

Audio Watermarking dengan Metode *Direct Sequence Spread Spectrum* untuk Konten Musik Digital

Rinanza Zulmy Alhamri
Program Studi Informatika
Universitas Sebelas Maret
Jl. Ir. Sutami No 36 A, Surakarta
rinanza.z.alhamri@gmail.com

Esti Suryani, S.Si, M.Kom
Program Studi Informatika
Universitas Sebelas Maret
Jl. Ir. Sutami No 36 A, Surakarta
suyapalapa@yahoo.com

Wisnu Widiarto, S.Si, M.T
Program Studi Informatika
Universitas Sebelas Maret
Jl. Ir. Sutami No 36 A, Surakarta
bethoro_wisnu@uns.ac.id

ABSTRAK

Berkembangnya teknologi menimbulkan pelanggaran hak cipta pada konten musik digital semakin rawan terjadi. Sebagai usaha perlindungan hak cipta terhadap konten musik digital, diperkenalkan teknik audio watermarking untuk menyisipkan tanda bukti kepemilikan pada konten musik digital. Penerapan audio watermarking dengan metode *Direct Sequence Spread Spectrum (DSSS)* memungkinkan tanda kepemilikan bisa dibubuhkan ke dalam konten musik digital. Metode DSSS menerapkan encoding terhadap tanda bukti kepemilikan menggunakan *Pseudo Random-Noise Sequence* untuk memperoleh data watermark. Pembangkitan *PN sequence* menggunakan algoritma *Linear Congruential Generator (LCG)*. *PN sequence* yang telah dibangkitkan digunakan sebagai kunci watermark. Ekstraksi watermark dari konten musik digital dilakukan secara *informed detection*. Musik digital yang telah diberi watermark memiliki kualitas suara yang baik dengan rata-rata *Peak Signal to Noise Ratio (PSNR)* sebesar 42,7 dB. Hasil ekstraksi watermark terhadap musik digital yang telah mengalami pengujian kompresi, resampling, serta cropping memiliki tingkat kesesuaian karakter sebesar 100% terhadap isi text tanda bukti kepemilikan yang telah dibubuhkan sebelumnya.

Kata Kunci

Audio Watermarking, *Direct Sequence Spread Spectrum*, *Pseudo Random – Noise Sequence*

1. PENDAHULUAN

Mudahnya penggandaan terhadap konten musik digital tanpa mengalami penurunan kualitas [1] serta dengan fasilitas internet sebagai jaringan yang menghubungkan masyarakat luas [2] membuat distribusi konten musik digital semakin sulit dikendalikan. Pelanggaran terhadap konten musik digital tetap saja dimungkinkan untuk terjadi meskipun konten musik digital tersebut diproduksi menggunakan teknologi tinggi terutama pelanggaran yang menyangkut hak cipta yang saat ini sering terjadi. *Audio watermarking* merupakan proses pembubuhan suatu informasi ke dalam suatu berkas audio digital. Pembubuhan tanda digital dilakukan secara sembunyi dengan tujuan tidak mengurangi kualitas audio [3]. Dengan algoritma *audio watermarking* maka suatu berkas digital seperti tanda bukti kepemilikan dapat dibubuhkan ke dalam konten musik digital. Meskipun secara langsung tidak diperuntukkan sebagai proteksi penggandaan ilegal ataupun aksi pembajakan, namun setidaknya tanda bukti

kepemilikan ini bisa membantu sebagai identifikasi sumber kepemilikan [4].

Salah satu metode yang digunakan pada proses *audio watermarking* adalah metode *direct sequence spread spectrum (DSSS)*. Metode DSSS merupakan metode yang terlebih dahulu melakukan *encoding* pada tanda digital menggunakan *Pseudo Random – Noise Sequence (PN sequence)* untuk memperoleh data watermark, untuk kemudian watermark tersebut dibubuhkan ke dalam berkas audio [5]. Untuk memperoleh tanda digital kembali maka proses ekstraksi dilakukan dengan menghitung selisih nilai sinyal musik digital watermark dengan konten musik digital yang asli [6]. Sinyal watermark yang telah diperoleh di-decode untuk didapatkan berkas teks digital kembali berupa tanda bukti kepemilikan.

Tanda bukti kepemilikan yang digunakan sebagai watermark merupakan berkas teks digital dengan format *text* berekstensi *.txt*. Konten musik digital yang digunakan merupakan berkas audio digital dengan format *wave* berekstensi *.wav*. Tujuan dari penelitian ini adalah penerapan *audio watermarking* dengan metode DSSS yang dapat membubuhkan tanda bukti kepemilikan ke dalam konten musik digital. Diharapkan dengan penelitian ini kepemilikan yang telah dibubuhkan pada konten musik digital memiliki ketahanan yang cukup kuat sehingga tidak akan mudah rusak akibat manipulasi terhadap sinyal audio serta tidak mengurangi kualitas dari konten musik digital secara signifikan.

2. TINJAUAN PUSTAKA

Beberapa penelitian *audio watermarking* telah dikembangkan oleh para peneliti. Penelitian untuk memperhalus hasil dari watermark yang akan dibubuhkan dalam file *cover* pernah dilakukan sehingga diperoleh *audio watermarked* yang *inaudible*. Pada penelitian ini diperoleh $K_{\mu}(x)$ dimana K merupakan parameter penyeimbang fungsi pada *chip sequence* [7].

Implementasi *audio watermarking* juga telah dikembangkan dengan hadirnya penelitian implementasi steganografi pada berkas audio digital berformat MP3 menggunakan teknik *spread spectrum*. Penelitian ini memperoleh keberhasilan dalam mengimplementasikan proses steganografi pada audio. Hasil pengujian kualitas audio berdasarkan nilai PSNR (*Peak Signal to Noise Ratio*) diperoleh sebagai berikut, untuk file pesan berupa teks memiliki nilai PSNR 41.75dB sedangkan file pesan berupa gambar bernilai 40.02 dB [8].

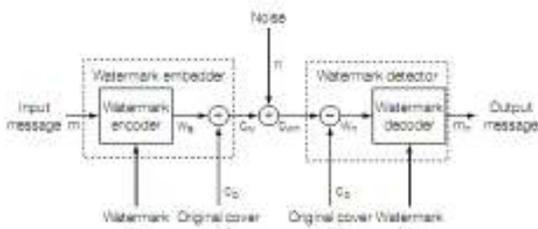
Pembangunan aplikasi *audio watermarking* menggunakan metode selain DSSS antara lain adalah perangkat lunak steganografi pada audio Mp3 pada lingkungan *mobile phone* dengan menggunakan metode *pharity coding*. Data dienkripsi terlebih dahulu

menggunakan teknik Hash dengan bantuan kunci enkripsi. Kunci pembangkit bilangan *pseudo-random* menggunakan algoritma *pseudorandom generator* untuk merubah urutan bit pada file *chip*. Semakin besar nilai PSNR maka semakin bagus pula nilai subjektif berdasarkan pendengaran. Nilai PSNR besar dapat diperoleh bila ukuran data pesan yang disembunyikan sesuai dengan frame ukuran *cover*. Nilai PSNR cenderung menurun seiring dengan bertambahnya ukuran pesan tanpa diperbesar ukuran *cover* [9].

Aplikasi *audio watermarking* dengan menggunakan kombinasi metode enkripsi telah dikembangkan juga seperti pada metode *parity coding* dengan kombinasi enkripsi *Rjindel*. Sebelum data dimasukkan ke dalam *cover*, data mengalami proses enkripsi dengan teknik *Rjindel*. Secara kualitas, hasil file MP3 yang dimasukkan data dapat diambil kesimpulan bahwa semakin kecil data yang disimpan dan semakin besar ukuran media *cover* maka nilai SNR semakin tinggi serta *noise* semakin tidak terdengar. Dari keseluruhan hasil uji coba diperoleh *ratio embedding* yang proporsional sebanyak 1 : 18968,049 bit [10]. Begitupula dengan penelitian implementasi *audio watermarking* dengan metode *phase coding* ketahanan *audio* terhadap pemampatan cukup tinggi namun tidak cukup tahan dengan serangan *cropping*, *time stretching*, dan *multiple watermarking* [11].

2.1 Audio Watermarking

Dasar dari proses *watermarking* pada data digital serupa dengan sistem komunikasi dasar yaitu mengomunikasikan pesan dari *transmitter* menuju ke *receiver* [6]. Teori tersebut memiliki kesamaan konsep dasar dengan sebuah sistem komunikasi secara konvensional. Konsep dasar *digital watermarking* mempunyai prinsip yang sama dengan konsep transmisi komunikasi berbasis enkripsi. Perbedaannya hanya pada implementasinya saja. Pada kosep sistem komunikasi mengimplementasikan *channel encoder* serta *channel decoder*. Sedangkan pada proses *digital watermarking*, *encoder* dan *decoder* ini mengimplementasikan prinsip *watermark embedder* dan *watermark extraction*.



Gambar 1 Model Konsep Dasar Digital Watermarking (Cox et. al. 2008, h. 68)

2.2 Direct Sequence Spread Spectrum

Metode DSSS memiliki dua tahapan yaitu tahap *encoding* serta tahap *decoding*. Prinsip dasar metode DSSS pada saat tahap *encoding* adalah sebagai berikut [12].

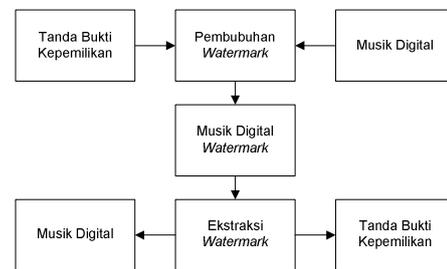
- Pesan diubah ke dalam bentuk biner.
- Menentukan nilai *chip rate*. *Chip rate* merupakan jumlah deret *chip*er untuk merepresentasikan satu bit pesan. Nilai *chip rate* menentukan besar jumlah data *PN sequence* yang dikalikan terhadap satu bit pesan.
- Membangkitkan *PN sequence* berdasarkan jumlah data pesan dan nilai *chip rate*.
- Watermark* diperoleh dengan mengalikan antara data pesan dengan nilai *PN sequence*.

Berikut ini merupakan prinsip dasar metode DSSS pada tahap *decoding*.

- Hasil perkalian antara *PN sequence* yang nilainya sama menghasilkan nilai absolut satu.
- Pesan diperoleh kembali dengan mengalikan deret *watermark* dengan *PN sequence* yang sama seperti dihasilkan saat proses *encoding*.
- Hasil perkalian saling dijumlahkan sesuai dengan jumlah *chip rate* yang telah ditentukan sebelumnya. Hasil penjumlahan kemudian dianalisa sesuai dengan *threshold* untuk diperoleh nilai biner pesan.

3. METODOLOGI

Ada dua tahap utama pada *audio watermarking* yaitu tahap pembubuhan *watermark* ke dalam berkas audio (*watermark embedding*) serta tahap ekstraksi *watermark* dari berkas audio (*watermark extraction*). berikut merupakan gambar skema *audio watermarking* secara garis besar.

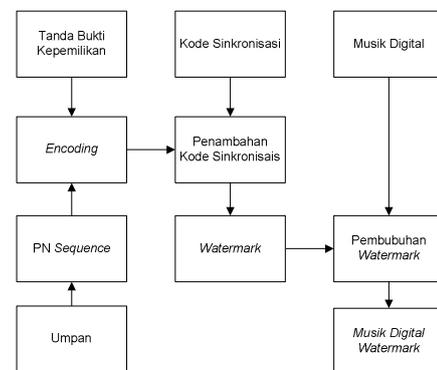


Gambar 1 Skema audio watermarking

Konten musik digital masuk menjadi *carrier watermark*. Kemudian dibubuhkan berkas digital berupa tanda bukti kepemilikan yang sudah diubah sebelum menjadi bentuk *watermark*. Hasil dari pembubuhan *watermark* pada musik digital ini adalah musik digital yang telah memiliki *watermark (watermarked signal)* [13].

3.1 Tahap Pembubuhan

Tahap pembubuhan melakukan pembubuhan berkas *text* berupa tanda bukti kepemilikan ke dalam musik digital berupa audio berformat *wave*. Pada tahap pembubuhan *watermark* ini secara lebih rinci ada 4 proses, yaitu proses pembangkitan *PN Sequence*, proses *encoding*, proses penambahan kode sinkronisasi, serta proses pembubuhan.



Gambar 2 Skema Tahap Pembubuhan Watermark

3.1.1 Pembangkitan PN Sequence

Pseudo-Noise Sequence atau *Pseudo Random Noise Sequence* merupakan deret bilangan acak yang berperan sebagai derau [14]. *PN Sequence* dibangkitkan untuk mengacak nilai data *plain* sehingga data *plain* tersebut bisa menjadi derau sehingga sulit dimengerti. Pembangkitan *PN Sequence* pada penelitian ini menggunakan algoritma *Linear Congruential Generator (LCG)*. Algoritma LCG melakukan iterasi menggunakan persamaan seperti berikut.

$$X_{n+1} = ((Pengali \times X_n) + Inkremen) \text{ mod } Modulator \quad (1)$$

x_n merupakan *PN Sequence* yang dibangkitkan dimana n bernilai lebih dari 0 dan x_0 adalah umpan. Umpan merupakan nilai awal dari *PN Sequence* yang akan dihasilkan. Sedangkan *Modulator*, *Pengali*, serta *Inkremen* merupakan nilai konstanta yang bisa diatur sesuai kebutuhan.

3.1.2 Proses Encoding

Proses *encoding* dilakukan untuk menghasilkan nilai *watermark* berdasarkan metode *DSSS*. Berkas digital *text* sebagai data *plain* berupa tanda bukti kepemilikan terlebih dahulu diubah ke dalam bentuk biner. Selanjutnya data *plain* bentuk biner tersebut kemudian mengalami proses *encoding* menggunakan *PN Sequence* yang telah dibangkitkan menjadi sebuah *watermark*.

Proses *encoding* menentukan nilai dari *chip rate*. Jika jumlah *PN Sequence* adalah jpn , cr adalah *chip rate* dan jp adalah jumlah data *plain* maka.

$$jpn = cr \times jp \quad (2)$$

Nilai data *plain* dikonversi menjadi nilai dengan simbol antara +1 dan -1. Begitupula dengan deret *PN Sequence* yang dibangkitkan. Nilai pada *PN Sequence* dikelompokkan sesuai jumlah *chip rate* untuk kemudian dikalikan setiap satu bit data *plain* seperti persamaan berikut.

$$w = p \times pn \quad (3)$$

w merupakan *watermark* yang diperoleh dari setiap perkalian satu bit *plain* p dengan *PN sequence* pn .

3.1.3 Proses Penambahan Kode Sinkronisasi

Menentukan kode sinkronisasi untuk ditambahkan ke data *watermark* yang telah terbentuk sebelumnya. Hal tersebut untuk memudahkan dalam mendeteksi apakah *watermark* benar atau tidak.

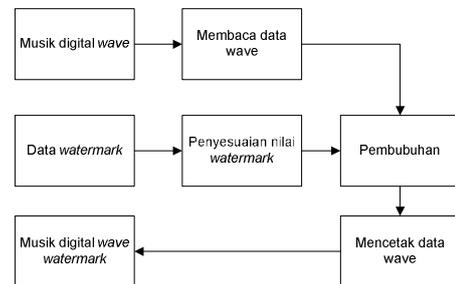
$$w' = k + w \quad (4)$$

Watermark yang terdapat kode sinkronisasi berupa w' diperoleh dengan menambahkan kode sinkronisasi berupa k ke dalam *watermark* w .

3.1.4 Proses Pembubuhan

Musik digital yang digunakan sebagai *carrier watermark* adalah berkas audio digital berupa berkas *wave*. Sebelum *watermark* dibubuhkan ke dalam *wave*, terlebih dahulu *wave* dibaca keseluruhan data utamanya untuk kemudian dibagi menjadi beberapa segmen frame. Setiap satu frame terdapat nilai sampel

audio. *Watermark* dibubuhkan ke dalam sampel sesuai dengan nilai skala audio.



Gambar 3 Skema Proses Pembubuhan

Setiap frame terdapat data sampel audio yang memiliki skala nilai audio antara +1 sampai dengan -1. Nilai *watermark* mengalami penyesuaian nilai audio yang bertujuan untuk mengkonversi nilai *watermark* menjadi nilai audio dengan menyesuaikan nilai *watermark* itu sendiri menggunakan faktor skala tertentu.

$$\text{nilai audio watermark} = \frac{\text{watermark}}{\text{faktor skala}} \quad (5)$$

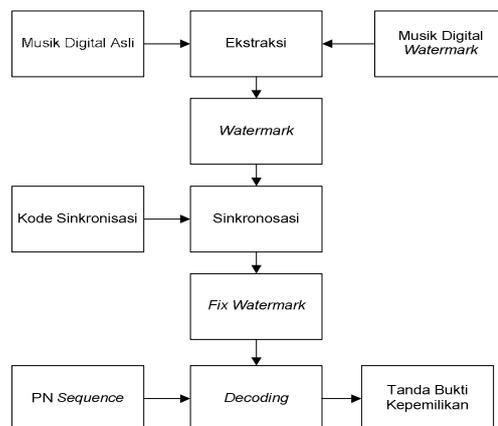
Proses pembubuhan dilakukan dengan menambahkan nilai audio *watermark* ke dalam sampel audio pada setiap frame. Proses pembubuhannya dilakukan seperti persamaan berikut.

$$y' = y + aw \quad (6)$$

Bila y merupakan berkas audio dan w adalah *watermark* dengan koefisien a maka sampel baru tersebut adalah y' yang akan dicetak kembali menjadi berkas audio melalui proses cetak data *wave*.

3.2 Tahap Ekstraksi

Proses ekstraksi *watermark* dilakukan untuk memisahkan *watermark* dari berkas digital audio *wave* serta menyusun kembali *watermark* menjadi berkas *text* tanda bukti kepemilikan. Ekstraksi *watermark* dilakukan dengan cara *informed detection* yaitu membandingkan *wave watermark* dengan *wave* asal [13]. Rincian prosesnya dimulai dengan proses ekstraksi untuk kemudian dilanjutkan dengan proses sinkronisasi, dan yang terakhir adalah proses *decoding*.

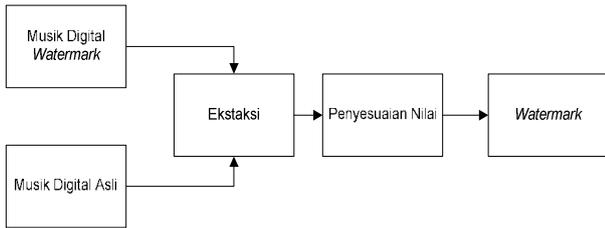


Gambar 4 Skema Proses Ekstraksi

PN *sequence* yang digunakan pada tahap ekstraksi *watermark* sama dengan PN *sequence* yang dihasilkan saat tahap pembubuhan *watermark*. PN *sequence* berfungsi sebagai kunci untuk melakukan proses *decoding* pada *watermark* agar bisa berentuk kembali menjadi berkas *text* tanda bukti kepemilikan.

3.2.1 Proses Ekstraksi

Proses ekstraksi melakukan pengambilan data *watermark* pada suatu berkas *wave*. Proses ekstraksi pada penelitian ini menggunakan proses ekstraksi secara *informed detection*. Proses ekstraksi dengan *nformed detection* dilakukan dengan cara membandingkan *wave* yang telah menjadi *watermarked signal* dengan *wave* yang asli (*original signal*).



Gambar 5 Skema Proses Ekstraksi

Bila y' adalah nilai sampel frame milik *wave watermark* dan y adalah nilai frame milik *wave* asli maka untuk memperoleh *watermark* w adalah.

$$w = y' - y \tag{7}$$

Nilai *watermark* yang masih dalam bentuk nilai audio disesuaikan menjadi nilai simbolik (+1/-1) kembali dengan persamaan.

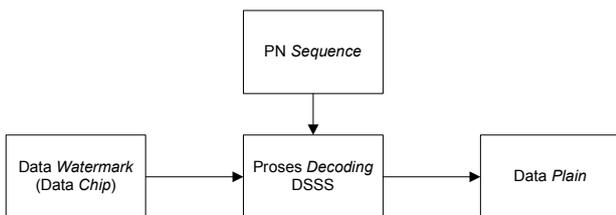
$$\text{nilai watermark} = \text{nilai audio} \times \text{faktor skala} \tag{8}$$

3.2.2 Proses Sinkronisasi

Nilai *watermark* hasil proses ekstraksi diperikasa deret awalnya apakah terdapat nilai kode sinkronisasi. Nilai kode sinkronisasi harus sama pada saat tahap pembubuhan. Bila terdapat kode sinkronisasi maka nilai kode sinkronisasi tersebut dihilangkan dari deret *watermark*. Kemudian nilai *watermark* masuk pada tahap *decoding* untuk dicetak data *plain* kembali.

3.2.3 Proses Decoding

Proses *decoding* berfungsi untuk mengubah data *watermark* menjadi data *plain* kembali. Proses *decoding* yang menerapkan metode DSSS ini memiliki langkah-langkah kebalikan dari proses *encoding* pada saat tahap pembubuhan *watermark*.



Gambar 6 Skema Proses Decoding Metode DSSS

Agar *watermark* bisa diubah nilai *plain*-nya kembali maka data *watermark* dikalikan dengan PN *Sequence*. PN *Sequence* yang digunakan pada proses *decoding* ini bernilai sama seperti PN *Sequence* yang digunakan pada saat melakukan proses *encoding* pada tahap pembubuhan *watermark*.

$$\begin{aligned} d &= w \times pn_d \\ d &= (p \times pn_e) \times pn_d \\ d &= p \end{aligned} \tag{9}$$

p merupakan nilai data *plain* saat *encoding*. Sedangkan pn_e merupakan PN *Sequence* saat *encoding* dan pn_d merupakan PN *Sequence* saat *decoding*. Nilai d merupakan *plain* hasil *decode* seperti nilai p . Data *plain* yang diperoleh ini untuk kemudian dicetak menjadi berkas *text* kembali berupa tanda bukti kepemilikan.

3.3 Pengujian

Tahap pengujian ada dua kali tahapan yaitu pertama adalah pengujian kualitas suara musik digital yang telah memiliki *watermark* dan kedua adalah pengujian ketahanan *watermark* pada musik digital. Pengujian dilakukan pada enam buah data sampel meliputi tiga buah berkas musik digital serta tiga buah berkas *text* tanda bukti.

3.3.1 Pengujian Kualitas Suara Musik Digital yang Telah Memiliki Watermark

Pengujian terhadap kualitas suara musik digital yang telah memiliki *watermark* diukur secara subyektif serta obyektif. Pengukuran secara subyektif merupakan pengukuran berdasarkan pada penilaian indera pendengar manusia secara langsung pada konten musik digital yang telah memiliki *watermark* tersebut. Penilaian dilakukan oleh sejumlah responden yang ahli di bidang musik untuk kemudian ditarik kesimpulan. Sedangkan pengukuran secara obyektif dilakukan dengan cara menghitung suatu nilai. Pengukuran dilakukan dengan menghitung nilai PSNR (*Peak Signal to Noise Ratio*). Nilai PSNR dihitung menggunakan satuan desibel (dB) dengan rumus:

$$PSNR = 10 \log_{10} \left(\frac{P_1^2}{P_1^2 + P_0^2 - 2P_1P_0} \right) \tag{10}$$

P_0 menyatakan kekuatan sinyal awal (berkas *wave* asli) dan P_1 menyatakan kekuatan sinyal akhir (berkas *wave* yang memiliki *watermark*). Nilai PSNR yang memiliki kualitas suara baik berada pada kisaran antara 30 dB sampai dengan 50 dB [8].

3.3.2 Pengujian Ketahanan Watermark pada Musik Digital

Pengujian ketahanan *watermark* dimaksudkan untuk menguji seberapa tahankah *watermark* berada pada musik digital sehingga pada saat mengalami ekstraksi, isi dari *watermark* tersebut masih sesuai dengan data sampel *text* yang telah dibubuhkan sebelumnya meskipun musik digital telah mengalami manipulasi sinyal audio [3].

1. Kompresi
Manipulasi berupa kompresi serta de-kompresi terhadap berkas *wave* yang telah memiliki *watermark* pada beberapa tingkatan rasio sebelum berkas *wave* tersebut diekstraksi menggunakan *audio watermarking*.

2. Resampling
Merubah *sample rate* pada berkas *wave* yang telah memiliki *watermark* pada beberapa tingkatan sebelum berkas *wave* tersebut mengalami proses ekstraksi.
3. *Cropping*
Memotong bagian *wave* yang telah memiliki *watermark* pada beberapa bagian yang berbeda – beda.
4. Pemberian efek *noise*
Memberikan efek *white noise* pada berkas *wave* yang telah memiliki *watermark* sebelum mengalami proses ekstraksi.

Tingkat ketahanan diukur berdasarkan pengukuran prosentase kesesuaian nilai byte ASCII setiap karakter pada *watermark* yang telah diekstrak terhadap berkas *text* tanda bukti kepemilikan yang telah dibubuhkan sebelumnya.

4. PEMBAHASAN

Data sampel pada penelitian ini terdiri dari tiga buah berkas *wave* dan tiga buah berkas *text* tanda bukti kepemilikan. Berikut ini akan dijelaskan atribut dari ke-enam sampel berkas digital tersebut.

Tabel 1 Atribut data sampel *wave*

Nama File	Durasi (menit)	Ukuran (Mb)	Bit Rate (kbps)	Sample Rate (MHz)
Pergi Cinta.wav	03:27	17,4	705	44100
Beware.wav	02:33	12,9	705	44100
Lelah.wav	03:36	18,2	705	44100

Tabel 2 Atribut data sampel berkas *text*

Nama File	Ukuran (Kb)	Jumlah Karakter
The Password.txt	570	562
Salahudin Al Ayubi.txt	321	322
Aurora.txt	389	389

Uji coba fungsionalitas aplikasi dilakukan dengan melakukan proses pembubuhan berkas *text* ke dalam berkas *wave*. Pada saat proses ekstraksi, *watermark* yang telah diekstrak dari berkas *wave* memiliki isi yang sesuai dengan isi *text* dengan kesesuaian karakter mencapai prosentase 100%. Tabel 3 merupakan kinerja performa berdasarkan kecepatan eksekusi pada proses pembubuhan serta proses ekstraksi berdasarkan.

Tabel 3 Performa kinerja fungsionalitas aplikasi *audio watermarking*

Proses	Ukuran Wave Rata-rata (Mb)	Waktu Eksekusi Rata-rata (detik)	Mb/ Detik
Proses Pembubuhan	16,16	8,30	1,94
Proses Ekstraksi	16,16	8,73	1,90

Berdasarkan tabel 3, proses pembubuhan pada *audio watermarking* mampu melakukan *encode* sebesar 1,94 Mb per detik. Sedangkan pada proses ekstraksi mampu melakukan *decode* sebesar 1,90 Mb per detik.

4.1 Pengujian Kualitas Suara Musik Digital

4.1.1 Pengukuran secara Obyektif

Pengukuran kekuatan sinyal audio pada berkas *wave* menggunakan *sound level meter* Extech tipe 407736.

Tabel 4 Hasil pengukuran secara obyektif

Nama File	Masih Original (dB)	Telah Memiliki Watermark (dB)	PSNR (dB)
Pergi Cinta.wav	81,7	81,4	39,8
Beware.wav	86,5	85,6	39,6
Lelah.wav	81,8	82,1	48,7

Wave watermark merupakan *wave* yang telah memiliki *watermark*, sedangkan *wave original* merupakan berkas *wave* yang asli (belum diberi *watermark*). Berdasarkan tabel 4, secara keseluruhan berkas *wave* yang memiliki *watermark* memiliki nilai PSNR berada antara 30 dB sampai dengan 50 dB. Dapat disimpulkan bahwa berkas *wave watermark* hasil proses pembubuhan pada aplikasi *audio watermarking* ini mempunyai kualitas suara yang baik.

4.1.2 Pengukuran secara Subyektif

Pengujian kualitas suara dengan pengukuran secara subyektif melibatkan 10 responden yang ahli di bidang musik. Mereka bersedia untuk mendengarkan secara langsung berkas *wave* yang telah memiliki *watermark*. Kemudian para responden dimintai pendapat tentang kejelasan suara pada berkas *wave*. Hasilnya adalah 93% dari responden menilai bahwa musik digital yang telah memiliki *watermark* tersebut memiliki kualitas suara yang baik. Hal tersebut didukung oleh Bapak Drs. Iwan Yahya, M.,Sc. Beliau adalah dosen bidang fisika akustik dan geofisika Jurusan Fisika, Fakultas MIPA, Universitas Sebelas Maret Surakarta. Menurut beliau bahwa indera pendengaran manusia memiliki karakteristik yang berbeda, apalagi bila data *watermark* ditambahkan secara sembunyi akan sulit untuk membedakan musik digital yang asli dengan musik digital yang telah memiliki *watermark*.

4.2 Pengujian Ketahanan *Watermark* pada Musik Digital

4.2.1 Pengujian Kompresi

Kompresi dilakukan dengan bantuan aplikasi Monkey's Audio 4.10. Pengujian kompresi dilakukan pada tiga level kompresi yaitu normal, *extra high*, dan *insane*.

Tabel 6 Hasil pengujian kompresi

Level	Rata – rata Rasio (%)	Rata-rata Waktu Ekstraksi (detik)	Rata-rata Kesesuaian Watermark (%)
Normal	30,73	7,8	100
<i>Extra High</i>	36,45	7,3	100
<i>Insane</i>	36,58	6,7	100

Tabel 6 menjelaskan bahwa kesesuaian karakter pada isi *watermark* dengan *text* tanda kepemilikan yang telah dibubuhkan sebelumnya mencapai prosentase rata-rata sebesar 100% pada level normal, *extra high*, maupun level *insane*. Hal ini mengindikasikan bahwa manipulasi kompresi tidak merusak keberadaan *watermark*.

4.2.2 Pengujian Resampling

Resampling dilakukan pada dua nilai *sample rate* yang berbeda yaitu pada *sample rate* 16000 dan *sample rate* 48000.

Tabel 7 Hasil pengujian resampling

Sample Rate	Bit Rate (kbps)	Rata-rata Durasi (Menit)	Rata-rata Waktu Ekstraksi (detik)	Rata-rata Kesesuaian Watermark (%)
16000	256	08:51	8,1	100
48000	768	02:57	7,6	100

Isi *watermark* hasil ekstraksi pada berkas *wave* yang mengalami manipulasi resampling dengan *sample rate* sebesar 16000 memiliki kesesuaian karakter terhadap isi *text* tanda bukti kepemilikan dengan prosentase rata-rata 100%. Begitupula dengan berkas *wave* yang mengalami resampling dengan *sample rate* sebesar 48000. *Watermark* tetap tahan meskipun mengalami manipulasi resampling.

4.2.3 Pengujian Cropping

Proses pemotongan dilakukan dengan menggunakan aplikasi Shuang's Audio Editor 3.2. Pemotongan dilakukan secara tiga tahap yaitu pemotongan ukuran 50%; 25%; dan 12,5%. Tabel 8 merupakan hasil pengujian *cropping* secara keseluruhan.

Tabel 8 Hasil pengujian cropping

Ukuran Cropping (%)	Rata-rata Durasi (menit)	Rata-rata Ukuran (Mb)	Rata-rata Waktu Ekstraksi (detik)	Rata-rata Kesesuaian Watermark (%)
50	01:36	8,05	7,7	100
25	00:48	4,06	6,6	100
12,5	00:24	2,02	5,8	100

Isi *watermark* memiliki kesesuaian karakter mencapai 100% terhadap isi *text* tanda bukti kepemilikan meskipun berkas *wave* telah mengalami manipulasi pemotongan hingga 12,5% dari keseluruhan ukuran data *wave*.

4.2.4 Pengujian Noise

Efek *noise* yang diberikan berjenis *white noise* dengan kisaran amplitudo bernilai 0,01. Pemberian efek *noise* dilakukan dengan bantuan aplikasi Audacity 1.3 Beta.

Tabel 14 Hasil pengujian pemberian noise

Wave Watermark	Wave Cropping	Durasi (menit)	Ukuran (Mb)	Waktu Decode (detik)	Kesesuaian (%)
Pergi Cinta Watermark .wav	Pergi Cinta Wm Noise.wav	03:27	17,4	13,4	0

Tabel 14 Lanjutan

Wave Watermark	Wave Cropping	Durasi (menit)	Ukuran (Mb)	Waktu Decode (detik)	Kesesuaian (%)
Beware Watermark .wav	Beware Wm Noise.wav	02:33	12,9	5,4	0
Lelah Watermark .wav	Lelah Wm Noise.wav	03:36	18,2	8,0	0

Berdasarkan tabel 14, isi *watermark* hasil ekstraksi dari *wave* yang mengalami manipulasi penambahan *noise* tidak memiliki kesesuaian sama sekali terhadap isi *text* tanda bukti kepemilikan yang telah dibubuhkan sebelumnya. *Audio watermarking* tidak mampu untuk melakukan ekstraksi secara benar terhadap nilai audio pada berkas *wave* karena telah tercampur dengan nilai audio *noise*.

Noise muncul saat berkas audio mengalami transmisi terutama transmisi pada media gelombang mikro. Transmisi gelombang mikro sangat rentan dengan gangguan cuaca sehingga rawan adanya *jamming* pada *band* gelombang. Kemunculan *noise* dapat terjadi pada berbagai keadaan. *Noise* dapat muncul saat sinyal audio sedang ditransmisikan atau dapat juga karena kualitas buruk dari *receiver* sinyal. Kemunculan *noise* pada keadaan yang berbeda diperlukan algoritma yang berbeda pula untuk bisa menyaring nilai sinyal yang terbebas dari *noise*. Perlu adanya analisa mendalam terhadap algoritma penyaringan *noise* pada berbagai keadaan kemunculan *noise*. *Audio watermarking* tidak disertai dengan algoritma penyaringan *noise* karena analisa terhadap transmisi data di luar dari penelitian ini.

5. KESIMPULAN

Audio watermarking dapat bekerja dengan baik. Pada proses pembubuhan, *audio watermarking* mampu melakukan pembubuhan *watermark* pada berkas *wave* sebesar 1,94 Mb per detik. Sedangkan pada proses ekstraksi, *audio watermarking* mampu melakukan ekstraksi *watermark* dari berkas *wave* sebesar 1,90 Mb per detik.

Watermark hasil ekstraksi dari *wave* yang telah mengalami manipulasi audio berupa kompresi, resampling, serta *cropping* memiliki kesesuaian isi terhadap *text* tanda bukti kepemilikan dengan tingkat kesesuaian mencapai 100%. Sehingga dapat disimpulkan bahwa *watermark* memiliki ketahanan terhadap manipulasi audio berupa kompresi, resampling, serta *cropping*. Sedangkan untuk manipulasi penambahan *noise*, *watermark* tidak mampu diekstraksi dengan baik oleh *audio watermarking* dengan kesesuaian karakter pada isi *watermark* terhadap isi *text* tanda kepemilikan sebesar 0%.

Kualitas suara berkas *wave* yang telah memiliki *watermark* memiliki kualitas yang baik berdasarkan penilaian responden. Rata – rata nilai PSNR pada berkas *wave* yang telah memiliki *watermark* sebesar 42,7 dB yang berarti pada batas kewajaran.

6. DAFTAR PUSTAKA

- [1] Nuryani. 2007. *Digital Right Management (DRM) dan Audio Watermarking untuk Perlindungan Hak Cipta pada Konten Musik Digital*. Jurnal INKOM Vol. 1 No. 1.
- [2] Turban, E., Rainer, R. K. JR., & Potter, R. E. 2006. *Introduction to Information Technology: Pengantar Teknologi Informasi*. Edisi Ketiga. Jakarta: Salemba Infotek.

- [3] Seitz, J. 2005. *Digital Watermarking for Digital Media*. Pennsylvania: Information Science Publishing.
- [4] Hartung, F. & Kutter, M. 1999. *Multimedia Watermarking Techniques*. Proceedings of the IEEE Vol. 87 No. 7.
- [5] Cvejić, N. & Seppänen, T. 2008. *Digital Audio Watermarking Techniques and Technologies : Applications and Benchmarks*. Pennsylvania: Information Science Publishing.
- [6] Cox, I. J. et. al. 2008. *Digital Watermarking and Steganography*. Second Edition. Massachusetts: Morgan Kuffman Publishers.
- [7] Malvar, H.S. & Florencio D. A. 2003. *Improved Spread Spectrum: A New Modulation Technique for Robust Watermarking*. IEEE Transactions on Signal Processing, Vol. 51, No. 4.
- [8] Baskara, T. 2008. *Studi dan Implementasi Steganografi pada MP3 dengan Teknik Spread Spectrum*. Makalah Tugas Akhir. Bandung: Institut Teknologi Bandung.
- [9] Herianto. 2009. *Pembangunan Perangkat Lunak Steganografi Audio MP3 dengan Teknik Parity Coding pada Perangkat Mobile Phone*. Makalah Tugas Akhir. Bandung: Institut Teknologi Bandung.
- [10] Wulandari, R. 2010. *Pembuatan Aplikasi Steganografi pada File Audio MP3 dengan Metode Parity Coding dan Enkripsi Rijndael*. Makalah Tugas Akhir. Surabaya: Institut Teknologi Sepuluh Nopember.
- [11] Dianitasari, A. 2011. *Implementasi Teknik Audio Watermarking dengan Metode Phase Coding*. Makalah Tugas Akhir. Bogor: Institut Pertanian Bogor.
- [12] Meel, J. 1999. *Spread Spectrum (SS): Introduction*. Rotselaar: Sirius Communications.
- [13] Cvejić, N. 2004. *Algorithms for Audio Watermarking and Steganography*. Oulu: Oulu University Press.
- [14] Kirovski, D. & Malvar H. S. 2003. *Spread-Spectrum Watermarking of Audio Signals*. IEEE Transactions on Signal Processing Vol. 51 No. 4 PP. 10230-33.