ANALISIS STEGANOGRAFI CITRA DIGITAL MENGGUNAKAN METODE *SPREAD SPECTRUM* BERBASIS*ANDROID*

ANALYSIS OF DIGITAL IMAGE STEGANOGRAPHY USING SPREAD SPECTRUM METHOD BASED ON ANDROID

[1]Ari Septayuda[2]Dr., Ir. Bambang Hidayat, DEA[3]Hilal Hudan Nuha,MT
[1,2,3] Fakultas Teknik Elektro Universitas Telkom, Bandung
Jl. Telekomunikasi, Dayeuh Kolot Bandung 40257 Indonesia
[1]ariseptayuda@yahoo.com, [2] bbhtelkom@gmail.com, [3]hilalnuha@gmail.com

ABSTRAK

Pertukaran informasi mengalami perkembangan yang signifikan, dapat terlihat dari media yang banyak dipergunakan saat ini adalah media digital, seperti internet. Salah satu perangkat yang menawarkan penggunaan internet secara mudah adalah Android. Saat ini smartphone berbasis Android sedang *booming* dikalangan masyarakat dunia dikarenakan banyak kemudahan yang ditawarkan.Kemudahan yang seharusnya memberikan keuntungan bagi kita ternyata memiliki sisi negatif. Misalnya pencurian data digital yang dikirim lewat internet dapat disalahgunakan oleh oknum yang tidak bertanggung jawab. Dengan adanya kemudahan tersebut seseorang dapat dengan mudah menyalin, mendistribusikan dan/atau mengubah isi dari data digital tersebut. Oleh karena itu dibutuhkan suatu teknik yang dapat menangani permasalahan tersebut, terutama hal yang menyangkut label hak cipta. Salah satu teknik yang dapat digunakan adalahsteganografi.Steganografi merupakan suatu teknik untuk menyamarkanatau menyembunyikan data atau informasi citra (*image*) ke dalam suatu data digital utama, yang disebut dengan citra *host*, dengan tujuan tertentu. Ada empat jenis metode steganografi, yaitu *Least Significant Bit Insertion* (LSB), *Algorithms and Transformation*, *Redundant Pattern Encoding*, *Spread Spectrummethod*.

Dalam tugas akhir ini telah dirancang steganografi berbasis Android dimana metode yang digunakan adalah *Spread Spectrum* dan diimplementasikan pada sistem operasi Android. Android merupakan sistem operasi yang berbasis Linux untuk telepon seluler. Dengan metode ini pesan dikodekan dan disebar ke setiap spektrum frekuensi yang memungkinkan.

Dari hasil penelitian, sistem steganografi menggunakan metode Spread Spectrum menghasilkan performansi imperceptibilityantara citra cover dan citra stego sangatlah mirip.Kesimpulanini ditunjukan dengan hasil nilai PSNR sebesar 59,153 dan nilai MSE sebesar 0,079 pada citra cover yang disisipi ukuran citra rahasia 16x16 *pixel*. Performansi *robustness* pada citra stego mempunyai nilai BER sebesar 0,154 artinya bit *error* yang terjadi akibat perubahan *pixel* pada citra stego sangat kecil. Namun sistem yang telah dibuat tidak memiliki performansi yang baik ketika diberikan serangan berupa *noise*, *cropping* dan proses kompresi. Hal ini dibuktikan dengan besarnya nilai BER yang berada pada kisaran 0,995.

Kata kunci*:*Steganografi, Spread Spectrum, Android

ABSTRACT

Exchange of information has improved significantly, it can be seen from the many media used today are digital media, such as the internet. One device that offers easy internet use is Android. Currently android-based smartphones is booming among the people of the world due to the many conveniences it offers. Ease that should give you an advantage for us turned out to have a negative side. For instance the data digitalstealing which send through the internet can be abused by person who are not responsible. With the ease, person can be easily copied, distributed, and/or changed the contents of digital data. Therefore we need a technique that can deal with these issues, especially matters related to copyright label. One technique that can be used is steganography. Steganography is a technique to disguise or hide the data or information (image) into a primary digital data, which is called the the host image, with a specific purpose. There are four different methods of steganography, which is the Least Significant Bit insertion (LSB), Algorithms and Transformation, Pattern Redundant Encoding, Spread Spectrum method.

In this final project has been designed steganography based on android where the method used is Spread Spectrum and implemented on the Android operating system.Android is a Linux-based operating system for mobile phones.With this method the message is encoded and distributed to each frequency spectrum allows.

The research says, steganography system using Spread Spectrum methode give the imperceptibility performance between cover image and stego image is very similar. This conclusion is indicated by the results of 59,153 PSNR and 0,079 value of MSE which is cover image embedded by the secret image with size 16x16 pixels. Robustness performance of the stego image has a value of 0,154 BER it means the bit error that occurs due to changes in the pixels of the stego image is very small.But the system that has been created does

not have a good performance when given in the form of attacks such as noise, cropping and compression process. This is proven by the value of the BER in the range of 0,995.

Keywords : Steganography , Spread Spectrum , Android

1. Pendahuluan

1.1 Latar Belakang

Pesatnya perkembangan ilmu teknologi di berbagai bidang berdampak pula pada proses pertukaran informasi. Pertukaran informasi mengalami perubahan yang signifikan dalam jangka waktu yang relatif singkat. Dapat terlihat dari media yang digunakan saat ini bukan lagi media konvensional seperti surat-menyurat via pos atau semacamnya tetapi sudah menggunakan media digital yaitu internet. Melalui internet jarak dan waktu tidak lagi menjadi kendala yang harus dikhawatirkan. Pertukaran informasi menjadi sangat mudah dimanapun dan kapanpun kita inginkan. Kemudahan yang seharusnya memberikan keuntungan bagi kita ternyata juga memiliki sisi negatif yaitu kemungkinan pencurian data juga semakin besar karena media yang digunakan merupakan media publik yang dapat digunakan oleh siapa saja.

Jenis pertukaran informasipun dari hari ke hari semakin beragam, yang pada awalnya hanya terbatas pada surat menyurat dan membutuhkan waktu kirim cukup lama saat ini sudah berkembang menjadi metode digital seperti teks, audio, video, dan citra yang dapat dikirim dalam hitungan detik. Pertukaran informasi digital menggunakan internet sangat riskan mengalami pencurian yang kemudian diubah sedemikian rupa lalu disebarkan kembali. Dengan semakin banyak serangan yang mungkin terjadi dalam proses pertukaran data menyebabkan perlunya suatu metode agar dapat meningkatkan keamanan informasi. Salah satu metode yang dapat digunakan yaitu teknik steganografi. Steganografi memberikan solusi untuk penyembunyian pesan yang sering digunakan dalam proses pengiriman data. Steganografi adalah teknik menyisipkan pesankedalam suatu media[1], dimana pesan rahasiayang akan dikirimkan tidak diubah bentuknya,melainkan disisipkan pada sebuah media lain(*cover-image*) yang digunakan dalam kehidupansehari-hari. Media baru yang telah disisipi pesanrahasia (*stego-image*) kemudian dikirim kepadapenerima tanpa menimbulkan kecurigaan daripihak luar, karena perbedaan dari media asli(*cover- image*) dengan media yang telah disisipipesan rahasia (*stego- image*) tidak dapat disadarisecara langsung oleh manusia. Steganografi padamasa kini dilakukan pada media digital berupacitra, audio, maupun video.

Pada tugas akhir sebelumnya sudah banyak yang mengambil tema steganografi dengan menggunakan metode *Discrete Wavelet Transform*(DWT) maupun *Discrete Cosine Transform* (DCT) *image* to *image*. Teknik DWT telah banyak digunakan pada watermarking karena kemampuan multiresolusi yang dimilikinya[14]. Meski demikian, DWT pun memiliki kekurangan, seperti shift sensitivity dan poor directionality[15]. Shift sensitivity adalah sebuah properti yang tidak diinginkan karena menunjukkan bahwa koefisien DWT gagal untuk membedakan antara pergeseran sinyal input, sedangkan poor directionality adalah directionality yang buruk ketika transformasi mengungkapkan hanya beberapa orientasi fitur dalam domain spasial. Dalam tugas akhir ini digunakansteganografidengan metode *Spread Spectrum* dan diimplementasikan pada Android. Dengan metode ini pesan dikodekan dandisebar ke setiap spektrum frekuensi yangmemungkinkan[2].

Metode *Spread Spectrum*mentransmisikan sebuah sinyal pita informasi yangsempit ke dalam sebuah kanal pita lebar denganpenyebaran frekuensi. Penyebaran ini bergunauntuk menambah tingkat redundansi. Tujuan menambah tingkat redudansi adalah agar kode tidak mudah dipecahkan. Dalam tugas akhir ini juga akan dibahas mengenaidampak perubahan kualitas dari citra yangdihasilkan setelah penyisipan, yang akan diukursecara subjektif dan objektif.

2. Landasan Teori

2.1 Steganografi

Steganografi adalah suatu teknik untuk menyembunyikan informasi yang bersifat pribadi dengan sesuatu yang hasilnya akan tampak seperti informasi normal lainnya. Media yang digunakan umumnya merupakan suatu media yang berbeda dengan media pembawa informasi rahasia, dimana disinilah fungsi dari teknik steganografi yaitu sebagai teknik penyamaran menggunakan media lain yang berbeda sehingga informasi rahasia dalam media awal tidak terlihat secara jelas[3].

Kata *steganografi* berasal dari bahasa Yunani, yaitu dari kata *Stegos* (*covered*/tersembunyi) dan *Graptos* (*writing*/tulisan). Steganografi di dunia modern biasanya mengacu pada informasi atau suatu arsip yang telah disembunyikan ke dalam suatu arsip citra digital, audio, atau video. Teknik *Steganografi* ini telah banyak digunakan dalam strategi peperangan dan pengiriman sandi rahasia sejak jaman dahulu kala. Dalam perang Dunia II, teknik *Steganografi* umum digunakan oleh tentara Jerman dalam mengirimkan pesan rahasia dari atau menuju Jerman[3].

Steganografi dalam penerapannya terhadap data digital, dapat diterapkan pada berbagai

metode. Ada empat jenis metode Steganografi, yaitu Least Significant Bit Insertion (LSB), Algorithms and Transformation, Redundant Pattern Encoding, Spread Spectrum. Metode *Steganografi* yang digunakanadalah Metode *Spread Spectrum.* Metode tersebut dibagi menjadi dua proses utama, yaitu proses *encode* dan*decode*. Pada proses *encode* dilakukan operasi penyisipan pesan *embedded-image* kedalam *cover-image*. Sedangkan proses *decode* dilakukan proses penyaringan hasil penyisipandan kemudian dikembalika menjadi pesan awal. Metode *Spread Spectrum* memiliki keunggulan dalamketangguhan terhadap berbagai serangan, meskipun di lain sisi metode ini memiliki kompleksitas yang tinggi[5].

2.2 Citra *Digital*

Citra (*image*) adalah suatu persepsi visual hasil dari pantulan cahaya yang menerangi objek dan dipantulkan kembali sebagian dari berkas cahaya tersebut. Alat-alat optik seperti mata manusia, kamera, *scanner* menangkap pantulan cahaya tadi sehingga bayangan objek yang disebut citra terekam. Secara sederhana dapat dikatakan sebagai suatu gambar pada bidang dua dimensi.

Citra digital direpresentasikan sebagai sebuah *matriks* yang indeks baris dan kolomnya mengidentifikasikan sebuah titik pada citra dan nilai dari elemen *matriks* yang bersangkutan merupakan tingkat warna pada titik tersebut. Elemen tersebut disebut elemen citra, elemen gambar (*picture elements*), *pixels*, atau *pels*. Resolusi citra pada sebuah citra digital ditentukan oleh *pixels*.

Semakin tinggi resolusi yang dihasilkan, semakin kecil ukuran *pixel*nya yang berarti bahwa citra yang dihasilkan semakin halus. Dalam komputer, citra disimpan dalam bentuk *array* dua dimensi yang berukuran M x N dimana M menyatakan jumlah baris dan N menyatakan jumlah kolom. Oleh sebab itu, sebuah citra digital dapat ditulis dalam bentuk matiks berikut:

$$f(x,y)=\begin{bmatrix} f(0,0) & f(0,1) & \dots & f(0,N-1) \\ f(1,0) & f(1,1) & \dots & f(1,N-1) \\ \dots & \dots & \dots & \dots \\ \dots & \dots & \dots & \dots \\ f(M-1,0) & f(M-1,1) & \dots & f(M-1,N-1) \end{bmatrix} \quad (2.1)$$

2.3 *Spread Spectrum*[5]

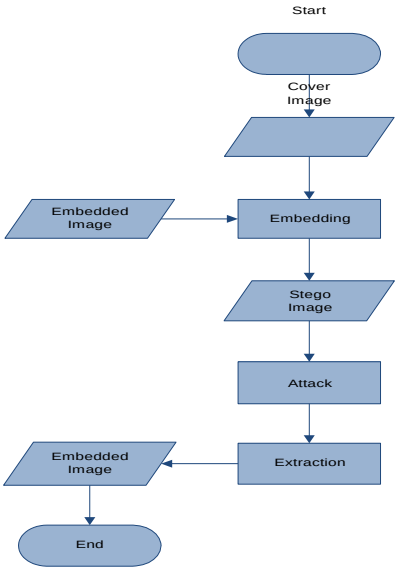
Metode*Spread Spectrum*adalahsebuahteknik pentransmisiandenganmenggunakan*pseudonoise code*,yang independenterhadap data informasi, sebagai modulatorbentukgelombanguntukmenyebarkanenergysinyaldalamsebuahjalurkomunikasi (*bandwidth*) yanglebihbesardaripadasinya jalurkomunikasiinformasi.Olehpenerima,sinyaldikumpulkankembalimenggunakanreplica*pseudonoise code*tersinkronisasi.Berdasarkandefinisi,dapatdikatakanbahwasteganografimenggunakanMetode*SpreadSpectrum*memperlakukan*cover-image* baiksebagaiderau (*noise*) ataupunsebagaiusahauntukmenambahkanderausemu (*pseudonoise*) kedalam*cover-image*.

Proses penyisipanpesanmenggunakanMetode*Spread Spectrum* initerdiridaritiga proses, yaitu*spreading*,modulasi, danpenyisipanpesankecitra. Sedangkan proses ekstraksipesanmenggunakanMetode*SpreadSpectrum* initerdiridaritiga proses, yaitupengambilanpesandarimatriksfrekuensi, demodulasi, dan de-*spreading*.

4. Perancangan dan Realisasi Sistem

3.1 Perancangan Sistem

Berikut adalah blok diagram steganografi pada citra digital :



Gambar 3.1 Diagram Alir Sistem Secara Umum

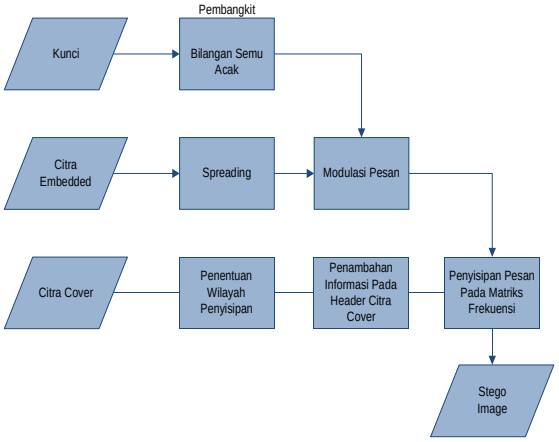
Adapun penjelasan dari diagram alir di atas adalah sebagai berikut :

1. Di sisi pengirim, dilakukan pemilihan media *cover-image* yang akan digunakan kemudian memasukkan pesan rahasia (*embedded-image)* yang akan disisipkan.
2. Kemudian melalui *embedding process* untuk disisipkan ke dalam *cover-image* dengan menggunakan metode *Spread Spectrum.* Keluaran proses tersebut yaitu *stego-image* dimana pesan rahasia (*embedded-image)* telah disisipkan dan yang terlihat hanya citra yang telah tersisipisaja.
3. Selanjutnya citra stego diuji kehandalannya dengan beberapa serangan. Keluaran dari proses tersebut adalah citra stego yang telah diserang.
4. Keluaran dari block ini adalah citra stego yang telah diserang. Di sisi penerima, *stego-image* tersebut diekstraksi untuk menghasilkan pesan rahasia yang tersimpan di dalam *cover-image*. Pada tahap ini pesan rahasia dan *cover-image* dalam keadaan terpisah sehingga pihak

penerima mendapatkan pesan rahasia yang dimaksud.

3.2 Penyisipan Pesan

Sistem untuk menyisipkan pesan pada citra covermembutuhkan masukan berupacitra citra cover, citra embedded yang ingindisisipkan dan kata kunci "sonny" yang akan digunakan untukproses modulasi pesan. Skema penyisipan pesandapat dilihat pada Gambar 3.2.



Gambar 3.2 Skema Penyisipan Pesan

3.2.1 Pembangkitan Bilangan Semu Acak

Pada steganografi, pembangkitan bilangan semu acakdapat digunakan untuk menentukan kuncipenyisipan dan ekstraksi data dari berkas media.Komputer mampu menghasilkan bilangan semuacak (*pseudorandom*). Deret bilangan*pseudorandom* adalah deret bilangan bilangan yangkelihatan acak dengan kemungkinan pengulanganyang sangat kecil atau periode pengulangan yangsangat besar.

Salah satu algoritma pembangkitan bilanganpseudorandom adalah *Linear CongruentialGenerator* (LCG). Algoritma LCG ini diciptakanoleh D.H. Lehmer pada tahun 1951. Deret bilanganbulat dalam LCG dirumuskan pada persamaan 2.2.

Untuk memulai bilangan acak ini dibutuhkansebuah bilangan bulat X_1yaitu yang berasal dari kata kunci dan akandijadikan sebagainilai awal (bibit pembangkitan). Bilangan acakpertama yang dihasilkan selanjutnya menjadi bibitpembangkitan bilangan bulat acak selanjutnya.Jumlah bilangan acak yang tidak sama satu samalain (unik) adalah sebanyak *m*. Semakin besar nilai*m* , semakin kecil kemungkinan akan dihasilkannilai yang sama.

3.2.2 *Spreading*

Proses spreading dilakukan sesuai dengan bilangan pengali skalar yang ditentukan, pada tugas akhir ini menggunakan bilangan pengali skalar 4. Pada proses ini citra rahasia diambil nilai intensitas per-*pixel* nya, lalu diubah kedalam bilangan biner. Kemudian bilangan biner tersebut disebar sesuai bilangan pengali skalar yang telah ditentukan, maka hasil keluaran dari proses spreading ini adalah deret bilangan biner yang telah tersebar dengan panjang setiap deretnya sebesar 32 bit.

3.2.3 Modulasi Pesan

Proses inimerupakan proses pengacakan pesan yang telahdisebar dengan bilangan *pseudonoise* yang telahdibangkitkan menggunakan algoritma LCG pada persamaan 2.3.Panjang dari bilangan *pseudonoise* ini disesuaikandengan panjang dari pesan. Jika panjang pesanlebih kecil dari panjang bilangan *pseudonoise*,bilangan *pseudonoise* tersebut akan dipotongsesuai dengan ukuran pesan. Sebaliknya, jikapanjang pesan lebih besar dari panjang bilangan*pseudonoise*, maka bilangan tersebut akan diulangsampai panjangnya sama dengan panjang pesan.Proses modulasi tersebut dilakukan denganmenggunakan fungsi XOR (*Exclusive OR*). Nilaiyang dihasilkan dari proses modulasi inilah yangkemudian akan disisipkan ke dalam berkas citra *cover*.

3.2.4 Penentuan Wilayah Penyisipan

Untuk mendapatkan hasil yang efektif, penyisipanhanya dapat dilakukan pada *layer blue* yang terdapat pada matriks frekuensi tersebut.

3.2.5 Penambahan Informasi Pada *Header* Citra Cover

Setelah wilayah penyisipan didapatkan, selanjutnyadilakukan penambahan informasi pada *header* dariberkas citra*cover* yang dijadikan media penyisipanpesan. Informasi yang ditambahkan adalah ukuran berkas yangakan disisipkan, wilayah penyisipan pesan, danfaktor bilangan pengali yang digunakan. Sedangkan ukuran maksimum citra rahasia yang dapat disisipkan adalah seperti persamaan berikut.

$I_{(I,J)}$ x *cr* x 8<$K_{(I,J)}$**.....(3.1)**

Keterangan :

I(i,j) : ukuran citra rahasia
K(i,j) : ukuran citra cover
cr : besar pengali skalar

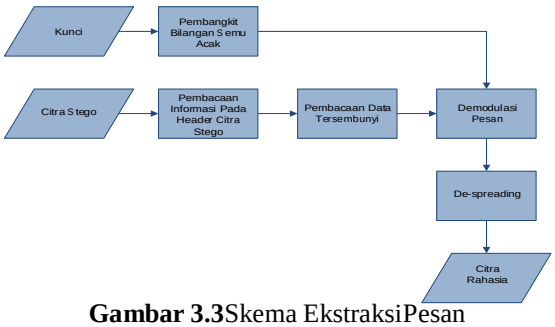
3.2.6 Penyisipan Pesan

Setelah proses penambahan informasi selesaidilakukan, selanjutnya dilakukan tahap terakhirdalam penyisipan pesan pada citra citra*cover*. Pesan yang akandisisipkan dalam tahap ini adalah hasil dari prosesmodulasi yang telah dilakukan sebelumnya.Penyisipan pesan pada matriks frekuensi dilakukandengan cara menyisipkan bit pesan pada bitterakhir dari nilai yang terdapat di matriksfrekuensi. Hal lain yang perlu diperhatikan dalammenyisipkan pesan pada matriks frekuensi adalahpembagian penyisipan yang merata pada seluruhmatriks frekuensi yang

terdapat pada berkas citra*cover*.Untuk itu penyisipan akan dilakukan secara selang-selingberdasarkan jumlah matriks frekuensi yangada pada berkas citra*cover* tersebut. Keluaran dari proses ini adalah citra stego yang telah tersisipi citra rahasia.

3.3 Ekstraksi Pesan

Sistem untuk mengekstraksi pesan pada citra stegomembutuhkan masukan berupacitra stegoyang telah disisipkan pesandan kata kunci "sonny" yang akan digunakan untuk prosesdemodulasi pesan. Skema penyisipan pesan dapatdilihat pada Gambar 3.3.Proses ekstraksi pesan menggunakan metode*Spread Spectrum* ini terdiri dari tiga proses, yaitupengambilan pesan dari matriks frekuensi,demodulasi, dan de-*spreading*.



Gambar 3.3 Skema EkstraksiPesan

3.3.1 Pembacaan Informasi Pada *Header* Citra Stego

Proses pengambilan pesan dari matriks frekuensi diawali dengan pembacaan informasi yang terdapat pada *header*citra stegoyang telah didefinisikan khusus sebelumnya. Adapun informasi yang didapatkan dari pembacaan ini adalah ukuran berkas yang disisipkan, wilayah penyisipan pesan, dan faktor bilangan pengali yang digunakan.

3.3.2 Pembacaan DataTersembunyi

Kemudian dilakukan pembacaan data tersembunyi berdasarkan informasi wilayah penyisipan pesan yang didapatkan dari tahap sebelumnya, yaitu pembacaan informasi pada *header*citra stego. Pembacaan akan dilakukan secara berselang-seling pada matriks frekuensi yang terdapat pada citra dan berlangsung sampai data yang dibaca besarnya sama dengan informasi ukuran berkas yang disisipkan**.**

3.3.3 Demodulasi

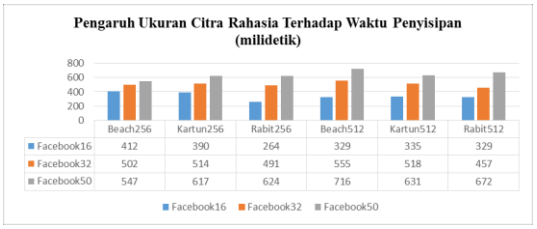
Setelah data tersembunyi berhasil dikumpulkan, dilakukan proses demodulasi terhadap data tersebut. Proses demodulasi ini melibatkan bilangan acak yang dibangkitkan dari kunci masukan menggunakan algoritma LCG pada persamaan 2.3. Adapun proses pembangkitan bilangan acak yang dilakukan pada tahap ekstraksi pesan sama seperti proses pembangkitan bilangan acak pada tahap penyisipan pesan.

3.3.4 *De-spreading*

Hasil dari proses demodulasi tersebut akan mengalami proses *de-spreading*. Proses *de-spreading* ini bekerja menggunakan faktor besaran pengali yang dimasukkan oleh pengguna pada proses penyisipan pesan. Proses *de-spreading* ini adalah proses yang dilakukan untuk mendapatkan bit-bit dari pesan tersembunyi, maka hasil keluaran dari proses *de-spreading* ini adalah deret bilangan biner yang telah disusutkan dengan panjang setiap deretnya sebesar 8 bit. Lalu bit-bit tersebut dikonversi kedalam bilangan desimal, yang selanjutnya akan disusun sebagai nilai intensitas tiap *pixel* pada citra rahasia.

4. Pengujian Sistem Dan Analisis Hasil

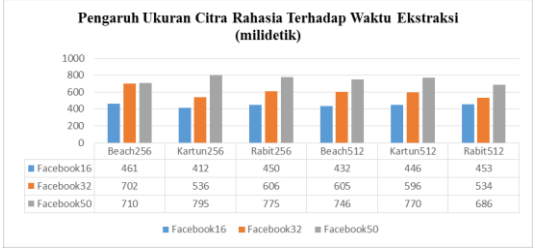
4.1 Analisis Waktu Penyisipan Terhadap Ukuran Citra *Embedded*



Gambar 4.1 Waktu Penyisipan Terhadap Ukuran Citra Rahasia (milidetik)

Berdasarkan data pada Gambar 4.1 ukuran citra rahasia 50x50 *pixel* mempunyai waktu penyisipan yang lebih lama dibanding ukuran citra rahasia yang lain. Dari data tersebut, dapat kita ketahui bahwa semakin besar ukuran citra rahasia yang disisipkan maka semakin lama proses penyisipannya. Hal ini dikarenakan jika ukuran citra rahasia yang akan disisipkan semakin besar maka pada proses *spreading* dan modulasi akan membutuhkan waktu yang lebih lama.

4.2 Analisis Waktu Ekstraksi Terhadap Ukuran Citra *Embedded*

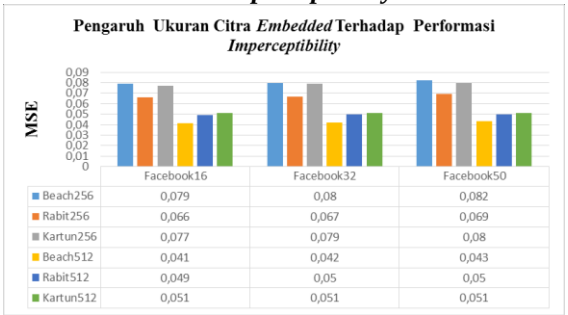


Gambar 4.2 Waktu Ekstraksi Terhadap Ukuran Citra Rahasia(milidetik)

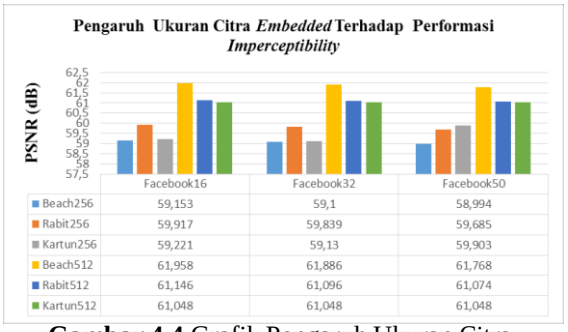
Dari Gambar 4.2 di atas, ukuran citra rahasia 50x50 *pixel* memerlukan waktu yang lebih

lama dibanding dengan ukuran citra rahasia 32x32 *pixel* dan 16x16 *pixel*. Dari data tersebut dapat kita ketahui bahwa semakin besar ukuran citra rahasia yang disisipkan maka waktu ekstraksinya semakin lama. Hal ini terjadi karena sistem membutuhkan waktu untuk membaca posisi bit-bit yang tertanam di dalam citra cover. Jika semakin banyak bit yang tertanam maka sistem juga akan membutuhkan waktu yang semakin lama. Selain itu jika ukuran citra rahasia yang disisipkan semakin besar maka pada proses *de*-spreading dan demodulasi akan membutuhkan waktu yang lebih lama.

4.3 Analisis Pengaruh Ukuran Citra *Embedded* Terhadap Performansi*Imperceptibility*



Gambar 4.3 Grafik Pengaruh Ukuran Citra Embeded Terhadap Nilai MSE

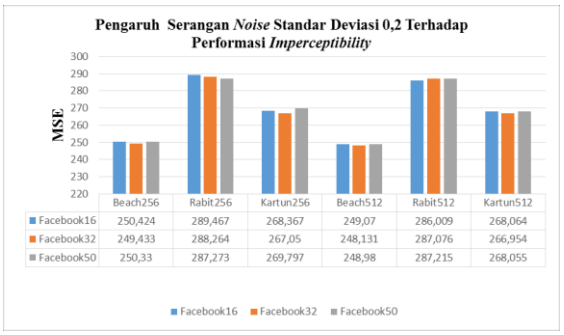


Gambar 4.4 Grafik Pengaruh Ukuran Citra Embeded Terhadap Nilai PSNR

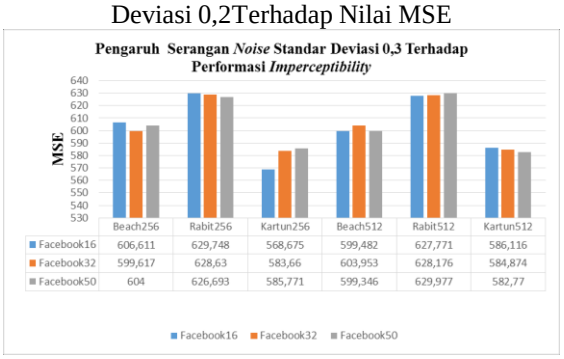
Dari Gambar 4.3 dan 4.4 diatas dapat diketahui bahwa semakin besar ukuran citra rahasia yang disisipkan maka semakin besar tingkat kesalahan/*error* yang terjadi pada citra stego sehingga nilai MSE semakin besar dan nilai PSNR semakin kecil. Hal ini disebabkan karena semakin besar ukuran citra rahasia maka semakin banyak pula bit-bit yang ditanam dalam citra cover sehingga kemiripan citra cover dan citra stego akan semakin kecil. Nilai MSE berbanding tebalik dengan nilai PSNR. Semakin turun nilai PSNR berarti semain turun kualitas suatu citra.

4.4 Analisis Pengaruh Serangan Terhadap Performansi*Imperceptibility*

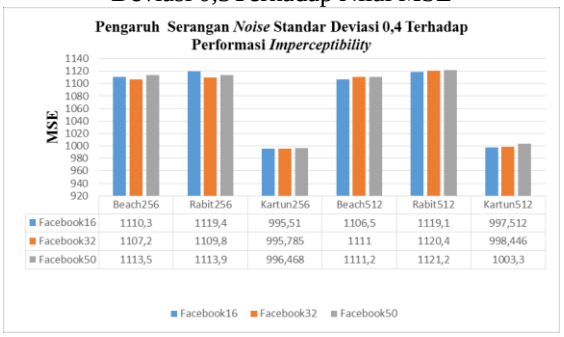
4.4.1 Serangan Noise Guassian



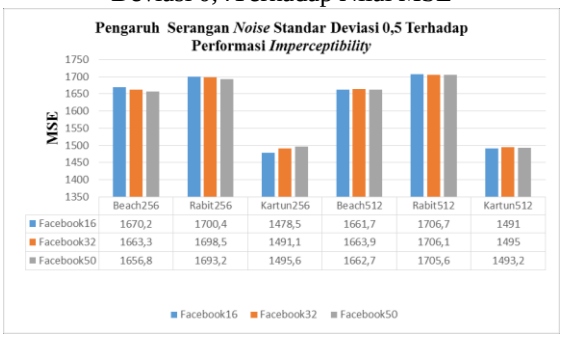
Gambar 4.5 Pengaruh Serangan *Noise* Standar Deviasi 0,2Terhadap Nilai MSE



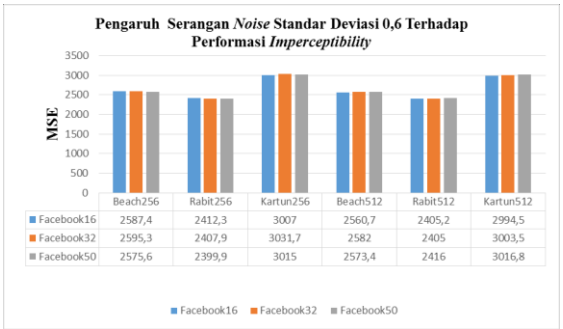
Gambar 4.6 Pengaruh Serangan *Noise* Standar Deviasi 0,3Terhadap Nilai MSE



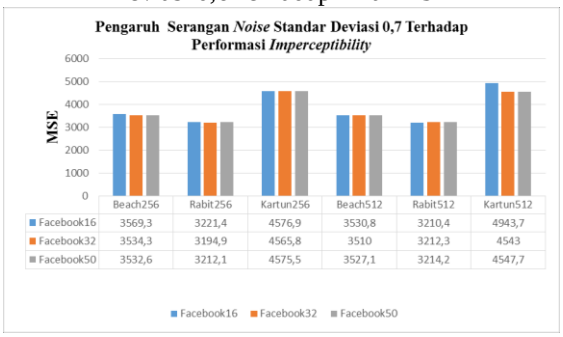
Gambar 4.7 Pengaruh Serangan *Noise* Standar Deviasi 0,4Terhadap Nilai MSE



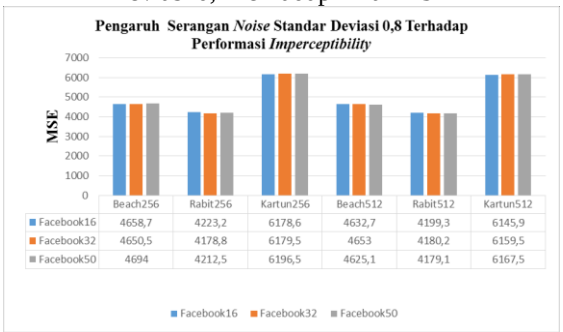
Gambar 4.8 Pengaruh Serangan *Noise* Standar Deviasi 0,5Terhadap Nilai MSE



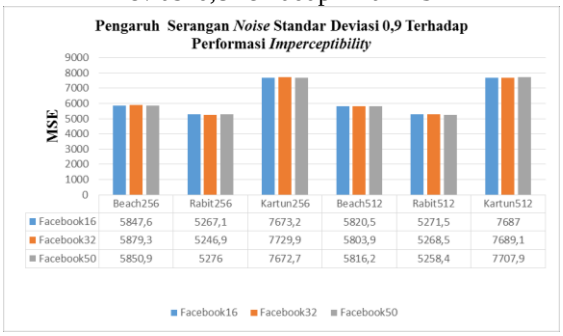
Gambar 4.9 Pengaruh Serangan *Noise* Standar Deviasi 0,6Terhadap Nilai MSE



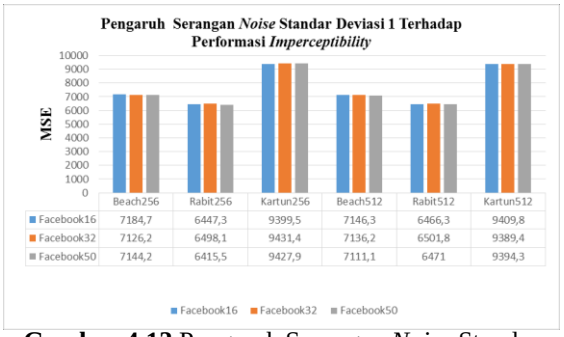
Gambar 4.10 Pengaruh Serangan *Noise* Standar Deviasi 0,7Terhadap Nilai MSE



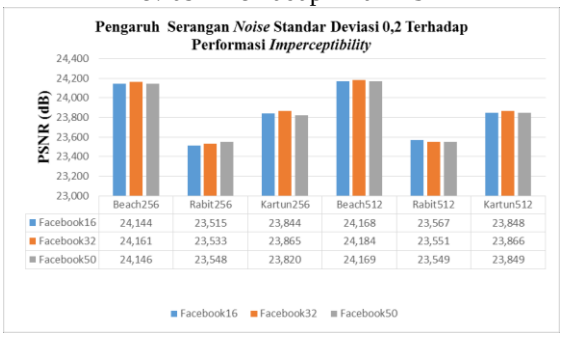
Gambar 4.11 Pengaruh Serangan *Noise* Standar Deviasi 0,8Terhadap Nilai MSE



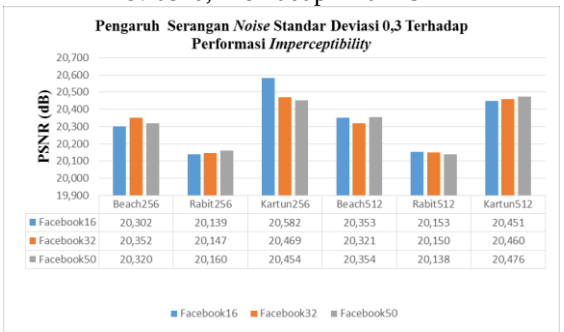
Gambar 4.12 Pengaruh Serangan *Noise* Standar Deviasi 0,9Terhadap Nilai MSE



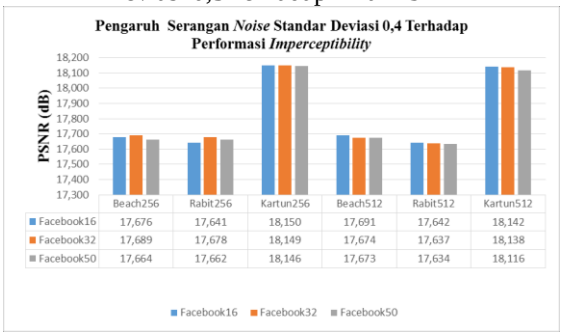
Gambar 4.13 Pengaruh Serangan *Noise* Standar Deviasi 1Terhadap Nilai MSE



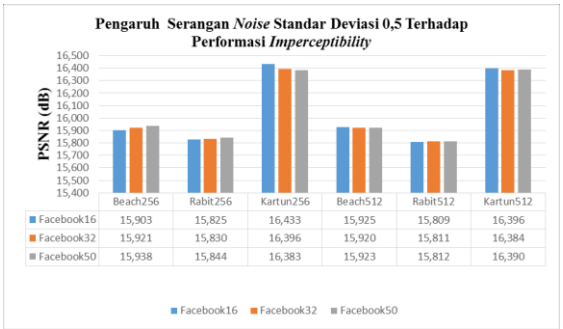
Gambar 4.15 Pengaruh Serangan *Noise* Standar Deviasi 0,2Terhadap Nilai PSNR



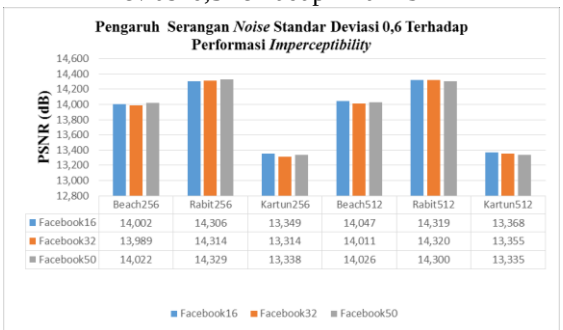
Gambar 4.16 Pengaruh Serangan *Noise* Standar Deviasi 0,3Terhadap Nilai PSNR



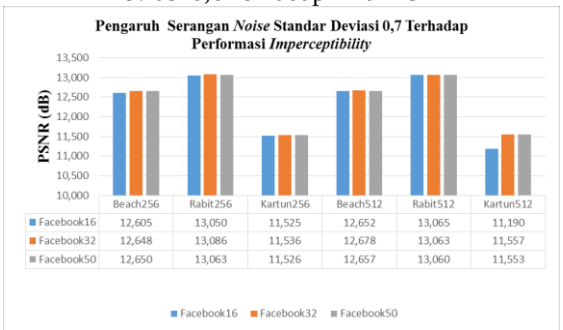
Gambar 4.17 Pengaruh Serangan *Noise* Standar Deviasi 0,4Terhadap Nilai PSNR



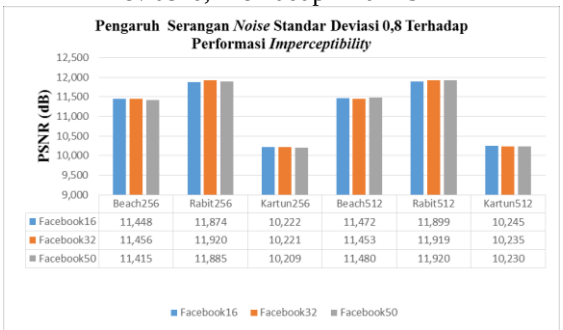
Gambar 4.18 Pengaruh Serangan *Noise* Standar Deviasi 0,5Terhadap Nilai PSNR



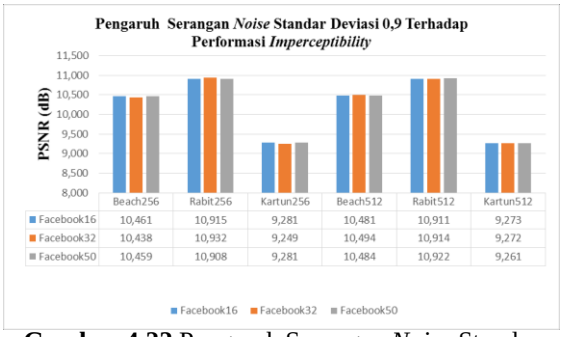
Gambar 4.19 Pengaruh Serangan *Noise* Standar Deviasi 0,6Terhadap Nilai PSNR



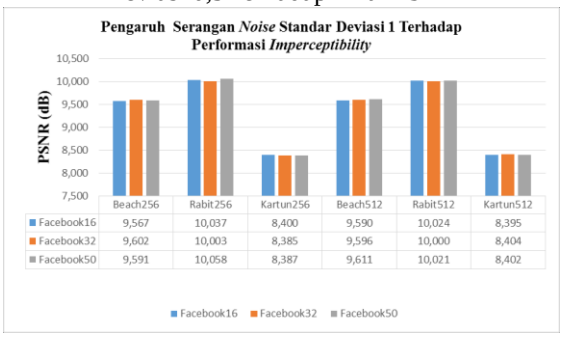
Gambar 4.20 Pengaruh Serangan *Noise* Standar Deviasi 0,7Terhadap Nilai PSNR



Gambar 4.21 Pengaruh Serangan *Noise* Standar Deviasi 0,8Terhadap Nilai PSNR



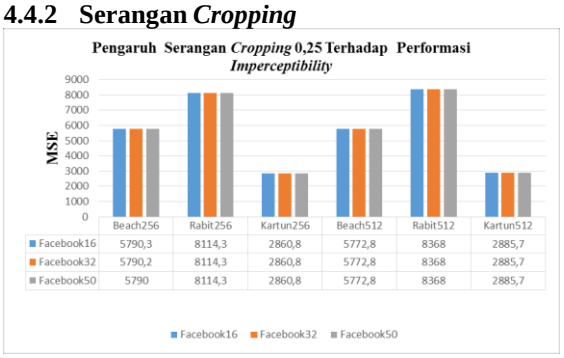
Gambar 4.22 Pengaruh Serangan *Noise* Standar Deviasi 0,9Terhadap Nilai PSNR



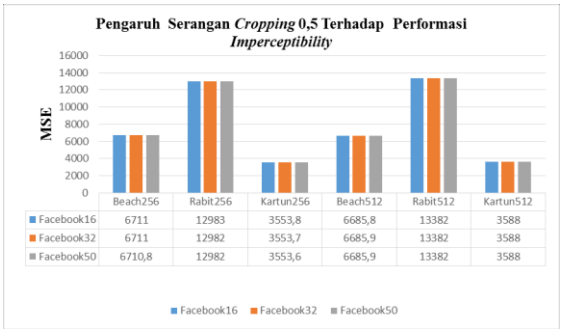
Gambar 4.23 Pengaruh Serangan *Noise* Standar Deviasi 1Terhadap Nilai PSNR

Dalam proses pengujian diberikan *Noise Gaussian* dengan berbagai nilai standar deviasi (sigma) mulai dari 0,2 sampai 1. Terlihat dari Gambar 4.5 sampai 4.8 diatas bahwa nilai sigma 0,2 mempunyai nilai MSE paling kecil dan nilai PSNR paling besar dibandingkan dengan nilai sigma yang lain. Selain itu besar ukuran citra rahasia tidak berpengaruh signifikan terhadap nilai MSE dan PSNR. Semakin besar nilai sigma maka semakin besar pula nilai MSE dan semakin kecil nilai PSNR. Hal ini terjadi karena semakin besar sigma maka penyebaran *noise* semakin luas sehingga menyebabkan nilai *error* yang semakin besar dan nilai PSNR semakin kecil. Selain itu persebaran *noise* dilakukan pada citra stego, sehingga tidak berpengaruh terhadap besar ukuran citra rahasia.

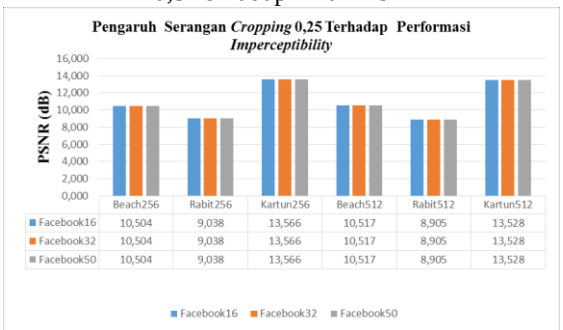
4.4.2 Serangan *Cropping*



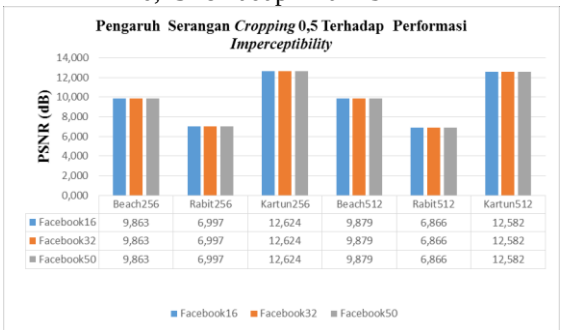
Gambar 4.24 Pengaruh Serangan *Cropping* 0,25Terhadap Nilai MSE



Gambar 4.25 Pengaruh Serangan *Cropping* 0,5Terhadap Nilai MSE



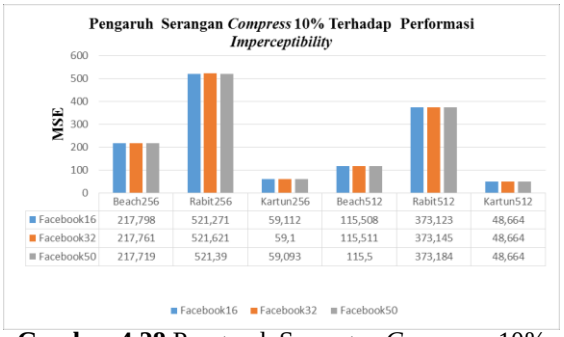
Gambar 4.26 Pengaruh Serangan *Cropping* 0,25Terhadap Nilai PSNR



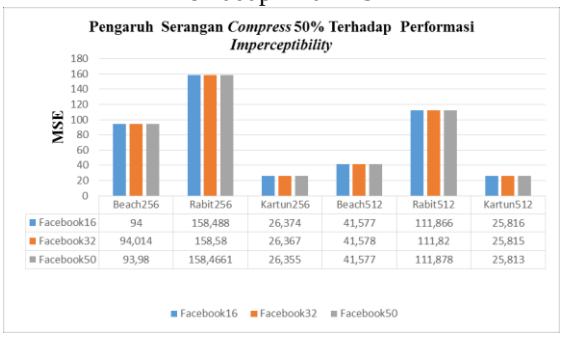
Gambar 4.27 Pengaruh Serangan *Cropping* 0,5Terhadap Nilai PSNR

Proses *cropping* yang dilakukan pada pengujian ini adalah mula-mula meng-*crop* ukuran *pixel* citra sebesar 0,25 atau 0,5 kali ukuran *pixel* citra stego, kemudian dikembalikan lagi ke ukuran semula dengan melakukan proses *resize*. Pengujian dilakukan untuk membuktikan seberapa besar pengaruh proses *crop* tersebut terhadap nilai MSE dan PSNR. Pada serangan *cropping* dengan *cropping* 0,25 kali nilai MSE yang lebih kecil dan nilai PSNR yang lebih besar dibandingkan dengan serangan *cropping* dengan *cropping* 0,5 kali. Hal ini terjadi karena semakin besar ukuran *rasiocropping* maka semakin besar nilai MSE sehingga semakin kecil pula nilai PSNR. Tentunya karena citra terkena *crop* maka terdapat *pixel* yang hilang dari citra stego. Selain itu besar ukuran citra rahasia tidak berpengaruh terhadap nilai MSE dan PSNR. Proses *cropping* dilakukan pada citra stego, sehingga tidak berpengaruh terhadap besar ukuran citra rahasia.

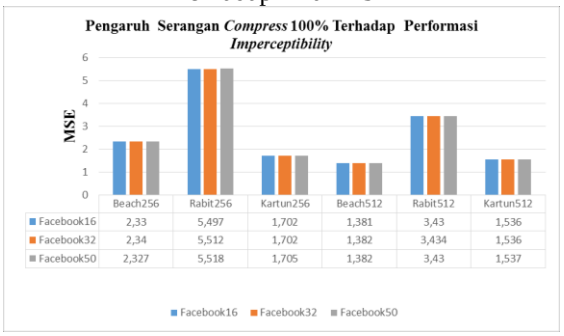
4.4.3 Serangan *Compress*



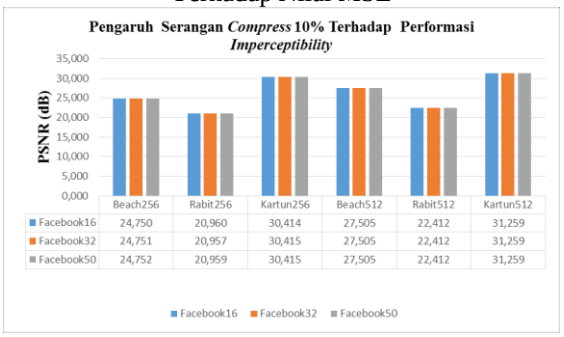
Gambar 4.28 Pengaruh Serangan *Compress* 10% Terhadap Nilai MSE



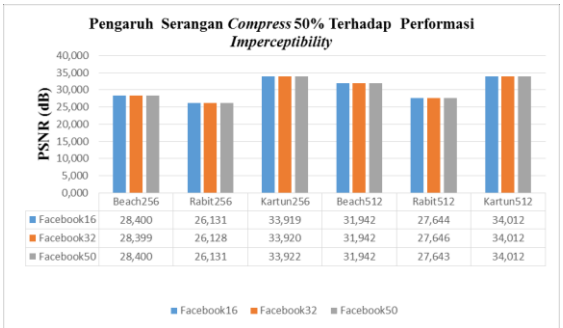
Gambar 4.29 Pengaruh Serangan *Compress* 50% Terhadap Nilai MSE



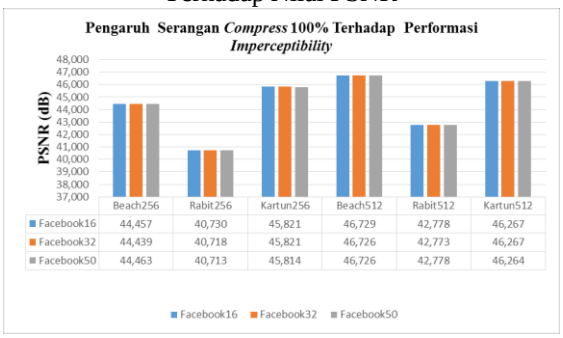
Gambar 4.30 Pengaruh Serangan *Compress* 100% Terhadap Nilai MSE



Gambar 4.31 Pengaruh Serangan *Compress* 10% Terhadap Nilai PSNR



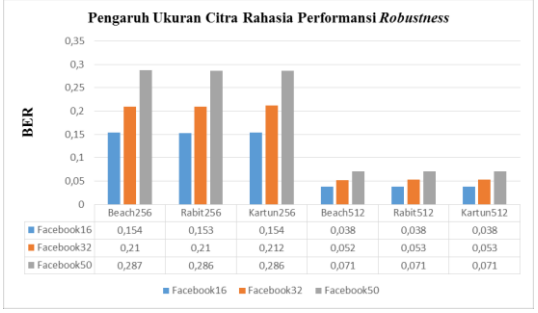
Gambar 4.32 Pengaruh Serangan *Compress* 50% Terhadap Nilai PSNR



Gambar 4.33 Pengaruh Serangan *Compress* 100% Terhadap Nilai PSNR

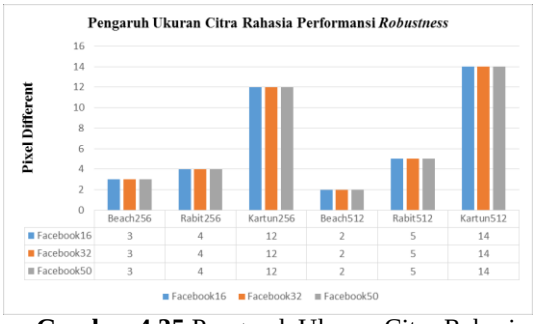
Proses *compress* yang dilakukan pada pengujian ini adalah dengan meng-*compress* citra berdasarkan kualitas kompresinya yaitu sebesar 10%, 50%, dan 100%. Pengujian dilakukan untuk membuktikan seberapa besar pengaruh perubahan proses *compress* tersebut terhadap nilai MSE dan PSNR. Pada serangan *compress* dengan kualitas 10% mempunyai nilai MSE yang lebih besar dan nilai PSNR yang lebih kecil dibandingkan dengan serangan *compress* dengan kualitas 50% dan 100%. Tentunya karena citra telah terkena serangan *compress* maka terdapat *pixel* yang berubah dari citra stego. Selain itu besar ukuran citra rahasia tidak berpengaruh signifikan terhadap nilai MSE dan PSNR. Hal ini terjadi karena semakin besar kualitas kompresinya maka semakin kecil nilai MSE sehingga semakin besar pula nilai PSNR. Selain itu proses *cropping* dilakukan pada citra stego, sehingga tidak berpengaruh signifikan terhadap besar ukuran citra rahasia.

4.5 Analisis Pengaruh Ukuran Citra *Embedded* Terhadap Performansi*Robustness*



Gambar 4.34 Pengaruh Ukuran Citra Rahasia Terhadap Nilai BER

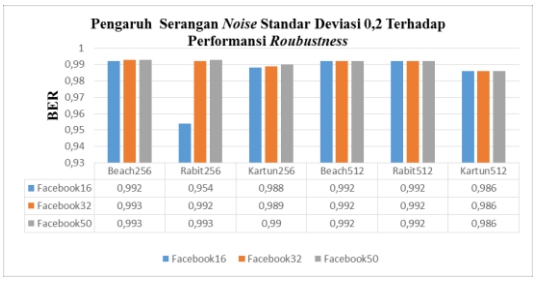
Berdasarkan data pada Gambar 4.34 ukuran citra rahasia 50x50 *pixel* mempunyai nilai BER yang lebih besar di banding ukuran citra rahasia yang lain. Dari data tersebut, dapat kita ketahui bahwa semakin besar ukuran citra rahasia yang disisipkan maka semakin besar nilai BER yang dihasilkan. Selain itu nilai BER yang dihasilkan akan semakin kecil jika citra cover yang digunakan semakin besar. Hal ini dikarenakan pada proses penyisipan terjadi perubahan nilai *pixel* yang semakin besar apabila ukuran citra yang disisipkan (citra rahasia) yang semakin besar. Selain itu semakin besar ukuran citra cover maka akan mengakibatkan jumlah *pixel* yang tidak disisipkan semakin besar.



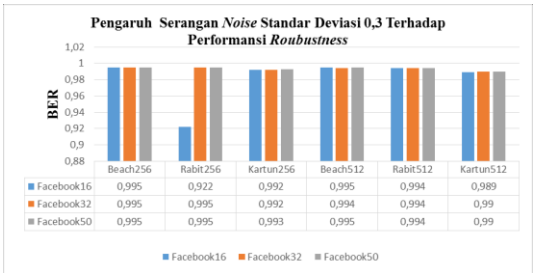
Gambar 4.35 Pengaruh Ukuran Citra Rahasia Terhadap Nilai *PixelDifferent*

Berdasarkan data pada Gambar 4.35 ukuran citra rahasia tidak berpengaruh terhadap nilai *pixel different* yang dihasilkan. Hal ini dikarenakan nilai maksimum perbedaan *pixel* antara citra cover dengan citra stego yang didapat hanya berpengaruh terhadap besar ukuran dan nilai intensitas tiap *pixel*.

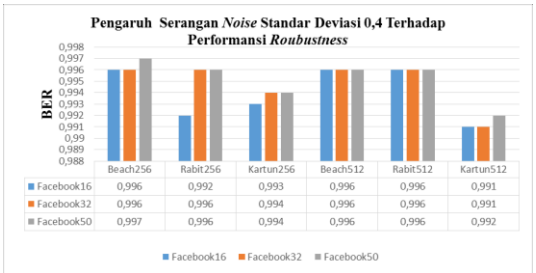
4.6 Analisis Pengaruh Serangan Terhadap Performansi*Robustness*



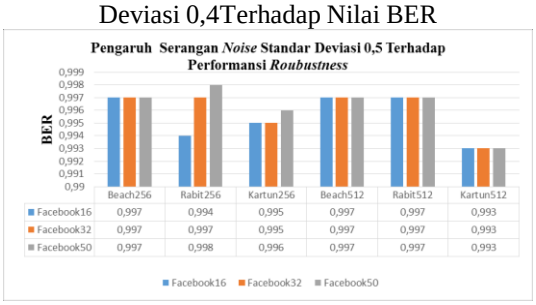
Gambar 4.36 Pengaruh Serangan *Noise* Standar Deviasi 0,2Terhadap Nilai BER



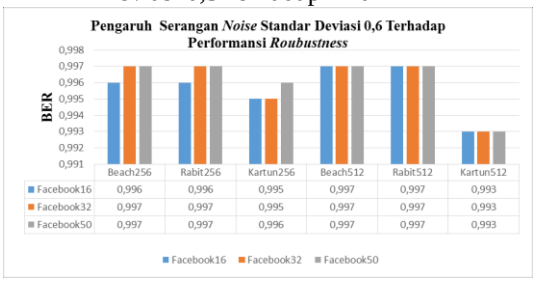
Gambar 4.37 Pengaruh Serangan *Noise* Standar Deviasi 0,3Terhadap Nilai BER



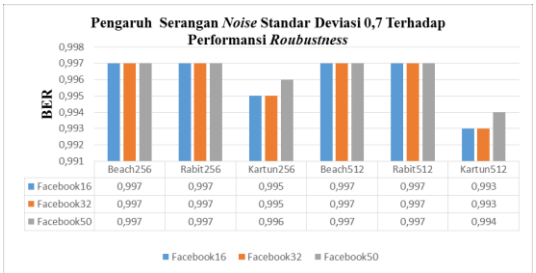
Gambar 4.38 Pengaruh Serangan *Noise* Standar Deviasi 0,4Terhadap Nilai BER



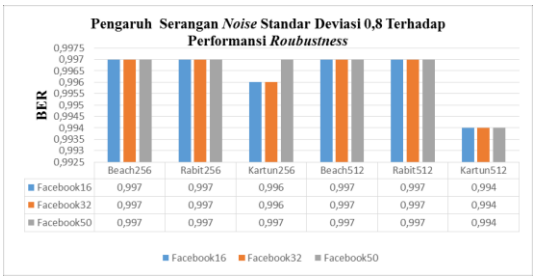
Gambar 4.39 Pengaruh Serangan *Noise* Standar Deviasi 0,5Terhadap Nilai BER



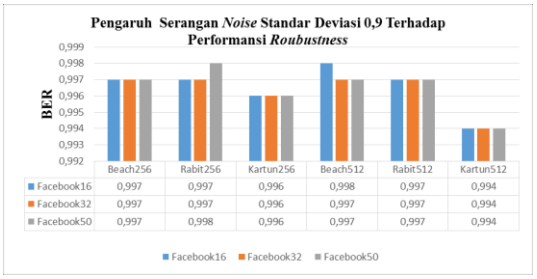
Gambar 4.40 Pengaruh Serangan *Noise* Standar Deviasi 0,6Terhadap Nilai BER



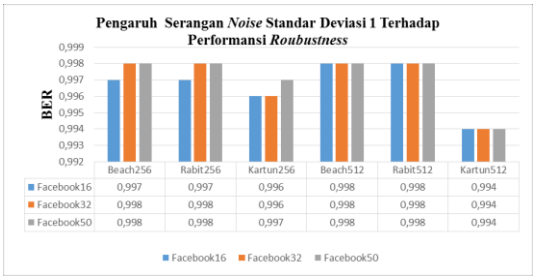
Gambar 4.41 Pengaruh Serangan *Noise* Standar Deviasi 0,7Terhadap Nilai BER



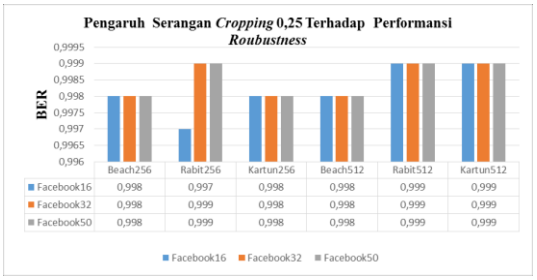
Gambar 4.42 Pengaruh Serangan *Noise* Standar Deviasi 0,8Terhadap Nilai BER



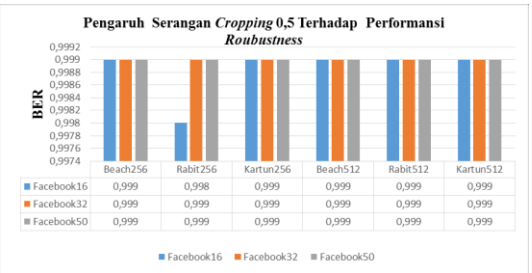
Gambar 4.43 Pengaruh Serangan *Noise* Standar Deviasi 0,9Terhadap Nilai BER



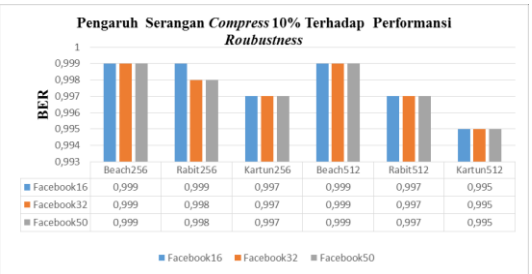
Gambar 4.44 Pengaruh Serangan *Noise* Standar Deviasi 1Terhadap Nilai BER



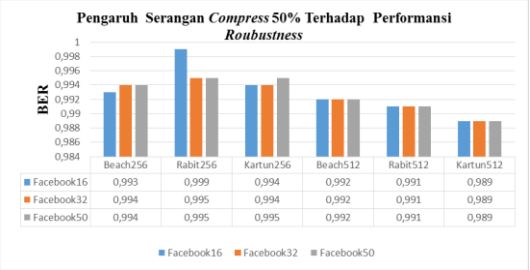
Gambar 4.45 Pengaruh Serangan *Cropping* 0,25Terhadap Nilai BER



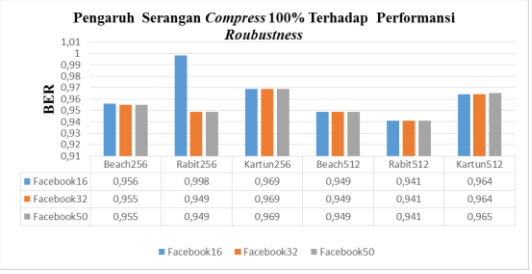
Gambar 4.46 Pengaruh Serangan *Cropping* 0,5Terhadap Nilai BER



Gambar 4.47 Pengaruh Serangan *Compress* 10%Terhadap Nilai BER



Gambar 4.48 Pengaruh Serangan *Compress* 50%Terhadap Nilai BER

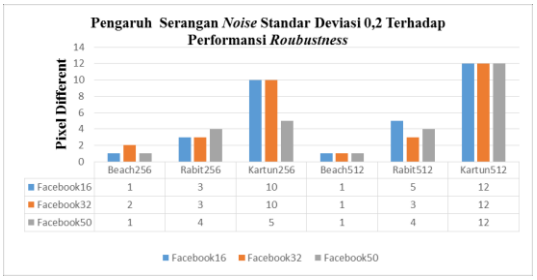


Gambar 4.49 Pengaruh Serangan *Compress* 100%Terhadap Nilai BER

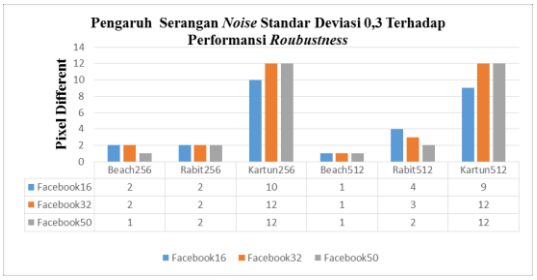
Pada Gambar diatas menunjukkan bahwa ketika sistem mendapat serangan *noisegaussian* dengan sigma sebesar 0,2 mempunyai rata-rata nilai BER sebesar 0,988. Namun pada saat sigma sebesar 0,3 rata-rata nilai BER yang didapat lebih besar dari 0,989. Semakin besar nilai sigma semakin besar pula nilai BER, hal ini terjadi karena semakin besar nilai sigma maka semakin luas pula penyebaran *noise* sehingga membuat sistem tidak tahan terhadap serangan.

Ketika sistem mendapat serangan *cropping* dengan *cropping* sebesar 0,25 kali mempunyai nilai BER terkecil sebesar 0,998 dan yang terbesar sebesar 0,999. Namun pada *cropping* sebesar 0,5 kali mempunyai nilai BER terkecil sebesar 0,998 dan yang terbesar sebesar 0,999. Sehingga dapat disimpulkan bahwa sistem tidak tahan tehadap serangan. Hal ini ditunjukkan dengan besarnya nilai BER yang di dapat maka dapat disimpulkan bahwa sistem tidak tahan terhadap serangan *cropping* dengan cara meng-*crop* ukuran *pixel* citra sebesar 0,25 atau 0,5 kali ukuran *pixel* citra stego.

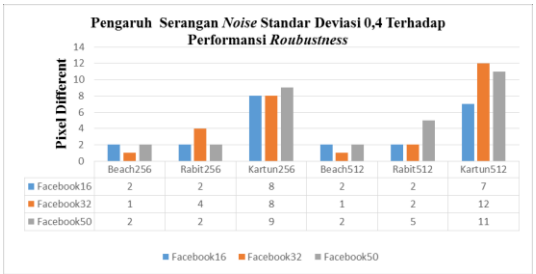
Pada saat sistem mendapat serangan *compress* dengan cara me meng-*compress* citra berdasarkan kualitas kompresi sebesar 10% mempunyai nilai BER terkecil sebesar 0,995 dan yang terbesar sebesar 0,999. Sedangkan kualitas kompresi sebesar 50% mempunyai nilai BER terkecil sebesar 0,989 dan yang terbesar sebesar 0,999. Sedangkan kualitas kompresi sebesar 100% mempunyai nilai BER terkecil sebesar 0,941 dan yang terbesar sebesar 0,998. Sehingga dapat disimpulkan bahwa sistem tidak tahan tehadap serangan. Hal ini ditunjukkan dengan besarnya nilai BER yang di dapat maka dapat disimpulkan bahwa sistem tidak tahan terhadap serangan *compress* dengan cara meng-*compress* citra berdasarkan kualitas kompresinya.



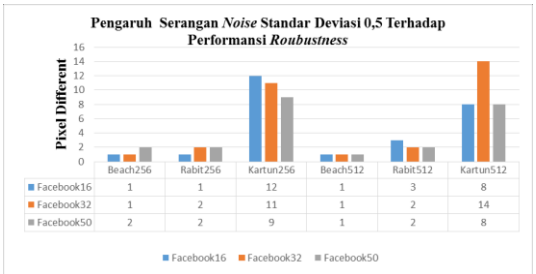
Gambar 4.50 Pengaruh Serangan *Noise* Standar Deviasi 0,2Terhadap Nilai *Pixel Different*



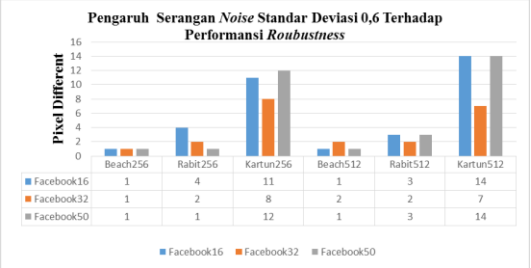
Gambar 4.51 Pengaruh Serangan *Noise* Standar Deviasi 0,3Terhadap Nilai *Pixel Different*



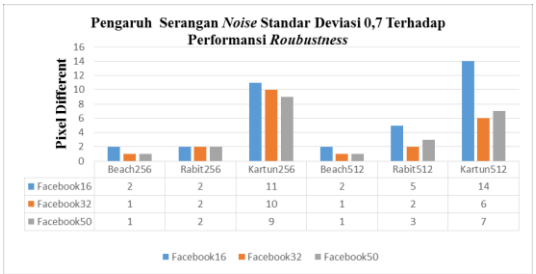
Gambar 4.52 Pengaruh Serangan *Noise* Standar Deviasi 0,4Terhadap Nilai *Pixel Different*



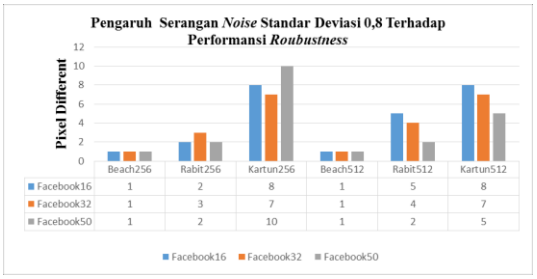
Gambar 4.53 Pengaruh Serangan *Noise* Standar Deviasi 0,5Terhadap Nilai *Pixel Different*



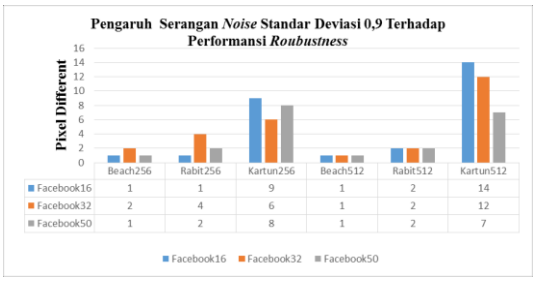
Gambar 4.54 Pengaruh Serangan *Noise* Standar Deviasi 0,6Terhadap Nilai *Pixel Different*



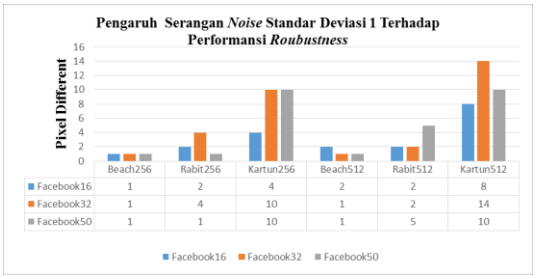
Gambar 4.55 Pengaruh Serangan *Noise* Standar Deviasi 0,7Terhadap Nilai *Pixel Different*



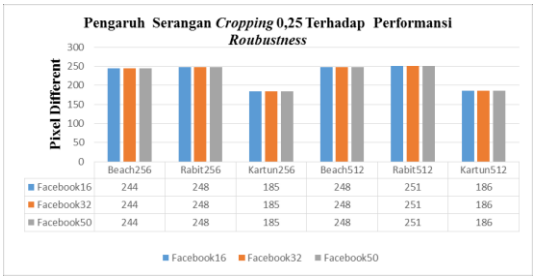
Gambar 4.56 Pengaruh Serangan *Noise* Standar Deviasi 0,8Terhadap Nilai *Pixel Different*



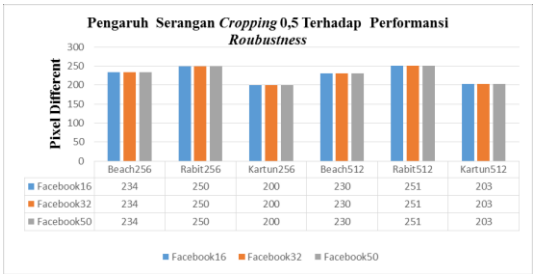
Gambar 4.57 Pengaruh Serangan *Noise* Standar Deviasi 0,9Terhadap Nilai *Pixel Different*



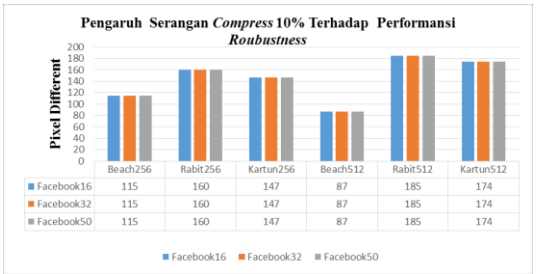
Gambar 4.58 Pengaruh Serangan *Noise* Standar Deviasi 1Terhadap Nilai *Pixel Different*



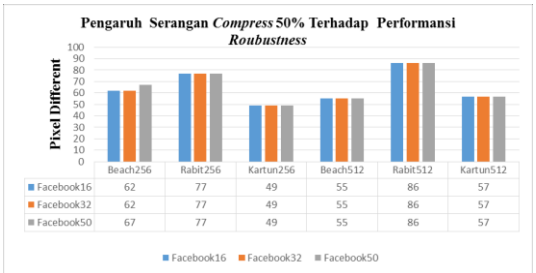
Gambar 4.59 Pengaruh Serangan *Cropping* 0,25Terhadap Nilai *Pixel Different*



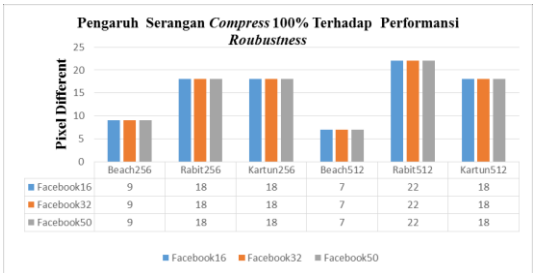
Gambar 4.60 Pengaruh Serangan *Cropping* 0,5Terhadap Nilai *Pixel Different*



Gambar 4.61 Pengaruh Serangan *Compress* 10%Terhadap Nilai *Pixel Different*



Gambar 4.62 Pengaruh Serangan *Compress* 50%Terhadap Nilai *Pixel Different*



Gambar 4.63 Pengaruh Serangan *Compress* 100%Terhadap Nilai *Pixel Different*

Pada Gambardiatasmenunjukkan bahwa ketika sistem mendapat serangan *noisegaussian* dengan sigma sebesar 0,2 mempunyai rata-rata nilai *pixel different* sebesar 5,73. Namun pada saat sigma sebesar 0,3 rata-rata nilai *pixel different* yang didapat lebih besar dari 5,67. Sistem tidah tahan terhadap serangan ini walaupun nilai *pixel different* yang didapat cukup kecil, hal ini terjadi karena semakin besar nilai sigma maka semakin luas pula penyebaran *noise* sehingga membuat sistem tidak tahan terhadap serangan.

Ketika sistem mendapat serangan *cropping* dengan *cropping* sebesar 0,25 kali mempunyai nilai *pixel different* terkecil sebesar 185 dan yang terbesar sebesar 251. Namun pada *cropping* sebesar 0,5 kali mempunyai nilai *pixel different* terkecil sebesar 200 dan yang terbesar sebesar 251. Sehingga dapat disimpulkan bahwa sistem tidak tahan tehadap serangan. Hal ini ditunjukkan dengan besarnya nilai *pixel different* yang didapat maka dapat disimpulkan bahwa sistem tidak tahan terhadap serangan *cropping* dengan cara meng-*crop* ukuran *pixel* citra sebesar 0,25 atau 0,5 kali ukuran *pixel* citra stego.

Pada saat sistem mendapat serangan *compress* dengan cara me meng-*compress* citra berdasarkan kualitas kompresi sebesar 10% mempunyai nilai *pixel different* terkecil sebesar 87 dan yang terbesar sebesar 185. Sedangkan kualitas kompresi sebesar 50% mempunyai nilai *pixel different* terkecil sebesar 49 dan yang terbesar sebesar 86. Sedangkan kualitas kompresi sebesar 100% mempunyai nilai *pixel different* terkecil sebesar 7 dan yang terbesar sebesar 22. Sehingga dapat disimpulkan bahwa sistem tidak tahan tehadap serangan. Hal ini ditunjukkan dengan besarnya nilai *pixel different* yang di dapat maka dapat disimpulkan bahwa sistem tidak tahan terhadap serangan *compress* dengan cara meng-*compress* citra berdasarkan kualitas kompresinya.

4.6 Analisis Pengujian BerdasarkanNilai *Mean Opinion Score*(MOS)

Selain pengukuran yang bersifat objektif dilakukan juga pengukuran subjektif yaitu berdasarkan opini dari pengamat menggunakan MOS. Ada 30 orang yang di jadikan sebagai pengamat dalam penilaian opini MOS ini.

Tabel 4.1 Penilaian MOS

Citra Rahasia		Facebook16	Facebook32	Facebook50
Citra Stego		4,589	4,594	4,572
Noise	0,2	2,706	2,706	2,656
	0,3	2,133	2,111	2,156
	0,4	1,906	1,878	1,928
	0,5	1,594	1,617	1,611
	0,6	1,550	1,556	1,528
	0,7	1,450	1,444	1,444
	0,8	1,444	1,439	1,430
	0,9	1,272	1,294	1,285

	1	1,211	1,194	1,201
Cropping	0,25	3,117	3,106	3,128
	0,5	2,389	2,394	2,394
Compress	10%	2,467	2,467	2,489
	50%	3,833	3,944	3,917
	100 %	4,539	4,583	4,572

Dari tabel 4.1 diatas dapat diketahui bahwa *imperceptibility* citra stego masih dapat diterima oleh penglihatan manusia, hal ini dapat dilihat dari hasil penilaian koresponden yang menilai citra stego memiliki nilai > 4,5. Hal ini sesuai dengan kriteria penilaian MOS, jika suatu citra diberi nilai 4 maka citra tersebut terdapat perbedaan dengan citra aslinya namun tidak mengganggu.

Pada saat citra stego di beri serangan berupa *rotate* sebesar 90, 180, dan 270 derajat, penilaian citra stego berada pada *range* 4,5≤ citra stego ≤ 4,6 yang berarti gambar bagus. Hal ini sesuai dengan nilai PSNR citra stego yang mendapat serangan *rotate* hasilnya sama dengan citra stego tanpa serangan dan mempunyai kualitas gambar yang bagus.

Pada saat citra stego di beri serangan berupa *noise* dengan *standar deviasi* sebesar 0,2 sampai dengan 1, mempunyai nilai MOS < 2,7. Sesuai dengan kriteria penilaian maka citra stego tersebut terlihat jelek dan banyak gangguan dibandingkan dengan citra aslinya.

Pada saat citra stego di beri serangan berupa *cropping* , mempunyai nilai MOS rata-rata < 3,2. Sesuai dengan kriteria penilaian maka citra stego tersebut terlihat jelek dan banyak gangguan dibandingkan dengan citra aslinya, karena gambar telah terpotong.

Pada saat citra stego di beri serangan berupa *compress* sebesar 10%, penilaian citra stego berada pada range 2,46 ≤ citra stego ≤ 2,48 yang berarti gambar jelek. Berbeda ketika di beri serangan *compress* sebesar 50% penilaian berada pada *range* 3,83 ≤ citra stego ≤ 3,94dan pada saat mendapat serangan *compress* sebesar 10% memiliki nilai > 4,5. Hal ini sesuai dengan nilai PSNR citra stego yang mendapat serangan *compress* sebesar 10% lebih kecil dibanding dengan yang lain,dan mempunyai kualitas gambar paling jelek dari yang lain.

4. Kesimpulan dan Saran

5.1 Kesimpulan

Dari pengujian dan analisis sistem yang telah dilakukan, maka dapat diambil kesimpulan sebagai berikut :

1. Telah berhasil dikembangkan perangkat lunak yang dapat melakukan steganografi pada citra berwarna. Kebutuhan fungsional dari perangkat lunak, seperti proses penyisipan dan ekstraksi pesan sudah dapat dilakukan dengan benar.
2. Ukuran citra rahasia yang disisipkan pada citra cover mempengaruhi waktu penyisipan dan ekstraksi. Semakin besar ukuran citra rahasia yang disisipkan maka semakin lama waktu yang diperlukan untuk penyisipan dan ekstraksi. Pada citra cover Beach256 yang disisipi citra rahasia Facebook50 mempunyai waktu penyisipan terlama yaitu sebesar 547 milidetik.
3. Ukuran citra rahasia yang disisipkan mempengaruhi kualitas citra stego. Semakin besar ukuran citra rahasia yang disisipkan maka semakin besar pula nilai MSE dan BER yang didapat sehingga kualitas citra stego semakin turun. Pada citra cover Beach256 yang disisipi citra rahasia Facebook50 mempunyai nilai MSE terbesar yaitu 0,082 dan PSNR terkecil yaitu58,994 dB.
4. Citra stego tidak tahan terhadap serangan *noisegaussian*, serangan ini pada citra stego di pengaruhi oleh nilai standar deviasi (sigma). Semakin besar nilai sigma maka semakin besar pula nilai MSE dan BER sehingga tingkat kesalahan/*error* semakin besar. Pada citra cover Kartun256 yang disisipi citra rahasia Facebook32 dengan diberikan serangan *noise* standar deviasi 1 mempunyai nilai MSE terbesar dibanding standar deviasi lainnya yaitu 9431,4 dan PSNR terkecil dibanding standar deviasi lainnya yaitu 8,385 dB.
5. Citra stego tidak tahan terhadap serangan *cropping* dengan meng-*crop* ukuran *pixel* citra sebesar 0,25 atau 0,5 kali ukuran *pixel* citra stego, kemudian dikembalikan lagi dengan cara melakukan proses *resize* ke ukuran semula.Semakin besar ukuran *rasiocropping* maka semakin besar nilai MSE dan BER sehingga semakin kecil pula nilai PSNR.Pada citra cover Rabit512 yang disisipi citra rahasia Facebook16 dengan diberikan serangan *cropping* 0,5 mempunyai nilai MSE lebih besar dibanding *rasiocropping* 0,25 yaitu 13382 dan PSNR lebih kecil dibanding *rasiocropping* 0,25 yaitu6,866 dB.
6. Citra stego tidak tahan terhadap serangan *compress* dengan cara meng-*compress* citra berdasarkan kualitas kompresinya yaitu sebesar 10%, 50%, dan 100%. Semakin besar kualitas kompresinya maka semakin kecil nilai MSE dan BER sehingga semakin besar pula nilai PSNR. Pada citra cover Rabit256 yang disisipi citra rahasia Facebook32 dengan diberikan serangan *compress* 10% mempunyai nilai MSE terbesar dibanding kualitas kompresi lainnya yaitu521,621 dan PSNR terkecil dibanding kualitas kompresi lainnya yaitu20,957 dB.

5.2 Saran

Adapun saran untuk pengembangan tugas akhir selanjutnya adalah:

1. Menggunakan algoritma steganografi *Spread Spectrum* yang di padukan dengan teknik kriptografi agar pesan yang disisipkan lebih terjamin tingkat keamanannya. Oleh karena itu untuk kedepannya sistem ini dapat diterapkan dan dikembangkan lagi agar kemanan pesan yang disisipkan lebih terjamin.
2. Memilih metode penyisipan lain yang dikombinasi dengan algoritma BPCS **(**Bit-Plane Complexity Segmentation) agar ukuran pesan yang dapat disisipkan lebih besar dari kemampuan algoritma yang sudah ada.
3. Menggunakan data dengan format yang lain, misalnya audio atau video.
4. Membuat agar aplikasi bisa digunakan secara *realtime*, yaitu dengan menggunakan kamera *handphone*.

DAFTAR PUSTAKA

[1] Cole, Eric. 2003. *Hiding in Plainsight : Steganography and the Art of Covert Communication*. Wiley Publishing,Inc.

[2] Torrieri, Don. 2005. *Principles of Spread Spectrum Communications System*. Springer.

[3] Morkel, T., JHP. Eloff, dan MS. Olivier. *An Overview of Image Steganography*. Pretoria. Information and Computer Security Architecture (ICSA) Research Group, Department of Computer Science, University of Pretoria.

[4] Marvel, Lisa M., Charles G. Boncelet, dan Charles T. Retter. 1999. *Spread Spectrum Image Steganography*. IEEE Transaction on Image Processing.

[5] Permatasari, Novia. 2013. *Aplikasi Steganografi Citra Dengan Format JPEG Menggunakan Metode Spread Spectrum Pada Smartphone Berbasis Android 4.0*. Jakarta. Universitas Gunadarma.

[6] Winanti, Winda. 2008. *Penyembunyian Pesan pada Citra Terkompresi JPEG Menggunakan Metode Spread Spectrum*. Bandung. Institut Teknologi Bandung.

[7] M.A. Ineke Pakereng, Yos Richard Beeh, dan Sonny Endrawan. 2010. *Perbandingan Steganografi Metode Spread Spectrum dan Least Significant Bit (LSB) Antara Waktu Proses dan Ukuran File Gambar*. Yogyakarta. Tugas Akhir UK Duta Wacana

[8] Munir, Rinaldi. 2004. *Steganografi dan Watermarking*. Bandung. Institut Teknologi Bandung.

[9] Vembrina, Yus Gias . 2006. *Spread Spectrum Steganography*, Bandung. Institut Teknologi Bandung.

[10] Wenny. 2011. *Perancangan Program Aplikasi Steganography pada Digital Video Berbasis Metode Singular Value Decomposition dan Discreate Wavelet Transform*. Jakarta. Universitas Bina Nusantara.

[11] Cox, Ingemar J. 2008. *Digital Watermarking and Steganography*. Burlington. Morgan Kaufmann Publisher.

[12] Pratiarso, Aries., Yuliana, Mike., Hadi, M. Zen Samsono., Bari, Fatchul., Brahim. 2012. *Analisa PSNR Pada Teknik Steganografi Menggunakan Spread Spectrum*. Surabaya.Politeknik Elektronika Negeri Surabaya.

[13] M. Alam, W. Badawy, and J. Graham, "A new time distributed DCT architecture for MPEG-4 hardware reference model," IEEE Transactions on Circuits and Systems for Video Technology, vol. 15, no. 5, pp. 726-730, May 2005.

[14] M. A. Mohamed, M. E.-D. A. Abou-Soud, and M. S. Diab, "Fast Digital Watermarking Techniques for Still Images," International Conference on Networking and Media Convergence, pp. 122-129, Mar. 2009.

[15] F. C. A. Fernandes, R. L. C. van Spaendonck, and C. S. Burrus, "A New Framework for Complex Wavelet Transforms," IEEE Transactions on Signal Processing, vol. 51, no. 7, pp. 1825-1837, Jul. 2003.