Implementasi *Robust Video Watermarking* berbasis DCT pada *Video Copyright* Media Sosial

Diki Ardian Wirasandi-13515092
Program Studi Teknik Informatika
Sekolah Teknik Elektro dan Informatika
Institut Teknologi Bandung, Jl. Ganesha 10 Bandung 40132, Indonesia
dikiwirasandi@gmail.com

Abstrak—Untuk mencegah penggunaan kembali informasi atau konten digital tanpa izin pada media sosial diperlukan suatu mekanisme pengecekan kepemilikan informasi atau konten tersebut. Salah satu teknik yang dapat digunakan adalah dengan menambahkan *copyright*. *Copyright* yang ditambahkan berupa *watermark* yang disisipkan pada informasi digital tersebut. Pada makalah ini akan diajukan solusi permasalahan *copyright* pada berkas video yang diunggah pada media sosial. Solusi tersebut mengimplementasi *robust video watermarking* berbasis DCT. *watermarking* ini akan memproses *frame-frame* penyusun video dan disisipkan *watermark* di dalamnya. Solusi ini juga mampu mendeteksi *copyright* secara parsial dari suatu video.

Kata Kunci—*Digital Watermarking*, *Watermark*, domain spasial, domain frekuensi, transformasi, *Watermark Attacks*

I. Pendahuluan

Media sosial merupakan sarana komunikasi dan berbagi informasi yang sudah umum digunakan di era digital ini. Informasi yang dibagikan pun beragam. Mulai dari informasi berupa teks, foto atau gambar, audio hingga video. Media sosial yang memiliki tujuan spesifik membagikan informasi berupa jenis informasi tertentu, seperti gambar, video dan sebagainya, dapat disebut sebagai media *sharing*.

Telah banyak *platform* media *sharing* yang memiliki banyak pengguna. Sebagai contoh, pada media *sharing* video terdapat beberapa *platform* yang tersedia, seperti YouTube, Instagram dan lain-lain. Pada dasarnya, media *sharing* dapat digunakan sebagai wadah pengguna untuk berbagi informasi yang dapat diakses oleh pengguna lainnya. Misalkan pada YouTube, pengguna dapat mengunggah video yang ingin dibagikan kepada pengguna lainnya. Pengguna lain dapat menonton, memberikan respon berupa komentar, *like* dan sebagainya terhadap video tersebut.

Salah satu permasalahan yang timbul pada media *sharing* yaitu pengguna yang bukan merupakan pemilik dari informasi dapat mengunduh informasi atau konten tertentu meskipun tidak disediakan fitur unduh pada *platform* tersebut. Terlebih jika informasi atau konten tersebut diunggah dan dibagikan kembali tanpa seizin dari pemilik asli informasi. Oleh karena itu, dibutuhkan suatu metode pengenalan informasi yang dibagikan. Metode pengenalan tersebut digunakan untuk mengidentifikasi pihak mana yang merupakan pemilik asli dari informasi tersebut. Sehingga, pihak lain yang ingin membagikan kembali informasi tanpa seizin pemilik asli dari informasi tidak akan dapat melakukannya. Hal tersebut biasa dikenal sebagai *copyright*.

Teknik pemberian identitas terhadap suatu informasi digital disebut dengan *digital watermarking*. Dengan adanya *digital watermarking*, informasi digital dapat dibuktikan kepemilikannya. *Watermarking* dapat diaplikasikan terhadap berbagai informasi digital, salah satunya adalah informasi berupa video. Video *watermarking* cukup banyak diimplementasikan pada berbagai bidang, seperti rekaman CCTV dan lain-lain. Tujuannya tidak lain adalah untuk menjamin keaslian atau identitas pemilik dari video tersebut.

Berdasarkan permasalahan yang telah disebutkan sebelumnya, video *watermarking* cocok diimplementasikan untuk menyelesaikan permasalahan video *copyright* pada *platform* video media *sharing*. Diharapkan dengan diterapkannya teknik ini, video-video yang telah dibagikan dapat dikenali pemilik aslinya dan dapat mencegah pengunggahan ulang tanpa seizin dari pemillik asli dari video tersebut.

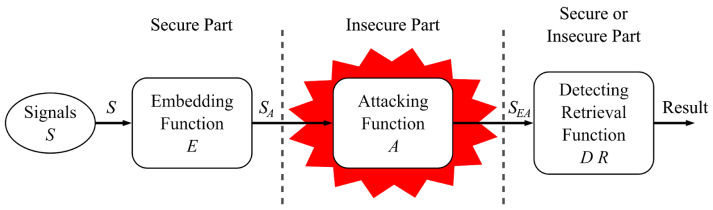
II. Dasar Teori

A. *Digital Watermarking*

Digital watermarking adalah penyisipan informasi digital ke dalam dokumen atau berkas digital. Berkas atau dokumen digital dapat berupa video, audio, citra, dokumen teks dan sebagainya[1]. Tujuan dari *digital watermarking* yaitu untuk perlindungan *copyright*, *fingerprinting*, pembuktian kepemilikan dan sebagainya[2]. Secara umum, alur pemrosesan *digital watermarking* seperti yang disajikan pada Gambar 1.

Pada Gambar 1, proses *digital watermarking* diawali dengan tahap *embedding*. Pada tahap ini, informasi digital yang asli akan disisipkan sinyal *watermark* di dalamnya.

Setelah disisipkan, maka informasi digital tersebut dapat disebarluaskan. Setelah disebarluaskan, informasi digital kemungkinan mendapatkan beberapa 'serangan' untuk merusak sinyal *watermark* yang telah disisipkan. Serangan dapat berupa manipulasi informasi atau dengan menyisipkan *noise-noise* tertentu. Tujuannya adalah agar sinyal *watermark* tidak dapat diekstrak kembali seperti semula dan tidak dapat dibuktikan.



Gambar 1. Tahapan pada proses *digital watermarking* secara umum.

Tahap yang terakhir adalah ekstraksi sinyal *watermark* dari informasi digital yang telah disisipi sebelumnya. Jika hasil ekstraksi yang dihasilkan adalah sama dengan sinyal *watermark* semula, maka informasi digital tersebut terbukti kepemilikan dan keasliannya.

B. *Digital Watermarking* pada Video

Digital watermarking dapat diterapkan pada berbagai berkas digital, salah satunya adalah video (*video watermarking*). Penerapan *digital watermarking* pada video, pada dasarnya hampir sama dengan penerapan *digital watermarking* pada citra atau gambar (*image watermarking*), perbedaannya pada jumlah citra yang diproses. Pada *video watermarking*, setiap *frame* dianggap sebagai satu citra, sehingga untuk setiap *frame* akan dilakukan *image watermarking*.

Pada *image watermarking* atau *video watermarking*, terdapat beberapa persyaratan yang dipenuhi[2], yaitu:

1. *Imperceptible*

   Pada informasi digital asli maupun hasil *embedding* sinyal *watermark* tidak dapat dibedakan secara kasat mata. Artinya, kualitas dari berkas citra atau video tidak berkurang secara signifikan dari sebelum dilakukan *watermarking* dengan setelahnya. Untuk itu, proses *embedding* memerlukan teknik khusus dalam penyisipan sinyal *watermark*.

2. *Robustness*

   Sinyal *watermark* yang telah disisipkan pada berkas, diharapkan dapat tahan terhadap berbagai serangan terhadap *watermark*. Untuk berkas citra, serangan-serangan tersebut antara lain modifikasi berkas, seperti *crop*, *filter*, *flip* dan sebagainya. Aksi-aksi tersebut tentunya berpeluang dapat mengubah sinyal *watermark* yang ada.

3. *Secure*

   Selain tahan terhadap serangan perusakan, diharapkan sinyal *watermark* juga aman dan tidak dapat diakses atau dilakukan ekstraksi oleh publik. Artinya, dibutuhkan skenario pengamanan pengaksesan sinyal *watermark*. Pada implementasinya, dapat menggunakan *private key* pada proses *embedding* dan ekstraksi.

Pada implementasinya, terdapat dua jenis teknik pada *image watermarking*[1][2], yaitu:

1. *Fragile Watermarking*

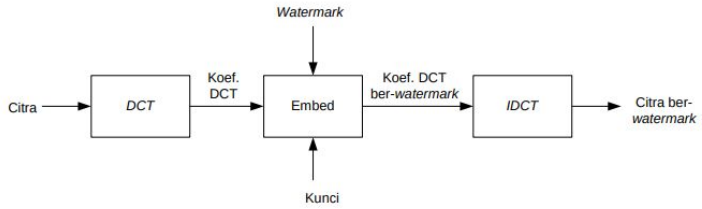
   Tujuan utama dari *fragile watermarking* adalah untuk menjaga keaslian berkas digital. Pembuktian keaslian dapat diperoleh dari keutuhan sinyal *watermark*. Jika pada citra dilakukan modifikasi seperti penyisipan objek, penghapusan objek dan sebagainya, maka sinyal watermark hasil ekstraksi akan mengalami kerusakan. Citra dapat dikatakan asli atau orisinil jika sinyal *watermark* hasil ekstraksi tetap utuh tanpa kerusakan. Metode yang digunakan pada proses *watermarking* yaitu menyisipkan *watermark* pada domain spasial (*pixel* citra).

Gambar 2. Contoh hasil ekstraksi *fragile watermarking.*

2. *Robust Watermarking*

   Fokus dari *robust watermarking* adalah untuk membuktikan kepemilikan. Untuk itu, *watermark* harus benar-benar dilindungi dari manipulasi citra, karena mengandung tanda pengenal pemilik. Metode penyisipan pada domain spasial tidak digunakan pada *watermarking* ini, melainkan menggunakan penyisipan pada domain frekuensi. Oleh karena itu, diperlukan transformasi dari domain spasial ke domain frekuensi terlebih dahulu untuk menyisipkan *watermark*. Setelah disisipkan *watermark*, perlu dilakukan invers transformasi dari domain frekuensi kembali ke domain spasial. Beberapa teknik

transformasi domain antara lain Fourier Transform, Discrete Cosine Transform (DCT), Discrete Wavelet Transform (DWT) dan sebagainya.



Gambar 3. Tahap *embedding* pada *robust watermarking* menggunakan DCT.

C. Discrete Cosine Transform

Discrete Cosine Transform (DCT) merupakan salah satu metode transformasi domain spasial ke domain frekuensi. DCT biasa digunakan pada pemrosesan citra. DCT juga dapat dimanfaatkan pada *digital watermarking* khususnya pada *robust watermarking*. Untuk melakukan sebaliknya, yaitu transformasi domain frekuensi menuju domain spasial, digunakan Inverse Discrete Cosine Transform (IDCT). Untuk menghitung koefisien DCT dan inversnya (IDCT) pada citra dua dimensi (MxN), dapat menggunakan rumus sebagai berikut.

$$C(u,v) = \alpha_u \alpha_v \sum_{x=0}^{M-1} \sum_{y=0}^{N-1} I(x,y) \cos\frac{\pi(2x+1)u}{2M} \cos\frac{\pi(2y+1)v}{2N}$$

$$I(x,y) = \alpha_u \alpha_v \sum_{u=0}^{M-1} \sum_{v=0}^{N-1} C(u,v) \cos\frac{\pi(2x+1)u}{2M} \cos\frac{\pi(2y+1)v}{2N}$$

$$\alpha_u = \begin{cases} \frac{1}{\sqrt{M}} & ,u=0 \\ \sqrt{\frac{2}{M}} & ,1 \le u \le M-1 \end{cases} \qquad \alpha_v = \begin{cases} \frac{1}{\sqrt{N}} & ,v=0 \\ \sqrt{\frac{2}{N}} & ,1 \le v \le N-1 \end{cases}$$

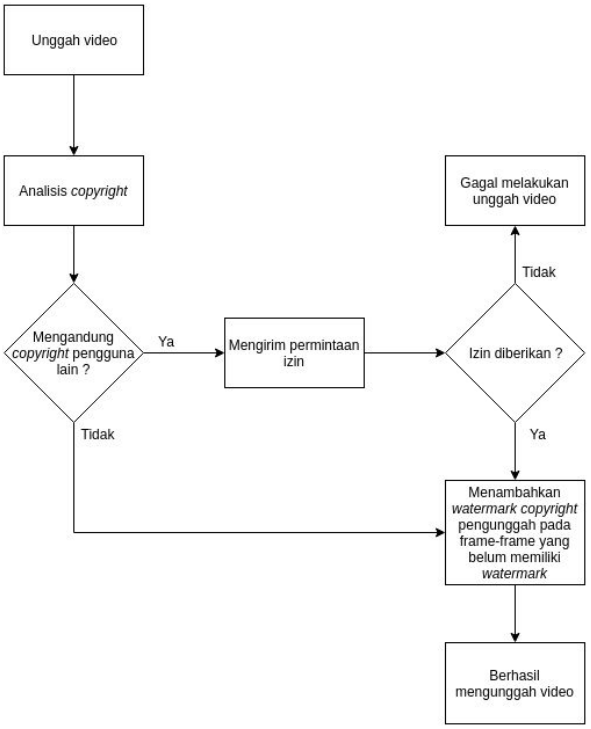
C(u,v) merupakan koefisien DCT yang merupakan representasi citra pada domain frekuensi. *I(x,y)* merupakan representasi berupa *pixel* pada domain spasial. Pada penerapannya di *robust watermarking*, *watermark* akan disisipkan pada koefisien-koefisien DCT pada tahap *embedding* (Gambar 3). Penyisipan tersebut dilakukan dengan operasi sebagai berikut.

$$c'i = ci + wi$$

Pada formula di atas, *ci* merupakan koefisien DCT dan *wi* merupakan bit *watermark*. Operasi ini dilakukan untuk semua bit-bit *watermark*. Penyisipan pada koefisien DCT dapat dilakukan secara terurut berdasarkan posisi. Selain itu, juga dapat menggunakan teknik penyisipan lainnya, seperti *spread spectrum* (penyebaran).

III. Rancangan dan Implementasi Sistem

Sistem yang diajukan berupa modul pengecekan video *copyright* pada sistem video media *sharing*. Modul ini dapat dipasang setelah modul unggah video. Alur pemrosesan disajikan pada Gambar 4 berikut.



Gambar 3. Alur pemrosesan pengecekan *copyright* dan penambahan *watermark.*

Pada alur proses di atas, terbagi menjadi empat tahap utama, yaitu:

A. Analisis *Copyright*

Pada tahap ini, video akan diuraikan menjadi *frame-frame* penyusunnya. Setiap *frame* akan dilakukan pengecekan apakah *frame* tersebut telah memiliki *watermark* dari pengguna lain (*copyright*). Kemudian, akan dilakukan rekapitulasi hasil pengecekan untuk semua *frame*. Hasilnya akan berupa persentase untuk setiap pengguna. Sebagai contoh:

```
Copyright Pengguna A: 45%
Copyright Pengguna B: 10%
Copyright Pengguna C: 5%
```

Pada contoh di atas, artinya adalah pada video yang dianalisis, terkandung 45% *frame* yang terdapat pada video yang telah diunggah atau dimiliki oleh pengguna A. Selain itu, 10% *frame* dimiliki oleh pengguna B dan 5% *frame* oleh pengguna C.

B. Pengiriman Permintaan Izin

Setelah dilakukan analisis terhadap *copyright* video, langkah selanjutnya adalah pengiriman permintaan izin. Tahap ini dilakukan jika video yang dianalisis memiliki *frame* yang telah ber*watermark*. Permintaan izin akan dikirimkan ke seluruh pengguna yang terdapat *copyright*nya di video tersebut. Sebagai contoh, untuk contoh kasus pada tahap sebelumnya, permintaan izin akan dikirim kepada pengguna A, B dan C.

C. Pemberian Izin

Setelah izin dikirimkan, pengguna sebagai pemilik asli video akan menerima notifikasi jika ada pengguna lain yang meminta izin mengunggah bagian (*frame-frame*) yang telah diunggah sebelumnya. Jika pemilik memberikan izin, maka proses akan lanjut ke tahap berikutnya. Jika izin tidak diberikan, maka pengirim permintaan izin akan diberikan pemberitahuan penolakan izin dan proses unggah video gagal. Izin harus diberikan oleh semua pengguna terkait. Artinya, untuk contoh kasus sebelumnya, pengguna A, B dan C harus memberikan izin jika proses unggah video ingin dilanjutkan.

D. Penambahan *Watermark* Pengunggah

Jika izin telah diberikan oleh semua pengguna terkait, langkah selanjutnya adalah penambahan *watermark* pengunggah. Penambahan ini dilakukan terhadap *frame-frame* yang belum memiliki *watermark* sebelumnya. Sebagai contoh, untuk contoh kasus sebelumnya, pemberian *watermark* dilakukan terhadap 40% sisa *frame* yang belum memiliki *watermark*. Proses *watermarking* menggunakan metode *robust watermarking* berbasis DCT untuk tiap *frame*.

IV. Percobaan dan Analisis

Percobaan dilakukan dengan mengimplementasikan tahap analisis *copyright* dan penambahan *watermark* pengunggah saja, tidak mencakup keseluruhan proses unggah. Selain implementasi, juga dilakukan pengujian terhadap serangan *watermark*. Kemudian dilakukan analisis terhadap hasil pengujian tersebut.

A. Implementasi Modul

Implementasi menghasilkan dua modul program terpisah, yaitu:

1. Modul Analisis *Copyright*

   Modul analisis *copyright* menerima masukan berupa video. Video tersebut kemudian diuraikan menjadi *frame-frame* penyusunnya. Setiap *frame* akan dilakukan pengecekan apakah terdapat *watermark* di dalamnya. Pengecekan dilakukan dengan cara mengecek apakah ada *flag* khusus sebagai tanda terdapat *watermark* di dalamnya. *Flag* dan *watermark* disimpan pada domain frekuensi, untuk itu perlu dilakukan transformasi terlebih dahulu menggunakan DCT. Keluaran dari modul berupa persentase keberadaan *watermark* berdasarkan tiap *watermark* pengguna.

2. Modul Penambahan *Watermark* Pengunggah

   Masukan dari modul ini berupa *frame-frame* dalam domain frekuensi yang telah ditransformasikan pada modul analisis *copyright*. Namun, dilakukan penyaringan terhadap *frame-frame* yang belum memiliki *watermark* saja. Kemudian, dilakukan *embedding* terhadap *frame* dengan menyisipkan *watermark* dengan operasi penyisipan seperti yang dijelaskan pada Bab II poin C. Selanjutnya dilakukan invers transformasi ke domain spasial menggunakan IDCT. *Frame-frame* yang telah diurai digabungkan kembali sehingga menghasilkan video yang disimpan dan siap di-*publish* oleh sistem.

B. Pengujian Modul

Untuk menguji modul yang telah diimplementasikan, dilakukan beberapa skenario pengujian. Skenario dan hasil pengujian disajikan pada Tabel 1 berikut.

Tabel 1. Skenario dan hasil pengujian modul.

Modul yang diuji	Skenario	Hasil
Modul Analisis *Copyright*	**Input**: AVI video, 72 *frames*, 0% *copyright*	**Output**: 0% *copyright* **Waktu**: 10.9 detik
	Input: AVI video, 72 *frames*, 40% *copyright* A	**Output**: 40% *copyright* A **Waktu**: 22.1 detik
	Input: AVI video, 72 *frames*, 40% *copyright* A, 20% *copyright* B	**Output**: 40% *copyright* A, 20% *copyright* B **Waktu**: 49.6 detik
	Input: AVI video, 215 *frames*, 40% *copyright* A, 20% *copyright* B	**Output**: 40% *copyright* A, 20% *copyright* B **Waktu**: 215.5 detik

Modul Penambahan *Watermark* Pengunggah	**Input**: AVI video, 72 *frames*	**PSNR:** 41.12 dB **Waktu**: 22.5 detik
	Input: AVI video, 152 *frames*	**PSNR:** 39.78 dB **Waktu**: 40.1 detik
	Input: AVI video, 312 *frames*	**PSNR:** 40.60 dB **Waktu**: 71.9 detik

Berdasarkan hasil pengujian modul, pada modul analisis *copyright*, pendeteksian *copyright* sesuai yang diharapkan untuk semua skenario, namun waktu pemrosesan yang dibutuhkan cukup lama dan berbanding lurus dengan jumlah *frame* yang diproses. Untuk modul penambahan *watermark* pengunggah, hasil penyisipan *watermark* memerlukan waktu yang cukup singkat dengan kualitas video setelah penyisipan yang tergolong baik (30 < PSNR < 50).

C. Pengujian Serangan *watermark*

Serangan *watermark* bertujuan untuk merusak atau memanipulasi *watermark* yang telah disisipkan. Serangan ini diujikan pada modul analisis *copyright*. Beberapa serangan *watermark* yang diujikan pada hasil implementasi beserta hasilnya adalah sebagai berikut.

1. *Simple attacks*

   Simple attacks merupakan jenis serangan yang mencoba merusak *watermark* yang telah disisipkan dengan memodifikasi *frame,* seperti penambahan *noise*, *cropping* dan sebagainya[3].

Tabel 2. Skenario dan hasil pengujian *simple attacks*.

Skenario	Hasil
Input: AVI video, 72 *frames*, 10 *frame* dengan *copyright* ***Attacks***: Menambah *noise* 5 dari 10 *frame* dengan *copyright*	**Output**: 9 *frame copyright*
Input: AVI video, 72 *frames*, 15 *frame* dengan *copyright* ***Attacks***: *Cropping* 5 dari 15 *frame* dengan *copyright*	**Output**: 13 *frame copyright*

Berdasarkan hasil pengujian *simple attacks*, dari skenario pertama terdapat 4 dari 5 *frame* yang ditambahkan *noise* berhasil dideteksi dan pada skenario kedua, 3 dari 5 *frame* yang dilakukan *cropping* berhasil dideteksi. Artinya, serangan *simple attacks* mampu merusak *watermark* meskipun tingkat keberhasilannya kecil.

2. *Detection-disabling attacks*

   Serangan ini bertujuan untuk menggagalkan proses ekstraksi *watermark* dengan beberapa modifikasi domain spasial, seperti rotasi, *zooming* dan sebagainya[3]. Skenario dan hasil pengujian yang dilakukan adalah sebagai berikut.

Tabel 3. Skenario dan hasil pengujian *detection-disabling attacks*.

Skenario	Hasil
Input: AVI video, 72 *frames*, 10 *frame* dengan *copyright* ***Attacks***: Rotasi 90 derajat searah jarum jam 7 dari 10 *frame* dengan *copyright*	**Output**: 3 *frame copyright*
Input: AVI video, 72 *frames*, 15 *frame* dengan *copyright* ***Attacks***: *Mirroring* 10 dari 15 *frame* dengan *copyright*	**Output**: 5 *frame copyright*

Berdasarkan hasil pengujian pada Tabel 3, serangan *detection-disabling attacks* mampu bekerja secara efektif. Hal ini ditunjukkan dengan tidak ada *frame* ber*watermark* yang berhasil dideteksi pada skenario pertama dan kedua.

3. *Ambiguity attacks*

   Ambiguity attacks bertujuan untuk membingungkan proses ekstraksi *watermark* dengan membentuk dan menyisipkan *watermark* palsu[3].

Tabel 4. Skenario dan hasil pengujian *ambiguity attacks*.

Skenario	Hasil
Input: AVI video, 72 *frames*, 10 *frame* dengan *copyright* ***Attacks***: Menambah *copyright* palsu pada 5 *frame*	**Output**: 10 *frame copyright*
Input: AVI video, 72 *frames*, 15 *frame* dengan *copyright* ***Attacks***: Menambah *copyright* palsu pada 8 *frame*	**Output**: 15 *frame copyright*

Hasil pengujian *ambiguity attacks* pada Tabel 4 menunjukkan bahwa serangan ini tidak bisa mengganggu proses pendeteksian *watermark*.

V. Kesimpulan dan Saran

Pemanfaatan *robust watermarking* untuk pembuktian *copyright* video pada media *sharing* cukup efektif diaplikasikan. Terlebih, metode ini termasuk *robust* terhadap serangan *watermarking* berupa *simple attacks* dan *ambiguity attacks*. Tetapi belum mampu menangani serangan *detection-disabling attacks.*

Saran untuk pengembangan selanjutnya yaitu dapat dilakukan percobaan terhadap metode *watermarking* atau metode transformasi selain DCT lainnya untuk membandingkan kinerja dan ketahanan terhadap serangan *watermark*. Selain itu, dapat dilakukan pula pengujian terhadap jenis serangan *watermark* lainnya.

References

[1] Chandramouli, R.; Memon, N.; Rabbani, M. (2002). Digital Watermarking

[2] Munir, Rinaldi. (2019). Slide Kuliah IF4020 Kriptografi: Digital Watermarking.

[3] Hood, A., A.; Janwe, N., J. (2013). Robust Video Watermarking Techniques and Attacks on Watermark – A Review.

PERNYATAAN

Dengan ini saya menyatakan bahwa makalah yang saya tulis ini adalah tulisan saya sendiri, bukan saduran, atau terjemahan dari makalah orang lain, dan bukan plagiasi.

Bandung, 10 Mei 2019

Diki Ardian W.
(13515092)