

# Studi Mengenai Teknik *Digital Watermarking* pada Berkas MP3

Muhammad Dhito Prihardhanto - 13507118

Prodi Teknik Informatika, Sekolah Teknik Elektro dan Informatika (STEI)  
Institut Teknologi Bandung, Jalan Ganeca 10 Bandung, email: [muh\\_dhito@students.itb.ac.id](mailto:muh_dhito@students.itb.ac.id)

**Abstrak** – *Digital watermarking* saat ini dirasakan menjadi suatu kebutuhan, terutama bagi perindustrian dalam bidang multimedia yang dihadapkan permasalahan yang cukup sulit diatasi yaitu masalah pembajakan karya orang lain (*copyright piracy*). Untuk itulah terus dilakukan riset mengenai algoritma digital watermarking yang mampu memenuhi konstrain-konstrain yang ada. Untuk itulah ditentukanlah beberapa karakteristik yang harus dimiliki oleh suatu algoritma. Pada akhirnya teknik digital watermarking yang ada sampai sekarang ini terus dikembangkan dan berusaha memenuhi karakteristik yang diharapkan tersebut dengan menggunakan berbagai algoritma. Dalam makalah ini akan diuraikan beberapa teknik digital watermarking yang ada saat ini. Teknik digital watermarking yang bisa sebagian besar adalah teknik-teknik yang sifatnya berupa hasil riset yang terus dikembangkan.

**Kata kunci:** teknik digital watermarking, berkas audio mp3

## 1. PENDAHULUAN

Dewasa ini pelanggaran hak cipta dengan melakukan pembajakan (piracy) atau penggandaan karya orang lain, terutama yang berbentuk digital, menjadi isu yang santer dibicarakan di dalam masyarakat. Pembajakan terhadap karya-karya digital tersebut begitu mudah terjadi karena seseorang cukup tinggal melakukan klik kanan copy pada suatu berkas digital, baik itu gambar, audio, video, dsb. dan melakukan klik kanan paste ke tempat yang lain. Itulah yang menjadi kelemahan pada berkas-berkas digital, yaitu masalah perlindungan hak ciptaan begitu leluasanya masyarakat melakukan penggandaan terhadap berkas digital.

Digital watermarking merupakan cara yang kini banyak digunakan untuk menjawab persoalan tersebut. Watermarking tersebut bertujuan untuk memberikan perlindungan hak cipta (copyright) serta menunjukkan bukti kepemilikan (ownership) yang sah terhadap suatu berkas digital.

Dalam perkembangannya banyak teknik atau metode yang dihasilkan sekarang untuk melakukan *watermarking* tersebut dan setiap teknik berbeda bergantung pada tipe

berkas yang akan diberikan *watermark*. Teknik *watermarking* pada suatu jenis berkas juga terdapat banyak variasinya.

Namun, khusus *watermarking* pada berkas-berkas audio, seperti mp3 memiliki tantangan tersendiri. Diperlukan kehati-hatian dalam melakukan perancangan algoritma *watermarking* karena suara memang lebih sensitive daripada citra. Sehingga berkas mp3 akan rentan sekali untuk rusak bila ditambahkan *watermarking*. Microsoft sendiri terus melakukan penelitian demi mengembangkan sistem *watermarking* untuk audio digital. Dengan sistem *watermarking* tersebut informasi lisensi disisipkan ke dalam sebuah berkas musik, sehingga aplikasi *Windows media player* hanya akan memutar berkas music yang terdapat *watermark* tersebut.

## 2. FORMAT MP3

*Layout* dari berkas audio *filestream* MPEG telah didefinisikan di dalam [ISO1993] untuk MPEG I dan di dalam [ISO1995] untuk MPEG II. MPEG *Audio Layer 3* atau yang biasa dikenal dengan MP3 merupakan *filestream* yang terdiri atas objek-objek kecil yang disebut *frame*. Setiap *frame* memiliki *frame header* masing-masing di mana *header* tersebut merupakan bagian terkecil dari berkas MP3 yang dapat diinterpretasi tanpa informasi tambahan lagi.

Untuk semua layer MPEG, khususnya untuk MPEG I atau MPEG II, *frame* audio selalu memiliki struktur yang terdiri atas header, CRC *Error Check* (opsional), *audio data*, *ancillary data* (opsional). *Header* digunakan untuk proses sinkronisasi perangkat lunak *media player* untuk *frame start* pada *data stream* dan untuk proses dekode *audio data* secara umum. *Header* mengandung informasi tentang *sample rate*, mode (stereo/mono), MPEG ID (MPEG-1/MPEG-2), banyaknya layer MPEG, dsb.

Sedangkan *audio data* pada frame MP3 mengandung layer khusus dengan *header* yang disempurnakan yang disebut dengan *side info*, sampel kode Huffman, dan faktor skala yang terkait. Faktor skala digunakan untuk merekonstruksi *multiplier* untuk kuantisasi terbalik (*inverted quantization*) dari sampel yang telah terkodekan. Setiap faktor skala disimpan dalam 0-4 bit yang berurutan.

### 3. KARAKTERISTIK DAN ISU TEKNIK WATERMARKING BERKAS AUDIO

Sebelum masuk ke penjelasan beberapa teknik *watermarking* berkas mp3 yang ada saat ini, akan disebutkan terlebih dahulu mengenai beberapa atribut yang harus dimiliki oleh suatu algoritma *watermarking* berkas mp3 atau audio pada umumnya agar efisien. Setelah dirangkum dari beberapa literature, beberapa atribut tersebut yaitu:

- *Inaudibility*, tanda air (*watermark*) yang disisipkan dalam suatu berkas audio tidak sampai menurunkan kualitas dari berkas audio itu sendiri, dan tidak mudah diketahui.
- *Statistical invisibility*, algoritma tersebut harus mampu mencegah berkas audio dari penyisipan/penghapusan *watermark* yang tidak sah, atau dengan kata lain dia mampu mendeteksi usaha perbahaan yang terjadi. Di samping itu, seorang user yang mengetahui algoritma *watermarking* yang digunakan tidak serta merta membuatnya dapat menghapus *watermark* tersebut.
- Karakteristik kompresi mirip dengan sinyal yang asli.
- Langsung tersisipkan pada data.
- *Robustness*, algoritma harus memiliki ketahanan terhadap berbagai variasi serangan yang dilancarkan *user*.
- Mendukung banyak tanda air (*watermark*).
- *Low redundancy*.
- *Self-clocking*.

Dari beberapa literature yang ada juga menyebutkan bahwa kebanyakan teknik yang diajukan dalam melakukan *watermarking* suatu berkas audio adalah dengan cara menyisipkan *watermark* pada PCM-data (*raw audio data*). Hal ini dapat menjamin bahwa teknik ini akan mampu bekerja untuk semua format audio karena teknik ini mensyaratkan bahwa berkas audio akan dikompresi dalam rangka untuk melakukan penyisipan tanda air (*watermark*) dan kemudian akan melakukan dekompresi. Akan tetapi, pendekatan ini dapat menciptakan dua masalah yang cukup signifikan. Pertama, tidak terjamin apakah *watermark* dapat bertahan dalam melalui prosedur pengkodean (*encoding*) ataupun proses pengkodean (*decoding*) dan kedua, adalah proses keduanya tersebut akan memakan waktu, dimana waktu merupakan suatu kemewahan yang tidak tersedia di dalam aplikasi *real-time*.

Agar mencapai kemampuan pemutaran yang *real-time*, data audio pada umumnya disimpan dan ditransmisikan dalam format terkompresi, seperti format MPEG. Dengan

demikian, skema *watermarking* akan lebih suka menargetkan domain data terkompresi ketika mereka akan dijalankan pada aplikasi *real-time*. Ada masalah terkait dengan kemampuan adaptasi (*adaptability*) teknik *watermarking* dengan pendekatan ini karena teknik ini berorientasi kepada format audio tertentu sehingga terpaksa teknik tersebut harus berubah agar mampu beroperasi dengan format audio lainnya.

Isu yang cukup penting dalam konteks *watermarking* audio adalah apakah sinyal asli harus digunakan untuk mendeteksi *watermark* atau tidak. Penggunaan sinyal asli untuk pendeteksian bukanlah hal yang ingin dilakukan karena akan membuang-buang sejumlah ruang penyimpanan yang besar serta menambah ancaman bahaya dari penggunaan oleh *malicious user*.

Isu lain adalah perbedaan teknik-teknik *watermarking* adalah apakah *watermark* tersisipkan di dalam domain ruang atau frekuensi. Biasanya, ketika digunakan domain waktu, maka distorsi waktu dapat diatasi dengan baik, namun harus dilakukan penanganan khusus untuk distorsi spektral. Dalam domain frekuensi yang terjadi sebaliknya. Beberapa metode yang menggunakan domain frekuensi mengeksplorasi karakteristik frekuensi sinyal audio dalam rangka menyisipkan *watermark*, sehingga meminimalkan distorsi suara bahkan untuk *watermark* beramplitudo tinggi sekalipun.

### 4. BEBERAPA TEKNIK DIGITAL WATERMARKING UNTUK BERKAS MP3

#### 4.1 Digital Watermarking dengan Kunci Watermark oleh Dimitrios Koukopoulos, and Yiannis Stamatou

Salah satu teknik *digital watermarking* untuk berkas audio MP3 adalah dengan bekerja di dalam domain terkompresi, membuat manipulasi dalam waktu dan domain *channel/subband* dan tidak membutuhkan sinyal asli untuk melakukan pendeteksian.

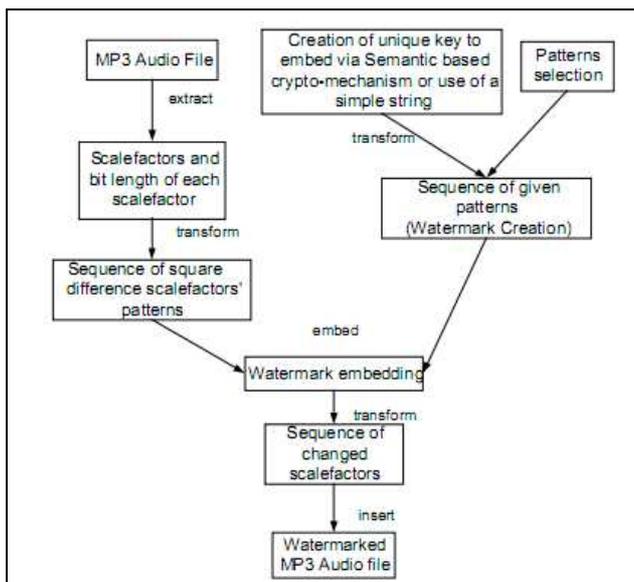
Teknik ini diterapkan dalam lingkungan *real-time* melakukan penanganan khusus untuk penggunaan yang efisien terhadap dua sumber daya yang bersifat terbatas dari sistem komputer, yaitu waktu dan ruang.

Skema ini dirancang dan dilaksanakan sedemikian rupa sehingga penundaan (*delay*) menjadi berkurang dan persyaratan memori dapat sekecil mungkin. Algoritma ini mampu mengatasi masalah algoritma yang beroperasi di domain PCM-data yang sangat rentan terhadap serangan kompresi/rekompresi karena menyisipkan *watermark* dalam domain factor skala dan bukan di bagian data audio yang dibayangi atau tidak dapat dimengerti karena hukum *psycho-acoustic*. Selain itu juga tidak ada kebutuhan untuk melakukan pengkodean (*encoding*) secara khusus

pada data audio agar menjadi berkas MP3 seperti yang dilakukan oleh aplikasi *MP3Stego*.

Algoritma ini menerapkan suatu mekanisme control untuk memantau perubahan-perubahan pada factor skala pada berkas sumber saat tahap penyisipan *watermark*. Algoritma ini juga memberikan kepada pelaku industri penyempurnaan kemampuan penerapan mekanisme kriptografi pada bagian penciptaan *watermark* dari algoritma. Mekanisme ini menciptakan sebuah kunci *watermarking* dengan arti semantik yang unik untuk masing-masing pengguna dan ketahanan yang kuat yang memiliki kapabilitas dalam bertahan terhadap serangan dengan potensi kerusakan yang signifikan.

Algoritma yang telah dirancang ini bisa dikatakan merupakan salah satu algoritma pertama yang dapat digunakan untuk otentikasi dan perlindungan hak cipta tanpa penggunaan sinyal asli. Hal ini karena kemampuan yang diberikan algoritma ini kepada pengguna industri dengan menggunakan mekanisme kriptografi untuk menciptakan kunci *watermarking*.



Gambar 1: Algoritma utama

Dalam algoritma ini, masukannya adalah berkas MP3, kunci unik yang dihasilkan melalui algoritma penciptaan kunci *watermarking* atau berupa teks sederhana dan satu set dari tiga pola yang mewakili bit “0” dan “1” dan sebuah bit sinkronisasi, digunakan antara bit-bit tertentu dan pada akhir kunci penyisipan (*embedding key*) untuk *self-clocking* dan ketahanan terhadap *cropping*.

Langkah selanjutnya adalah metransformasikan string biner yang mewakili kunci *watermarking* menjadi sekuens pola yang diberikan dan ekstraksi faktor skala dari frame-frame pada berkas MP3. Faktor skala pada berkas MP3

memeiliki beberapa varian bit yang tergantung dari 0-4 bit, sedangkan pada format MPEG audio layer 1 dan 2 terdiri dari jumlah bit yang tetap yaitu 6.

Karena itu bersama dengan faktor skala, arus informasi yang lain diekstrak oleh berkas MP3, yang menentukan jumlah bit yang digunakan dalam perwakilan bit dari masing-masing faktor skala. Berdasarkan faktor-faktor skala, perbedaan pola dihitung. Pola-pola ini berkorespondensi dengan perbedaan antara faktor skala pertama yang dapat dianggap sebagai titik awal dengan faktor skala berikutnya pada *stream* faktor skala. Sebagai contoh, asumsikan ada daftar faktor skala {6,8,5,3}, faktor skala pertama dianggap sebagai titik awal dan polanya adalah{2, -1, -3}.

Pada langkah berikutnya pola urutan yang diproduksi oleh kunci *watermarking* disisipkan ke dalam urutan (*sequence*) dari pola-pola berbeda yang dihasilkan oleh faktor skala pada berkas MP3. Penyisipan dapat dicapai setelah melalui mekanisme yang mengubah pola-pola berbeda faktor skala sehingga didapatkan sejumlah dari mereka yang sesuai dengan pola-pola urutan yang dikehendaki. Mekanisme tersebut menjamin bahwa tidak akan pernah terdengar distorsi pada berkas MP3 yang ber-*watermark* dan setiap pola yang diubah tidak akan melanggar jumlah bit yang didasari pada faktor skala.

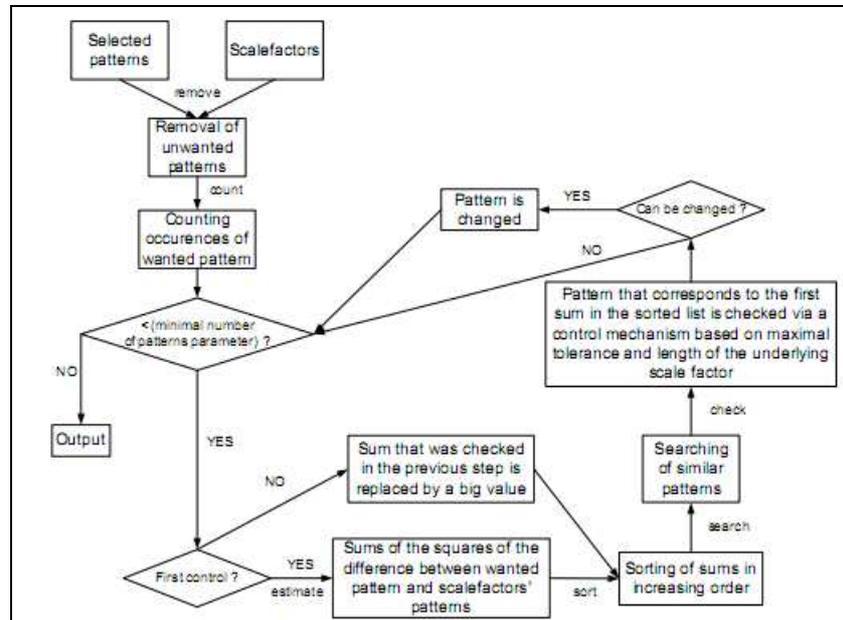
*Goal* kedua dari diterapkannya mekanisme tersebut adalah dicapainya dengan menerapkan kontrol terhadap sub-mekanisme yang menangani input *stream* yang menghantarkan informasi mengenai banyaknya bit yang terdiri pada setiap faktor skala. Terakhir, urutan pola yang dihasilkan dari langkah sebelumnya tadi ditransformasi menjadi urutan faktor skala yang disisipkan dalam berkas MP3 sumber, terciptalah berkas MP3 yang telah ber-*watermark*.

### Mekanisme Penyisipan

Di samping mengetahui algoritma utama yang telah diuraikan di atas, sebaiknya juga perlu diketahui mengenai mekanisme penyisipan *watermark* yang ada di dalamnya. Mekanisme penyisipan ini menerima input pola yang ingin disisipkan, pola-pola yang tidak diinginkan, dan faktor skala dari frame-frame di mana telah ditentukan sebelumnya untuk disisipkan dalam pola tersebut. Mekanisme penyisipan juga menggunakan parameter yaitu jumlah minimum pola pada area frame-frame harus sama dengan pola bit informasi yang ingin disisipkan agar dapat mempertimbangkan pola yang telah disisipkan, toleransi maksimum yang menentukan perubahan yang dapat dialami oleh pola, untuk mencocokkan pola yang diinginkan dan aliran dari bit faktor skala, atas pola yang diperiksa.

Langkah pertama adalah menyelidiki aliran pola faktor-faktor skala untuk menemukan pola yang tidak diinginkan dan melakukan penghapusan dengan mengubah mereka.

Langkah berikutnya adalah mencari pola-pola yang mirip dengan pola yang diinginkan dan menghitung jumlahnya



Gambar 2: Mekanisme penyisipan watermark

Jika pola yang diinginkan muncul beberapa kali dan melebihi parameter yang menspesifikasikan jumlah minimal atas kemunculannya maka mekanisme tersebut telah memenuhi dan mekanisme tersebut akan mengekstrak pola sebagai hasilnya.

Jika tidak, maka mekanisme tersebut akan mencoba menemukan pola-pola serupa dengan pola yang diinginkan. Spesifikasi dari pola yang serupa itu didiapatkan dengan menggunakan mekanisme kontrol yang menggunakan parameter toleransi maksimal mengenai *audibility* dan jumlah bit yang terdiri pada setiap faktor skala. Pertama, jumlah kuadrat dari selisih pola dari semua faktor skala yang diberikan dengan yang diinginkan diperkirakan dan diurutkan membesar.

Kemudian, pola yang jumlahnya ada pada awal daftar yang telah diurutkan, dipilih sebagai kandidat untuk berubah. Pola ini diperiksa oleh mekanisme kontrol yang dispesifikasikan jika parameter toleransi dipenuhi dan ukuran bit yang didasarkan pada faktor skala tidak dilanggar. Jika hasil pemeriksaan tersebut sukses, maka pola tersebut digantikan oleh pola yang dikehendaki.

Kemudian, diperiksa lagi apakah jumlah minimal parameter pola dipenuhi. Jika tidak, jumlah yang bersesuaian pada daftar terurut tadi akan digantikan dengan nilai yang besar, kemudian daftar tadi akan diurutkan lagi dan prosedur yang sama diulangi. Prosedur

ini dapat dianggap sebagai satu langkah untuk membandingkan prosedur-prosedur di mana pencarian pola-pola yang serupa terdiri atas sebuah loop yang didasari pada peningkatan satu per satu dari parameter toleransi sampai toleransi maksimum tercapai.

Dengan demikian, sebagian besar perubahan akan mempengaruhi pola-pola yang berbeda sedikit mungkin dari pola yang diinginkan dan meminimisasi distorsi pada kualitas suara, mengingat jumlah bit yang didasarkan pada faktor skala tidak dilanggar.

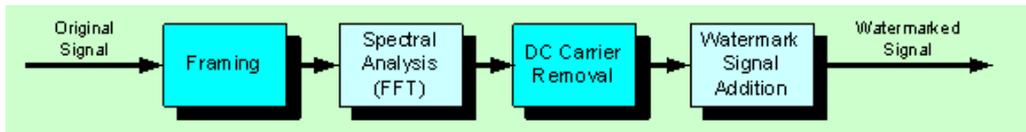
#### 4.2 Teknik Digital Watermarking yang Digunakan oleh Perusahaan Audiobox

**Audiobox** adalah sebuah proyek yang mengembangkan produk-produk *perceptual audio*. Salah satu proyek yang dijalankannya adalah pembuatan *watermarking* untuk berkas musik digital.

Skema *watermarking* yang digunakan adalah dengan menyembunyikan data *watermark* pada komponen berfrekuensi rendah pada sinyal suara, di mana berada di bawah batas sistem pendengaran manusia.

Proses penyisipan *watermark* pada sebuah berkas audio dapat dibagi menjadi 4 proses (gambar 3). Berkas asli audio menjadi masukan bagi sistem, diaman secara subsekuen dilakukan *framing*, *analyzing*, dan *processing*

sehingga dihasilkan sinyal output dengan *audible watermark* yang telah tersisipkan.



Gambar 3: Proses penyisipan watermark

Berkas audio dibagi-bagi menjadi frame-frame dengan durasi masing-masing 90 mili second. Ukuran frame ini dipilih sedemikian rupa agar *watermark* yang telah tersisipi nantinya tidak akan menimbulkan distorsi pada berkas.

Dengan ukuran 90 ms setiap *frame*-nya, maka *bit rate* untuk data ber-*watermark* adalah sama dengan  $1 / 0.09 = 11.1$  b/s (bits per second).

Setelah proses *framing*, selanjutnya adalah melakukan analisis spectral pada sinyal, terdiri atas sebuah *fast Fourier Transform (FFT)*, di mana akan memungkinkan untuk menghitung frekuensi rendah pada setiap frame dengan baik. Pemrosesan FFT tersebut dilakukan dengan menggunakan aplikasi *Mathlab*, dengan persamaan sebagai berikut:

$$F(k) = \sum_{n=1}^N f(n) e^{\frac{-j2\pi(n-1)(k-1)}{N}} \quad k = 1, 2, \dots, N$$

(*N* adalah frame terakhir pada berkas audio)

Dari FFT, selanjutnya akan dapat ditentukan frekuensi rendah (DC) pada frame juga daya spektral frame (*frame spectral power*). Daya frame dapat dihitung dengan:

$$P_{Frame}(n) = \frac{1}{\left(\frac{3969}{2} + 1\right)} \sum_{k=1}^{3969+1} F(k)^2 \quad n = 1, 2, \dots, N$$

Dari perhitungan tersebut akan didapatkan grafik yang menggambarkan frekuensi-frekuensi yang mungkin pada skala 0-4000 Hz. Setelah itu dapat ditentukan berapakah frekuensi rendah (DC) dari setiap frame yang kemudian akan diambil dengan melakukan pengurangan dari setiap frame menggunakan rumus:

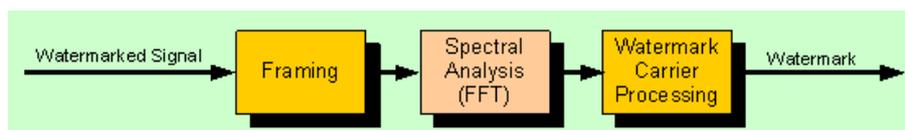
$$f(n) = \sum_{k=1}^{3969n} f(k) - F(1) \quad n = 1, 2, \dots, N$$

Setelah analisis spektral selesai, sekarang saatnya untuk melakukan penambahan sinyal *watermark* ke dalam frekuensi rendah (DC) yang telah diambil tadi. Ukuran *watermark* yang ditambahkan bergantung kepada rumus:

$$f(n) = \sum_{k=1}^{3969n} f(k) + K_s \times w(n) \times P_{Frame}(n) \quad n = 1, 2, \dots, N$$

Di mana *K* di atas adalah faktor skala, yang dapat menjamin bahwa *watermark* yang disisipkan akab berada di bawah ambang batas pendengaran manusia. Sementara itu *w(n)* merepresentasikan data sinyal *watermark*, yang hanya ada dua kemungkinan yaitu bernilai 1 atau -1.

Untuk proses ekstraksi *watermark* dari berkas audio yang telah ber-*watermark* langkah yang dilakukan adalah hampir sama dengan proses penyisipan *watermark* yaitu sebagai berikut:



Gambar 4: Proses ekstraksi watermark

## 5. KESIMPULAN

Dari ulasan di atas, ada suatu desain algoritma *watermarking* berkas audio termasuk MP3 memiliki karakteristik yang ingin dipenuhi, yaitu meliputi:

*Inaudibility, Statistical invisibility, Robustness, support multiple watermark, Low redundancy, dan Self-clocking.*

Di antara teknik-teknik yang ada semuanya melakukan pendekatan agar memenuhi karakteristik-karakteristik tersebut. Dalam contoh dua teknik yang telah diuraikan di atas, algoritma pertama menggunakan pendekatan penyisipan *watermark* dengan berusaha melakukan efisiensi agar tidak memakan ruang dan waktu yang besar. Teknik yang dilakukannya adalah menggunakan kunci *watermark* dan berdasarkan faktor skala. Sementara algoritma yang kedua lebih berusaha menekankan kepada pembuatan *watermark* yang tidak dapat dideteksi oleh pendengaran manusia, atau dengan kata lain tetap berusaha menjaga kualitas suara meskipun terdapat *watermark watermark* di dalamnya.

Sementara itu dari literature yang didapatkan penulis, riset mengenai algoritma *watermarking* untuk berkas audio ini terus dilakukan terutama, lebih ditekankan kepada aspek *robustness*-nya dan algoritma yang mampu diterapkan untuk semua jenis format berkas audio.

#### DAFTAR PUSTAKA

- [1] Munir, Rinaldi. 2004. Bahan Kuliah IF5054 Kriptografi. Departemen Teknik Informatika, Institut Teknologi Bandung.
- [2] <http://business.highbeam.com/436128/article-1G1-144932732/watermarking-scheme-mp-audio-files>, diakses pada tanggal 3 Maret 2010
- [3] <http://www.ece.uvic.ca/~aupward/w/watermarking.htm>, diakses pada tanggal 3 Maret 2010
- [4] <http://www.neurocode.de/?p=89>, diakses pada tanggal 3 Maret 2010
- [5] <http://www.petitcolas.net/fabien/steganography/mp3stego/>, diakses pada tanggal 23 Maret 2010
- [6] [http://www.sans.org/reading\\_room/whitepapers/steganography/mp3stego\\_hiding\\_text\\_in\\_mp3\\_files\\_550?show=550.php&cat=steganography](http://www.sans.org/reading_room/whitepapers/steganography/mp3stego_hiding_text_in_mp3_files_550?show=550.php&cat=steganography), diakses pada tanggal 23 Maret 2010
- [7] Koukopoulos, Dimitrios dan Stamatou, Yiannis. 2005. An Efficient Watermarking Method for MP3 Audio Files. The 7th Proceedings of World Academy of Science, Engineering and technology.