CARA MENCEGAH RUSAKNYA *WATERMARK* GAMBAR DAN MENGIDENTIFIKASI GAMBAR YANG ASLI BILA *WATERMARK* TIDAK DAPAT DIEKSTRAK KEMBALI

William – NIM : 13506085

Program Studi Teknik Informatika, Institut Teknologi Bandung
Jl. Ganesha 10, Bandung
E-mail : if16085@students.if.itb.ac.id

Abstrak

Makalah ini membahas tentang cara-cara pencegahan terhadap berbagai serangan terhadap *watermark* pada gambar ,serta cara membedakan gambar dengan *watermark* yang sudah rusak dengan gambar aslinya ,bila *watermark* sudah tidak dapat dikenali maupun diekstrak kembali.

Setelah mencoba serangan pada beberapa gambar dengan algoritma *watermark* yang lemah, saya menemukan bahwa serangan-serangan tersebut adalah serangan-serangan yang mudah dilakukan dengan *image editor* yang dimiliki orang banyak saat ini. Serangan-serangan tersebut mengakibatkan rusaknya *watermark* pada gambar ,sehingga sudah tidak mungkin lagi membuktikan mana gambar yang asli dengan gambar dengan *watermark*-nya yang sudah tidak dapat dikenali maupun diekstrak lagi.

Setelah mengidentifikasi berbagai serangan-serangan yang berbahaya tersebut ,maka saya dapat mengidentifikasikan ciri-ciri gambar yang *watermark*nya sudah tidak dapat diekstrak tersebut,baik itu dari sisi spektrum warna, jumlah *noise* dan tingkat intesitas cahaya, dengan gambar asli sebelum di-*watermark*. Selain itu berbagai cara untuk melakukan *watermarking* yang lebih kuat terhadap seranganSemua fakta ini dapat dibuktikan dengan beberapa perangkat lunak *watermark* yang populer, seperti Digimarc dan perangkat lunak *image editor*, seperti Adobe Photoshop.

Makalah ini diharapkan dapat meningkatkan keamanan dari hasil *watermarking* dari perangkat lunak *watermarking* yang murah, serta dapat membantu pihak yang mengalami kesulitan ,karena gambarnya yang sudah ia *watermark* diakuisisi oleh pihak lain dengan cara merusak *watermark* yang sudah ada dan pihak tersebut melakukan beberapa kali *watermarking* ulang, sehingga keaslian gambar sulit dibuktikan.

***Kata kunci**: Watermark, watermarking, noise,* spektrum warna, intesitas cahaya, serangan

1. Pendahuluan

Dengan perkembangan internet saat ini, maka tidaklah aneh bila banyak orang dari seluruh dunia berbagi dan memanfaatkan sumber daya digital mereka di internet. Sumber daya digital tersebut bisa berupa gambar, musik, video, perangkat lunak, dan berbagai jenis *file* lainnya. Karena keterbukaan sumber-sumber daya tersebut bagi orang banyak, maka tidaklah aneh bila hak cipta ataupun kepemilikan asli dari setiap sumber daya digital tersebut dipalsukan. Banyak pihak mengakui sumber daya digital yang mereka temukan di internet adalah milik mereka. Tentu saja hal ini membuat pihak sebenarnya yang sudah bekerja keras membuat sumber daya digital tersebut tidak mendapatkan penghargaan yang seharusnya ditujukan padanya.

Pengakuan kepemilikan sumber daya digital di internet juga dapat berdampak sangat buruk pada sektor ekonomi. Seperti yang kita semua ketahui bahwa saat ini banyak sekali bisnis atau usaha , seperti industri musik, industri film, industri fotografi dan lainnya yang mengalami kerugian besar karena hasil karya maupun produksinya dikopi dalam skala besar tanpa persetujuan pemiliknya dan disebarkan lewat internet. Oleh karena itu banyak pihak yang memerlukan tanda tangannya atau penandaan hak cipta pada sumber daya digital mereka, agar sumber-sumber daya digital yang mereka miliki tidak diakuisisi dan

dimanfaatkan oleh orang lain untuk meraup keuntungan.

2. *Watermark*

Steganografi adalah ilmu dan seni menyembunyikan pesan rahasia pada pesan lain agar, pesan rahasia tersebut tidak dapat dideteksi oleh orang yang tidak diinginkan. Salah satu aplikasi steganografi ini adalah *watermarking*. *Watermarking* adalah aplikasi steganografi yang diaplikasikan pada suatu media. Media digital menggunakan teknik *watermarking* ini yang fungsinya serupa dengan pemberian tanda tangan atau hak kepemilikan pada suatu media atau sumber daya digital. Pada dunia digital, *watermarking* dapat dilakukan pada semua jenis media digital mulai dari media tulisan sampai media video. Tentu saja *watermark* yang merupakan hasil *watermarking* pada media digital tersebut tidak merusak medianya, sehingga media digital tersebut dapat berjalan dengan baik, seperti sebelum diberi *watermark*. *Watermark* pada media digital dapat bersifat tersembunyi, dimana pihak lain yang menjalankan atau membaca media digital yang ter-*watermark* tidak akan sadar akan keberadaan *watermark* tersebut, maupun dengan cara tidak tersembunyi. *Watermark* yang bersifat tersembunyi pada media digital dapat diekstrak atau diambil dari media digital ter-*watermark*, dengan cara melakukan perbandingan dengan media digital aslinya yang belum diberi *watermark*.

Untuk mengaplikasikan *watermark* pada sebuah media digital agar media tersebut tidak rusak dan *watermark* tersembunyi diperlukan sebuah algoritma *watermarking*. Algoritma *watermarking* yang baik adalah algoritma yang menyediakan *watermark* yang tahan terhadap serangan-serangan pada media digital tempat *watermark* tersebut berada. Serangan ini dapat menyebabkan rusaknya *watermark* pada media digital, sehingga *watermark* tersebut tidakdapat diekstrak kembali ataupun bila dapat diekstrak *watermark* tersebut sudah tidak dapat dikenali lagi.

3. Serangan Pada *Watermark* Gambar

Sebenarnya banyak perangkat lunak pemberi *watermark* yang menggunakan algoritma *watermark* yang baik, tetapi perangkat-perangkat lunak ini kurang populer. Hal ini disebabkan oleh mahalnya perangkat lunak tersebut, memang sudah sewajarnya bahwa sesuatu yang bagus dan sulit pembuatannya memakan biaya yang tidak sedikit bila ingin kita gunakan. Tetapi tidak semua rela mengeluarkan biaya sebesar itu, sehingga mereka lebih memilih perangkat-perangkat lunak pemberi *watermark* yang relatif murah maupun gratis. Umumnya perangkat lunak ini tidak menggunakan algoritma *watermarking* yang kuat, sehingga rentan terhadap serangan.

Serangan-serangan terhadap *watermark* dapat disengaja maupun tidak disengaja. Serangan yang tidak disengaja umumnya diakibatkan karena perpindahan media digital ter*watermark* pada beberapa lingkungan yang berbeda sehingga terjadi berbagai konversi pada media digital tersebut. Konversi pada media digital umumnya dilakukan untuk menyesuaikan media digital pada lingkungannya berada. Pada media gambar tidak semua perangkat lunak, situs , maupun atau pesan digital dapat mengirim sebuah media gambar dengan format X , misalnya, sehingga media gambar ini harus dikompresi menjadi sebuah *file* dengan teknik kompresi atau format JPEG(Join Photograph Expert Group). Kompresi ini menyebabkan perubahan pada media gambar bila kita lihat dari sisi byte maupun bit media tersebut, walaupun mungkin dari mata kita tidak terlihat perbedaan apapun. Bayangkan bila kompresi ini terjadi beberapa kali pada media gambar yang ter-*watermark*, tentu saja akan terjadi perubahan yang mungkin cukup untuk merusak sebuah *watermark*, bila algoritma *watermarking* yang digunakan lemah atau tidak tahan terhadap serangan. Selain serangan yang tidak disengaja, ada pula serangan yang disengaja. Serangan ini biasanya dilakukan pihak-pihak yang ingin mengakuisisi media digital yang bukan miliknya ataupun pihak yang melakukan edit dan manipulasi gambar.

Secara garis besar serangan pada *watermark* media gambar dapat dikategorikan menjadi 2 kategori :

1. Serangan grafis
2. Serangan kriptorgrafi

3. 1 Serangan Grafis

Serangan grafis dilakukan melalui pendekatan sinyal grafis gambar. Serangan ini biasanya juga dapat terjadi secara tidak sengaja pada saat kita sedang melakukan manipulasi gambar. Ada banyak jenis serangan pada kategori serangan

grafis ini, beberapa serangan grafis yang sering digunakan adalah :

1. Kompresi gambar
2. Menambahkan *noise*
3. Mengurangi *noise*
4. *Blurring*
5. Penguat sinyal (*sharpening,contrast enhancement)*

Serangan kompresi seperti yang dijelaskan sebelumnya umumnya terjadi secara tidak disengaja walaupun tetap bisa dilakukan dengan sengaja. Serangan jenis ini dapat berhasil karena pembuat algoritma *watermarking* hanya mencoba, *watermarking* pada gambar dengan kompresi tertentu saja. Sebagai contoh adalah algoritma *watermarking* DCT(Discrete Cosine Transform) . Algoritma DCT ini hanya bekerja dengan baik pada media gambar dengan kompresi JPEG.

Serangan dengan menambahkan *noise* adalah serangan dengan menambahkan sinyal acak dalam jumlah besar tapi tidak kuat yang dapat mengenai bit *watermark*. Sehingga bit *watermark* tersebut dapat berubah. Contoh pada aplikasi *image editor* adalah dengan cara menaikkan tingkat *threshold*.

Serangan mengurangi *noise* dipakai dengan ide bahwa sebenarnya *watermark* pada gambar adalah pemberian *noise* pada media gambar. Cara melakukannya adalah berkebalikan dengan menambahkan *noise*, yaitu dengan cara mengurangi *noise-noise* yang terdeteksi pada media gambar. Contohnya pada aplikasi *image editor* adalah *local median* dan *soft thresholding.*

Blurring atau mengaburkan gambar dapat merusak *watermark* bila bit *watermark* pada gambar merupakan sinyal grafis yang lemah. Sehingga bila dikaburkan gambarnya sedikit, sinyal yang lemah itu akan makin lemah bahkan hilang. Kebalikannya adalah penguatan sinyal gambar, serangan ini selain dapat merusak *watermark* juga dapat memunculkan *watermark* pada gambar.

3. 2 Serangan Kriptografi

Serangan kriptografi bertukuan untuk memecahkan metode keamanan pada skema *watermarking* ,lalu menemukan jalan untuk menghilangkan informasi tersembunyi atau bahkan menambahkan *watermark* yang salah. Salah satu cara yang paling umum adalah pencarian informasi yang tersembunyi secara *brute force*. Cara ini sangat terbatas karena tingkat kompleksitas komputasi yang sangat tinggi. Serangan ini dapat menyerang secara langsung untuk mendapatkan kunci yang biasanya disebut *collusion attack*. Serangan kriptografis pada *watermark* sebenarnya mirip dengan serangan pada kriptografi. Mereka memiliki serangan *brute force* yang bertujuan mencari kunci dengan *exhautive search*. Serangan lain pada kategori ini adalah *oracle attack*. Serangan ini membuat gambar yang tidak ber-*watermark* dari gambar yang ber-*watermark* dengan cara mendeteksi *watermark* yang ada.

4. Eksperimen

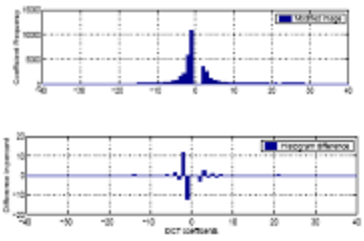
Saya melakukan sedikit eksperimen untuk mendeteksi *watermark* dan menerapkan serangan grafis.

4.1 Eksperimen Pendeteksian *Watermark*

Dalam penelitian ini saya memakai gambar hitam-putih di bawah ini dengan kompresi JPEG.

Gambar 1 Gambar Asli dan Gambar Ter-*watermark*

Image dengan resolusi 800x600 dan 24 bit warna, sebelah atas adalah image asli tapi sebelah bawah sudah disisipkan kata-kata “Hunting of the Snark”. Tetapi tidak ada perbedaan signifikan bila dilihat dengan system penglihatan manusia. Tanpa kompresi besarnya image 12Mb setelah dikompresi format JPEG ukuran *file* berubah menjadi 0.3Mb.Kami analisa dengan koefisien DCT dimana dalam histogram terdapat perbedaan signifikan seperti :



Gambar 2 Historgram Koefesiensi DCT

Di sini dilihat bahwa koefisien DCTnya berubah. Dari yang atas merupakan gambar asli dan yang bawah merupakan gambar ter-*watermark*.

Kita asumsikan X^2 adalah sebuah determinasi dari image yang menunjukkan distorsi dari penyisipan data. Karena test menggunakan stegoanalisis medium maka distribusi y*i untukX^2 test. Dengan ini frekuensi dari koefisien DCT dalam image kita asumsikan image yang sudah disisipkan mempunyai frekuensi yang sama dengan koefisien DCT hasilnya fungsi aritmatik [2].

$$y_i = n_{2i}.$$

$$y_i^* = \frac{n_{2i} + n_{2i+1}}{2}.$$

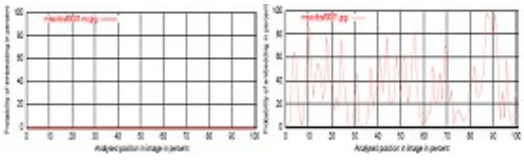
Nilai artimatik X^2 sebagai variable pembedanya dari distribusi berikut

$$\chi^2 = \sum_{i=1}^{\nu+1} \frac{(y_i - y_i^*)^2}{y_i^*}$$

Dimana v adalah derajat bebas nilai dari kategori yang berbeda dalam histogram minus satu.Kemungkinan dari penyisipan nilai p sebuah image yang diberikan komplement dari fungsi kumulatif distribusi.

$$p = 1 - \int_0^{\chi^2} \frac{t^{(\nu-2)/2} e^{-t/2}}{2^{\nu/2}\Gamma(\nu/2)} dt$$

Dimana r adalah fungsi euler gamma. Kita dapat melakukan perhitungan penyisipan data dari area yang berbeda pada sebuah image.



Gambar 3 Grafik Perbandingan Gambar Asli dan Gambar Modifikasi

Graf yang di atas menunjukkan hasil dari image yang tidak dimodifikasi.Sedangkan graf di bawahnya menunjukkan image yang sudah dimodifikasi. Kita dapat melakukan perhitungan penyisipan data dari area yang berbeda pada sebuah image. Dari sini kita dapat mengetahui system steganografi yang digunakan. Untuk image yang tidak mengandung data tersembunyi pasti akan terlihat graf kosong atau perhitungan dengan nilai nol.

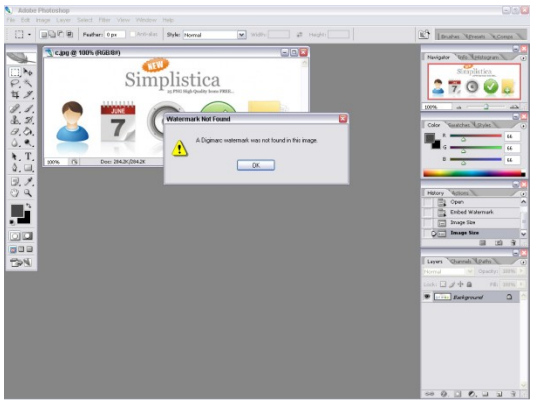
4.2 Eksperimen Serangan Pada *Watermark*

Untuk mengidentifikasi serangan apa saja pada *watermark*, saya melakukan beberapa pengetesan ringan dengan menggunakan perangkat lunak Adobe Photoshop CS 2 dengan bantuan fasilitas pemberi *watermark* pada perangkat lunak ini, Digimarc. Saya memakai Digimarc, karena aplikasi ini merupakan aplikasi *watermarking* yang paling banyak digunakan orang banyak dan skema *watermarking* yang digunakan adalah penyisipan informasi bukan penyisipan gambar pada gambar.

1. Serangan Kompresi
   Gambar asli yang saya gunakan memiliki tipe kompresi JPEG. Lalu saya kompresi gambar tersebut sebanyak 12 kali dengan sambil merubah-rubah ukuran dan mengembalikan ke ukuran semula pada kompresi terakhir. Kompresi dilakukan terus-menerus sampai akhirnya *watermark* sudah tidak dapat dibaca kembali. Dengan urutan kompresi : PBM, TIFF, GIF, ICO, BMP, PNG, PBM, ICO, TIFF, PBM, GIF, JPEG. Serangan ini tidak efektif untuk skema *watermarking* dengan menyisipkan informasi pada gambar.
2. Menambah *Noise*
   Dengan cara Filter->*Noise*->Median-> 1 pixel. Hasilnya *watermark* masih dapat dikenali. Saya tidak menambahkan *noise* lagi karena saya merasa akan memperburuk kualitas gambar. Tapi setelah saya tambahkan *noise* Filter->*Noise*->Add *Noise*-> 20%., ternyata *watermark* sudah tidak dapat ditemukan, walaupun menurut saya kurang relevan pada gambar yang saya pakai karena tampilan gambar

menjadi berubah. Sehingga serangan ini akan bermanfaat pada gambar tertentu saja.

3. Menurunkan *Noise*
   Dengan cara Filter->*Noise*->Reduce *Noise*-> (Strength 10, Reduce Colour *Noise* 100%). Cara ini sedikit bermasalah pada gambar berwarna, karena mengurangi kualitas warna gambar, jadi lebih baik menggunakan gambar hitam-putih. Tapi setelah diterapkan pada gambar ,baik itu gambar yang berwarna atau hitam-putih, *watermark* sudah tidak dapat ditemukan pada kedua gambar.
4. Blurring
   Dengan cara Filter->Blur->Gaussian Blur. Sebenarnya kalau dilihat dari mata telanjang, cara ini sangat mirip dengan serangan menurunkan *noise*. Tapi diperlukan tingkat pengkaburan yang cukup tinggi untuk membuat *watermark* tidak dapat dibaca, akibatnya gambar tidak akan terlihat sama seperti semula. Sehingga saya tidak menganjurkan serangan ini.
5. Menguatkan Sinyal
   Saya tidak bisa membuktikan serangan ini karena keterbatasan percobaan pada algoritma *watermarking*. Tapi dari yang saya lihat serangan ini akan efektif pada skema *watermarking* dengan menyisipkan gambar pada gambar, dan algoritma *watermarking*nya menyembunyikan pixel gambarnya dengan cara di-blur cukup tinggi pada gambar asli. Pixel ter-blur ini akan berubah pada suatu tingkatan tertentu dengan caraFilter->Sharpening->Smart Sharpening.



Gambar 4 Hasil Serangan yang Menyebabkan *Watermark* Tidak Dapat Ditemukan

Untuk serangan kriptografi pada gambar sangat sulit dilakukan karena kompleksitas komputasi yang sangat tinggi, oleh karena keterbatasan waktu dan tenaga saya belum berhasil melakukan eksperimen mengenai serangan ini. Tapi secara teori serangan ini mungkin dilakukan.

5. Pencegahan dan Pendeteksian Serangan

Dari serangan yang sudah dieksperimenkan, maka sebenarnya ada beberapa tips mudah agar kita dapat menggunakan *watermark* murah tapi tahan serangan , serta mengidentifikasi bila gambar yang asli bila serangan sudah terjadi.

Tips-tips yang dianjurkan :

- Gunakan *watermark* dengan skema penyembunyian informasi pada gambar berwarna. Bila ingin pada gambar hitam-putih sebaiknya menggunakan skema *watermarking* dengan penyisipan gambar pada gambar asli. Ini adalah cara untuk menangani serangan dengan menurunkan *noise*.
- Bila menggunakan skema *watermarking* dengan cara menyisipkan gambar, maka sebaiknya gambar yang disisipkan bersifat berulang dan ukurannya sama dengan gambar asli. Hal ini untuk mengatasi serangan dengan menambahkan *noise* pada gambar, karena bila gambar yang disisipkan relative kecil dari gambar asli makadengan pemberian *noise* yang sebenarnya besar tetapi relative kecil pada gambar asli tidak akan mempengaruhi kualitas gambar ,tapi dapat merusak pixel dari gambar yang disisipkan.
- Bila ingin menggunakan aplikasi *watermarking* dengan skema penyisipan informasi, seperti Digimarc, gunakan kompresi gambar GIF, karena GIF memiliki fasilitas penguncian sehingga gambar bertipe ini tidak dapat dikonversi lagi kompresi atau tipe lain. Hal ini untuk mengatasi serangan kompresi. Cara ini hanya dianjurkan pada skema *watermarking* dengan penyisipan informasi saja, karena serangan kompresi ini tidak efektif

terhadap skema *watermarking* dengan penyisipan gambar pada gambar.

- Untuk mengatasi serangan kriptografi, bagi algoritma *watermarking* menggunakan kunci dengan panjang yang cukup aman, panjangnya lebih besar dari 32 bit. Agar pencarian kunci *watermark* menjadi sulit dan memakan biaya yang tinggi.

Lalu, cara-cara mengidentifikasi gambar yang *watermark*nya sudah tidak dapat dikenali lagi tidaklah sulit. Dengan mengidentifikasi kelima serangan grafis yang ada kita dapat melihat ciri-ciri gambar yang *watermark*nya sudah dirusak. Bila dibandingkan dengan gambar yang asli, mungkin gambar dengan *watermark* yang rusak tersebut terkadang masih sulit dibedakan dengan mata telanjang. Tapi bila dibandingkan secara lebih teliti lagi, maka cirri-cirinya adalah sebagai berikut :

- Tingkat blur dari gambar yang diserang dengan gambar yang asli akan lebih tinggi gambar yang diserang. Hal ini disimpulkan dari gambar yang terkena serangan pengkaburan.
- Tingkat *noise* secara keseluruhan akan lebih tinggi gambar yang diserang dengan gambar yang asli. Hal ini disimpulkan dari gambar yang terkena serangan penambahan *noise* secara acak, tapi hal ini terbatas pada gambar berwarna.
- Gambar yang diserang memiliki sinyal yang lebih tinggi secara keseluruhan dibandingkan gambar yang asli. Hal ini disimpulkan dari serangan penambahan sinyal.
- Ada bagian yang kecil pada gambar yang diserang yang pixelnya sedikit pecah (walaupun samar) dan pada gambar yang asli bagian ini tidak pecah. Hal ini disimpulkan dari serangan kompresi yang mengakibatkan hilangnya data saat kompresi berkali-kali.

6. Kesimpulan

Banyak algoritma *watermarking* yang masih digunakan banyak orang ,seperti Digimarc, rentan terhadap serangan-serangan yang mudah dilakukan orang banyak, seperti serangan grafis. Serangan kriptografi dapat berbahaya ,tapi serangan ini jarang dilakukan ,karena kompleksitas komputasi yang tinggi ,serta memakan biaya yang tinggi juga.

DAFTAR PUSTAKA

[1] Munir, Rinaldi. (2004). Bahan Kuliah IF5054 Kriptografi. Departemen Teknik Informatika, Institut Teknologi Bandung.

[2] J. Fridrich, "*Image Watermarking for Tamper Detection*", Proc. ICIP '98, Chicago, Oct 1998.

[3] PETITCOLAS, F.—ANDERSON, R.—KUHN, M. : *Attacks on Copyright Marking Systems, in Lecture Notes on Computer Science*, pp. 218–238, April 1998.