STUDI WATERMARKING DAN SERANGAN-SERANGAN TERHADAP WATERMARKING

Sanrio Hernanto

13507019
Teknik Informatika Institut Teknologi Bandung
e-mail: if17019@students.if.itb.ac.id; x_tetra@yahoo.com

ABSTRAK

Makalah ini membahas tentang studi terhadap *watermarking* pada *file* digital dan serangan-serangan terhadap *watermarking* tersebut. *Watermarking* merupakan teknik untuk menyisipkan informasi ke dalam sebuah *file* digital. Penerapan *watermarking* ini dilakukan dalam berbagai jenis *file* digital. *Watermarking* mempunyai beberapa klasifikasi, antara lain *imperceptible*, *robustness*, dan *capacity*. Serangan pada *watermark* yang sudah diketahui saat ini ada berbagai macam dan sangatlah beragam dengan menyerang berbagai sudut lemah dari *watermarking*. Terdapat juga beberapa metode serangan yang sudah umum yang akan dibahas pada makalah ini seperti *Copy Attack*.

Kata kunci: *Watermarking*, *Imperceptible*, *Robustness*, *Capacity*, *Copy Attack*

1. PENDAHULUAN

Seiring dengan perkembangan teknologi, perputaran informasi yang terjadi pada masa ini terjadi dengan sangat cepat. Salah satu teknologi yang mempercepat perputaran informasi ini adalah internet. Dengan cepatnya perputaran informasi ini, seseorang dapat mendapatkan informasi dengan sangat cepat, tetapi informasi yang didapat bisa saja diragukan kepemilikannya. Selain itu informasi yang kita miliki bisa saja dipergunakan orang dengan seenaknya. Untuk mengatasinya, terdapat berbagai teknik agar dapat memberikan keabsahan dan *copyright* pada sebuah informasi yang berupa data digital. Salah satu teknik yang dapat digunakan adalah *watermarking* yang merupakan salah satu pengaplikasian dari *steganografi*.

Steganografi adalah seni dan ilmu menulis atau menyembunyikan pesan tersembunyi dengan suatu cara sehingga selain si pengirim dan si penerima, tidak ada seorangpun yang mengetahui atau menyadari bahwa ada suatu pesan rahasia. Sebaliknya, *kriptografi* menyamarkan arti dari suatu pesan, tapi tidak menyembunyikan bahwa ada suatu pesan.

Pada umumnya, pesan *steganografi* muncul dengan rupa lain seperti gambar, artikel, daftar belanjaan, atau pesan-pesan lainnya. Pesan yang tertulis ini merupakan tulisan yang menyelubungi atau menutupi.

Teknik *steganografi* meliputi banyak sekali metode komunikasi untuk menyembunyikan pesan rahasia (teks atau gambar) di dalam file-file lain yang mengandung teks, image, bahkan audio tanpa menunjukkan ciri-ciri perubahan yang nyata atau terlihat dalam kualitas dan struktur dari file semula. Metode ini termasuk tinta yang tidak tampak, *microdots*, pengaturan kata, tanda tangan digital, jalur tersembunyi dan komunikasi spektrum lebar. Tujuan dari *steganografi* adalah merahasiakan atau menyembunyikan keberadaan dari sebuah pesan tersembunyi atau sebuah informasi. Dalam prakteknya kebanyakan diselesaikan dengan membuat perubahan tipis terhadap data digital lain yang isinya tidak akan menarik perhatian dari penyerang potensial, sebagai contoh sebuah gambar yang terlihat tidak berbahaya. Perubahan ini bergantung pada kunci (sama pada *kriptografi*) dan pesan untuk disembunyikan. Orang yang menerima gambar kemudian dapat menyimpulkan informasi terselubung dengan cara mengganti kunci yang benar ke dalam algoritma yang digunakan.

Pada metode *steganografi* cara ini sangat berguna jika digunakan pada cara *steganografi* komputer karena banyak format file digital yang dapat dijadikan media untuk menyembunyikan pesan.

Kelebihan *steganografi* daripada *kriptografi* adalah pesan-pesannya tidak menarik perhatian orang lain. Pesan-pesan berkode dalam kriptografi yang tidak disembunyikan, walaupun tidak dapat dipecahkan, akan menimbulkan kecurigaan. Seringkali, *steganografi* dan *kriptografi* digunakan secara bersamaan untuk menjamin keamanan pesan rahasianya.

Sebuah pesan *steganografi* (*plaintext*), biasanya pertama-tama dienkripsikan dengan beberapa arti tradisional, yang menghasilkan *ciphertext*. Kemudian,

covertext dimodifikasi dalam beberapa cara sehingga berisi *ciphertext*, yang menghasilkan *stegotext*. Contohnya, ukuran huruf, ukuran spasi, jenis huruf, atau karakteristik *covertext* lainnya dapat dimanipulasi untuk membawa pesan tersembunyi; hanya penerima (yang harus mengetahui teknik yang digunakan) dapat membuka pesan dan mendekripsikannya.

Watermarking yang merupakan bagian dari *steganografi* adalah suatu cara penyembunyian atau penanaman data atau informasi tertentu (baik hanya berupa catatan umum maupun rahasia) kedalam suatu data digital lainnya, tetapi tidak diketahui kehadirannya oleh indera manusia (indera penglihatan atau indera pendengaran), dan mampu menghadapi proses-proses pengolahan sinyal digital sampai pada tahap tertentu.

Watermarking ini berbeda dengan *watermark* pada uang kertas. Tanda air pada uang kertas masih dapat kelihatan oleh mata telanjang manusia, tetapi *watermarking* pada media digital dimaksudkan agar tidak dapat dirasakan kehadirannya oleh manusia tanpa alat bantu mesin pengolah digital seperti komputer, dan sejenisnya.

Watermarking memanfaatkan kekurangan-kekurangan sistem indera manusia seperti mata dan telinga. Dengan adanya kekurangan inilah, metoda *watermarking* dapat diterapkan pada berbagai media digital.

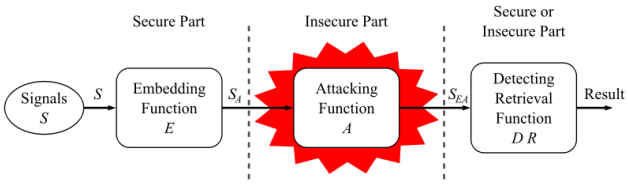
2. WATERMARKING

Watermarking adalah sebuah teknik untuk menyisipkan informasi tertentu ke dalam sebuah data digital dengan suatu cara sehingga *watermark* tersebut sulit untuk dirusak atau dihapus. *Watermark* sendiri adalah informasi yang disisipkan pada saat *Watermarking*. *Watermarking*, merupakan salah satu bidang ilmu yang populer untuk authentikasi dan proteksi *copyright*. Hal ini disebabkan kebutuhan akan perlindungan hak milik dari hak cipta.

Watermarking sudah ada sejak 700 tahun yang lalu. Pada akhir abad 13, pabrik kertas di Fabriano, Italia, membuat kertas yang diberi *watermark* atau tanda-air dengan cara menekan bentuk cetakan gambar atau tulisan pada kertas yang baru setengah jadi. Ketika kertas dikeringkan terbentuklah suatu kertas yang ber-*watermark*. Kertas ini biasanya digunakan oleh seniman atau sastrawan untuk menulis karya mereka. Kertas yang sudah dibubuhi tanda-air tersebut sekalius dijadikan identifikasi bahwa karya seni di atasnya adalah milik mereka.

Digital watermarking bermulai dari Ide *watermarking* pada data digital yang dikembangkan di Jepang tahun 1990 dan di Swiss tahun 1993. *Digital watermarking* semakin berkembang seiring dengan semakin meluasnya penggunaan internet, objek digital seperti video, citra, dan suara yang dapat dengan mudah digandakan dan disebarluaskan. *Watermark* pada Digital watermarking dapat berupa teks, logo, data audio, hingga rangkaian bit yang tidak berarti.

Terdapat dua macam *watermarking*, yaitu *visible watermarking* dan *invisible watermarking*. Pada *visible watermarking*, informasi yang ditambahkan akan terlihat pada gambar atau video. Biasanya informasi yang ditambahkan pada *visible watermarking* adalah text atau logo yang mengidentifikasi pemilik dari data. Pada *invisible watermarking*, informasi ditambahkan pada data digital, tetapi informasi tidak dapat dirasakan kehadirannya sedemikian rupa. *Watermark* dapat ditujukan untuk digunakan secara luas sehingga dibuat mudah diambil.



Gambar 1. Fase life-cycle dari watermark, dengan fungsi embedding, attacking dan detection/retrieval
Sumber: http://en.wikipedia.org/wiki/Digital_watermarking

Sebuah sistem *watermarking* biasanya dibagi menjadi tiga tahap, yaitu *embedding*, *attack* dan *detection*.

Pada *embedding*, sebuah algoritma menerima *host* dan data yang akan dimasukkan dan menghasilkan data yang telah diberi *watermark*. Data kemudian ditransmisikan atau disimpan. Jika seseorang melakukan modifikasi, ini yang disebut attack yaitu ketika modifikasi yang dilakukan bertujuan untuk merusak atau menghilangkan *watermark* yang terdapat pada data. Terdapat banyak kemungkinan modifikasi seperti *lossy compression*, *image* atau *video cropping*, dan *noise adding*.

Detection, sering disebut juga *extraction*, adalah sebuah algoritma yang diaplikasikan kepada data yang mungkin telah di attack untuk berusaha mengekstrak *watermark* dari data. Jika data tidak dimodifikasi ketika transmisi, maka *watermark* akan tetap ada dan dapat di ekstrak. Pada aplikasi *watermarking* bersifat *robust*, algoritma ekstraksi harus tetap dapat menghasilkan *watermark* yang tepat, bahkan ketika modifikasi yang dilakukan cukup kuat. Jika

watermarking yang dilakukan bertipe *fragile*, algoritma ekstraksi akan gagal jika terdapat perubahan pada data.

Digital watermarking dapat di klasifikasi dalam beberapa cara antara lain:

- **Robustness**
  Klasifikasi berdasarkan kekokohan dari *watermark*. Sebuah *watermark* dikatakan *fragile* jika gagal untuk di deteksi setelah data dimodifikasi sedikit, *semi-fragile* jika cukup kuat dan *robust* jika terbilang sangat kuat.

- **Perceptibilty**
  Klasifikasi berdasarkan penyembunyian *watermark*. Sebuah *watermark* dikatakan *imperceptible* jika data watermark tidak dapat dirasakan perbedaannya dengan data aslinya. Sebuah watermark dikatakan *perceptible* jika watermark dapat dirasakan.

- **Capacity**
  Klasifikasi berdasarkan kapasitas ukuran informasi yang dapat disembunyikan kedalam data digital.

Selain itu, *watermark* dapat di klasifikasikan berdasarkan metode *embed* dan *retrieve* antara lain:

- **Spread-spectrum**
  Watermark di *embed* dengan menggunakan modifikasi '*additive'*. *Watermark Spread-spectrum* dikenal sebagai *watermark* yang cukup kokoh, tetapi hanya dapat menampung sedikit informasi karena inferensi dari *host*.

- **Quantization**
  Watermark di *embed* dengan '*quantization'*. *Watermark Quantization* tidak kokoh, tetapi mempunyai kapasitas informasi yang besar karena inferensi *host* yang di '*reject'*

- **Amplitude Modulation**
  Watermark di *embed* dengan modifikasi '*additive'* tetapi dilakukan di '*spatial domain'*

Bila dilihat *watermarking* memiliki kemiripan dengan *steganografi*. *Watermarking* merupakan aplikasi dari *steganografi*, namun ada perbedaan antara keduanya. Jika pada *steganografi* informasi rahasia disembunyikan di dalam media digital dimana media penampung tidak berarti apa-apa, maka pada *watermarking* justru media digital tersebut yang akan dilindungi kepemilikannya dengan pemberian label hak cipta (*watermark*).

Meskipun *steganografi* dan *watermarking* tidak sama, namun secara prinsip proses penyisipan informasi ke dalam data digital tidak jauh berbeda.

3. SERANGAN-SERANGAN PADA WATERMARKING

3.1 Kategori-Kategori Serangan

- **Unauthorized Embedding**
  Serangan pada *fragile watermark*, menambah suatu *watermark* kedalam sebuah file digital yang telah diberi *watermark* sebelumnya. Serangan ini dapat menyebabkan keambiguitasan pada kepemilikian file digital.

- **Unauthorized Detection**
  Melakukan deteksi dan ekstraksi *watermark* pada suatu file digital. Hal ini dapat menyebabkan pencurian data *watermark* yang terdapat pada file digital.

- **Unauthorized Removal**
  Melakukan penghapusan atau perusakan suatu *watermark* pada sebuah file digital. Hal ini dapat menyebabkan hilangnya *copyright* yang sudah di embed pada *waternark* sebelumnya.

- **System Attack**
  Mengeksploitasi kelemahan dari penggunaan *watermark* dengan menyerang sistem, dalam hal ini menyerang fungsi *embed* dan/atau *detection retrieval*.

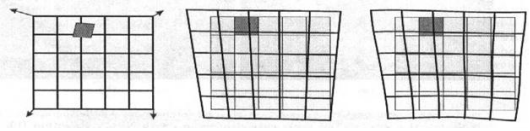
3.2 Berbagai Macam Serangan Pada Watermarking dan Countermeasure-nya

- **Scrambling Attack**
  Serangan dengan melakukan pengacakan pada file yang telah diberi *watermark*. *Scramble* (pengacakan) dilakukan sebelum dilakukan *detection* dan dilakukan *de-scrambled* setelah di *detect*. Dengan cara ini *detector* bisa tidak mendeteksi adanya *watermark* pada file.

  Contoh serangan ini adalah *Mosaic Attack* oleh PetitColas, pada *Mosaic Attack* sebuah file dipecah menjadi banyak bagian kecil yang masing-masing terlalu kecil untuk detektor mendeteksi *watermark*.

Penanganannya adalah dengan pengurangan ukuran minimum yang dibutuhkan pada *robust watermark embedding*

- **Synchronization Attack**
  Serangan dengan melakukan transformasi bentuk pada file yang telah diberi *watermark*. Dengan cara ini detektor tidak dapat mendeteksi *watermark*.



Gambar 2. StirMark Attack
Sumber:
www.ece.wisc.edu/~hu/ece738/notes/watermarkattack.ppt

Contoh serangan ini adalah *StirMark Attack* oleh PetitColas, Transformasi pada *StirMark Attack* dapat dilihat pada gambar 2 , gambar pertama menunjukkan file asli, 2 gambar lainnya menunjukkan tranformasi yang berubah *bending* dan *randomisation*.

Penangannnya adalah dengan menambah sebuah *pattern* registerasi, melakukan registerasi image sebelum melakukan deteksi jika memiliki gambar asli. Penanganan lain adalah dengan menambahkan *watermark* dengan *domain* yang tersebar.

Gambar 3. Hasil StirMark Attack
Sumber:
www.ece.wisc.edu/~hu/ece738/notes/watermarkattack.ppt

Gambar 3 menunjukkan hasil *stirmark attack*, dengan melakukan transformasi, output yang diberikan tidak sesuai.

- **Linear Filtering dan Noise Removal**
  Melakukan serangan pada file yang telah diberi *watermark* dengan berusaha mengurangi/menghapus *noise*/*frequency* yang tidak diinginkan pada file(dalam hal ini adalah *watermark* pada file).

  Contoh aplikasi serangan ini adalah *Host Data Estimation* yang dilakukan oleh Kutter.

- **Copy Attack**
  Serangan yang dilakukan terhadap *fragile watermark*. Serangan ini dilakukan dengan cara mencari *pattern watermark* pada sebuah file dan melakukan *embed pattern watermark* tersebut kedalam file lain. Hal ini menyebabkan *watermark* yang valid di-*embed* kepada file yang berbeda

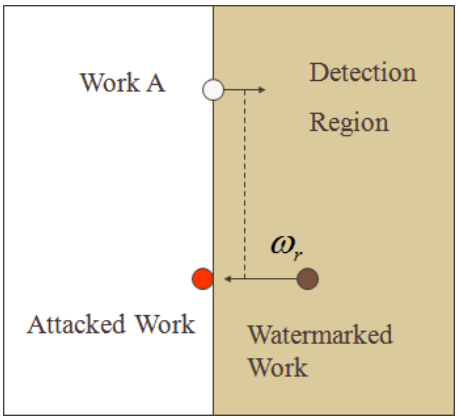
  Contoh serangan ini adalah *Collage* Attack oleh Holliman. *Collage Attack* melakukan attack terhadap algoritma *embed* file yang melakukan *embed* file secara *block independent* melakukan kopi *watermark* pada file baru per blok
  Penanganan serangan ini adalah dengan menambah penanaman *watermark* secara *file dependant* sehingga file yang berbeda akan menghasilkan nilai *watermark* yang di *embed* kedalam fila akan berbeda.

  Penanganan serangan *collage attack* dilakukan dengan cara penambahan info blok kepada blok setelahnya sehingga satu blok mempengaruhi blok lainnya seperti pada *Chiper Block Chaining* (CBC), *Chiper Feed Back* (CFB), atau *Output Feed Back* (OFB)

- **Ambiguity/Deadlock Attack**
  Serangan pada file yang telah diberi *watermark* dengan menambahkan *watermark* sendiri kedalam file yang telah diberi *watermark* sehingga file memiliki dua *watermark* dan dapat memberi ambiguitas kepemilikan dari file

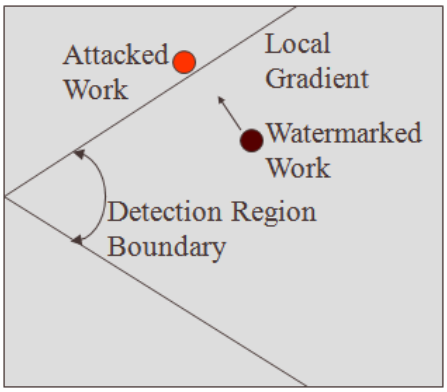
  Penanganan serangan ini dilakukan dengan cara penggunaan skema *embed watermark* yang *non-invertible*

- **Sensitivity Analysis Attack**
  Serangan pada file yang telah diberi *watermark* dengan mencari bagian dari file yang tidak terkena *watermark*. Dengan *Sensitivity Analysis Attack*, file tanpa *watermark* yang merupakan bagian dari file yang telah diwatermark bisa didapatkan. *Sensitivity Analysis Attack* ditemukan oleh Kalker.



Gambar 4. Tahap-tahap Sensitivity Analysis Attack
Sumber:
www.ece.wisc.edu/~hu/ece738/notes/watermarkattack.ppt

- **Gradient Descent Attack**
  Serangan pada file yang telah diberi *watermark* dengan pemisahan gradien, dari info gradien yang didapatkan, area yang dideteksi lebih ringan oleh detektor dapat diketahui dan diserang. *Gradient Descent Attack* ditemukan oleh Kalker.



Gambar 5. Tahap-tahap Gradient Descent Attack
Sumber:
www.ece.wisc.edu/~hu/ece738/notes/watermarkattack.ppt

3.3 Watermarking Copy Attack

Watermarking Copy Attack merupakan salah satu penyerangan terhadap gambar yang telah diberi *watermark*. Seperti yang sudah dijelaskan pada sub-bab sebelumnya, *watermarking copy attack* mencari *pattern watermark* dan melakukan *embed pattern watermark* tersebut kedalam file lain.

Tujuan serangan ini sendiri dilakukan adalah untuk melakukan *copy* sebuah *watermark* kedalam file lain. Selain itu serangan ini berguna untuk mengidentifikasi aplikasi yang digunakan oleh *watermark*. Keunggulan *watermark copy attack* salah satunya adalah tidak perlunya diketahui teknik atau algoritma *watermarking* yang digunakan pada saat meng-*embed watermark* kedalam suatu file. Pada sub-bab ini, akan dibahas *Watermarking Copy Attack* yang diperkenalkan oleh M. Kutter, S. Voloshynovskiy, dan A. Herriage dalam sebuah *paper* yang diperkenalkan pada Januari 2000 di Photonics West SPIE convention.

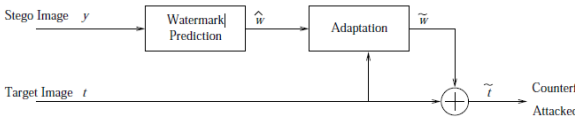
Pada *watermarking copy attack* tersebut, terdapat 3 langkah utama dari proses serangan ini, yaitu melakukan prediksi terhadap *watermark* yang ada pada file yang telah *diwatermark*, pemrosesan dari *watermark* hasil prediksi, dan proses prediksi dari *watermark* ke file target.

Pada proses pertama, melakukan prediksi terhadap *watermark* yang ada, digunakan teknik untuk prediksi melalui proses *denoising.* Dengan kata lain, prediksi dari *watermark* dikomputasi dengan mengambil perbedaan antara file watermark sebelum dan sesudah *denoising*. Untuk menampilkan *denoising*, dilakukan *ML-estimates* dan *MAP-estimates* terhadap *file* dan mengajukan bentuk tertutup dan solusi iteratif untuk kasus-kasus tertentu darii noise dan statistik *file*.

Pada proses kedua, dilakukan pemrosesan *watermark* hasil prediksi. Proses ini berguna untuk meningkatkan *imperceptibility*. Untuk mengadaptasi *watermark* ke file tujuan, diajukan fungsi visibilitas.

Pada proses ketiga, proses prediksi dari *watermark* ke file target, dilakukan pemasukan hasil prediksi yang sudah diproses kedalam sebuah file yang ditujukan.

Secara umum, teknik serangan ini melakukan *estimasi* dari *watermark* yang diletakkan ke dalam file lalu melakukan pengisian *watermark* tersebut kedalam file lain yang belum diisi *watermark*.



Secara umum proses dari *watermark copy attack* dapat dilihat seperti di atas. Input dari proses adalah *stego image*, yaitu file yang mengandung *watermark*, dan *target image*, file yang akan diserang. Serangan ini terbagi atas tiga langkah utama. Di langkah pertama *watermark* dari *stego image* diprediksi, menghasilkan w. Prediksi ini kemudian diproses ke langkah berikutnya. Tujuan dari proses ini adalah untuk mengadaptasi *watermark* ke *target image* dengan tujuan untuk membuatnya dapat dideteksi setelah penyisipan ke *target image*. Pada langkah terakhir, gambar yang telah terprediksi dan dan diproses diisi menjadi *target image*.

Memprediksi *watermark* yang ada di dalam gambar merupakan kunci utama serangan ini. Prediksi bisa dibagi menjadi dua cara, yaitu prediksi langsung dan *denoising*.

Secara umum teknik *watermarking* bisa dimodelkan dalam persamaan matematis

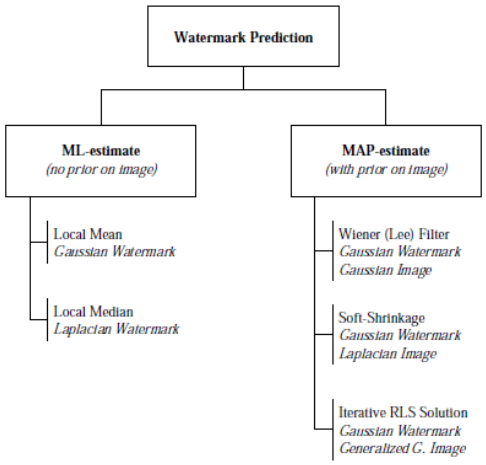
$$y = x + w$$

Di mana y adalah *stego image*, x adalah gambar asli, dan w adalah *watermark*. Bila diumpamakan *stego image* adalah sebuah *image* yang *noisy*, maka *watermark* adalah *noise* dan kita bisa melaukan estimasi mengenai *noise* atau *watermark* w dengan mengambil perbedaan antara perkiraan x dari gambar asli dan stego image seperti berikut:

$$w' = y - z'$$

Terdapat dua pendekatan dan teknik dari rumus-rumus diatas, jika kita tidak memiliki informasi mengenai statistik dari *stego image*, kita bisa menggunakan teknik *maximum likelihood* (*ML*)*-estimate* dari *watermark*. Namun jika tidak, kita bisa menggunakan *Maximum a posteriori Probability* (*MAP*) *estimate*.

Secara umum Kutt menggambarkan estimasi dalam gambar berikut:



Gambar 6 Bagan umum estimasi, ML-estimate dan MAP-estimate

Gambar tersebut menjelaskan bahwa untuk dalam melakukan prediksi ada dua alternatif, yaitu *ML Estimate* dan *MAP-Estimate*, dan setiap alternatif tersebut memiliki prinsip penyelesaiannya tergantung jenis *watermark* dan gambarnya. *ML-estimate* seperti telah dijelaskan sebelumnya memiliki *Local Mean* dan *Local Median*, sementara *MAP-Estimate* memiliki *Wiener* (*Lee Filler*), *Soft Shrinkage*, dan *Iterative* RLS Solution.

Pada *ML-Estimation*, estimasi dirumuskan sebagai:

$$x' = \arg\max \{\ln pw (y|x')\}$$

Di mana pw adalah probabilitasi kedalaman fungsi *watermark*. *ML-estimate* memiliki dua solusi saat apakah *watermark* tergolong *Gaussian* atau *Laplacian*. Jika *watermark* merupakan distribusi *Gaussian*, maka *ML-estimate* diberikan sebagai local mean oleh y:

$$x' = \text{localmean}(y); \text{ Gaussian } watermark$$

Namun jika *watermark* memiliki *Laplacian distribution*, solusi dari *ML-estimate* diberikan oleh *local median*

$$x' = \text{localmedian}(y); \text{ Laplacian } watermark$$

Melakukan komputasi *ML-estimate* dari cover image mengurangi perhitungan local mean atau local median. Untuk melakukan hal ini, beberapa pendekatan muncul dengan hanya menghitung rata-rata atau median di dalam sebuah gambar.

Jika kita mengasumsikan untuk bekerja dengan *natural image*, kita bisa melakukan komputasi untuk estimasi yang lebih akurat mengenai *local mean* atau *median* dengan hanya memperhitungkan *pixel* di *cross-shaped neighberhood*. Hal ini didasarkan fakta bahwa *natural image* menunjukkan korelasi yang tinggi dalam pengarahan *horizontal* dan *vertikal*.

Setelah kita berhasil memprediksi *watermark* dari metode sebelumnya, hasil prediksi tersebut dapat dimasukkan ke dalam gambar tujuan. Namun pengisian ini membutuhkan sebuah proses agar tidak membuat hasil yang tidak baik pada gambar tujuan. Seperti dijelaskan sebelumnya, tujuan dari memproses adalah untuk mengadaptasikan salinan dari *watermark* tersebut ke gambar tujuan. Prosesnya sama dengan metode penyisipan biasa.

Pada *paper*-nya Kutter mencontohkan dengan metode *Noise Visibility Function* (NVF) yang diperkenalkan oleh Voloshynovskiy.

NVF mengkarakteristikkan tekstur lokal dari gambar atas 0 dan 1. Di mana 1 untuk area flat dan 0 untuk area yang bertekstur. Ini bisa digunakan untuk mendeskripsikan tekstur lokal untuk fenomena masking *watermark*, yakni di mana area yang paling banyak memiliki *watermark*, di situlah akan dilakukan paling banyak *masking*.

4. KESIMPULAN

Dari studi mengenai *watermarking* yang dilakukan dapat diambil beberapa kesimpulan, antara lain:

Watermarking adalah sebuah teknik untuk menyisipkan informasi tertentu ke dalam sebuah data digital dengan tujuan menjaga keaslian dari data digital dengan melindungi hak cipta

Pada *watermarking* terdapat tiga tahap utama yaitu *embedding*, *attack*, dan *detection and retrieval*, *embedding* merupakan pengisian *watermark* kedalam sebuah file digital, *attack* merupakan modifikasi yang mungkin terjadi pada file digital, dan *detection and retrieval* adalah pendeteksian dan pengambil kembalian *watermark* dari file digital.

Digital watermarking dapat dinilai berdasarkan tiga klasifikasi yaitu *robustness* yang merupakan kekokohan *watermark*, *perceptibility* yang merupakan penyembunyian watermark dari indera dan *capacity* yang merupakan kapasitas penyimpanan yang bisa ditampung untuk informasi yang akan disembunyikan.

Terdapat empat kategori serangan berdasarkan daerah yang diserang, yaitu *unauthorized embedding* – penambahan watermark, *unauthorized detecition* – pendeteksian dan pengekstraksian *watermark*, *unauthorized removal* – penghapusan atau perusakan *watermark*, dan *system attack* – pengeksploitasian *detection* dan *retrieval*.

Terdapat berbagai macam serangan pada *watermarking* yang telah diketahui sehingga pada pembuatan algoritma untuk melakukan *embedding* pada *watermark* serta *detection* dan *retrieval* dapat mempertimbangkan serangan-serangan ini dan menangkalnya, walaupun masih sangat mungkin terdapat serangan-serangan lain pada *watermark* yang saat ini belum diketahui.

Watermark *copy attack* merupakan salah satu penyerangan yang telah diberi *watermark* dengan tujuan mendapatkan file baru yang diberi *watermark* yang berasala dari watermark pada file lain yang telah diberi *watermark*. Dari kedua file ini dapat diidentifikasikan aplikasi yang digunakan untuk melakukan *embedding watermark*.

REFERENSI

[1]Munir, Rinaldi. (2004). Bahan Kuliah IF5054 Kriptografi. Departemen Teknik Informatika, Institut Teknologi Bandung.
[2]Voloshynovskiy, S. Attacks on Digital Watermarks: Classification, Estimation-based Attacks and Benchmarks. University of Geneva.
[3]Kutter, Martin. The Watermark Copy Attack. University of Geneva.
[4]Samcovic, Sandreja. Attack on Digital Wavelet Image Watermarks.
[5]
www.ece.wisc.edu/~hu/ece738/notes/watermarkattack.ppt
tanggal akses: 22 Maret 2010
[6]
en.wikipedia.org/wiki/Digital_Watermarking
tanggal akses: 22 Maret 2010