informatika, Institut Teknologi Bandung. Diakses dari http://informatika.stei.itb.ac.id/~rinaldi.munir/Kriptografi/Steganografi%20dan%20Watermarking.pdf.

Pressman, R. S. (2010). *Software Engineering: A Practitioner's Approach Seventh Edition.* New York: McGraw-Hill.

Winanti, W. (2008). Penyembunyian Pesan pada Citra Terkompresi JPEG Menggunakan Metode Spread Spectrum. Undergraduate Theses. Institut Teknologi Bandung, Bandung. Diakses dari http://irmanf.blogspot.com/2010_02_01_archive.html.

Yusron, R. (2003, Desember 29). *Pengenalan Citra*. Diakses 9November 2011, dari http://blog.stikom.edu.