Aplikasi Perlindungan Hak Cipta Digital dengan Kriptografi dan Stenografi

DIGITAL COPYRIGHT PROTECTION APPLICATION USING CRYPTOGRAPHY AND STENOGRAPHY

Alfred Tenggono*[1], Ronal Fernando Simamora[2], Setia Budi[3], Steven Theodorus[4]
[1,2,3,4]STMIK PalComTech; Jln.Basuki Rahmat No.05, Telp:0711-358916, Fax:0711-359089
[1,2,3,4]Program Studi Teknik Informatika STMIK PalComTech Palembang
e-mail: *[1]alfred.tenggono@gmail.com

Abstrak

Seiring berkembanganya teknologi banyak karya seni yang dihasilkan para seniman dalam bentuk digital. Karya ini dihasilkan dari perangkat digital yang memungkinkan penciptaan citra, baik kamera digital, aplikasi pengolahan gambar, maupun hasil gambar/lukisan yang langsung dituangkan ke media digital.Citra digital yang dihasilkan tentunya memiliki hak cipta yang melekat pada citra tersebut. Perlu adanya perlindungan terhadap karya citra digital dikarenakan bentuk penyimpanan karya citra digital yang rentan terhadap pembajakan, klaim pihak tidak berwenang, pengandaan secara ilegal, ataupun modifikasi yang tidak berizin. Pengamanan citra digital dapat dilakukan dengan menggabungkan teknik kriptografi dan stegnografi. Kedua teknik ini memungkinkan pemberian ciri pengaman (watermark) untuk melindungi citra digital. Untuk itu dibutuhkan aplikasi yang dapat melindungi citra digital dari klaim (plagiarisme) yang dilakukan. Penulis mencoba untuk menggunakan penggabungan Algoritma Rivest Code 4 dan metode Least Significant Bit. Dimana teknik ini diyakini mampu melindungi citra digital lebih cepat daripada metode sebelumnya seperti RSA. Dimana aplikasi ini diharapkan dapat membantu melindungi hak cipta para pelaku industri kreatif khususnya yang menghasilkan karya berupa citra digital.Dari penelitian ini dihasilkan perangkat lunak yang digunakan untuk menyembunyikan tanda pengenal rahasia di dalam sebuah citra digital dan menggacak gambar tanda pengenal digital (watermark) sehingga tidak dapat dikenali.
***Kata kunci**—Hak Cipta,Kriptografi, Stenografi, Watermark, Citra Digital*

Abstract

Along with the development of technology, many artists produced art works in digital form. This work generated from digital devices that allow the creation of the image, either a digital camera, image processing applications, as well as the results of pictures/paintings are instantly created in digital media. The resulting digital image is certainly out of copyright attached to the image. protection of works of the digital image is required, due to the form of storage of digital images of works that are prone to piracy, unauthorized parties claim, pengandaan illegally, or modifications that are not licensed. Safeguards digital image can be done by combining the techniques of cryptography and stegnografi. Both of these techniques allows the granting of safety characteristics (watermark) to protect digital images. Required application that can protect the digital imagery of claims (plagiarism). The author tries to use the Merge Algorithm Rivest Code 4 and the method of Least Significant bits. Where this technique is believed to be able to protect digital images faster than previous methods such as RSA. Where the application is expected to help protect copyrights the perpetrators of particularly creative industries that produce the work in the form of a digital image. This research generated from software used to hide secret identification in a digital image and digital image ID menggacak (watermark) so as not to be recognised.
***Keywords**—Mobile Application, Google Maps, Food, Android*

1. PENDAHULUAN

Banyak karya seni yang dihasilkan pada seniman sekarang ini dalam bentuk digital. Karya ini dihasilkan dari perangkat digital yang memungkinkan penciptaan citra, baik kamera digital, aplikasi pengolahan gambar, maupun hasil gambar/lukisan yang langsung dituangkan ke media digital. Citra adalah suatu representasi (gambaran), kemiripan, atau imitasi dari suatu objek. Citra terbagi 2 yaitu ada citra yang bersifat analog dan ada citra yang bersifat digital. Citra analog adalah citra yang bersifat kontinu seperti gambar pada monitor televisi, foto sinar X, hasil CT Scan dan lain-lain. Sedangkan pada citra digital adalah citra yang dapat diolah oleh komputer [1]. Citra digital yang dihasilkan tentunya memiliki hak cipta yang melekat pada citra tersebut. Menurut undang-undang nomer 28 tahun 2014tentanghak cipta, Hak Cipta adalah hak eksklusif pencipta yang timbul secara otomatis berdasarkan prinsip deklaratifsetelah suatu ciptaan diwujudkan dalam bentuk nyata tanpa mengurangi pembatasan sesuai denganketentuan peraturan perundang-undangan.[2]

Berdasarkan pembentukannya, citra digital dapat dibagi menjadi dua jenis. Jenis pertama adalah citra digital yang dibentuk olehkumpulan pixel dalam array dua dimensi. Citra jenis ini disebut citra *bitmap (bitmap image)* atau citra *raster (raster image)*. Jenis citra yang kedua adalah citrayang dibentuk oleh fungsi-fungsi geometri dan matematika. Jenis citra ini disebut grafik vektor *(vector graphics)*. Dalam penelitian ini, yang dimaksud citra digital adalah citra *bitmap*.[3] Citra digital *(diskrit)* dihasilkan dari citra analog *(kontinu)* melalui digitalisasi Digitalisasi citra analog terdiri atas penerokan *(sampling)* dan kuantisasi *(quantization)* Penerokan adalah pembagian citra ke dalam elemen-elemen diskrit *(pixel)*, sedangkan kuantisasi adalah pemberian nilai intensitas warna pada setiap *pixel* dengan nilai yang berupa bilangan bulat [3]. Perlu adanya perlindungan terhadap karya citra digital dikarenakan bentuk penyimpanan karya citra digital yang rentan terhadap pembajakan, klaim pihak tidak berwenang, pengandaan secara ilegal, ataupun modifikasi yang tidak berizin.

Pengamanan citra digital dapat dilakukan dengan menggabungkan teknik kriptografi dan stegnografi, kriptografi berasal dari bahasa Yunani, *crypto* dan *graphis*. *Crypto* berarti *secret* (rahasia) dan *graphia* berarti *writing* (tulisan). Menurut terminologinya, kriptografi adalah ilmu dan seni untuk menjaga keamanan pesan ketika pesan dikirim dari suatu tempat ke tempat lain.[4]. Kata "Steganografi" berasal dari bahasa Yunani *steganos*, yang artinya "tersembunyi atau terselubung", dan *graphein*, "menulis". Steganografi adalah proses menyimpan pesan rahasia berupa teks dalam bentuk lain sehingga tidak mudah diketahui oleh orang lain.[5]. Dalam implementasinya pengamanan citra digital dilakukan dengan menyisipkan *Watermark* ke dalam gambar. *Watermarking* merupakan suatu cara untuk penyembunyian atau penanaman data/ informasi tertentu (baik hanya berupa catatan umum maupun rahasia) ke dalam suatu data digital lainnya, tetapi tidak diketahui kehadirannya oleh indera manusia (indera penglihatan atau pendengaran), dan mampu menghadapi proses-proses pengolahan sinyal digital sampai pada tahap tertentu. Jadi *watermarking* dapat juga diartikan sebagai suatu teknik penyisipan atau penyembunyian data atau informasi "umum maupun rahasia" ke dalam data digital lainnya (*host* data) tanpa diketahui adanya data tambahan pada *host* datanya oleh indera manusia seperti mata dan telinga.[4]. Kedua teknik ini memungkinkan pemberian ciri pengaman (*watermark*) untuk melindungi citra digital.

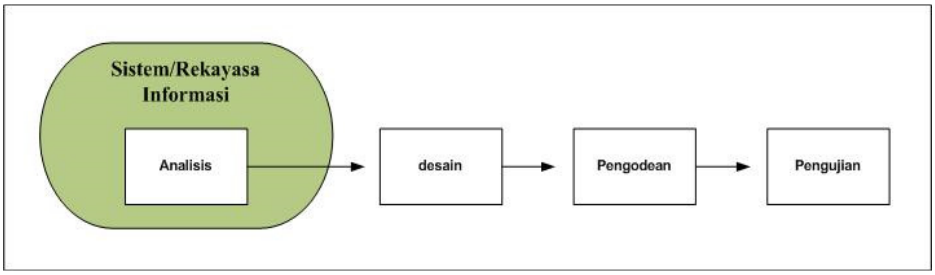
Untuk itu dibutuhkan aplikasi yang dapat melindungi citra digital dari klaim (*plagiarisme*) yang dilakukan. Penulis mencoba untuk menggunakan teknik gabungan kriptografi dan steganografi dengan *Algoritma Rivest Code*4 dan metode *Least Significant Bit*. Dimana teknik ini diyakini mampu melindungi citra digital lebih cepat dari pada metode sebelumnya seperti RSA. Dimana aplikasi ini diharapkan dapat membantu melindungi hak cipta para pelaku industri kreatif khususnya yang menghasilkan karya berupa citra digital.

Pada penelitian terdahulu yang dilakukan oleh Dwitya Putri[6], Semua format *file* dalam *steganography* pasti *invisible* atau tidak terlihat, seperti konsep dari *steganography* sendiri yaitu

agar pesan didalam *image* tidak terlihat atau terdeteksi oleh orang lain selain penerima pesan yang di maksud. *Watermark* harus tidak terlihat sehingga tidak berdampak pada kualitas dari data yang akan dilindungi, dengan kata lain pertahanan harus kuat dari perusakan atau pemanipulasian *image* oleh orang lain. Pada *watermarking visible* atau terlihat bertujuan untuk memperlihatkan identitas pemilik *image*. Ada dua persyaratan penting dalam penyembunyian data yaitu proses tidak terlihat dan kekuatan dari sebuah *image*. Untuk *Hidden* data mempunyai keunggulan tidak kasat mata dan punya kelemahan bila di *capture* / *screen shot* tidak berlaku dan bila gambar dimanipulasi maka data akan rusak. Untuk *Visible* data keunggulannya sulit dimanipulasi dan kelemahannya yaitumengubah *output image* kita. Baik dengan metode *steganography* atau pada teknik digital *watermarking* baik *visible* maupun *invisible* mencoba menyembunyikan *watermark* dalam sebuah *image* agar tidak dapat di ubah struktur *image* tersebut oleh pihak yang merugikan. Dan lebih kepada perlindungan hak cipta atau HAKI untuk kepentingan individu, maupun perusahaan bidang lebih banyak digunakan dalam e-*commerce* untuk melindungi brand-nya. Pada penelitian yang dilakukan oleh Dwitya Putri[7], berdasarkan analisa yang dilakukan, dengan adanya implementasi digital *watermarking* yang sudah dibuat serta pengujian yang dilakukan dapat diambil kesimpulan bahwa metode LSB (*Least Significant Bit)* cukup handal digunakan sebagai metode untuk digital *watermarking* dengan bahasa pemrograman Delphi6, dikarenakan tingkat invisibilitasnya yang tinggi. Dengan prosentase keberhasilan otentikasi sebesar 80%, maka dapat disimpulkan bahwasanya metode MD5 *(Message Digest 5)* cukup handal digunakan untuk otentikasi citra dengan bahasa pemrograman Delphi 6. Rata-rata keberhasilan pembacaan label hak cipta hasil digital *watermarking* dengan mengimplementasi metode LSB (*Least Significant Bit*) dan MD5 (*Message Digest* 5), setelah diberi serangan berupa *cropping*, kompresi GIF, perubahan kontras, penggandaan dan pemberian *noise* adalah 55,55% berarti cukup robust. Pada penelitian yang dilakukan oleh Fauzan Masykur[8], metode LSB yang digunakan dapat dengan mudah melindungi karya cipta dari pemalsuan. Secara kasat mata perubahan warna tidak akan terlihat oleh mata namun bisa dideteksi dengan perhitungan. Metode LSB juga dapat dikombinasikan dengan teknik-teknik steganografi untuk memperkuat ketahanan citra asli dari pemalsuan karena teknik LSB saja akan mudah ditembus untuk mengetahui keaslian citra. Pada penelitian yang dilakukan oleh Tri Prasetyo Utomo[9], dengan aplikasi steganografi yang telah dibuat dapat melindungi transaksi pengiriman pesan antara dua pihak yang saling bertukar pesan. Aplikasi yang dihasilkan juga dapat menyamarkan pesan, karena secara kasat mata pesan tidak akan terlihat, dan terlihat sebagai gambar biasa.

2. METODE PENELITIAN

Model air terjun (*Waterfall*) sering juga disebut model *sekuensial linear* (*squential linear*) atau alur hidup klasik (*classic life cycle*). Model air terjun menyediakan pendekatan alur hidup perangkat lunak secara sekuensial atau terurut dimulai dari analisis, desain, pengkodean, dan pengujian. Berikut adalah model air terjun yang tergambar pada gambar 1.[10]



Gambar1 Proses *Watefall*[10]

Pada analisis kebutuhan perangkat lunak proses pengumpulan kebutuhan dilakukan secara insentif untuk menspesifikasikan kebutuhan perangkat lunak agar dapat dipahami

perangkat lunak seperti apa yang dibutuhkan oleh *user.* Pada tahap ini penulis melakukan analisis kebutuhan dari kebutuhan dari *user* yang akan menggunakan aplikasi dari segi spesifikasi perangkat lunak. Spesifikasi kebutuhan perangkat lunak pada tahap ini perlu untuk didokumentasikan dengan melakukan kebutuhan user dalam berkonsultasi.

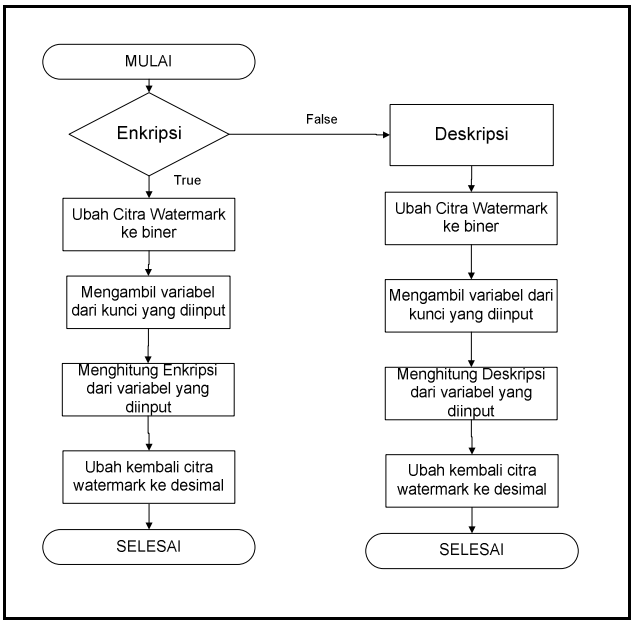
Desain perangkat lunak adalah proses multi langkah yang fokus pada desain pembuatan program perangkat lunak termasuk struktur data, arsitektur perangkat lunak, repsentasi antarmuka, dan prosedur pengkodean. Tahap ini mentranslasi kebutuhan ke representasi desain agar dapat diimplementasikan menjadi sebuah program.

Pengkodean dilakukan dengan mentranslasikan desain ke dalam program perangkat lunak. Hasil dari tahap ini adalah program komputer sesuai dengan desain yang telah dibuat pada tahap desain. Pada tahap ini penulis membuat aplikasi dan mulai memasuki tahap pengkodean sesuai dengan kebutuhan *user*.

Proses pengujian fokus pada perangkat lunak secara dari segi lojik dan fungsional untuk memastikan bahwa semua bagian sudah diuji. Hal ini dilakukan untuk meminimalisir kesalan (*eror*) dan memastikan keluaran yang dihasilkan sesuai kebutuhan *user*.

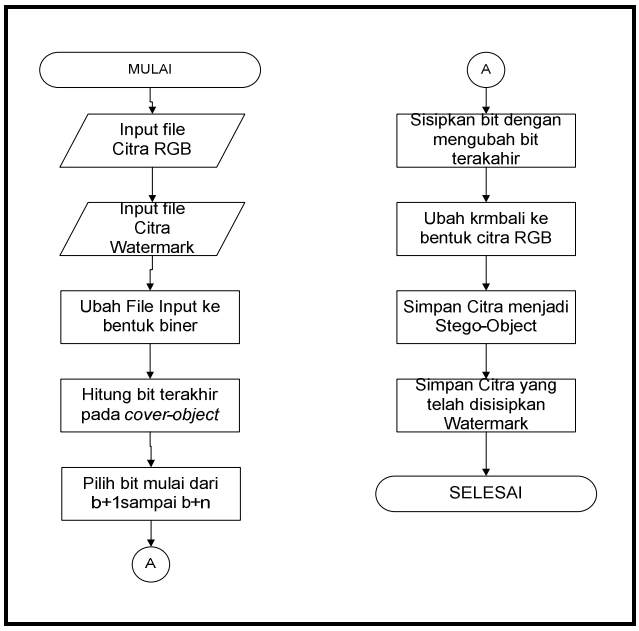
3. HASIL DAN PEMBAHASAN

Pada gambar 2 Sistem akan mengubah *file* citra digital ke dalam bentuk biner yang berupa susunan bit. kemudian bit yang telah didapat akan dipecah menjadi blok-blok kecil yang kemudian dapat di enkripsi atau deskripsi dengan variabel kunci yang di *input.*



Gambar 2 *Flowchart* Enkripsi dan Deskripsi RC4

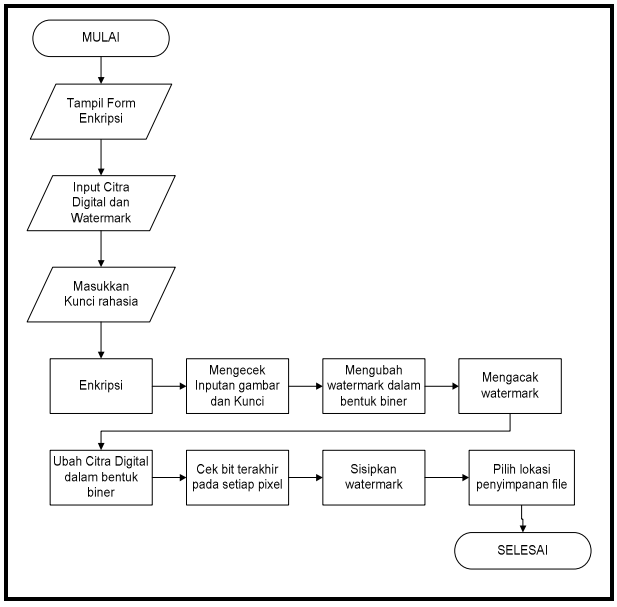
Gambar 3 merupakan proses memasukkan citra digital yang akan disisipkan *watermark.* Dimulai dengan *user* memasukkan citra digital yang akan disisipkan *watermark,* kemudian memasukkan kunci. lalu sistem akan mengubah *file* citra menjadi bentuk *biner* yang berupa susunan *bit,* kemudian sistem akan menghitung nilai *bit* terakhir pada *file* citra yang akan disisipi *watermark,* kemudian *bit* terakhir dari gambar tersebut akan diubah sesuai dengan *file watermark.* lalu nilai *bit* tersebut akan ditransformasikan kembali ke dalam bentuk citra RGB dan setelah proses penyisipan maka sistem akan menyimpan gambar yang telah disisipi *watermark.*



Gambar 3*Flowchar*t *Encoding* LSB

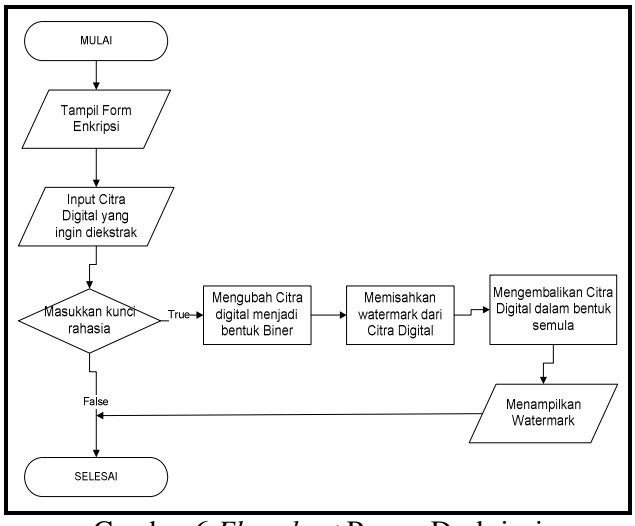
Gambar 4 *Flowchart Decoding* LSB

Gambar 4 memperlihatkan proses *decoding* LSB dimana proses ini dimulai dengan *user* memasukkan citra RGB yang telah disisipi *watermark* dan kunci. sistem akan mengubah *file* citra menjadi bentuk biner yang berupa susunan *bit*, kemudian sistem akan menghitung nilai *bit* terakhir pada citra yang telah disisipi *watermark* kemudian *bit* terakhir dari citra tersebut akan dikembalikan sesuai dengan *watermark.* setelah proses selesai, sistem akan menampilkan pesan rahasia/ *watermark* dari gambar.



Gambar 5 *Flowchart* Proses Enkripsi

Pada gambar 5 menjelaskan *flowchart* Proses enkripsi pada aplikasi, pada proses enkripsi ini, pertama akan muncul *form* enkripsi kemudian *user* diminta untuk memasukkan citra yang akan disisipi dan *watermark*. Kemudian *user* memasukan *password* berupa kunci-kunci. lalu ketika pilihan sisipkan dipilih, sistem akan mengecek semua keperluan yang dibutuhkan untuk mengacak data, bila semua data telah lengkap maka proses pengacakkan akan dimulai dengan mengubah *pixel* dalam gambar menjadi bentuk *biner* yang akan diacak. Setelah proses enkripsi *watermark*, maka sistem akan mengubah citra yang akan disisipi kedalam bentuk *biner* dan kemudian menjalankan proses LSB dimana *bit* terakhir akan diubah. setelah proses LSB selesai, maka *file* citra yang dipilih telah disisipi dengan *watermark.* lalu *user* akan diminta untuk menentukan lokasi untuk penyimpanan data yang telah dilindungi.



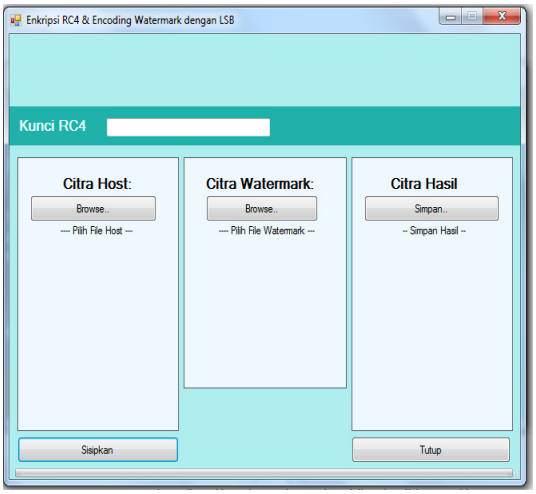
Gambar 6 *Flowchart* Proses Deskripsi

Pada gambar 6 menjelaskan *flowchart* Proses dekripsi dimana, pertama akan tampil *form* deskripsi. Kemudian *user* diminta memasukkan gambar yang akan ditampilkan *watermark*-nya serta *user* diminta memasukkan kunci rahasia. Setelah semua selesai, maka sistem akan

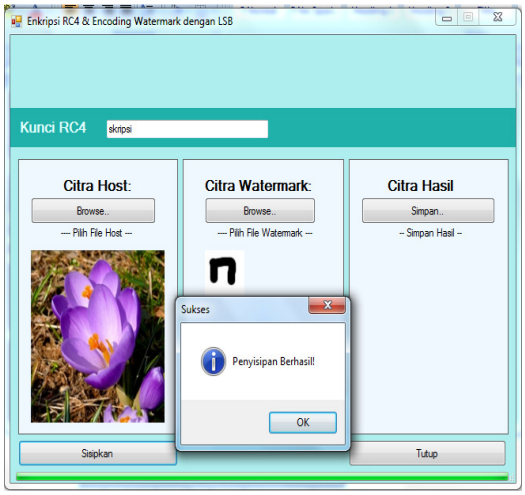
TEKNOMATIKA, Vol.06, No.02, September 2016
P-ISSN : 2087-9571, E-ISSN : 2541-335X

melakukan pengecekan bila data yang dimasukkan benar, maka proses akan dimulai dengan mengubah gambar menjadi bentuk *biner* yang berupa susunan *bit*, kemudian akan memisahkan gambar yang telah di*watermark*, kemudian gambar *watermark* yang telah mengalami pengacakkan sebelumnya akan dikembalikan menjadi bentuk semula. kemudian akan ditampilkan dalam aplikasi berupa *watermark* asli yang belum teracak.

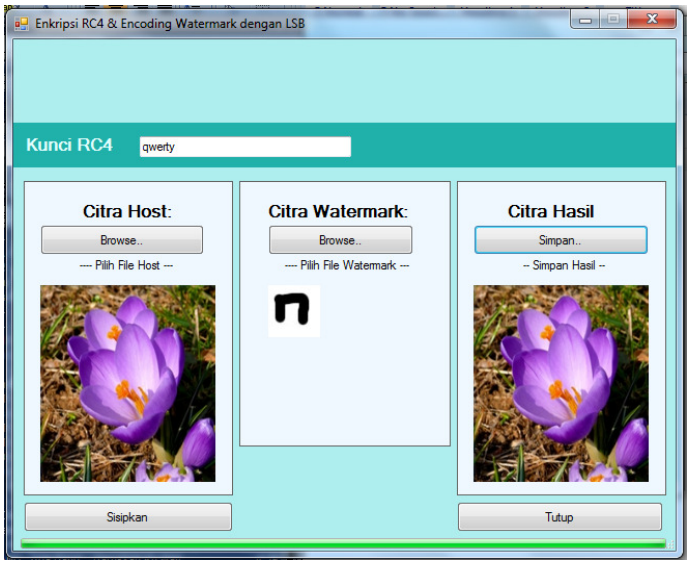
Pada gambar 7 yaitu *Form* penanaman *Watermark* ini, *user* diminta untuk memasukkan kunci rahasia serta gambar dengan memilih *button Browse* dimana setelah *button* dipilih akan terbuka *windows* untuk *user* memilih gambar baik tanda pengenal maupun gambar yang akan dilindungi. Setelah gambar yang terpilih semua baik tanda pengenal maupun gambar yang akan dilindungi pilih *button* sisipkan akan tampil pesan berhasil seperti pada Gambar 8, kemudian *user* diminta untuk memilih lokasi penyimpanan gambar yang telah disisipkan tanda pengenal.



Gambar 7 Tampilan *Form* Penanaman *Watermark*

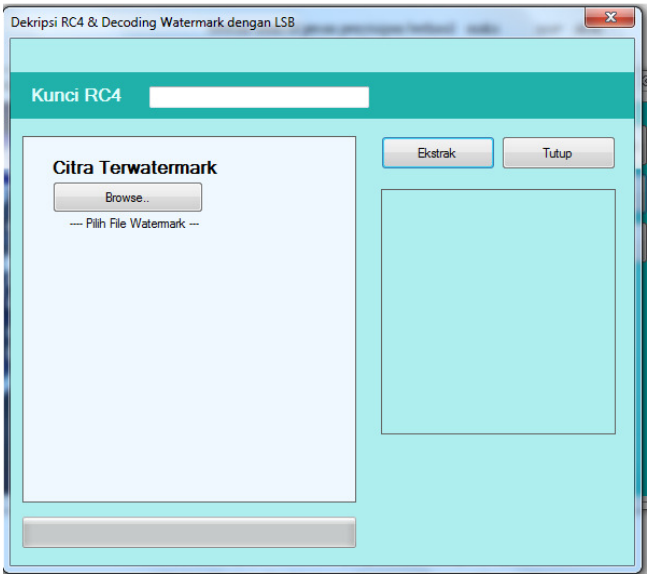


Gambar 8 Tampilan Penyisipan Berhasil Dilakukan



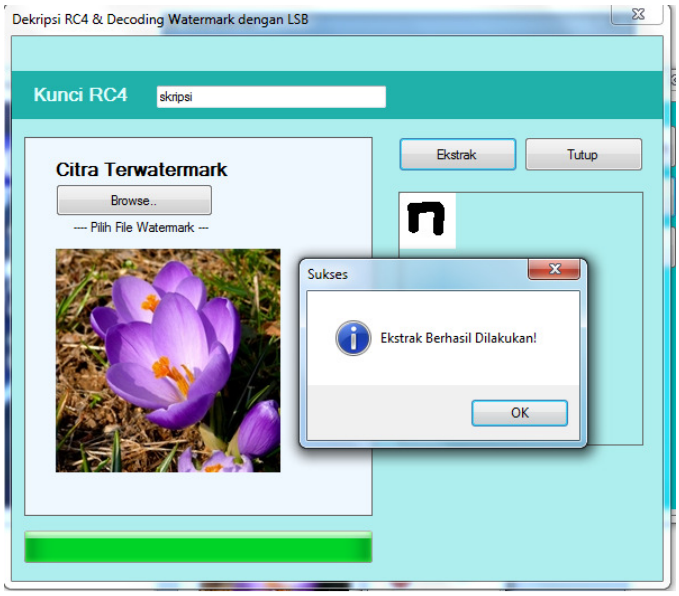
Gambar 9 Tampilan Hasil Setelah Penyisipan

Setelah muncul pesan penyisipan berhasil maka *user* akan memilih *button ok* dimana setelah *user* memilih *button ok* maka akan tampil *windows* untuk memilih tempat penyimpanan *file* gambar yang telah disisipkan tanda pengenal. *File* yang disisipkan akan disimpan dalam bentuk gambar yang kasat mata tidak memiliki perbedaan dengan gambar aslinya yang dapat dilihat pada Gambar 9.



Gambar 10 Tampilan *Form Ekstark Watermark*

Untuk melakukan proses pengekstrakan dibutuhkan kunci RC4 yang harus dimasukkan dengan benar yang kita masukkan saat penanaman tanda pengenal. Setelah itu *user* diminta untuk memasukkan gambar yang akan diungkap tanda pengenal yang ada didalamnya seperti terlihat pada gambar 10. Bila kata kunci rahasia tersebut cocok maka akan tampil gambar tanda pengenal dibawah *button* ekstrak dan *button* tutup yang dapat dilihat pada Gambar 11 tetapi bila kata kunci rahasia tersebut salah maka *watermark* yang akan ditampilkan menjadi tidak sesuai dengan *watermark* asli yang dimasukkan sebelumnya.



Gambar 11 Tampilan *Form* Pengekstrakan Berhasil

Aspek *Imperceptible* merupakan salah satu karakteristik dari metode *watermark* dimana aspek ini menekankan pada *watermark* agar sebisa mungkin harus tidak dapat terlihat atau berbeda dengan dokumen aslinya. Baik dari gambar, ukuran gambar dan dimensi dari gambar yang telah ditambah tanda pengenal.

Tabel 1 Pengujian Aspek *Imperceptible*

Nama Gambar	CitraAsli		Citra Watermark	
	Size (KB)	Dimensi (pixel)	Size (KB)	Dimensi (pixel)
A.bmp	118	200 x 200	118	200 x 200
B.bmp	184	250 x 250	184	250 x 250
C.bmp	264	300 x 300	264	300 x 300
D.bmp	469	400 x 400	469	400 x 400
E.bmp	733	500 x 500	733	500 x 500
F.bmp	1650	750 x 750	1650	750 x 750
G.bmp	2930	1000x 1000	2930	1000x 1000
H.bmp	6592	1500 x 1500	6592	1500 x 1500
I.bmp	178	300 x 202	178	300 x 202
J.bmp	174	295 x 200	174	295 x 200
K.bmp	192	300 x 218	192	300 x 218
L.bmp	1116	750 x 507	1116	750 x 507

Berdasarkan hasil Tabel 1, terlihat bahwa *file* gambar *Bitmap* yang sudah tersisipi tidak dapat mengalami perubahan baik dari tampilan, ukurangambar, dimensi gambar. Sehingga dapat dipastikan tidak ada yang dapatmengetahui bahwa gambar tersebut telah disisipkan tanda pengenal. Artinya,perangkat lunak ini mendukung aspek *Imperceptible*.

Pada tahapan pengujian aspek *Robustness* dilakukan dengan melakukan modifikasi *file* yang telah telah disisipin tanda pengenal, proses ini dilakukan dengan cara memotong *file* dan juga melakukan kompresi gambar.

Tabel 2 Hasil Pengujian Aspek *Robustness*

No	File	Pengujian	Hasil Pengungkapan
1	A.bmp	Pengubahan Ukuran Gambar	*Watermark* gagal
2	Pemotongan B	Pemotongan Gambar Bagian Kanan	*Watermark* gagal
3	Pemotongan B	Pemotongan Gambar Bagian Kiri	*Watermark* gagal
4	Pemotongan B	Penotongan Gambar Bagian Atas	*Watermark* gagal
5	Pemotongan B	Penotongan Gambar Bagian Bawah	*Watermark* gagal
6	Pembalikan C	*Rotate* gambar	*Error*

Dari tabel 2 terlihat bahwa *file* gambar *Bitmap* yang sudah tersisipi tidak dapat mengalami manipulasi ditempat dimana peletakkan tanda pengenal disisipkan, karena sistem ini tidak menggunakan algoritma pengacakkan maka tanda pengenal akan diletakkan diawal *pixel* gambar.

Pada tahap pengujian aspek *recovery*, pengujian dilakukan dengan melakukan pengecekan terhadap validasi kunci yang di masukkan user untuk menampilkangambar tanda pengenal dan juga hasil pengungkapan yang diharapkan sama dengan tanda pengenal yang dimasukkan.

Tabel 3 Hasil Pengujian Aspek *Recovery*

No	File	Pengujian	Hasil Pengungkapan
1	A.bmp	Kunci Salah	*Watermark* Gagal
2	A.bmp	Kunci Benar	*Watermark* Terungkap

Untuk hasil pengujian *recovery* dapat dilihat pada Tabel 3, dapat dinyatakan bahwa tingkat keberhasilannya adalah 100%. Dimana bila semua data yang diminta telah terpenuhi dan benar dapat dipastikan gambar tanda pengenal sama dengan aslinya. Artinya, perangkat lunak ini mendukung aspek *recovery*.

4. KESIMPULAN

1. Berdasarkan hasil implementasi dan pengujian yang telah dilakukan dapat ditarik kesimpulan perangkat lunak pengamanan pesan rahasia menggunakan algoritma RC4 (*Rivest Code 4)* dan metode LSB *(Least Significant Bit)* ini dapat digunakan dengan baik untuk menyembunyikan tanda pengenal rahasia di dalam sebuah citra digital.
2. Perangkat lunak ini menggacak gambar tanda pengenal digital (*watermark*) sehingga tidak dapat dikenali.

5. SARAN

1. Untuk proses penyisipan tanda pengenal hanya berupa gambar hitam putih dan tidak berwarna juga hanya berukuran 50x50 pixel sehingga kurang efisien bila citra digital yang digunakan dalam ukuran besar. Sehingga diharapkan dapat dikembangkannya perangkat lunak ini agar dapat menampung tanda pengenal dengan berbagai ukuran
2. Pesan rahasia yang di-enkripsi dan disisipkan ke dalam citra digital dalam perangkat lunak ini bisa menggunakan berbagai format gambar (*.bmp,*.jpg,*.png) tetapi format gambar untuk citra digital hanya bisa (*.bmp dan *.png) maka untuk itu pengembangannya agar dapat digunakan untuk semua format gambar.

UCAPAN TERIMA KASIH

Penulis mengucapkan terimakasih kepada Ronal Fernando Simamora, Setia Budi, Steven Theodorus yang telah membantu penulis dalam mengumpulkan data penelitian.

DAFTAR PUSTAKA

[1] Sutoyo, T., Mulyanto, E., Suhartono, V., Nurhayati, O. K. & Wijanarto. 2009.*TeoriPengolahan Citra Digital*. Andi Yogyakarta: Yogyakarta.

[2] http://www.dgip.go.id/images/ki-images/pdf-files/uu_pp1/uu_hc_%2028_2014.pdf. Diakses tanggal 11 Agustus 2016

[3] Awcock, G.W. 1996. *Applied Image Processing*. McGraw-Hill Book. Singapore.

[4] Ariyus, Dony. 2008. *PENGANTAR ILMU KRIPTOGRAFI Teori Analisis dan Implementasi*. Andi Offset. Yogyakarta.

[5] Zam, Efvy. 2013. *ANTI PRIVACY Melacak, Membajak, & Membobol Data Rahasia*. PT TransMedia.Jakarta.

[6] Dwitya Putri, Dkk, 2008, *Membandingkan Steganography Dan Watermarking Pada Keamanan File.*, Proceeding Seminar Ilmiah Nasional Komputer dan Sistem Intelijen (KOMMIT), ISSN : 1411-6286.

[7] Sri Esti T. S, Nikmatu Sholikhah, 2010, *Penerapan Teknik Digital Watermarking untuk Perlindungan Citra Digital BITMAP IMAGE dengan metode LSB dan MDS*, Dinamika DotCom, Vol 1. ISSN : 2086-2652.

[8] Shalahuddin, 2016, *Implementasi Watermarking Metode LSB Pada Citra Guna Perlindungan Karya Cipta.*, Indonesian Journal on Networking and Security, Volume 5 No 3, ISSN : 2302-5700 (Print) – 2354-6654 (Online).

[9] *http://jumadi.blog.ugm.ac.id/files/2012/05/trip.pdf.* Utomo, Tri Prasetyo. *Steganografi Gambar Dengan Metode Least Significant Bit Untuk Proteksi Komunikasi Pada Media Online.* Diakses pada tanggal 29 Mei 2015.Pukul 14.01 WIB.

[10] Shalahuddin, 2013, *Rekayasa Perangkat Lunak Tersruktur dan Berorientasi Objek*, Informatika, Bandung.