# Steganalisis :Teknik *Jitter Attack, StirMark,* dan *Mosaic Attack* untuk Merusak dan Menguji Ketangguhan *Watermark*

Edward Ferdian – NIM : 13503006

*Program Studi Teknik Informatika, Institut Teknologi Bandung*
*Jl. Ganesha 10, Bandung*
E-mail : if13006@students.if.itb.ac.id

## Abstrak

Steganalisis adalah teknik untuk mendeteksi atau memecahkan informasi tersembunyi yang disisipkan dengan teknik steganografi. Steganalisis tidak hanya berguna untuk merusak dan menyerang metode steganografi, tetapi juga dapat digunakan untuk menguji ketangguhan metode steganografi yang digunakan. Teknik-teknik steganalisis yang akan dibahas dalam makalah ini adalah teknik-teknik dasar dalam manipulasi citra digital dan juga serangan-serangan terhadap *watermark*, yaitu *Jitter Attack, StirMark,* dan *Mosaic Attack* untuk mendeteksi atau merusak *watermark*.

*Watermarking* telah banyak diterapkan dalam media digital, misalnya gambar, video, audio, dan objek multimedia lainnya. Tujuannya adalah untuk menyembunyikan informasi atau memberi tanda hak cipta pada media digital tersebut. Akan tetapi, tiap teknik memiliki kelemahannya masing-masing. Kelemahan tersebut dapat diukur dari kemudahan untuk mendeteksi atau merusak *watermark* dalam media digital. Beberapa teknik yang dapat digunakan dalam merusak atau menghilangkan *watermark* dalam media digital adalah teknik *Jitter Attack, StirMark,* dan *Mosaic Attack*. Untuk menghadapi teknik-teknik penyerangan ini dibutuhkan berbagai metode yang kuat. Setiap algoritma *watermarking* memiliki kelemahan masing-masing. Untuk itulah dilakukan pengujian untuk melihat seberapa tangguh suatu teknik *watermarking* dalam menghadapi berbagai jenis serangan.

***Kata kunci**: Jitter Attack, StirMark, Mosaic Attack, watermark,* steganalisis, steganografi.

### Steganografi dan *Watermarking*

Salah satu teknik untuk menyandikan pesan selain kriptografi adalah steganografi. Steganografi dapat dikatakan merupakan kelanjutan dari kriptografi. Jika dalam kriptografi, pesan rahasia dienkripsi menjadi *ciphertext*, sehingga akan menimbulkan kecurigaan pada pihak lawan, sedangkan dalam steganografi pesan rahasia disembunyikan pada media tertentu sehingga tidak menimbulkan kecurigaan.

Steganografi memerlukan dua atribut utama, yaitu media penampung (*covertext*) dan pesan rahasia (*hiddentext*). Media penampung dan pesan rahasia dapat berupa teks, gambar, suara, atau video. Cara berkomunikasi secara rahasia ini semakin berkembang, tidak hanya untuk menyembunyikan pesan rahasia saja, tetapi juga untuk menyembunyikan informasi rahasia seperti tanda hak cipta dalam suatu media digital. Teknik ini disebut *watermarking*, yang merupakan implementasi dari steganografi.

Teknik *watermarking* memiliki fungsi yang sama dengan steganografi, yaitu untuk menyembunyikan informasi, akan tetapi pada *watermarking*, media penampung menjadi hal yang sangat penting, sedangkan pada

steganografi media penampung hanya sebagai media pembawa pesan rahasia saja.

Pada *watermarking*, media penampung itulah yang harus dilindungi. *Watermark* diberikan pada suatu media digital untuk melindungi media tersebut dari pelanggaran hak cipta (*copyright*). Untuk itu, diperlukan suatu teknik yang kuat untuk menjaga hak cipta dalam sebuah media digital. *Watermark* yang kuat adalah *watermark* yang tidak mengalami kerusakan yang berarti jika dilakukan manipulasi terhadap media penampungnya. Hal ini sangat penting untuk dapat membuktikan kepemilikan hak cipta suatu media jika media tersebut dimanipulasi.

**Perlidungan Hak Cipta dengan *Digital Watermarking***

Media digital sangat rentan terhadap pembajakan. Oleh karena itu dibutuhkan sebuah mekanisme untuk menjaga masalah kepemilikan dan hak cipta media digital. Masalah ini terjadi pada media yang memang harus dilindungi. Masalah yang muncul misalnya, masalah kepemilikan suatu karya digital, masalah keaslian, ataupun masalah pelanggaran hak cipta (*copyright*).

Dengan *digital watermarking*, diharapkan media digital yang dilindungi tersebut dapat diberi tanda yang dapat dibuktikan jika terjadi pelanggaran terhadap masalah-masalah yang telah disebutkan di atas. *Watermark* harus dapat dimasukkan ke dalam media penampung tanpa membuat perubahan yang dapat diindera oleh manusia. Jika media penampungnya adalah sebuah *file* musik, maka setelah musik tersebut disisipi *watermark***,** musik tersebut harus dapat dimainkan sama seperti semula tanpa menimbulkan perbedaan yang dapat dibedakan oleh pendengaran manusia. Jika media penampung berupa citra / gambar, maka citra tersebut setelah disisipi *watermark* harus tidak dapat dibedakan dengan kondisi aslinya sebelum disisipi *watermark*.

**Teknik-teknik *Watermarking***

Teknik dalam *watermarking* dapat dibedakan ke dalam dua *domain*, yaitu domain spasial dan domain transform. Media yang dapat digunakan dapat berupa

- ***Image*** bitmap, GIF, *JPEG (compression)*
- ***Audio*** Raw, WAV*, MP3 (compression)*
- ***Video*** *MPEG (compression)*
- ***Text*** document, XML, program

Dalam makalah ini akan akan dibahas watermarking dalam citra digital saja sebagai contoh untuk membuktikan ketangguhan teknik penyisipan *watermark*.

Secara umum, skema penyisipan pesan dalam *watermarking* sama dengan steganografi. *Watermark* harus dapat diekstraksi atau dideteksi kembali.

*Watermark* ada dua jenis, yaitu :

- *Robust watermark*
  *Robust watermark* adalah *watermark* yang harus masih dapat terdeteksi setelah dilakukan pemrosesan yang rumit. Biasanya serangan terhadap *watermark* jenis ini bertujuan agar *watermark* tersebut tidak dapat terdeteksi lagi. *Watermark* jenis ini biasanya digunakan dalam *Copy Control, Evidence of Ownership, Fingerprinting*. Serangan terhadap *watermark* ini akan dibahas pada makalah ini.

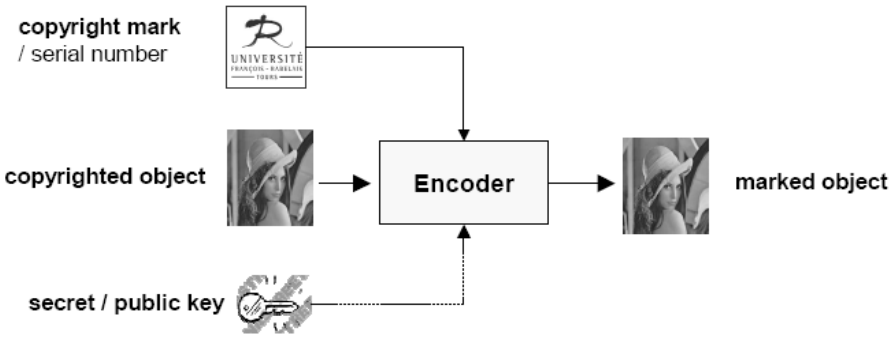
- *Fragile watermark*
  *Fragile watermark* bertujuan untuk memberi tanda jika media penampung telah diberi *watermark* atau tidak. Selain itu, *watermark* jenis ini berguna untuk menyulitkan orang yang tidak memiliki hak dalam memasukkan *watermark*. Serangan terhadap *watermark* jenis ini bertujuan untuk membuat *watermark* dapat diubah setelah terjadi manipulasi terhadap media *watermark*.

  Penyembunyian data dilakukan dengan mengganti bit-bit data di dalam segmen citra dengan bit-bit data rahasia. Salah satu

metode penyembunyian data yang sederhana adalah dengan menggunakan metode *LSB*.

Bit yang cocok untuk diganti adalah bit *LSB*, sebab perubahan tersebut hanya mengubah nilai *byte* satu lebih tinggi atau satu lebih rendah dari nilai sebelumnya. Misalkan *byte* tersebut menyatakan warna keabuan tertentu, maka perubahan satu bit LSB tidak mengubah warna keabuan tersebut secara berarti. Lagi pula, mata manusia tidak dapat membedakan perubahan yang kecil.

Untuk memperkuat teknik penyembunyian data, bit-bit data rahasia tidak digunakan mengganti *byte-byte* yang berurutan, namun dipilih susunan *byte* secara acak. Misalnya jika terdapat 50 *byte* dan 6 bit data yang akan disembunyikan, maka maka *byte* yang diganti bit *LSB*-nya dipilih secara acak.




**Gambar 1. Skema Watermarking**

**Syarat-syarat Teknik Watermarking yang Baik**

*Watermark* yang telah disisipkan ke dalam sebuah media digital harus dapat diekstraksi kembali. Akan tetapi, *watermark* tersebut harus kuat terhadap berbagai jenis serangan. Watermark yang baik harus memiliki syarat-syarat sebagai berikut:

1. *Imperceptibility*: keberadaan *watermark* tidak dapat diindera oleh pengamatan manusia.

2. *Key uniqueness*: kunci yang digunakan hanya satu, perbedaan kunci (*stegokey*) seharusnya menghasilkan *watermark* yang berbeda pula. Kesalahan pemilihan kunci pada saat ekstraksi akan menghasilkan hasil yang salah pula.

3. *Noninvertibility*: sangat sulit mendeteksi apakah citra tersebut ber-*watermark* atau tidak jika hanya diketahui citra ber-*watermark* saja.

4. *Image dependency*: *Watermark* bergantung dari isi citra.

5. *Robustness*: *Watermark* seharusnya dapat bertahan setelah dilakukan manipulasi. Teknik yang baik dapat mengatasi manipulasi sehingga tidak merusak *watermark* dan *watermark* masih dapat terdeteksi.

Pada makalah ini akan lebih banyak dibahas mengenai serangan terhadap kekokohan (*robustness*) teknik *watermarking*. Biasanya manipulasi yang dilakukan terhadap citra ber-*watermark* meliputi:

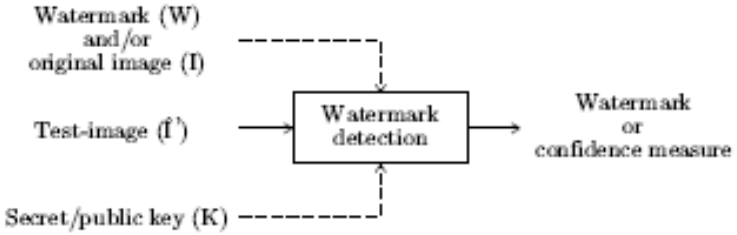
- Kompresi
- Transformasi geometri: rotasi, perbesaran, perkecilan
- Penambahan derau

Banyak teknik *watermark* dapat bertahan dari manipulasi secara individual, tetapi biasanya tidak ada yang mampu bertahan jika dilakukan gabungan dari manipulasi yang dapat dilakukan.

**Steganalisis**

Steganalisis bertujuan untuk mendeteksi atau memastikan keberadaan suatu informasi tersembunyi dalam sebuah data yang diamati dengan sedikit atau tanpa mengetahui algoritma steganografi. Steganalisis dapat dikatakan sebagai seni ataupun ilmu pengetahuan.

Walaupun teknik steganalisis yang baik dapat dirancang untuk melawan sebuah algoritma steganografi yang ada, namun tujuan utama dari steganalisis adalah untuk menciptakan teknik yang efektif untuk melawan algoritma steganografi yang sejenis.




**Gambar 2. Skema ekstraksi watermark**

**Berbagai Jenis Serangan terhadap Watermark**

Berbagai jenis serangan dapat dilakukan terhadap teknik *watermarking*. Dalam menghadapi berbagai serangan ini, penyerang (*attacker*) dapat diasumsikan sebagai berikut :

- Tidak mengetahui apapun mengenai teknik watermark yang digunakan. Dengan asumsi ini, penyerang diasumsikan menyerang kelemahan yang ada pada skema *watermarking* pada umumnya.

- Penyerang memiliki lebih dari satu benda ber-*watermark*. Dalam hal ini, dapat dianggap media yang berbeda dengan *watermark* yang sama, ataupun media yang sama dengan *watermark* yang berbeda.

- Penyerang mengetahui algoritma *watermarking* yang digunakan. Asumsi ini adalah asumsi yang paling banyak digunakan. Penyerang biasanya menyerang titik lemah dari algoritma tertentu. Kerahasiaan sebaiknya ada pada kunci yang digunakan, bukan pada algoritma yang digunakan, karena algoritma diasumsikan telah diketahui pihak lawan.

- Penyerang memiliki akses ke alat pendeteksi. Biasanya penyerang menggunakan *gradient descent attack* atau *sensitivity analysis attack*.

Serangan dalam steganalisis terhadap *watermark* dapat dibagi ke dalam dua jenis serangan, yaitu serangan pasif (*passive attacks*) dan serangan aktif (*active attacks*).

Dalam serangan pasif, teknik serangan yang digunakan hanya berfungsi untuk mendeteksi adanya pesan rahasia atau tanda hak cipta dalam sebuah media digital. Sedangkan pada serangan aktif, teknik yang digunakan bertujuan untuk mengekstraksi atau mengubah pesan rahasia dalam media penampungnya.

Berbagai macam serangan aktif yang ditujukan pada *watermarking* adalah sebagai berikut:

1. *Robustness attacks* – memproses media yang telah disisipi *watermark*, untuk menghilangkan atau merusak *watermark*, contohnya : *StirMark*

2. *Presentation attacks* – memodifikasi media yang telah disisipi *watermark* untuk menghindari pendeteksian terhadap

*watermark* yang telah disisipkan, contohnya *Mosaic Attack*.

3. *Interpretation attacks* – menyangkal bukti kepemilikan, contohnya penambahan tanda *watermark* palsu.

4. *Implementation attacks* – mengambil keuntungan dari software yang memiliki algoritma *watermarking* yang lemah.

Selain itu, serangan-serangan dalam *watermarking* dapat dibedakan ke dalam empat kategori. Kategori-kategori tersebut adalah:

- *Unauthorized Embedding*
  *Unauthorized Embedding* termasuk serangan pada fragile *watermark*. Biasanya dilakukan denggan cara memodifikasi media yang sudah diberi *watermark* agar dapat dijadikan media penampung yang baru. Selain itu, dapat dilakukan juga dengan cara menyalin blok media yang valid tanpa mengerti isi dari media tersebut.

- *Unauthorized Detection.*
  *Unauthorized Detection* dapat dilakukan dengan men-*decode* isi *watermark*. Contohnya pada data medis pasien rumah sakit yang diberi *watermark* berupa *ID* pasien. Selain itu dapat pula dilakukan dengan mendeteksi keberadaan *watermark*.

- *Unauthorized Removal*
  Biasanya diterapkan dalam *Elimination Attack*. Jenis serangan ini dapat menghapusa *watermark*, sehingga tidak dapat dideteksi lagi bahwa media tersebut pernah diberi *watermark*. Biasanya juga diterapkan dalam *Masking Attack*, sehingga *watermark* yang ada menjadi tersamarkan atau tidak terdeteksi. Namun, pendeteksi *watermark* yang bagus biasanya dapat mengalahkan metode ini.

- *System Attack*
  Serangan ini dilakukan dengan cara mencari kelemahan dari teknik *watermarking* yang digunakan.

Berikut ini akan dibahas mengenai beberapa teknik steganalisis dalam menguji algoritma steganografi pada citra dan audio. Teknik-teknik ini merupakan serangan yang dapat menghancurkan atau setidaknya memecahkan batasan yang dimiliki oleh skema *watermarking.*

**1. Transformasi geometris**

Jenis serangan ini adalah jenis serangan yang paling dasar dalam teknik serangan terhadap *watermark*. Serangan jenis ini banyak dilakukan pada serangan awal, namun banyak media ber-*watermark* yang tidak lolos dari serangan sederhana ini.

- *Horizontal Flip*
  Serangan ini dilakukan hanya dengan membalikkan gambar secara horizontal. Metode ini tampak sangat sederhana, namun beberapa skema *watermarking* tidak lolos dari serangan ini.

- Rotasi
  Rotasi dilakukan biasanya dengan derajat perputaran yang sangat kecil, sehingga citra tampak tidak berubah. Namun karena perputaran yang kecil tersebut, *watermark* tidak dapat terdeteksi lagi.

- Cropping
  Pemotongan citra menjadi bagian-bagian kecil. Hal ini mengakibatkan *watermark* tidak utuh dan kemudian menjadi tidak terdeteksi lagi. Cara ini menjadi landasan bagi metode serangan *Mosaic Attack* yang akan dibahas kemudian secara lebih mendalam.

- Scaling
  Penskalaan dapat dibedakan menjadi 2 jenis, *uniform* dan *non-uniform*. Penskalaan *uniform* mengubah ukuran citra dengan faktor skala yang sama, baik vertikal maupun horizontal. Sedangkan pada *non-uniform scaling* faktor skala vertikal dan horizontal berbeda.

- Penghapusan garis atau kolom
  Cara ini dilakukan dengan menghapus satu kolom atau satu baris pada citra. Cara ini

sangat efektif dilakukan untuk melawan teknik *spread-spectrum*.

- *Random geometric distortions*
  Melakukan distorsi geometris secara acak. Biasanya merupakan gabungan dari teknik-teknik dasar. Salah satu metode ini adalah *StirMark* yang akan dibahas secara mendalam dalam makalah ini.

Teknik dasar yang dapat digunakan masih banyak, akan tetapi hanya akan dibahas sedikit saja. Misalnya dengan meningkatkan atau mengurangi ketajaman dalam suatu citra digital. Hal ini akan mengubah media penampung *watermark.*

Penambahan *noise* juga merupakan salah satu pilihan jenis serangan yang baik, terutama dalam teori pemrosesan sinyal. Banyak skema *watermark* yang selamat dari serangan ini, tapi tentunya jika diberikan *noise* dalam jumlah besar, skema apapun tidak akan dapat bertahan. Contoh penggunaan *noise* akan dibahas dalam *Jitter Attack*.

Selain pemrosesan secara digital, pemrosesan secara manual pun dapat merusak *watermark*, misalnya dengan pencetakan dan *scanning*. Hal ini sebenarnya sama dengan transformasi geometris, hanya saja biasanya disamakan dengan *noise*.

### *2. Jitter Attack*

Langkah awal dalam membangun sebuah serangan yang sistematis pada *watermark* pada audio adalah dengan menebak metode *watermark* yang digunakan. Metode watermark yang paling mudah ditebak adalah metode LSB (*Least Significant Bit*). Metode ini merupakan metode yang paling sederhana dan paling mudah diimplementasikan. Metode ini dilakukan dengan cara mengubah bit yang paling tidak berarti dalam sebuah *byte*. Jika dilakukan perubahan terhadap sebuah bit pada citra atau audio, maka indera manusia tidak dapat membedakannya.

*Byte* yang diubah biasanya ditunjuk secara acak dengan pola tertentu, atau dibangkitkan dari kunci. Sebuah serangan yang sederhana dan merusak untuk metode LSB ini adalah dengan *jitter attack*. *Jitter Attack* biasanya dilakukan pada media audio.

Metode *jitter attack* dilakukan dengan cara menambahkan *jitter* pada sinyal audio yang mengandung *watermark*. Pertama-tama, dilakukan pemecahan sinyal tersebut menjadi beberapa pecahan yang terdiri dari masing-masing 500 *sample*. Lalu lakukan duplikasi atau penghapusan secara acak pada tiap pecahan. Maka saat ini, tiap pecahan akan memiliki 499 atau 501 *sample*. Lakukan penggabungan kembali pecahan-pecahan tersebut menjadi sinyal semula, tentunya dengan jumlah *sample* yang telah berubah pada tiap pecahannya.

Stelah penggabungan kembali dilakukan, sinyal audio tersebut akan sulit dibedakan dari sinyal sebelum dilakukan pemecahan. *Jitter* atau gangguan akan mencegah ditemukannya bit-bit yang telah diubah oleh LSB. Hal ini dapat terjadi, dikarenakan ada bit yang terbuang atau karena terjadi perubahan posisi pada bit-bit dalam sinyal audio.

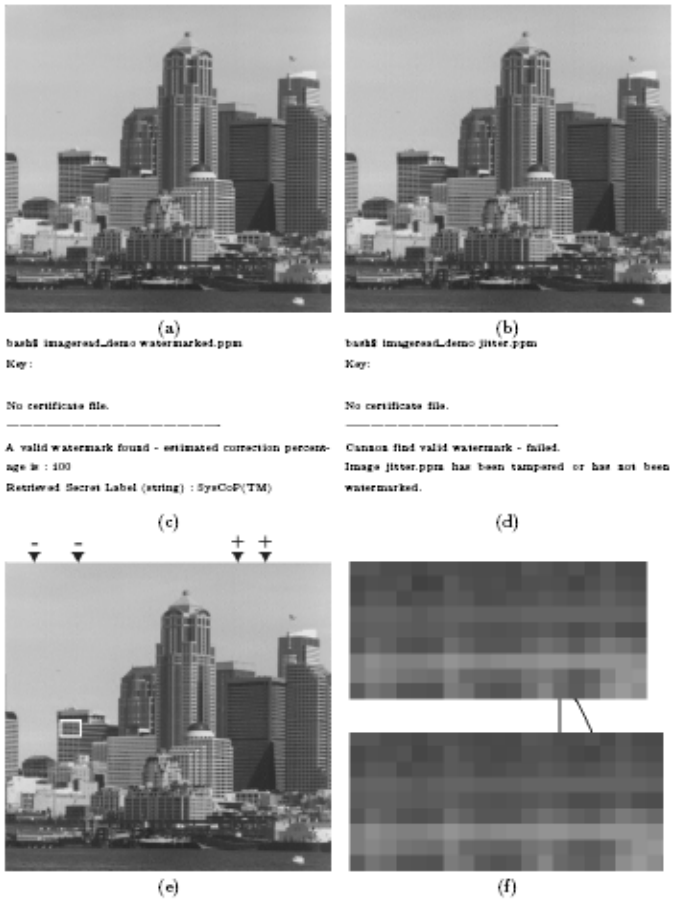
Dalam implementasi yang lebih rumit, dilakukan pengubahan terhadap pecahan-pecahan sinyal pada frekuensi yang berbeda dengan frekuensi asalnya.

Secara algoritmik dapat dijelaskan bahwa, jika $n_i$ adalah jumlah *sample* dalam pecahan ke-*i,* $n'_i$ adalah jumlah *sample* dalam pecahan ke-*i* setelah dilakukan perubahan, dan $\alpha$ adalah nilai maksimum relatif yang dibolehkan untuk perubahan frekuensi, maka $|\Delta n_i| \leq \alpha\, n_i$, karena $\alpha$ sangat kecil, dimana $\Delta n_i := n'_{i+1} - n'_i$

Persamaan ini dapat disederhanakan menjadi $0 < k < \alpha\, n/2$ ketika $n_i$ bernilai sama dan $k$ merupakan jumlah konstan banyaknya *sample* yang dihapus atau ditambahkan dalam sebuah pecahan. Strategi untuk memilih nilai $k$ dan $n$ dapat ditentukan bergantung dari sinyal masukan.

Walaupun teknik ini lebih spesifik diterapkan pada sinyal audio, tetapi *jitter attack* juga dapat diimplementasikan pada cita digital. Pada citra digital, *jitter attack* dapat diterapkan dengan cara menghapus sebuah *pixel* pada pecahan tertentu dan menggantikannya dengan *pixel* yang berbeda.

Hal ini dilakukan untuk menjaga agar ukuran citra tetap sesuai dengan citra masukannya. Dengan metode ini, akan terjadi perubahan bit, sehingga merusak *watermark* yang terdapat dalam media penampungnya.




**Gambar 3. Jitter Attack**

Gambar (a) merupakan gambar ber-*watermark*. Gambar (b) merupakan gambar yang sama, hanya saja gambar tersebut telah mengalami serangan. Gambar (c) watermark terdeteksi, gambar (d) watermark tidak terdeteksi. Gambar(e) menunjukkan kolom yang dihapus kemudian diduplikasi. Gambar (f) menunjukkan kotak putih di (e) yang diperbesar. Gambar pada bagian bawah adalah kolom yang asli.

Metode *jitter attack* berguna untuk merusak media ber-*watermark* dan tidak bertujuan untuk mendeteksi ataupun mengekstraksi *watermark* dari media penampungnya.

*Jitter attack* dapat dilakukan pada media digital yang lolos dari manipulasi sederhana, seperti rotasi, *resize*, ataupun distorsi amplitudo pada media audio.

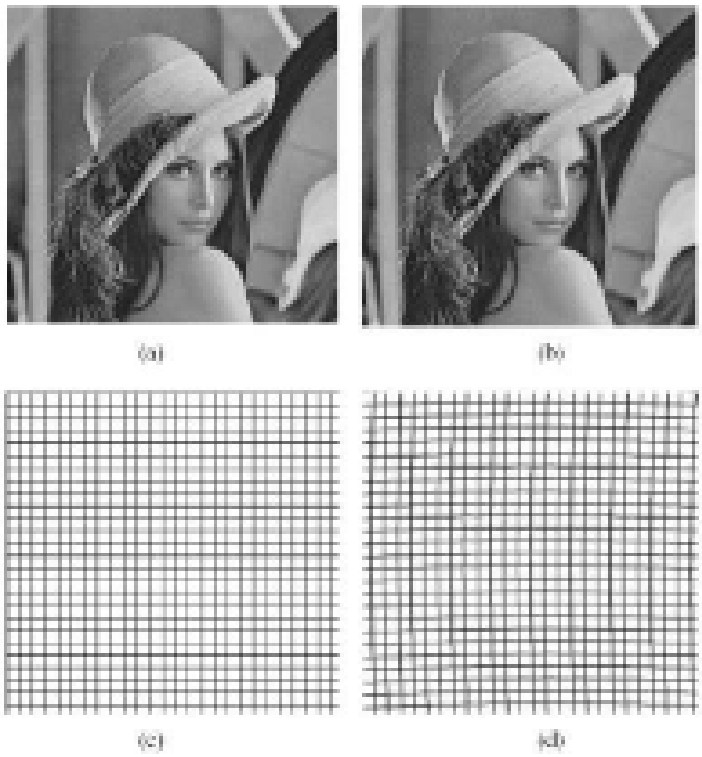
### *3. StirMark*

Pada umumnya, skema *watermarking* berhasil lolos dari manipulasi-manipulasi dasar (manipulasi yang dapat dilakukan dengan mudah menggunakan kakas standar, seperti rotasi, pemotongan, *resampling*, *resizing*, dan kompresi). Dan pada umumnya pula, skema *watermark* tidak dapat bertahan dari gabungan atau kombinasi dari

manipulasi-manipulasi dasar tersebut. Hal ini menjadi dasar pembangunan dari metode *StirMark.StirMark* sebenarnya merupakan kakas yang generik untuk menguji kekokohan dari algoritma *watermarking* pada citra.

Dalam versi yang paling sederhana, *StirMark* membangkitkan sebuah proses *resampling*. Proses resampling ini dilakukan secara digital. Proses ini dapat diumpamakan dengan cara mencetak sebuah citra ke kertas dan melakukan *scanning* terhadap citra tersebut, dalam hal ini *scanner* yang digunakan diasumsikan berkualitas sangat tinggi, sehingga hampir tidak ada perubahan jika hasil *resampling* dibandingkan dengan *sample* aslinya.

Dengan perlakuan seperti itu, citra hasil *resampling* akan memiliki distorsi secara geometris walaupun ukuran distorsinya sangat kecil. Dalam hal ini, citra akan mengalami goresan, potongan, pergeseran dan rotasi sejauh beberapa derajat yang nilainya dapat diabaikan karena nilainya sangat kecil.



**Gambar 4. StirMark**

Gambar (a) citra asli berwatermark, gambar (b) citra ber-watermark yang telah mengalami StirMark. Gambar (c) posisi pixel gambar asli berwatermawk, gambar (d) posisi pixel yang sudah mengalami distorsi. Dari gambar ini dapat dilihat bahawa pada bagian tengah terjadi pembengkokan sedangkan pada bagian pinggir, hampir tidak terjadi distorsi sama sekali.

Beberapa distorsi akibat *resampling* dengan menggunakan *StirMark*

- Distorsi geometri minor
- Deviasi *low frequency* secara random

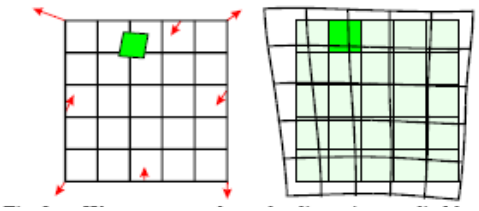
- Penempatan posisi yang salah dari *high frequency*
- *Error* yang terdistribusi
- Lengkungan saat *resampling*

Dalam proses *resampling*, dapat pula ditambahkan fungsi transfer yang akan menghasilkan *erorr-erorr* kecil yang terdistribusi di seluruh *sample*. Hal ini dapat menyamai ketidaksempurnaan proses *resample* dari proses konversi digital-analog pada *scanner* atau alat konversi lainnya.

*StirMark* sebenarnya juga mengurangi kualitas pada citra. Akan tetapi berkurangnya kualitas ini tidak dapat diindera jika hanya dilakukan sekali. Jika *StirMark* dilakukan berkali-kali, maka perbedaan kualitas cita asli dan citra hasil resampling akan tampak jelas.

Setelah beberapa kali iterasi, degradasi dari kualitas citra akan tampak. Dengan adanya distorsi geometris sederhana, dapat mengakibatkan kebingungan (*confusion*) pada sistem *marking* yang ada. Distorsi yang lebih banyak lagi dapat ditambahkan tanpa membuat perubahan yang berarti.

Jika pada citra diberikan pelengkungan secara global pada citra (*global bending*), maka citra akan mengalami deviasi pada bagian tengah citra dan hampri tidak ada pelengkungan pada bagian pinggir citra. Hal ini akan mengakibatkan pergeseran *pixel-pixel*, sehingga proses *resample* akan dilakukan berkali-kali sampai didapatkan posisi *pixel* yang jelas.



**Gambar 5. Distorsi yang diakibatkan oleh StirMark**

Dengan teknik *StirMark*, citra ber-*watermark* akan lebih sulit dideteksi, walaupun tidak merusak *watermark* di dalamnya. Kakas *watermarking*, yang tidak lolos dari *StirMark* dapat dikatakan mudah untuk dipecahkan. Kakas *watermark* yang baik setidaknya mampu lolos dari teknik *StirMark.*

Seorang yang ingin melindungi ciptaannya dengan menggunakan *watermark*, mungkin dapat berusaha untuk meningkatkan kekokohan sistem *watermarking*-nya dengan cara mengetahui teknik-teknik yang mungkin digunakan oleh para pembajak.

Sang pemilik hak cipta dapat mencoba dengan cara memasukkan (*embedding*), berbagai versi *watermark* dengan fungsi invers yang sesuai. Versi watermark di sini bukan berarti perbedaan *content,* namun versi adalah *watermark* yang dihasilkan dengan teknik yang berbeda-beda. Misalnya saja, Ó Ruanaidh dan Pereira menyarankan untuk menggunakan transformasi Fourier-Mellin untuk mengatasi rotasi dan skala.

Bagaimanapun, inti dari serangan *StirMark* adalah untuk menciptakan distorsi atau kombinasi dari distorsi yang akan merusak ataupun membuat *watermark* menajadi tidak terbaca (*unreadable*). Seperti yang telah disebutkan di atas, *StirMark* termasuk ke dalam katgegori *robustness attack*, yaitu serangan yang bertujuan untuk menguji kekokohan *watermark*, bahkan sampai merusak *watermark*.

Cara menghadapi *StirMark* adalah dengan cara :

- *Attach a registration pattern* (menempelkan pola registrasi)
- *Do image registration before detection if original image available* (melakukan registrasi sebelum terjadinya pendeteksian jika media asli dimiliki)
- *Embed watermark in the transform (e.g., RST) invariant domain* (memasukkan *watermark* dalam *domain* transformasi).

### *4. Mosaic Attack*

*Mosaic Attack* termasuk ke dalam *presentation attack*, yaitu serangan yang memodifikasi media yang telah disisipi *watermark* untuk menghindari pendeteksian terhadap *watermark* yang telah disisipkan

Pada umumnya, hal ini terjadi pada *browser*. Dalam mengakses gambar di *internet*, sering kali gambar tersebut dipecah dahulu menjadi beberapa bagian. Hal ini memang mempercepat waktu *download*, akan tetapi dapat berakibat *watermark* yang ada menjadi tidak terbaca. Walaupun pada dasarnya proses *rendering* dilakukan dengan cara yang sama satu demi satu *pixel*.
Biasanya serangan ini dilakukan secara tidak sengaja oleh pihak pengelola *web*, yang hanya bermaksud untuk mempercepat waktu akses ke situsnya. Dengan adanya pemotongan gambar, maka gambar dapat di-*load* bagian per bagian sehingga membentuk gambar aslinya.

Hal yang menjadi masalah adalah, setiap skema *watermarking* memiliki ukuran minimum untuk menyembunyikan sebuah informasi rahasia (Tidak mungkin pesan rahasia disembunyikan hanya dalam satu *pixel*). Dengan cara memecah gambar menjadi potongan=potongan kecil, maka alat pendeteksi *watermark* (*mark detector*) akan mengalami kebingungan (*confusion*). Hal yang dapat diharapkan dari sebuah alat pemberi *watermark* adalah adalah ukuran minimum dapat sekecil mungkin, sehingga jika terjadi pemecahan gambar menjadi bagian-bagian kecil, maka *watermark* masih dapat terdeteksi.

Cara menghadapi *mosaic attack* adalah dengan cara mengurangi ukuran *watermark* seminimum mungkin sehingga cocok dengan ukuran minimum yang masih dapat terdeteksi.

**Gambar 6. Mosaic Attack**

**Cara Mengatasi Serangan terhadap Teknik *Watermarking***

Dalam menghadapi berbagai serangan terhadap *watermark*, dapat dilakukan dengan berbagai cara, yaitu :

- *Preventing Unauthorized Embedding*
  Cara ini digunakan dengan menggunakan kakas *digital signature* untuk mencegah modifikasi. Selain itu, dapat juga digunakan metode *copy attack*, yaitu *watermark* bergantung kepada media penampungnya.

- *Preventing Unauthorized Detection*
  Men-*decode content*, dilakukan dengan cara mengenkripsi terlebih dahulu *watermark* sebelum dimasukkan ke dalam media penampung. Atau dapat pula dilakukan dengan mendeteksi keberadaan *watermark*.

- *Preventing Unauthorized Removal*
  Biasanya teknik ini bergantung pada jenis serangan tertentu.

Memasukkan versi *watermark* yang berbeda dengan berbagai macam teknik untuk menghindari manipulasi dengan berbagai kombinasi. Perbedaan versi ini maksudnya adalah dengan cara memasukkan *watermark* dengan teknik yang berbeda-beda.

**Kesimpulan**

Teknik *watermark* yang baik harus memenuhi persyaratan yang telah didefinisikan. Namun hal tersebut tidak mutlak adanya. Teknik atau skema yang baik, harus dapat lolos dari berbagai jenis serangan, setidaknya serangan-serangan dasar pada teknik *watermarking*.

Banyak skema *watermarking* dapat lolos dari serangan-serangan dasar, tetapi tidak dapat bertahan dari kombinasi serangan-serangan dasar. Untuk itu perlu diimplementasikan berbagai versi *mark* yang dihasilkan dari teknik yang berbeda, sehingga jika satu *mark* rusak karena sebuah serangan, masih terdapat *mark* lainnya yang dapat bertahan dari serangan tersebut.

**Daftar Pustaka**


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

[2] Johnson, Neil F., Jajodia, Sushil. Steganalysis : The Investigation of Hidden Information. http://citeseer.ist.psu.edu

[3] Peticolas, Fabien A.P., Anderson, Ross J., Kuhn, Markus G. Attacks on Copyright Marking Systems. http://citeseer.ist.psu.edu

[4] Avcibas, Ismail., Memon, Nasir., Sankur, Bulent. Steganalysis Based on Image Quality Metrics. http://citeseer.ist.psu.edu

[5] R.J. Anderson, F. Petitcolas (1998) On the limits of steganography, IEEE Journal of Selected Areas in Communications, 16(4), 474-481.

[6] E. Cole (2003) *Hiding in plain sight : steganoication*, Willey Publ. Inc.

[7] F. Petitcolas (2000) *Information hiding for steganography and digital watermarking*. Artech House Inc., Norwood, MA.

[8] F. Petitcolas, R. J. Anderson, M. G. Kuhn (1999) Information hiding : a survey, Proc. of the IEEE, 87(7), 1062-1078.

[9] Ross J. Anderson. Why cryptosystems fail. Communications of the ACM, 37(11):32{40, November 1994.

[10] Marc Cooperman and Scott A. Moskowitz. Steganographic method and device.US Patent 5,613,004, March 1995.

[11] Ingemar J. Cox, Joe Kilian, Tom Leighton, and Talal Shamoon. A secure, robust watermark for multimedia. In Anderson [3], pages 183{206.

[12] Ingemar J. Cox and Matt L. Miller. A review of watermarking and the importance of perceptual modeling. In Rogowitz and Pappas [57].

[13] E. Koch and J. Zhao. Towards robust and hidden image copyright labeling. In Workshop on Nonlinear Signal and Image Processing, pages 452{455, Neos Marmaras, Greece, June 1995. IEEE.

[14] Markus G. Kuhn and Fabien A. P. Petitcolas. StirMark. <http://www.cl.cam.ac.uk/~fapp2/watermarking/stirmark/>, November 1997.

[15] Chandramouli, R. A Mathematical Approach to Steganalysis. http://citeseer.ist.psu.edu

[16] Antoine, Jean-Yves. Networks Security. Université François Rabelais de Tours

[17] Kutter M., Petitcolas, F.A.P. A Fair Benchmark for Image Watermarking Systems. University of Cambridge.

[18] Zhao, Hong. Watermark Attacks.

[19] Langelaar, R. Langedijk, J. Biemond, *Removing Watermarks by Nonlinear Filtering*, Proc. European Signal Processing, Rhodes, Greece, Sept. 1998

[20] M. Kutter, S. Voloshynovskiy and A. Herrigel, *The watermark Copy Attack*, Security and Watermarking of Multimedia Contents, II, SPIE-3971: 371-280, 2000

[21] I. Cox, M. Miller, J. Bloom, *Digital Watermarking*, (chap. 9) Morgan Kaufmann Pub., 2001

[22] Lee, Han Ho., Lee, Jung Soo., Lee, Nam Yong., Kim, Jong Weon. Image Watermarking for Semi - fingerprinting