



TUGAS AKHIR - SM091332

WATERMARKING DENGAN METODE DEKOMPOSISI NILAI SINGULAR PADA CITRA DIGITAL

**LATIFATUL MACHBUBAH
NRP 1209 100 027**

**Dosen Pembimbing
Drs. Soetrisno, MI.Komp**

**JURUSAN MATEMATIKA
Fakultas Matematika dan Ilmu Pengetahuan Alam
Institut Teknologi Sepuluh Nopember
Surabaya 2014**



FINAL PROJECT - SM091332

***WATERMARKING WITH SINGULAR VALUE
DECOMPOSITION METHOD OF DIGITAL IMAGE***

**LATIFATUL MACHBUBAH
NRP 1209 100 027**

**Supervisor
Drs. Soetrisno, MI.Komp**

**MATHEMATICS DEPARTMENT
Faculty of Mathematics and Science
Sepuluh Nopember Institute of Technology
Surabaya 2014**

***Watermarking* dengan Metode Dekomposisi Nilai Singular pada Citra Digital**

Nama : Latifatul Machbubah
NRP : 1209100027
Jurusan : Matematika
Dosen Pembimbing : Drs. Soetrisno, MI.Komp

Watermarking merupakan salah satu cara untuk mengamankan citra yang disajikan dalam format digital. Dalam *watermarking* terdapat dua proses utama, yakni proses penyisipan dan proses ekstraksi. Pada Tugas Akhir ini, dibahas *watermarking* dengan metode dekomposisi nilai singular (*singular value decomposition*), yang selanjutnya disebut SVD pada citra digital. Dalam metode ini dibutuhkan informasi dari citra asli untuk proses ekstraksi.

Proses penyisipan dengan metode SVD ini menghasilkan citra ter-*watermark* yang digunakan sebagai citra utama pada proses ekstraksi. Proses ekstraksi dilakukan pada citra ter-*watermark* yang mengalami gangguan berupa *noise*. Nilai PSNR yang diperoleh dari hasil percobaan antara citra asli dan citra ter-*watermark* memiliki kualitas yang baik untuk nilai diatas 79dB pada setiap citra dan untuk nilai koefisien korelasi dari estimasi *watermark* memiliki kualitas baik untuk nilai diatas 0,5.

Kata kunci : Citra Digital, *Watermarking*, Dekomposisi Nilai Singular

Watermarking with Singular Value Decomposition Method at Digital Image

Name : Latifatul Machbubah
NRP : 1209100027
Department : Mathematics
Supervisor : Drs. Soetrisno, MI.Komp

Watermarking is one of method to secure image presented in a digital format. In watermarking, there are two main processes, that is the process of embedding and extraction process. In this final project, discussed watermarking with singular value decomposition method (singular value decomposition), which further called the SVD in digital image. In this method takes the information from original image for extraction process.

Embedding process with the SVD method produces a watermarked image is used as the main image on extraction process. Extraction process is performed on the watermarked image having a noise disturbance. PSNR values obtained from the experiment between original image and watermarked image have a good quality for value above 79dB in every image and for the value of the correlation coefficient of the estimated watermark is quite good for values above 0.5.

Keywords : Digital Image, Watermarking, Singular Value Decomposition

LEMBAR PENGESAHAN

WATERMARKING DENGAN METODE DEKOMPOSISI NILAI SINGULAR PADA CITRA DIGITAL

WATERMARKING WITH SINGULAR VALUE DECOMPOSITION METHOD AT DIGITAL IMAGE

TUGAS AKHIR

Diajukan untuk memenuhi salah satu syarat
memperoleh gelar Sarjana Sains
Pada bidang studi Ilmu Komputer
Program Studi S-1 Jurusan Matematika
Fakultas Matematika dan Ilmu Pengetahuan Alam
Institut Teknologi Sepuluh Nopember Surabaya

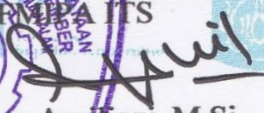
Oleh :
LATIFATUL MACHBUBAH
NRP. 1209 100 027

Menyetujui,
Dosen Pembimbing


Drs. Soetrisno, Ml.Komp
NIP. 19571103 198603 1 003

Mengetahui,

Ketua Jurusan Matematika
Fakultas Matematika dan Ilmu Pengetahuan Alam ITS


Dr. Erna Apriliani, M.Si

NIP. 19660414 199102 2 001
Surabaya, Januari 2014



KATA PENGANTAR

Segenap puji syukur penulis sampaikan kepada Allah SWT, karena atas rahmat dan hidayah-Nya, perencanaan, pelaksanaan, dan penyelesaian Tugas Akhir ini dapat diselesaikan dengan baik. Semoga sholawat serta salam tetap tercurahkan kepada junjungan Nabi Muhammad SAW yang telah membimbing umat-Nya dari zaman jahiliyah menuju zaman yang penuh ilmu.

Kesuksesan ini dapat penulis peroleh karena dukungan banyak pihak. Oleh karena itu, penulis sampaikan terima kasih yang sedalam-dalamnya kepada:

1. Ibu Dr. Erna Apriliani, M.Si selaku Ketua Jurusan yang memberikan dukungan dan kemudahan dalam penyelesaian Tugas Akhir ini.
2. Bapak Drs. Soetrisno, MI.Komp selaku Dosen Pembimbing yang senantiasa membimbing dan mengarahkan penulis sehingga Tugas Akhir ini dapat terselesaikan.
3. Bapak dan Ibu Dosen Penguji yang telah memberikan masukan berupa kritik dan saran yang bersifat membangun guna kesempurnaan Tugas Akhir ini.
4. Kedua orangtua penulis yang selalu memberi dukungan dan doa, sehingga penulis dapat menyelesaikan Tugas Akhir ini.
5. Seluruh keluarga besar Jurusan Matematika yang telah memberikan kemudahan dalam menyelesaikan Tugas Akhir ini.
6. Teman-teman, dan para sahabat serta semua pihak yang tidak dapat penulis sebutkan satu-persatu yang turut membantu hingga terselesainya Tugas Akhir ini.

Penulis menyadari bahwa penyusunan Tugas Akhir ini jauh dari sempurna oleh karenanya kepada semua pembaca Tugas Akhir ini diharapkan kritik dan saran yang membangun guna memberikan sebuah karya Tugas Akhir yang lebih baik kedepannya.

Akhirnya, semoga segala amal baik yang telah Bapak/Ibu/Saudara berikan kepada penulis mendapat balasan yang sebaik mungkin dari Allah, penguasa alam seisinya. Amin

Surabaya, Januari 2014

Penulis

DAFTAR ISI

	Halaman
HALAMAN JUDUL.....	i
LEMBAR PENGESAHAN	v
ABSTRAK	vii
ABSTRACT	ix
KATA PENGANTAR.....	xi
DAFTAR ISI.....	xiii
DAFTAR GAMBAR	xv
DAFTAR TABEL.....	xvii
BAB 1 PENDAHULUAN	1
1.1 Latar Belakang	1
1.2 Rumusan Masalah	2
1.3 Batasan Masalah	3
1.4 Tujuan	3
1.5 Manfaat	3
1.6 Sistematika Penulisan.....	4
BAB 2 TINJAUAN PUSTAKA.....	7
2.1 Citra Digital.....	7
2.1.1 Macam-Macam Citra	8
2.2 Watermarking	9
2.3 Singular Value Decomposition (SVD)	11
2.4 Watermarking berbasis Metode SVD	16
2.5 Matlab	20
BAB 3 ANALISIS SISTEM	21
3.1 Langkah-Langkah Sistem.....	21
3.2 Analisis Koefisien Korelasi.....	24
3.3 Analisis PSNR	33
3.4 Penambahan Distorsi	34
BAB 4 PERANCANGAN DAN IMPLEMENTASI SISTEM	35
4.1 Langkah-Langkah Program.....	35
4.1.1 Langkah-Langkah Program Watermarking dalam Proses Penyisipan Watermark pada Citra Digital ...	35
4.1.2 Langkah-Langkah Program Watermarking dalam Proses Ekstraksi Watermark pada Citra Digital	36
4.1.3 Langkah-Langkah Pengujian Hasil Nilai Koefisien Korelasi	37

4.1.4 Langkah-Langkah Pengujian Hasil Menggunakan PSNR	39
4.2 Perancangan Program	40
4.2.1 Data Flow Diagram (DFD)	40
a. Proses Watermarking pada Citra Digital Menggunakan Metode SVD.....	40
b. Proses Penambahan Noise.....	41
c. Proses Ekstraksi Watermark	42
d. Proses Menghitung Nilai Koefisien Korelasi ...	43
e. Proses Menghitung Nilai MSE dan PSNR	44
4.2.2 Fungsi-Fungsi yang Dibutuhkan Program Watermarking Menggunakan Metode SVD	44
4.3 Ruang Lingkup Implementasi	46
4.4 Desain Tampilan	46
4.4.1 Program Watermarking Metode SVD	47
4.4.2 Program Ekstraksi	48
4.4.3 Program Penghitungan PSNR	50
4.4.4 Program Penghitungan Koefisien Korelasi.....	51
4.5 Proses Pengambilan Citra	53
4.6 Proses Penyimpanan Citra	54
4.7 Proses Pembuatan Citra Ter-watermark.....	54
4.8 Proses Penambahan Noise	55
4.9 Proses Ekstraksi Watermark	56
4.10 Proses PSNR	57
4.11 Proses Koefisien Korelasi	57
BAB 5 HASIL DAN PEMBAHASAN PENGUJIAN SISTEM.....	59
5.1 Ruang Lingkup Uji Coba	59
5.2 Data Uji Coba.....	59
5.3 Pelaksanaan Uji Coba	61
5.3.1 Uji Coba dengan Nilai Alpha yang Berbeda	61
5.3.2 Uji Coba dengan Penambahan Noise	70
BAB 6 PENUTUP.....	81
6.1 Kesimpulan	81
6.2 Saran	81
DAFTAR PUSTAKA	83
LAMPIRAN.....	85
BIODATA PENULIS	95

DAFTAR GAMBAR

Gambar	Judul	Halaman
2.1	Koordinat Suatu Piksel pada Citra Digital	7
2.2	Proses <i>Watermarking</i>	10
2.3	Proses Penyisipan	17
2.4	Proses Ekstraksi	17
3.1	Diagram Alir Proses Penyisipan <i>Watermark</i>	22
3.2	Diagram Alir Proses Ekstraksi <i>Watermark</i>	23
3.3	Citra yang telah diberi <i>Noise</i> (a) <i>Gaussian</i> , (b) <i>Speckle</i> , (c) <i>Salt & Pepper</i>	34
4.1	Diagram Alir Program Penambahan <i>Noise</i>	37
4.2	Diagram Alir Program Koefisien Korelasi	38
4.3	Diagram Alir Program Penghitungan PSNR	39
4.4	Data Flow Diagram Proses Penyisipan <i>Watermark</i>	40
4.5	Data Flow Diagram Proses Penambahan <i>Noise</i>	41
4.6	Data Flow Diagram Proses Ekstraksi <i>Watermark</i>	42
4.7	Data Flow Diagram Koefisien Korelasi	43
4.8	Data Flow Diagram MSE dan PSNR	44
4.9	Desain Antarmuka Menu Utama	47
4.10	Desain Antarmuka Program <i>Watermarking</i> pada Citra Digital Menggunakan Metode SVD	48
4.11	Desain Antarmuka Program Ekstraksi Citra <i>Watermark</i>	49
4.12	Desain Antarmuka Program PSNR	51
4.13	Desain Antarmuka Program Koefisien Korelasi	52
4.14	Kode Program Pengambilan Citra Input	53
4.15	Kode Program Penyimpanan Citra Output	54
4.16	Kode Program Pembuatan Citra Ter- <i>watermark</i>	55
4.17	Kode Program Penambahan <i>Noise Gaussian</i>	55
4.18	Kode Program Penambahan <i>Noise Speckle</i>	56
4.19	Kode Program Penambahan <i>Noise Salt & Pepper</i>	56
4.20	Kode Program Ekstraksi <i>Watermark</i>	56
4.21	Kode Program Penghitungan PSNR	57
4.22	Kode Program Penghitungan Koefisien Korelasi	57

5.1	Citra Asli (a) Lena.jpg, (b) Cameraman.jpg dan (c) Baboon.jpg	60
5.2	Citra Watermark (a) watermark1.jpg dan (b) watermark2.jpg	60
5.3	Citra Lena Ter- <i>watermark</i> dengan Alpha (a) 0.001 (b) 0.01 (c) 0.03 dan (d) 0.1 dari Hasil Uji Coba 5.3.1.a	62
5.4	Citra Lena Ter- <i>watermark</i> dengan Alpha (a) 0.001 (b) 0.01 (c) 0.03 dan (d) 0.1 dari Hasil Uji Coba 5.3.1.b	63
5.5	Citra Cameraman Ter- <i>watermark</i> dengan Alpha (a) 0.001 (b) 0.01 (c) 0.03 dan (d) 0.1 dari Hasil Uji Coba 5.3.1.c	65
5.6	Citra Cameraman Ter- <i>watermark</i> dengan Alpha (a) 0.001 (b) 0.01 (c) 0.03 dan (d) 0.1 dari Hasil Uji Coba 5.3.1.c	66
5.7	Citra Baboon Ter- <i>watermark</i> dengan Alpha (a) 0.001 (b) 0.01 (c) 0.03 dan (d) 0.1 dari Hasil Uji Coba 5.3.1.e	67
5.8	Citra Baboon Ter- <i>watermark</i> dengan Alpha (a) 0.001 (b) 0.01 (c) 0.03 dan (d) 0.1 dari Hasil Uji Coba 5.3.1.f	69

DAFTAR TABEL

Tabel	Judul	Halaman
4.1	Ruang Lingkup Implementasi Program	46
5.1	Nilai PSNR dari Citra Asli dan Citra Ter- <i>watermark</i> pada Hasil Uji Coba 5.3.1.a	62
5.2	Nilai PSNR dari Citra Asli dan Citra Ter- <i>watermark</i> pada Hasil Uji Coba 5.3.1.b	64
5.3	Nilai PSNR dari Citra Asli dan Citra Ter- <i>watermark</i> pada Hasil Uji Coba 5.3.1.c	65
5.4	Nilai PSNR dari Citra Asli dan Citra Ter- <i>watermark</i> pada Hasil Uji Coba 5.3.1.d	66
5.5	Nilai PSNR dari Citra Asli dan Citra Ter- <i>watermark</i> pada Hasil Uji Coba 5.3.1.e	68
5.6	Nilai PSNR dari Citra Asli dan Citra Ter- <i>watermark</i> pada Hasil Uji Coba 5.3.1.f	69
5.7	Kualitas Estimasi <i>Watermark</i> dari Hasil Uji Coba 5.3.2.a	70
5.8	Kualitas Estimasi <i>Watermark</i> dari Hasil Uji Coba 5.3.2.b	71
5.9	Kualitas Estimasi <i>Watermark</i> dari Hasil Uji Coba 5.3.2.c	71
5.10	Kualitas Estimasi <i>Watermark</i> dari Hasil Uji Coba 5.3.2.d	72
5.11	Kualitas Estimasi <i>Watermark</i> dari Hasil Uji Coba 5.3.2.e	72
5.12	Kualitas Estimasi <i>Watermark</i> dari Hasil Uji Coba 5.3.2.f	73
5.13	Kualitas Estimasi <i>Watermark</i> dari Hasil Uji Coba 5.3.2.g	73
5.14	Kualitas Estimasi <i>Watermark</i> dari Hasil Uji Coba 5.3.2.h	74
5.15	Kualitas Estimasi <i>Watermark</i> dari Hasil Uji Coba 5.3.2.i	74

5.16	Kualitas Estimasi <i>Watermark</i> dari Hasil Uji Coba 5.3.2.j	75
5.17	Kualitas Estimasi <i>Watermark</i> dari Hasil Uji Coba 5.3.2.k	75
5.18	Kualitas Estimasi <i>Watermark</i> dari Hasil Uji Coba 5.3.2.l	75
5.19	Kualitas Estimasi <i>Watermark</i> dari Hasil Uji Coba 5.3.2.m	76
5.20	Kualitas Estimasi <i>Watermark</i> dari Hasil Uji Coba 5.3.2.n	76
5.21	Kualitas Estimasi <i>Watermark</i> dari Hasil Uji Coba 5.3.2.o	77
5.22	Kualitas Estimasi <i>Watermark</i> dari Hasil Uji Coba 5.3.2.p	77
5.23	Kualitas Estimasi <i>Watermark</i> dari Hasil Uji Coba 5.3.2.q	78
5.24	Kualitas Estimasi <i>Watermark</i> dari Hasil Uji Coba 5.3.2.r	78

BAB I

PENDAHULUAN

Pada bab ini dijelaskan mengenai latar belakang permasalahan, rumusan masalah, batasan penelitian, tujuan dan manfaat penelitian, serta sistematika penulisan.

1.1 Latar Belakang

Saat ini sebagian besar data multimedia baik berupa data citra, audio, maupun video, disajikan dalam format digital. Hal tersebut dikarenakan data digital mudah digandakan dan disimpan, mudah diolah lebih lanjut, serta mudah didistribusikan. Salah satu sarana pendistribusian data digital adalah melalui internet. Penggunaan internet dalam kehidupan sehari-hari telah menjadi hal yang umum. Setiap tahunnya pengguna internet kian bertambah. Pendistribusian data digital yang mudah ini ternyata tidak sepenuhnya memberikan dampak yang positif. Semakin berkembang dan populernya internet menyebabkan semakin tinggi pula pelanggaran-pelanggaran terhadap hak cipta karya digital sehingga perlindungan hak cipta telah menjadi bagian penting dalam dunia informasi.

Citra digital merupakan salah satu data digital yang paling banyak digunakan dalam aplikasi multimedia. Seperti data digital lainnya, citra digital memiliki karakteristik yang sekaligus menjadi kelemahan, yakni citra digital mudah digandakan dan hasilnya sama dengan citra aslinya serta mudah didistribusikan. Dalam beberapa kasus dibutuhkan keamanan terhadap citra digital agar tidak terjadi pelanggaran hak cipta dan kebocoran rahasia.

Selama ini penggandaan atas produk digital seperti citra digital dilakukan begitu bebas dan leluasa secara ilegal. Hasil penggandaan tersebut memiliki kualitas yang sama dengan produk digital aslinya. Namun, pemilik asli citra digital tidak mendapatkan royalti dari penggandaan tersebut. Akibatnya

pemegang hak cipta produk digital sangat dirugikan. Salah satu solusi efektif terhadap masalah distribusi yang tidak sah adalah dengan menanamkan *watermark* digital kedalam citra digital tersebut. *Watermark* adalah kode digital yang tidak bisa dibuang, kuat, dan tidak diketahui oleh indera manusia, yang tertanam dalam citra asli. Teknik yang digunakan untuk menyisipkan *watermark* disebut *watermarking*. Memang selama ini teknik *watermarking* digunakan sebagai salah satu cara alternatif untuk mengirimkan pesan rahasia. Namun tidak menutup kemungkinan bahwa teknik *watermarking* ini bisa digunakan sebagai alat untuk identifikasi kepemilikan, hak penggunaan, dan integritas dari citra digital.

Saat ini berbagai metode *watermarking* telah berkembang. Masing-masing metode tersebut memiliki kelebihan dan kelemahan. Salah satu metode yang digunakan adalah metode yang bekerja pada domain spasial. Pada umumnya, metode ini kurang kompleks sehingga tidak tahan terhadap gangguan-gangguan pada citra digital. Sedangkan metode *watermarking* yang bekerja pada domain transform lebih kompleks dan lebih tahan terhadap gangguan-gangguan pada citra digital. Oleh karena itu dalam penelitian ini dikembangkan sebuah metode yang bekerja pada domain transform, yakni metode dekomposisi nilai singular atau *Singular Value Decomposition*, yang untuk selanjutnya disebut SVD. Teknik *watermarking* dengan menggunakan metode SVD ini, pada umumnya adalah penyisipan dilakukan pada nilai-nilai singular. Hal ini berdasarkan pertimbangan bahwa nilai singular dari sebuah citra mempunyai stabilitas yang bagus artinya sebuah citra tidak akan mengalami perubahan signifikan jika terjadi sedikit gangguan.

1.2 Rumusan Masalah

Perumusan masalah yang dibahas dalam Tugas Akhir ini berdasarkan latar belakang di atas adalah :

1. Bagaimana membangun aplikasi untuk proses penyisipan *watermark* pada citra digital dengan metode SVD?

2. Bagaimana membangun aplikasi untuk proses ekstraksi *watermark* pada citra digital dengan metode SVD?
3. Bagaimana menguji ketahanan dari teknik *watermarking* menggunakan metode SVD?

1.3 Batasan Masalah

Pada Tugas Akhir ini ditentukan batasan masalahnya adalah sebagai berikut :

1. Metode *watermarking* yang digunakan adalah berbasis SVD.
2. Citra asli yang digunakan berukuran $N \times N$ berupa citra *grayscale* dalam format JPG atau BMP.
3. *Watermark* yang disisipkan adalah berupa citra *grayscale* yang ukurannya lebih kecil dari citra asli dalam format JPG atau BMP.
4. Gangguan atau distorsi yang diberikan pada citra yang telah disisipi *watermark* berupa pemberian *noise* seperti *noise Gaussian*, *speckle*, dan *salt & pepper*.
5. *Software* yang digunakan dalam implementasinya menggunakan program Matlab.

1.4 Tujuan

Tujuan dari Tugas Akhir ini adalah :

1. Membangun aplikasi dalam melakukan proses penyisipan *watermark* pada citra digital dengan metode SVD.
2. Membangun aplikasi dalam melakukan proses ekstraksi *watermark* pada citra digital dengan metode SVD.
3. Menguji ketahanan dari teknik *watermarking* menggunakan metode SVD

1.5 Manfaat

Manfaat dari Tugas Akhir ini adalah untuk membantu pengguna dalam menyisipkan dan mengekstraksi *watermark* yang dimilikinya sehingga dapat melindungi citra digital tersebut dari penyalahgunaan oleh pihak-pihak yang tidak bertanggung jawab. Selain itu, Tugas Akhir ini juga bermanfaat sebagai referensi

untuk penelitian pada pengembangan sistem keamanan dari citra digital selanjutnya.

1.6 Sistematika Penulisan

Tugas Akhir ini secara keseluruhan terdiri dari enam bab dan lampiran. Secara garis besar masing-masing bab akan membahas hal-hal sebagai berikut:

BAB I PENDAHULUAN

Bab ini berisi latar belakang permasalahan, perumusan masalah, batasan-batasan masalah, tujuan dan manfaat penelitian serta sistematika penulisan.

BAB II TINJAUAN PUSTAKA

Bab ini membahas tentang teori mengenai konsep dasar citra digital, *watermarking*, SVD, *watermarking* berbasis SVD dan Matlab.

BAB III ANALISIS SISTEM

Bab ini membahas tentang proses penyisipan *watermark* dengan metode SVD dan proses ekstraksinya.

BAB IV PERANCANGAN DAN IMPLEMENTASI SISTEM

Bab ini membahas tentang perancangan dan pengimplementasian rancangan program dari proses *watermark*, pemberian distorsi dan uji ketahanan dengan menggunakan *software* Matlab. Selain itu juga dibahas tentang bagaimana sistem akan berinteraksi dengan pengguna.

BAB V UJI COBA PROGRAM

Bab ini membahas tentang hasil uji coba program. Kualitas citra yang telah mengalami proses *watermarking* akan diberi gangguan pada citra ter-*watermark*, menghitung *Peak Signal to Noise Ratio (PSNR)* antara citra asli dan ter-*watermark* dan

menghitung kesamaan antara citra *watermark* dengan estimasi citra *watermark* dengan menggunakan koefisien korelasi.

BAB VI PENUTUP

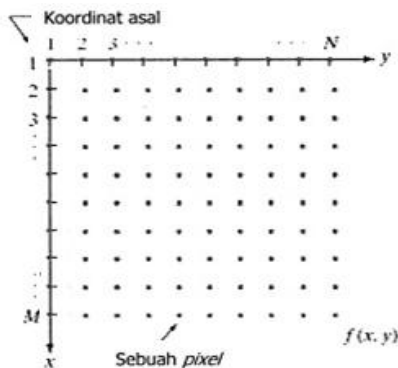
Bab ini berisi kesimpulan yang diambil dari hasil uji coba dan analisa. Selain itu juga terdapat saran-saran untuk pengembangan lebih lanjut dari Tugas Akhir ini.

BAB II TINJAUAN PUSTAKA

Pada bab ini dibahas teori-teori yang terkait dengan permasalahan dalam tugas akhir ini. Pertama, dibahas mengenai citra digital. Selanjutnya dibahas mengenai *watermarking*, SVD, *watermarking* berbasis pengembangan SVD dan yang terakhir dasar teori tentang Matlab.

2.1 Citra Digital

Sebuah citra dapat didefinisikan sebagai suatu fungsi dua dimensi $f(x, y)$ berukuran $M \times N$, dengan x dan y adalah koordinat spasial dan f merupakan amplitudo di titik koordinat (x, y) yang sering disebut sebagai intensitas atau tingkat keabuan dari citra pada titik tersebut[1]. Nilai dari intensitas berbentuk diskrit, mulai dari 0 sampai 255. Apabila nilai x, y dan nilai amplitudo f secara keseluruhan berhingga dan bernilai diskrit maka dapat dikatakan citra tersebut sebagai citra digital. Citra digital tersusun dari sejumlah nilai tingkat keabuan yang dikenal sebagai piksel (pixel) pada posisi tertentu. Gambar 2.1 menunjukkan posisi koordinat suatu piksel pada citra digital.



Gambar 2.1 Koordinat Suatu Piksel pada Citra Digital

Suatu citra digital dapat dinyatakan dalam bentuk matriks sebagai berikut[2]:

$$f(x, y) = \begin{pmatrix} f(1,1) & f(1,2) & \cdots & f(1, N) \\ f(2,1) & f(2,2) & \cdots & f(2, N) \\ \vdots & \vdots & \cdots & \vdots \\ f(M, 1) & f(M, 2) & \cdots & f(M, N) \end{pmatrix} \quad (2.1)$$

2.1.1 Macam-Macam Citra

Berdasarkan jumlah bit per piksel, citra dapat digolongkan menjadi empat tipe, yakni :

- a. Citra biner yang hanya terdiri dari dua nilai, yaitu '1' yang menunjukkan warna putih dan '0' yang menunjukkan warna hitam.
- b. Citra *grayscale* yang merupakan citra monokrom atau gambar yang terdiri dari warna hitam sampai warna putih. Nilai-nilai bit yang terdapat pada setiap piksel menentukan tingkat kecerahan yang berbeda pada citra. Citra ini direpresentasikan dengan data 8 bit per piksel yang menentukan 256 tingkat kecerahan yang berbeda, dimulai dari 0 sampai dengan 255. Pada aplikasi tertentu seperti citra medis atau astronomi terdiri dari 12 atau 16 bit per piksel karena adanya penambahan detail kejelasan citra.
- c. Citra warna yang terdiri dari tiga berkas monokrom dimana tiap berkasnya mewakili warna yang berbeda dan informasi mengenai kecerahan citra tergantung pada ketiganya. Setiap berkas direpresentasikan dengan data 8 bit per piksel sehingga citra warna ini memiliki 24 bit per piksel untuk data digitalnya. Tiga berkas data tersebut adalah merah (red), hijau (green) dan biru (blue) atau sering disebut RGB.
- d. Citra multispektral yang mengandung informasi diluar kemampuan persepsi visual manusia.

Jumlah bit per piksel akan mempengaruhi jumlah bit citra secara keseluruhan. Bila citra *grayscale* memiliki ukuran piksel

sebesar 256×256 , maka jumlah bit keseluruhan citra adalah $256 \times 256 \times 8 \text{ bit} = 524288 \text{ bit}$.

2.2 Watermarking

Watermark merupakan sebuah informasi yang disisipkan pada media lain dengan tujuan melindungi media yang disisipi oleh informasi tersebut dari pembajakan dan penyalahgunaan hak cipta. *Watermarking* sendiri adalah suatu cara untuk menyembunyikan atau menanamkan data tertentu ke dalam suatu data digital lainnya, tetapi tidak diketahui kehadirannya oleh indera manusia dan mampu menghadapi gangguan data digital sampai pada tahap tertentu.

Seiring dengan perkembangan zaman, *watermarking* muncul pada media digital atau yang biasanya disebut dengan digital *watermarking*. Digital *watermarking* sendiri adalah sebuah kode identifikasi yang secara permanen disisipkan ke dalam data digital.

Ada beberapa tujuan yang ingin dicapai dari penggunaan *watermarking* sebagai suatu teknik penyembunyian data pada suatu data digital lain yaitu:

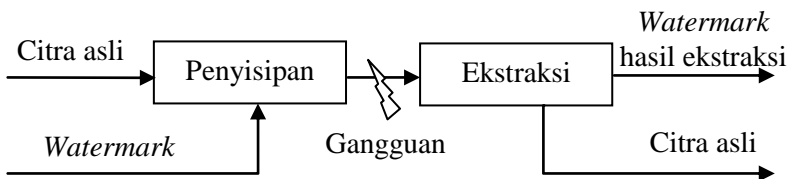
1. *Tamper-proofing*; *Watermarking* digunakan sebagai indikator yang menunjukkan apakah data digital yang asli telah mengalami perubahan (mengecek integritas data).
2. *Feature location*; *Watermarking* sebagai alat identifikasi isi dari data digital pada lokasi-lokasi tertentu, misalnya penamaan suatu objek tertentu dari beberapa objek yang ada pada suatu citra digital.
3. *Annotation / caption*; *Watermark* berisi keterangan tentang data digital itu sendiri. Selain itu *watermark* juga dapat digunakan untuk mengirimkan pesan rahasia.
4. *Copyright-labelling*; *Watermarking* digunakan sebagai metode untuk menyembunyikan label hak cipta pada data digital atau sebuah bukti otentik kepemilikan atas dokumen digital tersebut.

Untuk memenuhi tujuan diatas, teknik *watermarking* harus memenuhi beberapa kriteria berikut:

1. *Imperceptibility*; Antara citra asli dengan citra yang telah disisipi *watermark* secara persepsi tidak dapat dibedakan oleh mata manusia. *Watermark* tidak mengalami interferensi dengan mediana.
2. *Trustworthiness*; Menjamin tidak akan dapat dibangkitkan *watermark* yang sama dengan *watermark* asli dan menyediakan bukti terpercaya untuk melindungi hak kepemilikan.
3. *Robustness*; *Watermark* harus tahan terhadap berbagai macam serangan.
4. *Noninvertibility*; Secara komputasi sangat sukar menemukan *watermark* jika yang diketahui hanya citra ter*watermark* saja.

Berdasarkan analisis verifikasinya, *watermarking* dibedakan menjadi dua kategori, yakni sebagai *blind watermarking* (analisis verifikasi *watermark* yang tidak membutuhkan informasi dari citra asli) dan *non-blind watermarking* (analisis verifikasi *watermark* yang membutuhkan informasi dari citra asli)[3].

Proses *watermarking* perlu didukung dengan proses ekstraksi *watermark* dari citra ter*watermark*. Proses ekstraksi ini bertujuan untuk mendapatkan kembali citra *watermark* yang disisipkan dalam citra digital tersebut. Umumnya proses ekstraksi melibatkan proses perbandingan citra digital asal dengan citra ter*watermark* untuk mendapatkan *watermark* yang disisipkan[4]. Proses *watermarking* ditunjukkan dengan Gambar 2.2 berikut :



Gambar 2.2 Proses *Watermarking*

Biasanya teknik *watermarking* yang tahan dari berbagai serangan memiliki kualitas gambar ber-*watermark* yang kurang memuaskan, demikian juga sebaliknya, teknik *watermarking* yang menghasilkan kualitas gambar yang memuaskan biasanya kurang kuat atau tahan menghadapi gangguan. Secara umum jenis gangguan terhadap citra ber-*watermark* dibagi menjadi dua, yakni gangguan standar (*standard attack*) seperti *cropping* maupun kompresi atau penambahan *noise*, dan *malicious attack*.

2.3 Singular Value Decomposition (SVD)

Singular Value Decomposition (SVD) atau Dekomposisi Nilai Singular adalah salah satu teknik dalam analisis numerik yang digunakan untuk mendiagonalkan matriks. SVD merupakan salah satu alat yang dapat digunakan untuk merepresentasikan sebuah matriks. SVD juga merupakan suatu teknik yang handal dalam melakukan berbagai analisis dan komputasi matriks, yaitu dengan mengekspos struktur geometrinya, sehingga dapat diketahui beberapa properti penting dari matriks tersebut. Sebuah matriks yang direpresentasikan dengan SVD akan didekomposisi menjadi 3 komponen matriks, yaitu matriks vektor singular kiri, matriks nilai singular, dan matriks vektor singular kanan. Dalam persamaan ditulis sebagai berikut[5]:

$$A = U\Sigma V^T \quad (2.2)$$

dengan A adalah matriks berukuran $N \times N$, U dan V adalah matriks orthogonal atau vektor singular kiri dan kanan berukuran $N \times N$ dan Σ adalah matriks $N \times N$ yang semua elemen diluar diagonalnya adalah 0 dan elemen-elemen diagonalnya memenuhi persamaan:

$$\sigma_1 \geq \sigma_2 \geq \dots \geq \sigma_n \geq 0 \quad (2.3)$$

sehingga,

$$\Sigma = \begin{pmatrix} \sigma_1 & 0 & \dots & 0 \\ 0 & \sigma_2 & & \vdots \\ \vdots & & \ddots & 0 \\ 0 & \dots & 0 & \sigma_n \end{pmatrix} \quad (2.4)$$

dan T menunjukkan bahwa V adalah matriks transpose. Semua σ_i yang ditentukan dengan faktorisasi ini adalah tunggal dan disebut dengan nilai-nilai singular dari matriks A . Proses SVD dapat dilihat pada Contoh 2.1 berikut ini :

Contoh 2.1

Suatu citra disajikan dengan matriks $A = \begin{bmatrix} 2 & -2 \\ 1 & 2 \end{bmatrix}$, langkah-langkah dalam mencari matriks U, V dan Σ adalah sebagai berikut :

Langkah 1 : mencari matriks $A^T A$

$$A^T A = \begin{bmatrix} 2 & 1 \\ -2 & 2 \end{bmatrix} \begin{bmatrix} 2 & -2 \\ 1 & 2 \end{bmatrix} = \begin{bmatrix} 5 & -2 \\ -2 & 8 \end{bmatrix}$$

Langkah 2 : menghitung nilai eigen untuk mendapatkan nilai singularnya dari determinan $A^T A$ dan mencari matriks Σ

$$\begin{aligned} |A^T A - \lambda I| &= 0 \\ \left| \begin{bmatrix} 5 & -2 \\ -2 & 8 \end{bmatrix} - \begin{bmatrix} \lambda & 0 \\ 0 & \lambda \end{bmatrix} \right| &= 0 \\ \left| \begin{bmatrix} 5 - \lambda & -2 \\ -2 & 8 - \lambda \end{bmatrix} \right| &= 0 \\ ((5 - \lambda)(8 - \lambda)) - ((-2)(-2)) &= 0 \\ (40 - 5\lambda - 8\lambda + \lambda^2) - (4) &= 0 \\ \lambda^2 - 13\lambda + 36 &= 0 \\ (\lambda - 9)(\lambda - 4) &= 0 \end{aligned}$$

$\lambda_1 = 9$ dan $\lambda_2 = 4$ adalah nilai eigennya sehingga didapatkan nilai singularnya $\sigma_1 = \sqrt{\lambda_1} = \sqrt{9} = 3$ dan $\sigma_2 = \sqrt{\lambda_2} = \sqrt{4} = 2$

Maka matriks $\Sigma = \begin{bmatrix} 3 & 0 \\ 0 & 2 \end{bmatrix}$

Langkah 3 : mencari matriks V

Untuk $\lambda_1 = 9$ didapat

$$\begin{aligned} (A^T A - \lambda_1 I)v &= 0 \\ \left(\begin{bmatrix} 5 & -2 \\ -2 & 8 \end{bmatrix} - \begin{bmatrix} 9 & 0 \\ 0 & 9 \end{bmatrix} \right) \begin{pmatrix} v_1 \\ v_2 \end{pmatrix} &= 0 \\ \begin{pmatrix} -4 & -2 \\ -2 & -1 \end{pmatrix} \begin{pmatrix} v_1 \\ v_2 \end{pmatrix} &= 0 \end{aligned}$$

Sehingga,

$$\begin{aligned} -4v_1 - 2v_2 &= 0 & \text{atau} & & -2v_1 - v_2 &= 0 \\ v_1 &= -\frac{1}{2}v_2 & & & v_1 &= -\frac{1}{2}v_2 \end{aligned}$$

Proses normalisasi

$$\begin{aligned} v_1^* &= \frac{v_1}{(v_1^T v_1)^{1/2}} \\ &= \frac{\begin{pmatrix} -\frac{1}{2}v_2 \\ v_2 \end{pmatrix}}{\left(\begin{pmatrix} -\frac{1}{2}v_2 & v_2 \end{pmatrix} \begin{pmatrix} -\frac{1}{2}v_2 \\ v_2 \end{pmatrix} \right)^{1/2}} \\ &= \frac{\begin{pmatrix} -\frac{1}{2} \\ 1 \end{pmatrix} v_2}{\left(\left(\frac{1}{4} + 1 \right) v_2^2 \right)^{1/2}} \end{aligned}$$

$$\begin{aligned}
 v_1^* &= \frac{\begin{pmatrix} -\frac{1}{2} \\ 1 \end{pmatrix} v_2}{\left(\frac{5}{4} v_2^2\right)^{1/2}} \\
 &= \frac{\begin{pmatrix} -\frac{1}{2} \\ 1 \end{pmatrix} v_2}{\frac{\sqrt{5}}{2} v_2} \\
 &= \begin{pmatrix} -\frac{1}{\sqrt{5}} \\ \frac{2}{\sqrt{5}} \end{pmatrix}
 \end{aligned}$$

Untuk $\lambda_2 = 4$ didapat

$$\begin{aligned}
 (A^T A - \lambda_2 I)v &= 0 \\
 \left(\begin{bmatrix} 5 & -2 \\ -2 & 8 \end{bmatrix} - \begin{bmatrix} 4 & 0 \\ 0 & 4 \end{bmatrix} \right) \begin{pmatrix} v_1 \\ v_2 \end{pmatrix} &= 0 \\
 \begin{pmatrix} 1 & -2 \\ -2 & 4 \end{pmatrix} \begin{pmatrix} v_1 \\ v_2 \end{pmatrix} &= 0
 \end{aligned}$$

Sehingga,

$$\begin{aligned}
 v_1 - 2v_2 &= 0 & \text{atau} & \quad -2v_1 - 4v_2 = 0 \\
 v_2 &= \frac{1}{2}v_1 & & \quad v_2 = \frac{1}{2}v_1
 \end{aligned}$$

Proses normalisasi

$$v_2^* = \frac{v_2}{(v_2^T v_2)^{1/2}}$$

$$\begin{aligned}
 v_2^* &= \frac{\begin{pmatrix} v_1 \\ \frac{1}{2}v_1 \end{pmatrix}}{\left(\begin{pmatrix} v_1 & \frac{1}{2}v_1 \end{pmatrix} \begin{pmatrix} v_1 \\ \frac{1}{2}v_1 \end{pmatrix} \right)^{1/2}} \\
 &= \frac{\begin{pmatrix} 1 \\ \frac{1}{2} \end{pmatrix} v_1}{\left(\left(1 + \frac{1}{4}\right) v_1^2 \right)^{1/2}} \\
 &= \frac{\begin{pmatrix} 1 \\ \frac{1}{2} \end{pmatrix} v_1}{\left(\frac{5}{4} v_1^2 \right)^{1/2}} \\
 &= \frac{\begin{pmatrix} 1 \\ \frac{1}{2} \end{pmatrix} v_1}{\frac{\sqrt{5}}{2} v_1} \\
 &= \begin{pmatrix} \frac{2}{\sqrt{5}} \\ \frac{1}{\sqrt{5}} \end{pmatrix}
 \end{aligned}$$

Sehingga dapat dibentuk matriks orthogonal V sebagai berikut :

$$\begin{aligned}
 V &= [v_1^* \quad v_2^*] \\
 &= \begin{bmatrix} \frac{1}{\sqrt{5}} & \frac{2}{\sqrt{5}} \\ \frac{2}{\sqrt{5}} & \frac{1}{\sqrt{5}} \end{bmatrix}
 \end{aligned}$$

Langkah 4 : mencari matriks U yang dibentuk dari $u_i = \frac{1}{\sigma_i} Av_i^*$

$$u_1 = \frac{1}{\sigma_1} Av_1^* = \frac{1}{3} \begin{bmatrix} 2 & -2 \\ 1 & 2 \end{bmatrix} \begin{pmatrix} -\frac{1}{\sqrt{5}} \\ \frac{2}{\sqrt{5}} \end{pmatrix} = \begin{pmatrix} -\frac{2}{\sqrt{5}} \\ \frac{1}{\sqrt{5}} \end{pmatrix}$$

$$u_2 = \frac{1}{\sigma_2} Av_2^* = \frac{1}{2} \begin{bmatrix} 2 & -2 \\ 1 & 2 \end{bmatrix} \begin{pmatrix} \frac{2}{\sqrt{5}} \\ \frac{1}{\sqrt{5}} \end{pmatrix} = \begin{pmatrix} \frac{1}{\sqrt{5}} \\ \frac{2}{\sqrt{5}} \end{pmatrix}$$

Sehingga dapat dibentuk matriks orthogonal U sebagai berikut :

$$U = [u_1 \quad u_2] \\ = \begin{bmatrix} -\frac{2}{\sqrt{5}} & \frac{1}{\sqrt{5}} \\ \frac{1}{\sqrt{5}} & \frac{2}{\sqrt{5}} \end{bmatrix}$$

Dari sudut pandang aplikasi pengolahan citra digital, SVD memiliki tiga sifat utama, yakni[6] :

1. Nilai-nilai singular dari suatu citra memiliki stabilitas yang baik.
2. Setiap nilai singular menentukan cahaya dari layer citra sementara pasangan vektor singular yang sesuai menentukan geometri citra.
3. Nilai-nilai singular mewakili sifat intrinsik citra.

2.4 Watermarking berbasis SVD

Pada dasarnya, proses *watermarking* dengan metode SVD pada citra digital dapat dibagi menjadi dua bagian yakni proses penyisipan dan ekstraksi *watermark*[7].

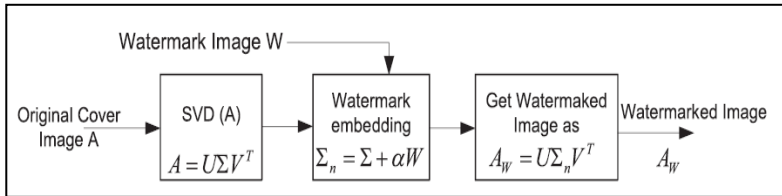
Pada proses penyisipan *watermark* ini terdapat tiga langkah untuk algoritma penyisipannya seperti pada Gambar 2.3, yaitu :

1. Melakukan SVD pada citra asli A seperti pada persamaan (2.2)
2. Menambahkan citra *watermark* W ke Σ , dengan konstanta positif α yang menunjukkan kekuatan dari *watermark* yang disisipkan[8].

$$\Sigma_n = \Sigma + \alpha W \quad (2.5)$$

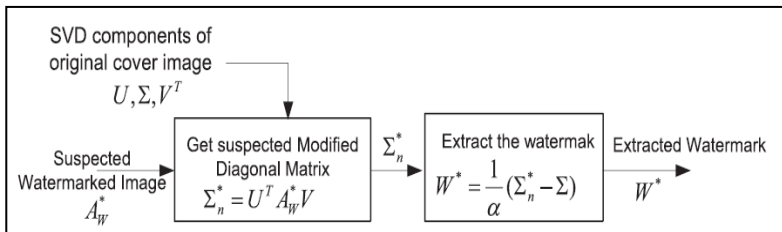
3. Mendapatkan citra ter-*watermark* A_W

$$A_W = U \Sigma_n V^T \quad (2.6)$$



Gambar 2.3 Proses Penyisipan

Sedangkan untuk proses ekstraksinya adalah seperti pada gambar 2.4 berikut ini :



Gambar 2.4 Proses Ekstraksi

Diberikan komponen-komponen SVD dari citra asli dan kemungkinan citra ter-*watermark* yang telah dimanipulasi A_W^* , sehingga :

1. Didapatkan Σ_n^* dari :

$$\Sigma_n^* = U^T A_W^* V \quad (2.7)$$

2. Didapatkan W^* dari :

$$W^* = \frac{1}{\alpha} (\Sigma_n^* - \Sigma) \quad (2.8)$$

Proses penyisipan dan ekstraksi di atas dapat dilihat dari contoh 2.2 berikut ini :

Contoh 2.2

Dari contoh 2.1, misalkan citra *watermark* disajikan dengan matriks $W = \begin{bmatrix} 0 & 1 \\ 1 & 1 \end{bmatrix}$ dan nilai $\alpha = 1$ maka langkah-langkah penyisipan adalah sebagai berikut :

Langkah 1 : melakukan SVD pada citra asli A seperti pada contoh 2.1, sehingga

$$A = U \Sigma V^T \\ = \begin{bmatrix} -\frac{2}{\sqrt{5}} & \frac{1}{\sqrt{5}} \\ \frac{1}{\sqrt{5}} & \frac{2}{\sqrt{5}} \end{bmatrix} \begin{bmatrix} 3 & 0 \\ 0 & 2 \end{bmatrix} \begin{bmatrix} -\frac{1}{\sqrt{5}} & \frac{2}{\sqrt{5}} \\ \frac{2}{\sqrt{5}} & \frac{1}{\sqrt{5}} \end{bmatrix}$$

Langkah 2 : menambahkan citra *watermark* W ke Σ , dengan faktor skala α

$$\Sigma_n = \Sigma + \alpha W = \begin{bmatrix} 3 & 0 \\ 0 & 2 \end{bmatrix} + 1 \begin{bmatrix} 0 & 1 \\ 1 & 1 \end{bmatrix} = \begin{bmatrix} 3 & 1 \\ 1 & 3 \end{bmatrix}$$

Langkah 3 : mendapatkan citra ter-*watermark* A_W

$$A_W = U \Sigma_n V^T \\ = \begin{bmatrix} -\frac{2}{\sqrt{5}} & \frac{1}{\sqrt{5}} \\ \frac{1}{\sqrt{5}} & \frac{2}{\sqrt{5}} \end{bmatrix} \begin{bmatrix} 3 & 1 \\ 1 & 3 \end{bmatrix} \begin{bmatrix} -\frac{1}{\sqrt{5}} & \frac{2}{\sqrt{5}} \\ \frac{2}{\sqrt{5}} & \frac{1}{\sqrt{5}} \end{bmatrix}$$

$$\begin{aligned}
 A_W &= \begin{bmatrix} -\sqrt{5} & \frac{1}{\sqrt{5}} \\ \sqrt{5} & \frac{7}{\sqrt{5}} \end{bmatrix} \begin{bmatrix} -\frac{1}{\sqrt{5}} & \frac{2}{\sqrt{5}} \\ \frac{2}{\sqrt{5}} & \frac{1}{\sqrt{5}} \end{bmatrix} \\
 &= \begin{bmatrix} \frac{7}{5} & -\frac{9}{5} \\ \frac{9}{5} & \frac{17}{5} \end{bmatrix}
 \end{aligned}$$

Setelah didapatkan citra ter-watermark A_W akan dilakukan proses

ekstraksi dengan mengaggap $A_W^* = A_W = \begin{bmatrix} \frac{7}{5} & -\frac{9}{5} \\ \frac{9}{5} & \frac{17}{5} \end{bmatrix}$

Berikut langkah-langkah proses ekstraksi :

Langkah 1 : mendapatkan Σ_n^* dari :

$$\begin{aligned}
 \Sigma_n^* &= U^T A_W^* V \\
 &= \begin{bmatrix} -\frac{2}{\sqrt{5}} & \frac{1}{\sqrt{5}} \\ \frac{1}{\sqrt{5}} & \frac{2}{\sqrt{5}} \end{bmatrix} \begin{bmatrix} \frac{7}{5} & -\frac{9}{5} \\ \frac{9}{5} & \frac{17}{5} \end{bmatrix} \begin{bmatrix} -\frac{1}{\sqrt{5}} & \frac{2}{\sqrt{5}} \\ \frac{2}{\sqrt{5}} & \frac{1}{\sqrt{5}} \end{bmatrix} \\
 &= \begin{bmatrix} \frac{1}{\sqrt{5}} & \frac{7}{\sqrt{5}} \\ -\frac{1}{\sqrt{5}} & \frac{2}{\sqrt{5}} \end{bmatrix} \begin{bmatrix} -\frac{1}{\sqrt{5}} & \frac{2}{\sqrt{5}} \\ \frac{2}{\sqrt{5}} & \frac{1}{\sqrt{5}} \end{bmatrix} \\
 &= \begin{bmatrix} 3 & 1 \\ 1 & 3 \end{bmatrix}
 \end{aligned}$$

Langkah 2 : mendapatkan W^* dari :

$$W^* = \frac{1}{\alpha} (\Sigma_n^* - \Sigma)$$

$$\begin{aligned} W^* &= \frac{1}{1} \left(\begin{bmatrix} 3 & 1 \\ 1 & 3 \end{bmatrix} - \begin{bmatrix} 3 & 0 \\ 0 & 2 \end{bmatrix} \right) \\ &= \begin{bmatrix} 0 & 1 \\ 1 & 1 \end{bmatrix} \end{aligned}$$

2.5 Matlab

MATLAB (*Matrix Laboratory*) adalah perangkat lunak yang berorientasi pada komputasi yang melibatkan penggunaan matriks dan vektor. MATLAB merupakan bahasa pemrograman dengan kemampuan tinggi dalam bidang komputasi. Matlab memiliki kemampuan mengintegrasikan komputasi, visualisasi, dan pemrograman. Tampilan layar untuk ruang kerja MATLAB meliputi, jendela utama yang terdiri dari *Current Directory*, *Command Window* (Jendela Perintah), *Workspace*, *Command History Window* dan menu utama. Selain itu MATLAB juga memiliki fasilitas GUI (*Graphical User Interface*) yang digunakan untuk membuat desain *interface* grafis dari program yang dibuat.

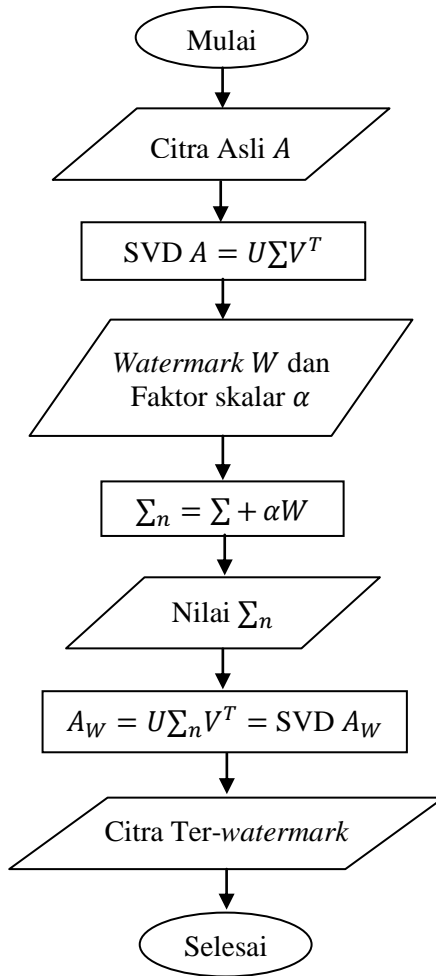
BAB III ANALISIS SISTEM

Pada bagian ini dibahas langkah-langkah sistem, mulai dari memasukan input data sampai menghasilkan output. Pembahasan tentang sistem ini meliputi langkah-langkah penyelesaian teknik penyisipan *watermark*, langkah-langkah teknik ekstraksi *watermark*, langkah-langkah pengujian citra ter-*watermark* dengan menggunakan pemberian distorsi dan langkah-langkah pengujian hasil menggunakan koefisien korelasi dan PSNR.

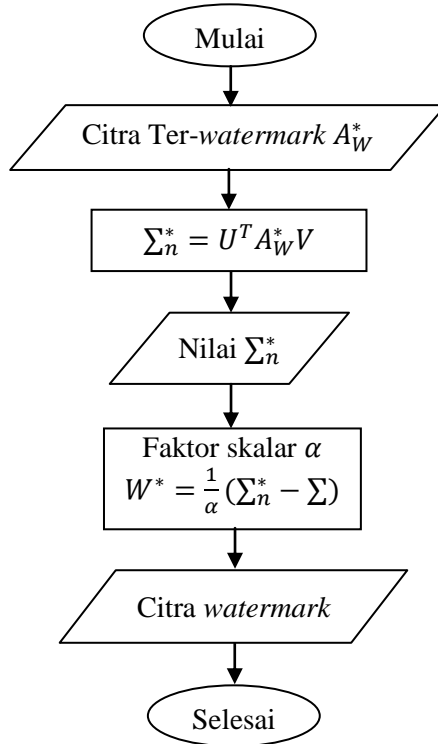
3.1 Langkah-Langkah Sistem

Sistem dalam Tugas Akhir ini terdiri dari dua proses utama yakni penyisipan *watermark* pada citra digital dengan menggunakan metode SVD dan proses ekstraksi *watermark*. Secara garis besar proses penyisipan *watermark* dalam program ini ditunjukkan oleh Gambar 3.1 dan proses ekstraksi *watermark* ditunjukkan oleh Gambar 3.2. Berikut ini adalah penjelasan langkah-langkah penyisipan *watermark* :

- Langkah 1 : Pada citra asli A dilakukan proses SVD, sehingga didapatkan nilai U , Σ dan V dari citra asli A .
- Langkah 2 : Setelah didapatkan nilai U , Σ dan V dari citra asli A , sistem akan mencari nilai singular dari citra ter-*watermark*. Untuk mendapatkan nilai singular tersebut, diinputkan citra *watermark* W dan nilai faktor skalar α yang bernilai positif.
- Langkah 3 : Setelah didapatkan nilai singular dari citra ter-*watermark*, dengan menggunakan cara SVD sistem akan mendapatkan citra ter-*watermark* A_W .



Gambar 3.1 Diagram Alir Proses Penyisipan *Watermark*



Gambar 3.2 Diagram Alir Proses Ekstraksi *Watermark*

Berikut ini adalah penjelasan langkah-langkah ekstraksi *watermark* :

Langkah 1 : Sistem akan memanggil kembali informasi dari citra asli A yang telah didekomposisi. Informasi ini digunakan untuk mendapatkan nilai singular *watermark*, $\Sigma_n^* = U^T A_W^* V$ dengan A_W^* adalah citra ter-*watermark* .

Langkah 2: Setelah didapatkan nilai Σ_n^* , nilai faktor skala α diinputkan berdasarkan proses penyisipan. Faktor

skala ini bernilai positif. Nilai ini digunakan untuk mendapatkan *watermark* W^* dari $W^* = \frac{1}{\alpha} (\Sigma_n^* - \Sigma)$

3.2 Analisis Koefisien Korelasi

Setelah proses ekstraksi yang menghasilkan estimasi *watermark*, dibutuhkan suatu cara untuk mengetahui kesamaan antara citra *watermark* dan estimasinya. Dalam hal ini dilakukan penghitungan koefisien korelasi. Koefisien korelasi digunakan untuk menghitung kesamaan antara dua citra. Berikut ini adalah bentuk umum dari koefisien korelasi[9] :

$$|R_{W,W^*}| = \frac{|S_{W,W^*}|}{\sqrt{S_{W,W}S_{W^*,W^*}}} \quad (3.1)$$

dengan :

$$S_{W,W^*} = \sum_{i=1}^M \sum_{j=1}^N (W_{(i,j)} - \bar{W})(W_{(i,j)}^* - \bar{W}^*) \quad (3.2)$$

$$S_{W,W} = \sum_{i=1}^M \sum_{j=1}^N (W_{(i,j)} - \bar{W})^2 \quad (3.3)$$

$$S_{W^*,W^*} = \sum_{i=1}^M \sum_{j=1}^N (W_{(i,j)}^* - \bar{W}^*)^2 \quad (3.4)$$

$$\bar{W} = \frac{1}{MN} \sum_{i=1}^M \sum_{j=1}^N W_{(i,j)} \quad (3.5)$$

$$\bar{W}^* = \frac{1}{MN} \sum_{i=1}^M \sum_{j=1}^N W_{(i,j)}^* \quad (3.6)$$

Contoh 3.1

Suatu citra *watermark* disajikan dengan matriks $W = \begin{bmatrix} 0 & 1 \\ 1 & 1 \end{bmatrix}$ dan estimasinya disajikan dengan matriks $W^* = \begin{bmatrix} 0 & 1 \\ 1 & 1 \end{bmatrix}$. Langkah-langkah dalam menghitung nilai koefisien korelasinya adalah sebagai berikut :

Langkah 1 : mencari nilai \bar{W} seperti pada persamaan (3.5)

$$\bar{W} = \frac{1}{MN} \sum_{i=1}^M \sum_{j=1}^N W_{(i,j)}$$

$$\bar{W} = \frac{1}{2.2} \sum_{i=1}^2 \sum_{j=1}^2 W_{(i,j)}$$

$$\bar{W} = \frac{1}{4} (0 + 1 + 1 + 1)$$

$$\bar{W} = \frac{3}{4}$$

Langkah 2 : mencari nilai \bar{W}^* seperti pada persamaan (3.6)

$$\bar{W}^* = \frac{1}{MN} \sum_{i=1}^M \sum_{j=1}^N W_{(i,j)}^*$$

$$\bar{W}^* = \frac{1}{2.2} \sum_{i=1}^2 \sum_{j=1}^2 W_{(i,j)}^*$$

$$\bar{W}^* = \frac{1}{4} (0 + 1 + 1 + 1)$$

$$\bar{W}^* = \frac{3}{4}$$

Langkah 3 : mencari nilai S_{W,W^*} seperti pada persamaan (3.2)

$$S_{W,W^*} = \sum_{i=1}^M \sum_{j=1}^N (W_{(i,j)} - \bar{W})(W_{(i,j)}^* - \bar{W}^*)$$

$$S_{W,W^*} = \sum_{i=1}^2 \sum_{j=1}^2 (W_{(i,j)} - \bar{W})(W_{(i,j)}^* - \bar{W}^*)$$

$$S_{W,W^*} = \left(0 - \frac{3}{4}\right) \left(0 - \frac{3}{4}\right) + \left(1 - \frac{3}{4}\right) \left(1 - \frac{3}{4}\right) + \left(1 - \frac{3}{4}\right) \left(1 - \frac{3}{4}\right) + \left(1 - \frac{3}{4}\right) \left(1 - \frac{3}{4}\right)$$

$$S_{W,W^*} = \left(-\frac{3}{4}\right) \left(-\frac{3}{4}\right) + \left(\frac{1}{4}\right) \left(\frac{1}{4}\right) + \left(\frac{1}{4}\right) \left(\frac{1}{4}\right) + \left(\frac{1}{4}\right) \left(\frac{1}{4}\right)$$

$$S_{W,W^*} = \frac{9}{16} + \frac{1}{16} + \frac{1}{16} + \frac{1}{16}$$

$$S_{W,W^*} = \frac{12}{16}$$

$$S_{W,W^*} = \frac{3}{4}$$

Langkah 4 : mencari nilai $S_{W,W}$ seperti pada persamaan (3.3)

$$S_{W,W} = \sum_{i=1}^M \sum_{j=1}^N (W_{(i,j)} - \bar{W})^2$$

$$S_{W,W} = \sum_{i=1}^2 \sum_{j=1}^2 (W_{(i,j)} - \bar{W})^2$$

$$S_{W,W} = \left(0 - \frac{3}{4}\right)^2 + \left(1 - \frac{3}{4}\right)^2 + \left(1 - \frac{3}{4}\right)^2 + \left(1 - \frac{3}{4}\right)^2$$

$$S_{W,W} = \left(-\frac{3}{4}\right)^2 + \left(\frac{1}{4}\right)^2 + \left(\frac{1}{4}\right)^2 + \left(\frac{1}{4}\right)^2$$

$$S_{W,W} = \frac{9}{16} + \frac{1}{16} + \frac{1}{16} + \frac{1}{16}$$

$$S_{W,W} = \frac{12}{16}$$

$$S_{W,W} = \frac{3}{4}$$

Langkah 5 : mencari nilai S_{W^*,W^*} seperti pada persamaan (3.4)

$$S_{W^*,W^*} = \sum_{i=1}^M \sum_{j=1}^N (W_{(i,j)}^* - \bar{W}^*)^2$$

$$S_{W^*,W^*} = \sum_{i=1}^2 \sum_{j=1}^2 (W_{(i,j)}^* - \bar{W}^*)^2$$

$$S_{W^*,W^*} = \left(0 - \frac{3}{4}\right)^2 + \left(1 - \frac{3}{4}\right)^2 + \left(1 - \frac{3}{4}\right)^2 + \left(1 - \frac{3}{4}\right)^2$$

$$S_{W^*,W^*} = \left(-\frac{3}{4}\right)^2 + \left(\frac{1}{4}\right)^2 + \left(\frac{1}{4}\right)^2 + \left(\frac{1}{4}\right)^2$$

$$S_{W^*,W^*} = \frac{9}{16} + \frac{1}{16} + \frac{1}{16} + \frac{1}{16}$$

$$S_{W^*,W^*} = \frac{12}{16}$$

$$S_{W^*,W^*} = \frac{3}{4}$$

Langkah 6 : mencari nilai koefisien korelasi seperti pada persamaan (3.1)

$$|R_{W,W^*}| = \frac{|S_{W,W^*}|}{\sqrt{S_{W,W}S_{W^*,W^*}}}$$

$$|R_{W,W^*}| = \frac{\frac{3}{4}}{\sqrt{\frac{3}{4} \cdot \frac{3}{4}}}$$

$$|R_{W,W^*}| = \frac{\frac{3}{4}}{\sqrt{\frac{9}{16}}}$$

$$|R_{W,W^*}| = \frac{\frac{3}{4}}{\frac{3}{4}}$$

$$|R_{W,W^*}| = 1$$

Contoh 3.2

Suatu citra *watermark* disajikan dengan matriks $W = \begin{bmatrix} 0 & 1 \\ 1 & 1 \end{bmatrix}$ dan estimasinya disajikan dengan matriks $W^* = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}$. Langkah-langkah dalam menghitung nilai koefisien korelasinya adalah sebagai berikut :

Langkah 1 : mencari nilai \bar{W} seperti pada persamaan (3.5)

$$\bar{W} = \frac{1}{MN} \sum_{i=1}^M \sum_{j=1}^N W_{(i,j)}$$

$$\bar{W} = \frac{1}{2 \cdot 2} \sum_{i=1}^2 \sum_{j=1}^2 W_{(i,j)}$$

$$\bar{W} = \frac{1}{4} (0 + 1 + 1 + 1)$$

$$\bar{W} = \frac{3}{4}$$

Langkah 2 : mencari nilai \bar{W}^* seperti pada persamaan (3.6)

$$\bar{W}^* = \frac{1}{MN} \sum_{i=1}^M \sum_{j=1}^N W_{(i,j)}^*$$

$$\bar{W}^* = \frac{1}{2.2} \sum_{i=1}^2 \sum_{j=1}^2 W_{(i,j)}^*$$

$$\bar{W}^* = \frac{1}{4} (0 + 1 + 1 + 0)$$

$$\bar{W}^* = \frac{1}{2}$$

Langkah 3 : mencari nilai S_{W,W^*} seperti pada persamaan (3.2)

$$S_{W,W^*} = \sum_{i=1}^M \sum_{j=1}^N (W_{(i,j)} - \bar{W})(W_{(i,j)}^* - \bar{W}^*)$$

$$S_{W,W^*} = \sum_{i=1}^2 \sum_{j=1}^2 (W_{(i,j)} - \bar{W})(W_{(i,j)}^* - \bar{W}^*)$$

$$S_{W,W^*} = \left(0 - \frac{3}{4}\right) \left(0 - \frac{1}{2}\right) + \left(1 - \frac{3}{4}\right) \left(1 - \frac{1}{2}\right) + \left(1 - \frac{3}{4}\right) \left(1 - \frac{1}{2}\right) + \left(1 - \frac{3}{4}\right) \left(0 - \frac{1}{2}\right)$$

$$S_{W,W^*} = \left(-\frac{3}{4}\right) \left(-\frac{1}{2}\right) + \left(\frac{1}{4}\right) \left(\frac{1}{2}\right) + \left(\frac{1}{4}\right) \left(\frac{1}{2}\right) + \left(\frac{1}{4}\right) \left(-\frac{1}{2}\right)$$

$$S_{W,W^*} = \frac{3}{8} + \frac{1}{8} + \frac{1}{8} - \frac{1}{8}$$

$$S_{W,W^*} = \frac{4}{8}$$

$$S_{W,W^*} = \frac{1}{2}$$

Langkah 4 : mencari nilai $S_{W,W}$ seperti pada persamaan (3.3)

$$S_{W,W} = \sum_{i=1}^M \sum_{j=1}^N (W_{(i,j)} - \bar{W})^2$$

$$S_{W,W} = \sum_{i=1}^2 \sum_{j=1}^2 (W_{(i,j)} - \bar{W})^2$$

$$S_{W,W} = \left(0 - \frac{3}{4}\right)^2 + \left(1 - \frac{3}{4}\right)^2 + \left(1 - \frac{3}{4}\right)^2 + \left(1 - \frac{3}{4}\right)^2$$

$$S_{W,W} = \left(-\frac{3}{4}\right)^2 + \left(\frac{1}{4}\right)^2 + \left(\frac{1}{4}\right)^2 + \left(\frac{1}{4}\right)^2$$

$$S_{W,W} = \frac{9}{16} + \frac{1}{16} + \frac{1}{16} + \frac{1}{16}$$

$$S_{W,W} = \frac{12}{16}$$

$$S_{W,W} = \frac{3}{4}$$

Langkah 5 : mencari nilai S_{W^*,W^*} seperti pada persamaan (3.4)

$$S_{W^*,W^*} = \sum_{i=1}^M \sum_{j=1}^N (W_{(i,j)}^* - \bar{W}^*)^2$$

$$S_{W^*,W^*} = \sum_{i=1}^2 \sum_{j=1}^2 (W_{(i,j)}^* - \bar{W}^*)^2$$

$$S_{W^*,W^*} = \left(0 - \frac{1}{2}\right)^2 + \left(1 - \frac{1}{2}\right)^2 + \left(1 - \frac{1}{2}\right)^2 + \left(0 - \frac{1}{2}\right)^2$$

$$S_{W^*,W^*} = \left(-\frac{1}{2}\right)^2 + \left(\frac{1}{2}\right)^2 + \left(\frac{1}{2}\right)^2 + \left(-\frac{1}{2}\right)^2$$

$$S_{W^*,W^*} = \frac{1}{4} + \frac{1}{4} + \frac{1}{4} + \frac{1}{4}$$

$$S_{W^*,W^*} = \frac{4}{4}$$

$$S_{W^*,W^*} = 1$$

Langkah 6 : mencari nilai koefisien korelasi seperti pada persamaan (3.1)

$$|R_{W,W^*}| = \frac{|S_{W,W^*}|}{\sqrt{S_{W,W} S_{W^*,W^*}}}$$

$$|R_{W,W^*}| = \frac{1}{\sqrt{\frac{3}{4} \cdot 1}}$$

$$|R_{W,W^*}| = \frac{\frac{1}{2}}{\sqrt{\frac{3}{4}}}$$

$$|R_{W,W^*}| = \frac{1}{\sqrt{3}}$$

$$|R_{W,W^*}| = 0,577$$

Contoh 3.3

Suatu citra *watermark* disajikan dengan matriks $W = \begin{bmatrix} 0 & 1 \\ 1 & 1 \end{bmatrix}$ dan estimasinya disajikan dengan matriks $W^* = \begin{bmatrix} 2 & 2 \\ 2 & 3 \end{bmatrix}$. Langkah-langkah dalam menghitung nilai koefisien korelasinya adalah sebagai berikut :

Langkah 1 : mencari nilai \bar{W} seperti pada persamaan (3.5)

$$\bar{W} = \frac{1}{MN} \sum_{i=1}^M \sum_{j=1}^N W_{(i,j)}$$

$$\bar{W} = \frac{1}{2.2} \sum_{i=1}^2 \sum_{j=1}^2 W_{(i,j)}$$

$$\bar{W} = \frac{1}{4} (0 + 1 + 1 + 1)$$

$$\bar{W} = \frac{3}{4}$$

Langkah 2 : mencari nilai \bar{W}^* seperti pada persamaan (3.6)

$$\bar{W}^* = \frac{1}{MN} \sum_{i=1}^M \sum_{j=1}^N W_{(i,j)}^*$$

$$\bar{W}^* = \frac{1}{2.2} \sum_{i=1}^2 \sum_{j=1}^2 W_{(i,j)}^*$$

$$\bar{W}^* = \frac{1}{4} (2 + 2 + 2 + 3)$$

$$\bar{W}^* = \frac{9}{4}$$

Langkah 3 : mencari nilai S_{W,W^*} seperti pada persamaan (3.2)

$$S_{W,W^*} = \sum_{i=1}^M \sum_{j=1}^N (W_{(i,j)} - \bar{W})(W_{(i,j)}^* - \bar{W}^*)$$

$$S_{W,W^*} = \sum_{i=1}^2 \sum_{j=1}^2 (W_{(i,j)} - \bar{W})(W_{(i,j)}^* - \bar{W}^*)$$

$$S_{W,W^*} = \left(0 - \frac{3}{4}\right) \left(2 - \frac{9}{4}\right) + \left(1 - \frac{3}{4}\right) \left(2 - \frac{9}{4}\right) + \left(1 - \frac{3}{4}\right) \left(2 - \frac{9}{4}\right) + \left(1 - \frac{3}{4}\right) \left(3 - \frac{9}{4}\right)$$

$$S_{W,W^*} = \left(-\frac{3}{4}\right) \left(-\frac{1}{4}\right) + \left(\frac{1}{4}\right) \left(-\frac{1}{4}\right) + \left(\frac{1}{4}\right) \left(-\frac{1}{4}\right) + \left(\frac{1}{4}\right) \left(\frac{3}{4}\right)$$

$$S_{W,W^*} = \frac{3}{16} - \frac{1}{16} - \frac{1}{16} + \frac{3}{16}$$

$$S_{W,W^*} = \frac{4}{16}$$

$$S_{W,W^*} = \frac{1}{4}$$

Langkah 4 : mencari nilai $S_{W,W}$ seperti pada persamaan (3.3)

$$S_{W,W} = \sum_{i=1}^M \sum_{j=1}^N (W_{(i,j)} - \bar{W})^2$$

$$S_{W,W} = \sum_{i=1}^2 \sum_{j=1}^2 (W_{(i,j)} - \bar{W})^2$$

$$S_{W,W} = \left(0 - \frac{3}{4}\right)^2 + \left(1 - \frac{3}{4}\right)^2 + \left(1 - \frac{3}{4}\right)^2 + \left(1 - \frac{3}{4}\right)^2$$

$$S_{W,W} = \left(-\frac{3}{4}\right)^2 + \left(\frac{1}{4}\right)^2 + \left(\frac{1}{4}\right)^2 + \left(\frac{1}{4}\right)^2$$

$$S_{W,W} = \frac{9}{16} + \frac{1}{16} + \frac{1}{16} + \frac{1}{16}$$

$$S_{W,W} = \frac{12}{16}$$

$$S_{W,W} = \frac{3}{4}$$

Langkah 5 : mencari nilai S_{W^*,W^*} seperti pada persamaan (3.4)

$$S_{W^*,W^*} = \sum_{i=1}^M \sum_{j=1}^N (W_{(i,j)}^* - \bar{W}^*)^2$$

$$S_{W^*,W^*} = \sum_{i=1}^2 \sum_{j=1}^2 (W_{(i,j)}^* - \bar{W}^*)^2$$

$$S_{W^*,W^*} = \left(2 - \frac{9}{4}\right)^2 + \left(2 - \frac{9}{4}\right)^2 + \left(2 - \frac{9}{4}\right)^2 + \left(3 - \frac{9}{4}\right)^2$$

$$S_{W^*,W^*} = \left(-\frac{1}{4}\right)^2 + \left(-\frac{1}{4}\right)^2 + \left(-\frac{1}{4}\right)^2 + \left(\frac{3}{4}\right)^2$$

$$S_{W^*,W^*} = \frac{1}{16} + \frac{1}{16} + \frac{1}{16} + \frac{9}{16}$$

$$S_{W^*,W^*} = \frac{12}{16}$$

$$S_{W^*,W^*} = \frac{3}{4}$$

Langkah 6 : mencari nilai koefisien korelasi seperti pada persamaan (3.1)

$$|R_{W,W^*}| = \frac{|S_{W,W^*}|}{\sqrt{S_{W,W} S_{W^*,W^*}}}$$

$$|R_{W,W^*}| = \frac{\frac{1}{4}}{\sqrt{\frac{3}{4} \cdot \frac{3}{4}}}$$

$$|R_{W,W^*}| = \frac{\frac{1}{4}}{\sqrt{\frac{9}{16}}}$$

$$|R_{W,W^*}| = \frac{\frac{1}{4}}{\frac{3}{4}}$$

$$|R_{W,W^*}| = \frac{1}{3}$$

$$|R_{W,W^*}| = 0,333$$

Dapat dilihat pada contoh-contoh di atas, misalkan terdapat dua citra W dan W^* dengan ukuran $M \times N$, dua citra tersebut dikatakan berbeda ketika nilai koefisien korelasinya mendekati nol seperti pada contoh 3.3, sedangkan dikatakan identik satu sama lain ketika nilai koefisien korelasinya mendekati satu seperti pada contoh 3.1.

3.3 Analisis PSNR

Selain dihitung koefisien korelasi antara citra *watermark* dan estimasinya, perbandingan antara citra ter-*watermark* dan citra asli juga diperlukan pada proses legalisasi ini. Untuk itu diperlukan PSNR (*Peak Signal to Noise Ratio*). PSNR adalah ukuran kesalahan yang digunakan untuk membandingkan kualitas antara dua citra. Untuk menghitung PSNR, dihitung terlebih dahulu *Mean Square Error* (MSE). MSE merupakan kuadrat kesalahan kumulatif antara dua citra. MSE dihitung dari sebuah contoh obyek yang kemudian dibandingkan dengan obyek aslinya sehingga dapat diketahui tingkat ketidaksesuaian antara obyek contoh dengan obyek aslinya. Semakin rendah nilai MSE maka semakin rendah kesalahan, begitu pula sebaliknya. Berikut adalah bentuk umum dari PSNR[9] :

$$PSNR(A_W, A_W^*) = 10 \log_{10} \frac{\text{Max}(A_W^2)}{MSE} \quad (3.7)$$

$$\text{dengan : } MSE = \frac{1}{MN} \sum_{i=1}^M \sum_{j=1}^N (A_{W(i,j)} - A_{W^*(i,j)})^2 \quad (3.8)$$

$A_{W(i,j)}$ adalah nilai piksel dari citra asli pada baris ke- i dan kolom ke- j , sedangkan $A_{W^*(i,j)}$ adalah untuk citra ter-*watermark*. M dan N adalah dimensi citra.

3.4 Penambahan Distorsi

Berbagai macam distorsi ditambahkan pada citra ter-*watermark*. Penambahan distorsi disini bertujuan untuk menguji kekuatan dari proses *watermarking* dengan metode SVD. Dalam hal ini distorsi yang digunakan berupa penambahan *noise* seperti *noise Gaussian*, *speckle*, dan *salt & pepper*.

Noise Gaussian merupakan model *noise* yang mengikuti distribusi normal standard dengan rata-rata nol dan standard deviasi satu. Efek dari *noise* ini adalah munculnya titik-titik berwarna yang jumlahnya sama dengan prosentase *noise*. Untuk piksel yang terkena *noise*, nilai fungsi citra ditambahkan dengan *noise* yang ada.

Noise speckle merupakan model *noise* yang memberikan warna hitam pada titik yang terkena *noise*. *Noise* ini dapat dibangkitkan dengan cara membangkitkan bilangan nol atau wana hitam pada titik-titik yang secara probabilitas lebih kecil dari nilai probabilitas *noise*.

Noise salt & pepper memberikan *noise* seperti halnya taburan garam, memberikan warna putih pada titik yang terkena *noise*. *Noise* ini dapat dibangkitkan dengan cara membangkitkan bilangan 255 atau warna putih pada titik-titik yang secara probabilitas lebih kecil dari nilai probabilitas *noise*.



(a)

(b)

(c)

Gambar 3.3 Citra yang telah diberi *Noise* (a) *Gaussian*, (b) *Speckle*, (c) *Salt & Pepper*

BAB IV

PERANCANGAN DAN IMPLEMENTASI SISTEM

Pada bab ini dibahas bagaimana sistem program akan berinteraksi dengan pengguna mulai dari data masukan sampai menghasilkan keluaran. Pembahasan tentang sistem program ini meliputi langkah-langkah penyelesaian teknik penyisipan *watermark*, langkah-langkah teknik ekstraksi *watermark*, langkah-langkah pengujian citra ter-*watermark* dengan menggunakan pemberian distorsi dan langkah-langkah pengujian hasil menggunakan koefisien korelasi dan PSNR.

4.1 Langkah-Langkah Program

Program dalam Tugas Akhir ini terdiri program utama yakni program *watermarking* pada citra digital dengan menggunakan metode SVD, yang berisi proses penyisipan *watermark*. Melalui program ini dapat dipanggil program lain yakni proses ekstraksi sebagai salah satu proses dalam *watermarking*. Sebelum proses ekstraksi terdapat proses penambahan *noise* pada citra ter-*watermark*. Selain program ekstraksi *watermark* juga terdapat program penghitungan koefisien korelasi dan PSNR. Adapun penjelasan mengenai sistem pada masing-masing program tersebut dijelaskan pada masing-masing subbab berikut ini.

4.1.1 Langkah-Langkah Program Watermarking dalam Proses Penyisipan Watermark pada Citra Digital

Program *watermarking* pada citra digital dengan menggunakan metode SVD merupakan program utama dalam program ini. Fungsi utamanya adalah membuat citra ter-*watermark* yang digunakan dalam proses ekstraksi. Proses pelaksanaan sistem dalam program ini adalah sebagai berikut :

- a. Ketika program dimulai, pengguna mendapat pilihan untuk memasukkan citra asli langsung dari file citra yang sudah ada. Citra asli yang digunakan adalah citra *grayscale* berukuran

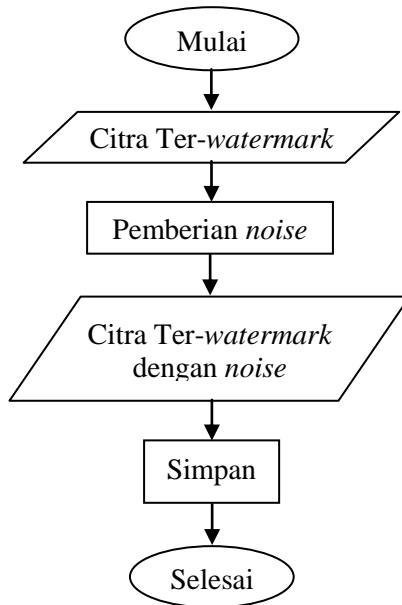
$N \times N$. Selain itu pengguna dapat memilih *watermark* yang berupa citra *grayscale* dengan ukuran yang lebih kecil dari citra asli.

- b. Setelah citra asli dan *watermark* diinputkan, pengguna juga harus menginputkan nilai faktor skalar α yang bernilai positif.
- c. Sistem akan melakukan pembuatan citra ter-*watermark* dari semua inputan yang telah diinputkan dengan menggunakan metode SVD. Setelah proses tersebut, program akan menampilkan citra ter-*watermark* yang nantinya akan digunakan pada proses ekstraksi.

4.1.2 Langkah-Langkah Program Watermarking dalam Proses Ekstraksi Watermark pada Citra Digital

Program ini bertujuan untuk mengekstraksi estimasi dari citra *watermark*. Input dalam program ini berupa citra asli dan citra ter-*watermark*. Sebelum dilakukan proses ekstraksi, citra ter-*watermark* dapat ditambahkan *noise*. Program untuk penambahan *noise* ini terdapat pada program ekstraksi *watermark*. Macam-macam *noise* dalam program ini adalah *noise Gaussian*, *speckle*, dan *salt & pepper*. Berikut adalah penjelasan proses penambahan *noise* yang ada pada Gambar 4.1:

- a. Citra ter-*watermark* yang diinputkan digunakan ketika pemberian *noise*. Pengguna dapat memilih jenis *noise* yang akan ditambahkan pada citra ter-*watermark*.
- b. Pengguna dapat memasukkan nilai parameter yang ada dalam setiap jenis *noise*.
- c. Setelah itu sistem akan menampilkan hasil citra pemberian *noise*, untuk selanjutnya pengguna dapat menyimpan hasil citra ter-*watermark* tersebut.



Gambar 4.1 Diagram Alir Program Penambahan *Noise*

Citra *ter-watermark* yang telah ditambahkan *noise* ini akan digunakan dalam proses ekstraksi. Berikut ini adalah penjelasan proses dari program ekstraksi dengan metode SVD :

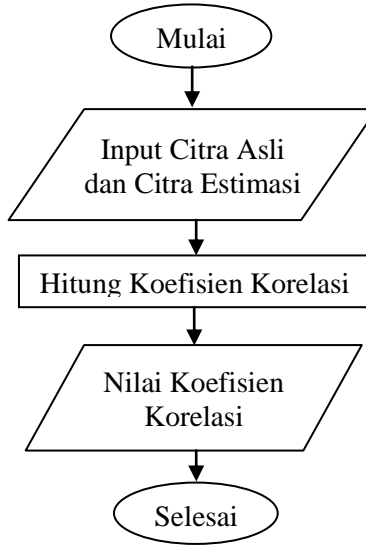
- a. Ketika program dimulai, pengguna menggunakan citra *ter-watermark* yang telah ditambahkan dengan *noise*.
- b. Sistem akan melakukan proses ekstraksi untuk mendapatkan estimasi dari citra *watermark* yang telah disisipkan dalam citra inputan dan menampilkannya.

4.1.3 Langkah-Langkah Pengujian Hasil Nilai Koefisien Korelasi

Tujuan utama program ini adalah untuk mengestimasi citra output dari proses metode SVD apakah sesuai dengan citra input. Masukan dalam program ini berupa dua citra yang akan

dibandingkan satu sama lain dengan menghitung koefisien korelasi.

Proses pembuatan sistem pada program ini ditunjukkan oleh Gambar 4.2 berikut ini :



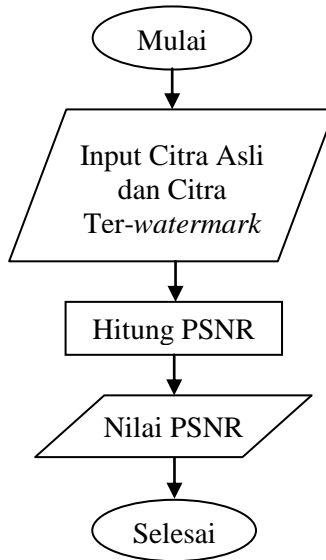
Gambar 4.2 Diagram Alir Program Koefisien Korelasi

Berikut ini adalah penjelasan dari program perhitungan nilai koefisien korelasi :

- Pengguna memberi input citra pertama yang merupakan citra asli yang digunakan sebelum proses penyisipan *watermark*.
- Pengguna memberi input citra kedua dari estimasi *watermark*.
- Selanjutnya nilai koefisien korelasi akan dihitung oleh sistem dan akan ditampilkan pada kotak dialog yang tersedia.

4.1.4 Langkah-Langkah Pengujian Hasil Menggunakan Peak Signal to Noise Ratio (PSNR)

Proses pengujian citra hasil *watermarking* menggunakan metode SVD dengan perhitungan MSE dan PSNR ditunjukkan pada Gambar 4.3 berikut ini :



Gambar 4.3 Diagram Alir Program Perhitungan PSNR

Berikut ini adalah penjelasan dari program perhitungan nilai PSNR :

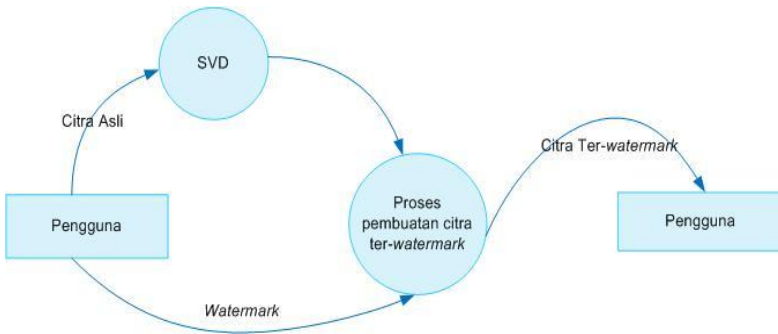
- Pengguna memberi input citra pertama yang merupakan citra asli yang digunakan sebelum proses penyisipan *watermark*.
- Pengguna memberi input citra kedua dari citra yang telah disisipi *watermark*.
- Selanjutnya nilai MSE dan PSNR akan dihitung oleh sistem dan akan ditampilkan pada kotak dialog yang tersedia.

4.2 Perancangan Program

4.2.1 Data Flow Diagram (DFD)

a. Proses Watermarking pada Citra Digital Menggunakan Metode SVD

Proses penyisipan *watermark* menggunakan metode SVD ini bertujuan untuk membentuk citra ter-*watermark* dari citra asli dan *watermark*. Gambaran proses penyisipan *watermark* dapat dilihat pada Gambar 4.4 berikut ini :



Gambar 4.4 Data Flow Diagram Proses Penyisipan *Watermark*

Penjelasan setiap proses pada Gambar 4.4 adalah sebagai berikut :

- i. Proses SVD
 - a. Uraian
Transformasi ini dilakukan pada nilai-nilai singular suatu citra.
 - b. Skenario Input
Proses menerima input berupa citra asli.
 - c. Syarat (Kondisi Awal)
Citra asli tersedia.
 - d. Hasil
Nilai-nilai singular dari citra asli.
- ii. Proses pembuatan citra ter-*watermark*
 - a. Uraian

Proses ini menyisipkan citra *watermark* ke dalam citra asli.

b. Skenario Input

Proses menerima input berupa citra asli, citra *watermark*, dan nilai α .

c. Syarat (Kondisi Awal)

Citra asli dan citra *watermark* dan nilai α tersedia.

d. Hasil

Citra Ter-*watermark*.

b. Proses Penambahan Noise

Pada proses ini dilakukan penambahan noise. Gambaran prosesnya dapat dilihat pada Gambar 4.5 berikut ini :



Gambar 4.5 Data Flow Diagram Proses Penambahan Noise

Penjelasan setiap proses pada Gambar 4.5 adalah sebagai berikut :

- i. Uraian

Dalam proses ini dilakukan penambahan noise.
- ii. Skenario Input

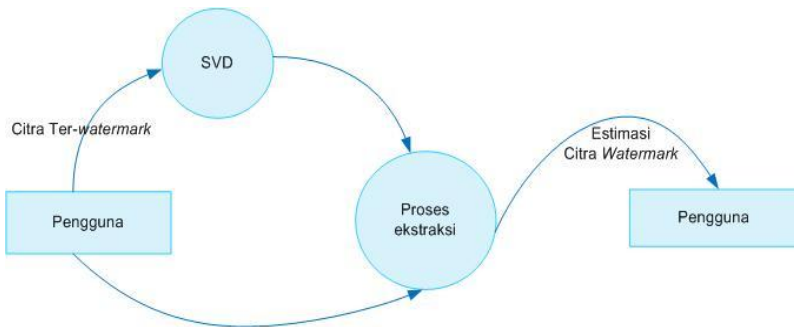
Proses menerima input berupa citra ter-*watermark*.
- iii. Syarat (Kondisi Awal)

Citra ter-*watermark* tersedia.
- iv. Hasil

Citra ter-*watermark* yang telah mengalami penambahan noise

c. Proses Ekstraksi Watermark

Pada proses ini akan dilakukan ekstraksi *watermark* dengan menggunakan metode SVD. Berikut adalah data flow diagram yang ditunjukkan pada Gambar 4.6 :



Gambar 4.6 Data Flow Diagram Proses Ekstraksi *Watermark*

Penjelasan setiap proses pada Gambar 4.6 adalah sebagai berikut :

- i. Proses SVD
 - a. Uraian
Transformasi ini dilakukan pada nilai-nilai singular suatu citra.
 - b. Skenario Input
Proses menerima input berupa citra ter-*watermark*.
 - c. Syarat (Kondisi Awal)
Citra ter-*watermark* tersedia.
 - d. Hasil
Nilai-nilai singular dari citra ter-*watermark*.
- ii. Proses ekstraksi *watermark*
 - a. Uraian
Dalam proses ini estimasi citra *watermark* dibentuk dari citra ter-*watermark*.
 - b. Skenario Input
Citra asli dan citra ter-*watermark* serta nilai α .

- c. Syarat (Kondisi Awal)
Citra asli dan citra ter-*watermark* serta nilai α tersedia.
- d. Hasil
Estimasi citra *watermark*.

d. Proses Menghitung Nilai Koefisien Korelasi

Penghitungan nilai koefisien korelasi antara citra *watermark* dan estimasinya diperlukan untuk mengetahui kualitas ekstraksi *watermark* menggunakan metode SVD. Gambar 4.7 berikut menunjukkan data flow diagram untuk penghitungan nilai koefisien korelasi :



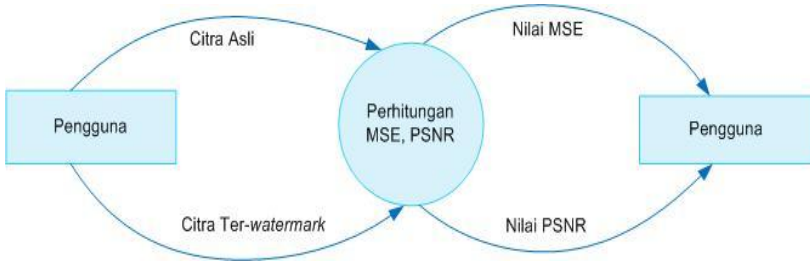
Gambar 4.7 Data Flow Diagram Koefisien Korelasi

Penjelasan setiap proses pada Gambar 4.7 adalah sebagai berikut :

- i. Uraian
Dalam proses ini dilakukan proses menghitung nilai koefisien korelasi.
- ii. Skenario Input
Proses menerima input berupa citra *watermark* dan estimasi citra *watermark*.
- iii. Syarat (Kondisi Awal)
Citra *watermark* dan estimasi citra *watermark* tersedia.
- iv. Hasil
Nilai koefisien korelasi.

e. Proses Menghitung Nilai MSE dan PSNR

Pada proses ini dilakukan penghitungan nilai MSE dan PSNR. Berikut adalah data flow diagram untuk penghitungan nilai MSE dan PSNR :



Gambar 4.8 Data Flow Diagram MSE dan PSNR

Penjelasan setiap proses pada Gambar 4.8 adalah sebagai berikut :

- i. Uraian
Dalam proses ini dilakukan proses menghitung nilai MSE dan PSNR seperti pada persamaan (3.7) dan (3.8).
- ii. Skenario Input
Proses menerima input berupa citra asli dan citra *ter-watermark*.
- iii. Syarat (Kondisi Awal)
Citra asli dan citra *ter-watermark* tersedia.
- iv. Hasil
Nilai MSE dan PSNR.

4.2.2 Fungsi-Fungsi yang Dibutuhkan Program Watermarking Menggunakan Metode SVD

Fungsi-fungsi yang dibutuhkan dalam pembuatan program perlu diketahui terlebih dahulu sebelum ke tahap pengimplementasian program. Hal ini dimaksudkan untuk memudahkan dalam perancangan dan analisis algoritma yang akan diimplementasikan ke dalam bahasa pemrograman matlab.

Adapun fungsi-fungsi yang dibutuhkan dalam pembuatan perangkat lunak ini antara lain adalah :

a. Pengambilan Citra

Proses ini digunakan untuk mengambil citra yang telah ada pada file komputer. Fungsi mengambil citra tersebut menggunakan toolbox dari matlab seperti berikut :

Nama fungsi : `[namafile1] = uigetfile`
`({'*.jpg'; '*.bmp'; '*.*'}, 'Buka Gambar');`

Tugas : Mengambil citra yang akan diproses.

b. Penyimpanan Citra

Proses ini digunakan untuk menyimpan citra yang dihasilkan oleh metode SVD. Fungsi menyimpan citra tersebut menggunakan toolbox dari matlab seperti berikut :

Nama fungsi : `[filename, user_cancel] =`
`imsave();`

Tugas : Menyimpan citra yang telah diproses.

c. Penambahan *Noise*

Gangguan yang dilakukan pada citra ter-*watermark* dalam Tugas Akhir ini adalah penambahan *noise* berupa *Gaussian*, *speckle*, dan *salt & pepper*.

1. Nama fungsi : `imnoise(I, 'gaussian', m, v);`

Tugas : Menambahkan *noise Gaussian* pada citra dengan menginputkan nilai mean dan varians.

2. Nama fungsi : `imnoise(I, 'speckle', v);`

Tugas : Menambahkan *noise speckle* pada citra dengan menginputkan nilai varians.

3. Nama fungsi : `imnoise(I, 'salt & pepper', d);`

Tugas : Menambahkan *noise salt & pepper* pada citra dengan menginputkan nilai kerapatan *noise*.

d. Koefisien Korelasi

Nama fungsi : R_{xy}

Tugas : Menghitung nilai koefisien korelasi.

e. PSNR (*Peak signal-to-noise ratio*)

Hasil pengujian secara numerik bisa didapatkan melalui penghitungan PSNR.

Nama fungsi : PSNR

Tugas : Menghitung nilai PSNR.

4.3 Ruang Lingkup Implementasi

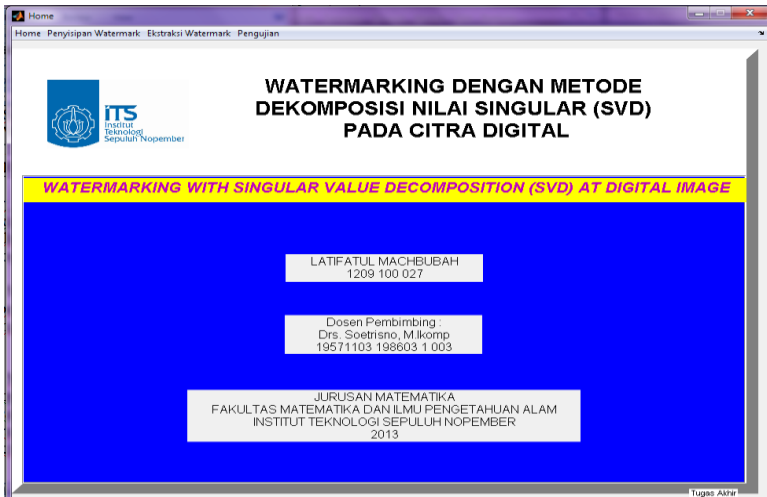
Ruang lingkup implementasi program yang dibangun dalam Tugas Akhir ini meliputi perangkat keras dan perangkat lunak yang digunakan. Spesifikasinya disajikan dalam Tabel 4.1 berikut ini :

Tabel 4.1 Ruang Lingkup Implementasi Program

Perangkat keras	Processor : Pentium(R) Dual-Core CPU T4300 @ 2.10GHz Memory : RAM 1.00 GB
Perangkat lunak	Sistem Operasi : Microsoft Windows 7 Ultimate 32-bit Tools Pembangun : Matlab R2010a

4.4 Desain Tampilan

Program dalam Tugas Akhir ini terdiri dari satu program utama yakni program *watermarking* pada citra digital dengan menggunakan metode SVD. Melalui program ini, program lain dapat dipanggil. Program tersebut adalah program ekstraksi *watermark* sebagai salah satu proses dalam *watermarking* dengan metode SVD yang di dalamnya terdapat proses pemberian gangguan pada citra ter-*watermark*, program penghitungan koefisien korelasi, dan program penghitungan PSNR. Gambar 4.9 menunjukkan halaman awal dari program ini.



Gambar 4.9 Desain Antarmuka Menu Utama

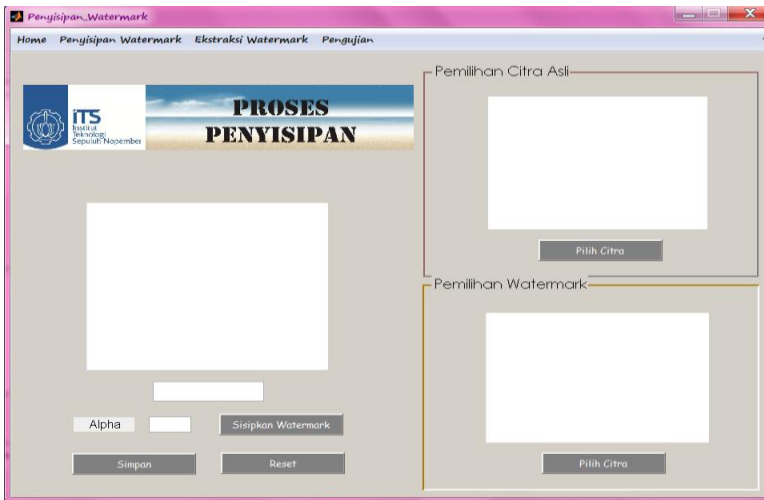
4.4.1 Program Watermarking Menggunakan Metode SVD

Program ini merupakan program utama yang berfungsi untuk menciptakan citra ter-*watermark*. Proses penyisipan *watermark* ini menghasilkan citra ter-*watermark* yang didapat dari pengolahan citra asli dan citra *watermark* menggunakan algoritma SVD.

Berikut ini akan dijelaskan masing-masing tools yang digunakan pada sistem penyisipan *watermark* menggunakan metode SVD yang ditunjukkan pada Gambar 4.10.

- i. Push Button
 - a. **Pilih Citra**, pada panel “Pemilihan Citra Asli” berfungsi untuk mengambil citra asli yang akan disisipi *watermark*, sedangkan pada panel “Pemilihan Citra *Watermark*” berfungsi untuk mengambil citra *watermark*.
 - b. **Sisipkan Watermark**, digunakan untuk menyisipkan citra *watermark* menggunakan metode SVD.
 - c. **Reset**, digunakan untuk menghapus semua input dan output pada program.

- d. **Simpan**, digunakan untuk menyimpan citra ter-*watermark*.
- ii. Menu
 - a. **Home**, digunakan untuk memanggil GUI dari program menu utama.
 - b. **Penyisipan Watermark**, digunakan untuk memanggil GUI dari program penyisipan ini.
 - c. **Ekstraksi Watermark**, digunakan untuk memanggil GUI dari program ekstraksi.
 - d. **Pengujian**, digunakan untuk memanggil GUI dari program PSNR dan koefisien korelasi.
 - iii. Edit Text
 - Alpha**, menyatakan nilai alpha.

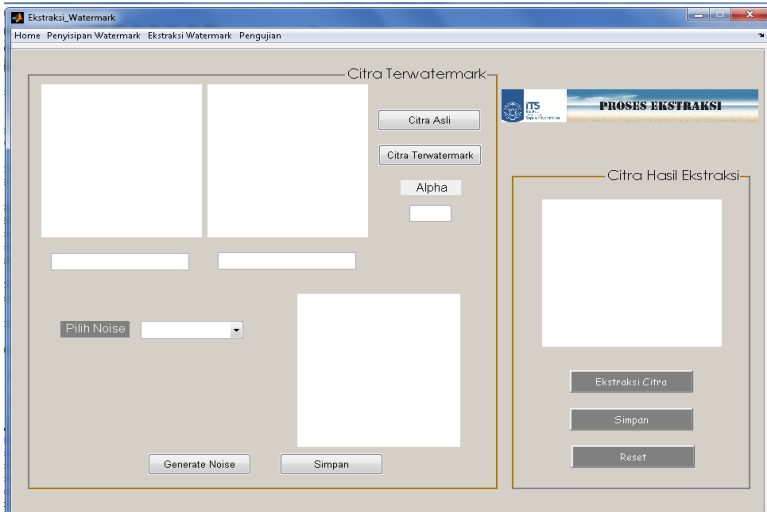


Gambar 4.10 Desain Antarmuka Program Watermarking pada Citra Digital Menggunakan Metode SVD

4.4.2 Program Ekstraksi

Program ini adalah salah satu rangkaian dari proses *watermarking* yang berfungsi untuk mengekstraksi citra

watermark dari citra ter-*watermark*. Gambar 4.11 berikut adalah desain antarmuka untuk program ekstraksi citra *watermark* :



Gambar 4.11 Desain Antarmuka Program Ekstraksi Citra *Watermark*

Berikut adalah tools digunakan pada sistem ekstraksi citra *watermark* :

- i. Push Button
 - a. **CitraAsli**, berfungsi untuk mengambil citra asli yang digunakan untuk mengambil informasi pada saat proses ekstraksi.
 - b. **Citra Terwatermark**, berfungsi untuk mengambil citra ter-*watermark* yang akan diekstraksi citra *watermark*-nya.
 - c. **Ekstraksi Citra**, berfungsi untuk memanggil proses ekstraksi.
 - d. **Reset**, digunakan untuk menghapus semua input dan output pada program.
 - e. **Simpan**, pada panel “Citra Terwatermark” berfungsi untuk menyimpan citra ter-*watermark* yang telah diberi

noise, sedangkan pada panel “Citra Hasil Ekstraksi” digunakan untuk menyimpan citra *watermark*.

ii. Menu

- a. **Home**, digunakan untuk memanggil GUI dari program menu utama.
- b. **Penyisipan Watermark**, digunakan untuk memanggil GUI dari program penyisipan.
- c. **Ekstraksi Watermark**, digunakan untuk memanggil GUI dari program ekstraksi ini.
- d. **Pengujian**, digunakan untuk memanggil GUI dari program PSNR dan koefisien korelasi.

iii. Edit Text

Alpha, menyatakan nilai alpha.

iv. Popup Menu

Pilih Noise terdiri dari pilihan *noise* yang akan ditambahkan pada citra ter-*watermark*.

4.4.3 Program Penghitungan PSNR

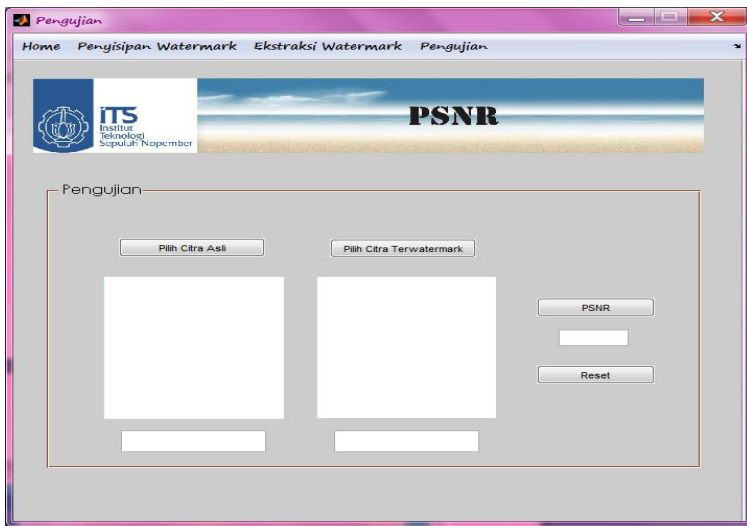
Program ini berfungsi untuk mengukur kualitas citra ter-*watermark* secara numerik. Cara mengukur kualitas citra ter-*watermark* ini dengan membandingkan antara citra ter-*watermark* dan citra asli, kemudian dicari nilai rasio antara kemungkinan maksimum dari kekuatan sinyal citra asli dengan sinyal yang terdegradasi pada citra hasil rekonstruksi. Desain antarmuka dari program PSNR ini ditunjukkan pada Gambar 4.12.

Berikut adalah tools digunakan pada program PSNR ini :

i. Push Button

- a. **Pilih Citra Asli**, berfungsi untuk mengambil citra asli.
- b. **Pilih Citra Terwatermark**, berfungsi untuk mengambil citra ter-*watermark*.
- c. **PSNR**, digunakan untuk menghitung nilai PSNR antara citra asli dan citra ter-*watermark*.
- d. **Reset**, digunakan untuk menghapus semua input dan output pada program.

- ii. Menu
 - a. **Home**, digunakan untuk memanggil GUI dari program menu utama.
 - b. **Penyisipan Watermark**, digunakan untuk memanggil GUI dari program penyisipan.
 - c. **Ekstraksi Watermark**, digunakan untuk memanggil GUI dari program ekstraksi.
 - d. **Pengujian**, digunakan untuk memanggil GUI dari program PSNR dan koefisien korelasi.
- iii. Edit Text
Hasil PSNR, menunjukkan nilai PSNR yang diperoleh.

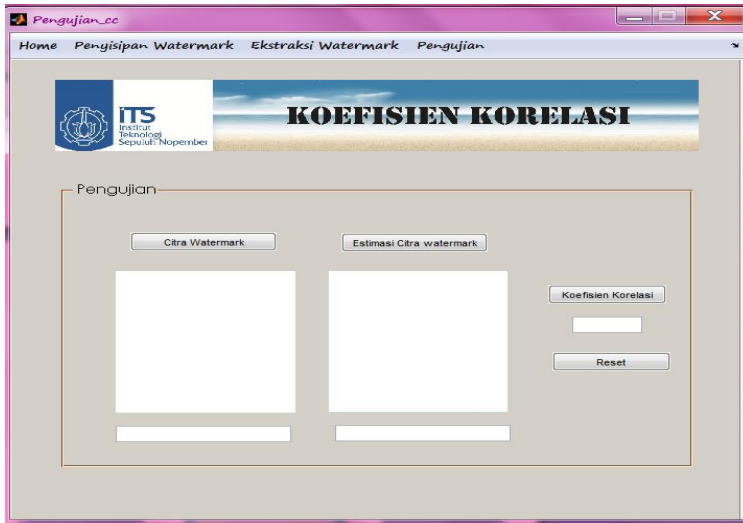


Gambar 4.12 Desain Antarmuka Program PSNR

4.4.4 Program Penghitungan Koefisien Korelasi

Program ini bertujuan untuk menghitung koefisien korelasi antara citra *watermark* dan estimasi citra *watermark*. Nilai koefisien korelasi untuk kedua citra tersebut dikatakan identik jika nilainya mendekati satu dan jika tidak maka nilainya

mendekati nol. Desain antarmuka dari program koefisien korelasi ditunjukkan oleh Gambar 4.13.



Gambar 4.13 Desain Antarmuka Program Koefisien Korelasi

Berikut adalah tools digunakan pada program Koefisien Korelasi:

- i. Push Button
 - a. **Pilih Citra Asli**, berfungsi untuk mengambil citra asli.
 - b. **Pilih Citra Terwatermark**, berfungsi untuk mengambil citra *ter-watermark*.
 - c. **Koefisien Korelasi**, digunakan untuk menghitung nilai kesamaan antara citra *ter-watermark* dan estimasinya.
 - d. **Reset**, digunakan untuk menghapus semua input dan output pada program.
- ii. Menu
 - a. **Home**, digunakan untuk memanggil GUI dari program menu utama.
 - b. **Penyisipan Watermark**, digunakan untuk memanggil GUI dari program penyisipan.

- c. **Ekstraksi Watermark**, digunakan untuk memanggil GUI dari program ekstraksi.
 - d. **Pengujian**, digunakan untuk memanggil GUI dari program PSNR dan koefisien korelasi.
- iii. Edit Text
 Hasil Koefisien Korelasi, menunjukkan nilai koefisien korelasi yang diperoleh.

4.5 Proses Pengambilan Citra

Dalam setiap proses pada *watermarking* menggunakan metode SVD memerlukan citra input untuk diproses. Sehingga diperlukan suatu fungsi untuk memanggil citra input tersebut. Fungsinya ditunjukkan dengan Gambar 4.14 berikut ini :

```
[namafile]=uigetfile({'*.jpg'; '*.bmp'; '*.*'}, 'Bu
ka Gambar');
if ~isequal (namafile,0)
N = imread(namafile);

[m n x] = size(N);
if x==3;
    M = rgb2gray(N);
else
    M = N;
end

A = im2double(M);
set(handles.figure1, 'CurrentAxes', handles.citraa
sli);

if m==n
    imshow(A), title('Citra Asli');
else
    msgbox('Ukuran Citra harus
NxN', 'Error', 'error');
end
end
```

Gambar 4.14 Kode Program Pengambilan Citra Input

4.6 Proses Penyimpanan Citra

Output yang dihasilkan pada tiap-tiap program berbeda-beda. Diperlukan suatu fungsi penyimpanan untuk menyimpan citra hasil proses, baik itu berupa estimasi dari citra *watermark*, citra ter-*watermark*, dan citra output yang lain. Gambar 4.15 berikut adalah fungsi untuk menyimpan citra output :

```
[namafile,direktori]=uiputfile({'*.jpg'; '*.bmp'}
,'Simpan Citra');
if ~isequal ([namafile,direktori],[0,0])
    set(handles.edit2, 'String',
[direktori,namafile]);
    imwrite(Aw,[direktori,namafile], 'jpg');
else
    msgbox('Tidak ada citra yang
disimpan', 'Warning', 'warn');
    return;
end
```

Gambar 4.15 Kode Program Penyimpanan Citra Output

4.7 Proses Pembuatan Citra Ter-watermark

Proses untuk membuat citra ter-*watermark* dilakukan dengan menyisipkan *watermark* pada citra asli dengan mengambil nilai - nilai singularnya dan fungsinya seperti pada Gambar 4.16.

```
[U,S,V] = svd(A);

fwm = zeros(m,n);
fwm(1:a, 1:b) = W;

alpha = str2double(get(handles.edit1,'string'));
if alpha > 0
    Sn = S + alpha.*fwm;
    Aw = U*Sn*V';

set(handles.figure1,'CurrentAxes',handles.citraterwatermark);
imshow(Aw),title('Citra Ter-watermark')
else
msgbox('Nilai alfa harus positif','Error','error');
end
end
```

Gambar 4.16 Kode Program Pembuatan Citra Ter-watermark

4.8 Proses Penambahan Noise

Penambahan *noise* dilakukan pada citra ter-watermark. *Noise* yang ditambahkan berupa *noise Gaussian*, *speckle*, dan *salt & pepper*. Gambar 4.17, 4.18, 4.19 berikut adalah kode program untuk penambahan *noise Gaussian*, *speckle*, dan *salt & pepper*.

a. *Noise Gaussian*

```
global Aw Aws
myhandles=guidata(gcbo);
m =str2double(get(myhandles.edit5,'string'));
v =str2double(get(myhandles.edit6,'string'));
Aws = imnoise(Aw,'gaussian',m,v);

set(myhandles.figure1,'CurrentAxes',myhandles.citraterwatermark);
imshow(Aws), title('Citra Ter-watermark dengan Noise');
```

Gambar 4.17 Kode Program Penambahan *Noise Gaussian*

b. *Noise speckle*

```

global Aw Aws
myhandles=guidata(gcbo);
v =str2double(get(myhandles.edit5,'string'));
Aws = imnoise(Aw,'speckle',v);
set(myhandles.figure1,'CurrentAxes',myhandles
.citraterwatermark);
imshow(Aws), title('Citra Ter-watermark
dengan Noise');

```

Gambar 4.18 Kode Program Penambahan *Noise speckle*c. *Noise salt & pepper*

```

global Aw Aws
myhandles=guidata(gcbo);
d =str2double(get(myhandles.edit5,'string'));
Aws = imnoise(Aw,'salt & pepper',d);
set(myhandles.figure1,'CurrentAxes',myhandles
.citraterwatermark);
imshow(Aws), title('Citra Ter-watermark
dengan Noise');

```

Gambar 4.19 Kode Program Penambahan *Noise salt & pepper***4.9 Proses Ekstraksi Watermark**

Pada proses ekstraksi *watermark* dibutuhkan informasi - informasi yang ada pada citra asli, seperti matriks - matriks dari hasil dekomposisi citra asli. Untuk itu citra asli akan dipanggil kembali pada proses ini. Sedangkan untuk proses ekstraksinya sendiri ditunjukkan dengan fungsi seperti pada Gambar 4.20.

```

myhandles=guidata(gcbo);

Sns=U'*Aws*V;
Ws=(Sns-S)./alpha;

set(myhandles.figure1,'CurrentAxes',myhandles.wa
termark);
imshow (Ws),title('Watermark')

```

Gambar 4.20 Kode Program Ekstraksi *Watermark*

4.10 Proses PSNR

Persamaan 3.7 dan 3.8 merupakan cara untuk mencari nilai MSE dan PSNR dan untuk implementasi dalam program ini ditunjukkan oleh gambar 4.21 berikut ini :

```
global A A2
myhandles=guidata(gcbo);

mseIm = (double(A) - double(A2)).^ 2;
[rows columns] = size(A);

mse = sum(sum(mseIm)) / (rows * columns);
hasil_psnr = 10 * log10((255^2) / mse);

set(myhandles.hslpsnr, 'String', hasil_psnr);
```

Gambar 4.21 Kode Program Penghitungan PSNR

4.11 Proses Koefisien Korelasi

Implementasi dari persamaan 3.1 adalah seperti Gambar 4.22 berikut ini :

```
uW = size(W);
uWS= size(Wstar);

S = sum(sum((W-(sum(sum(W))/(uW(1)*uW(2)))*
ones(uW(1),uW(2)))* (Wstar-sum(sum(Wstar))/
(uWS(1)*uWS(2)))*ones(uWS(1),uWS(2)))));
S1 = sum(sum((W-(sum(sum(W))/(uW(1)*uW(2)))*
ones(uW(1),uW(2))).^2));
S2 = sum(sum((Wstar-(sum(sum(Wstar))/
(uWS(1)*uWS(2)))*ones(uWS(1),uWS(2))).^2));

R = abs(S/sqrt(S1*S2));
```

Gambar 4.22 Kode Program Penghitungan Koefisien Korelasi

BAB V

HASIL DAN PEMBAHASAN PENGUJIAN SISTEM

Bab ini membahas mengenai hasil dan pengujian dari program *watermarking* dengan metode SVD yang telah dibuat. Pembahasan pengujian ini meliputi ruang lingkup dan data uji coba yang digunakan serta pelaksanaan uji coba dengan berbagai macam nilai parameter.

5.1 Ruang Lingkup Uji Coba

Ruang lingkup implementasi program yang dibangun dalam Tugas Akhir ini meliputi perangkat keras dan perangkat lunak yang digunakan. Spesifikasinya disajikan seperti pada Tabel 4.1

5.2 Data Uji Coba

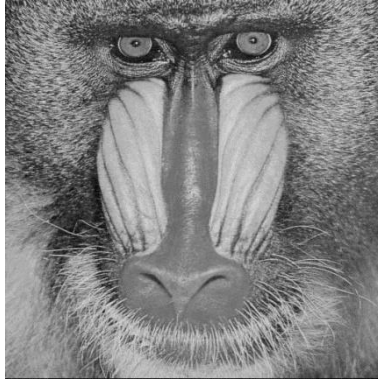
Uji coba pada program dalam Tugas Akhir ini dilakukan terhadap citra *grayscale* sebagai citra asli dengan ukuran 512×512 piksel dan citra *watermark*. Citra – citra uji coba tersebut antara lain disajikan dalam Gambar 5.1 dan Gambar 5.2.



(a)



(b)



(c)

Gambar 5.1 Citra Asli (a) Lena.jpg (b) Cameraman.jpg dan (c) Baboon.jpg

Setiap citra asli yang digunakan memiliki sifat yang berbeda-beda. Pada citra Lena memiliki kontras yang sedang namun gambarnya relatif konstan, jarang terdapat gradasi kontras yang drastis. Pada citra Cameraman memiliki kontras yang tinggi namun terdapat beberapa area luas dengan tingkat kontras yang sama membentuk blok-blok dengan gradasi konstan dan antar blok memiliki perbedaan kontras yang tinggi. Pada citra Baboon memiliki banyak garis-garis rapat dengan kontras yang sangat tinggi.



(a)



(b)

Gambar 5.2 Citra *watermark* (a) watermark1.jpg dan (b) watermark2.jpg

5.3 Pelaksanaan Uji Coba

Uji coba akan dilakukan dengan memperhatikan input citra, baik itu citra asli maupun citra *watermark*, dan nilai alpha. Jenis citra input akan diperhatikan untuk mengetahui apakah mempengaruhi atau tidak mempengaruhi kualitas *watermarking*. Parameter yang digunakan adalah :

1. Karakteristik dari citra asli.
2. Jenis *watermark*, berupa gambar logo atau gambar teks.

5.3.1 Uji Coba dengan Nilai Alpha yang Berbeda

Untuk mengetahui bahwa nilai alpha berpengaruh terhadap teknik *watermarking* dengan metode SVD, maka akan digunakan sepuluh nilai alpha yang berbeda pada setiap penyisipan *watermark*. Berikut ini adalah uji coba yang dilakukan:

- a. Uji coba pertama
Nama citra asli : Lena.jpg
Ukuran citra asli : 512×512
Nama watermark : watermark1.jpg
Ukuran watermark : 247×204

Untuk uji coba pertama dihasilkan citra ter-*watermark* pada Gambar 5.3. Untuk nilai PSNR dapat dilihat pada Tabel 5.1.



Gambar 5.3 Citra Lena Ter-watermark dengan Alpha
(a) 0.001 (b) 0.01 (c) 0.03 dan (d) 0.1 dari Hasil Uji Coba 5.3.1.a

Tabel 5.1 Nilai PSNR dari Citra Asli dan Citra Ter-watermark
pada Hasil Uji Coba 5.3.1.a

No.	Nilai α	PSNR
1.	0.0010	110.304
2.	0.0100	90.1464
3.	0.0300	85.8062
4.	0.1000	78.6772

- b. Uji coba kedua
Nama citra asli : Lena.jpg
Ukuran citra asli : 512×512
Nama watermark : watermark2.jpg
Ukuran watermark : 247×204

Hasil dari citra ter-*watermark* dari uji coba ini dapat dilihat pada Gambar 5.4. Untuk nilai PSNR dapat dilihat pada Tabel 5.2.



Gambar 5.4 Citra Lena Ter-*watermark* dengan Alpha
(a) 0.001 (b) 0.01 (c) 0.03 dan (d) 0.1 dari Hasil Uji Coba 5.3.1.b

Tabel 5.2 Nilai PSNR dari Citra Asli dan Citra Ter-*watermark* pada Hasil Uji Coba 5.3.1.b

No.	Nilai α	PSNR
1.	0.0010	Inf
2.	0.0100	96.7218
3.	0.0300	83.8368
4.	0.1000	75.4194

c. Uji coba ketiga

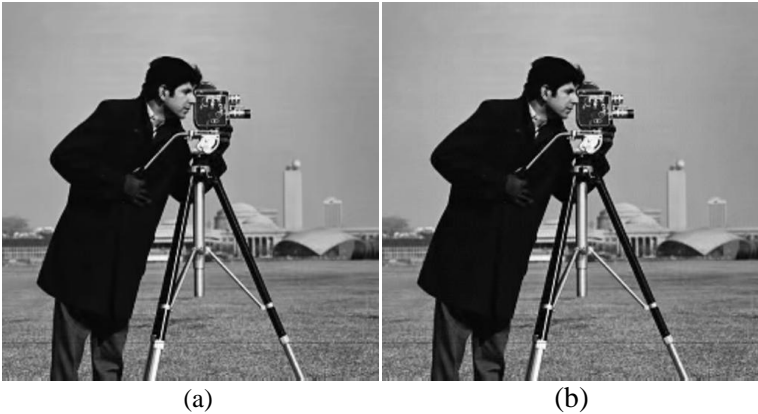
Nama citra asli : Cameraman.jpg

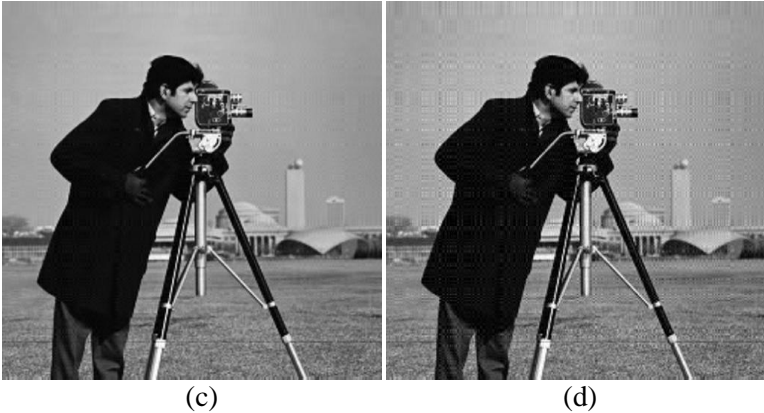
Ukuran citra asli : 512×512

Nama watermark : watermark1.jpg

Ukuran watermark : 247×204

Hasil dari citra ter-*watermark* dari uji coba ini dapat dilihat pada Gambar 5.5. Untuk nilai PSNR dapat dilihat pada Tabel 5.3.





Gambar 5.5 Citra Cameraman Ter-*watermark* dengan Alpha (a) 0.001 (b) 0.01 (c) 0.03 dan (d) 0.1 dari Hasil Uji Coba 5.3.1.c

Tabel 5.3 Nilai PSNR dari Citra Asli dan Citra Ter-*watermark* pada Hasil Uji Coba 5.3.1.c

No.	Nilai α	PSNR
1.	0.0010	90.9758
2.	0.0100	90.4179
3.	0.0300	87.2399
4.	0.1000	78.7861

d. Uji coba keempat

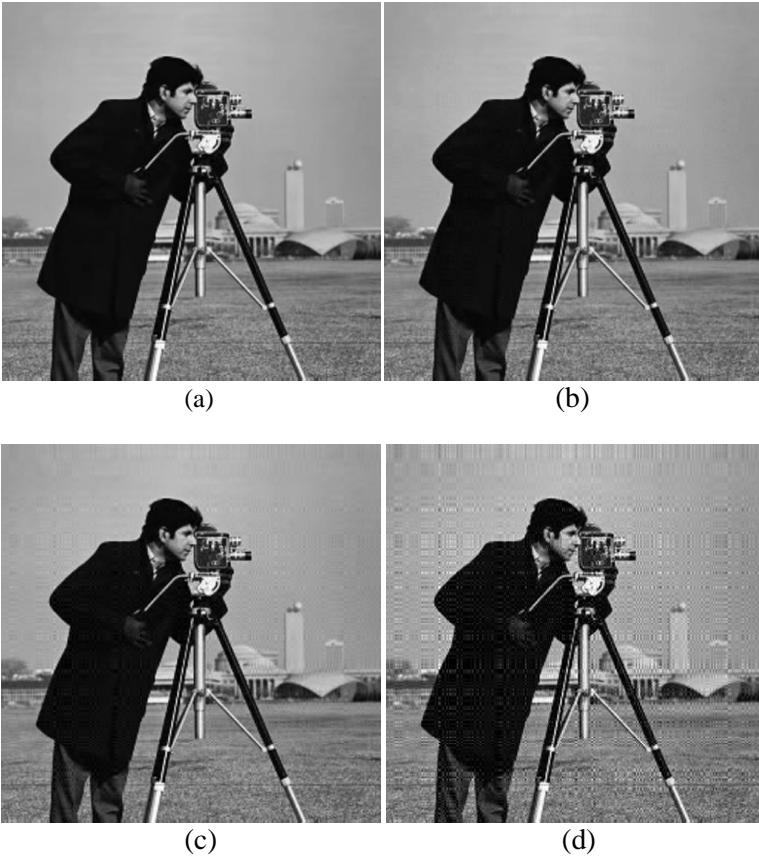
Nama citra asli : Cameraman.jpg

Ukuran citra asli : 512×512

Nama watermark : watermark2.jpg

Ukuran watermark : 247×204

Hasil dari citra ter-*watermark* dari uji coba ini dapat dilihat pada Gambar 5.6. Untuk nilai PSNR dapat dilihat pada Tabel 5.4.



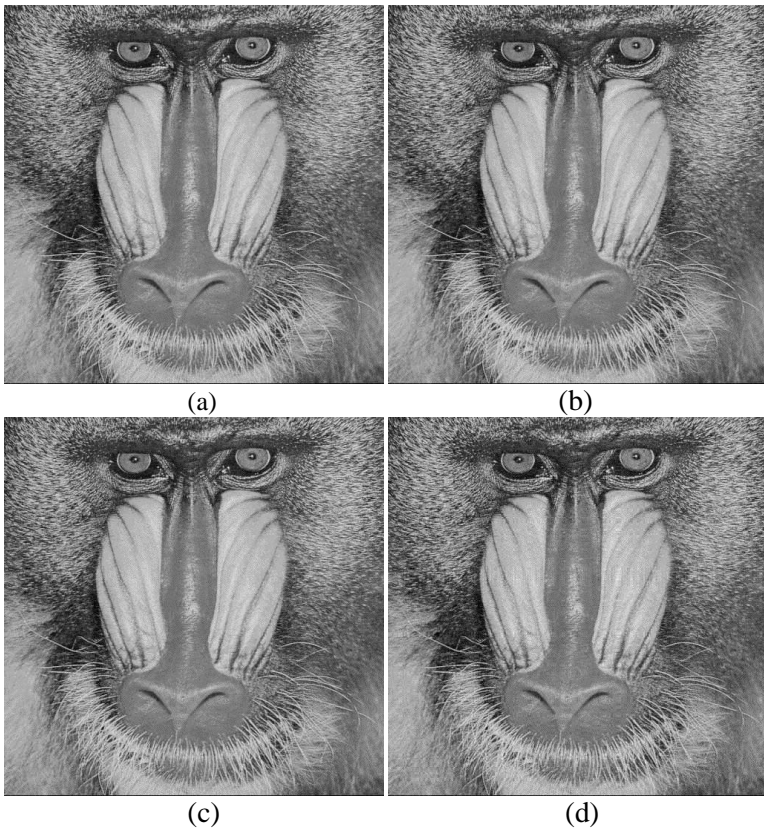
Gambar 5.6 Citra Cameraman Ter-watermark dengan Alpha (a) 0.001 (b) 0.01 (c) 0.03 dan (d) 0.1 dari Hasil Uji Coba 5.3.1.d

Tabel 5.4 Nilai PSNR dari Citra Asli dan Citra Ter-watermark pada Hasil Uji Coba 5.3.1.d

No.	Nilai α	PSNR
1.	0.0010	90.9743
2.	0.0100	89.8210
3.	0.0300	84.9791
4.	0.1000	75.7650

- e. Uji coba kelima
Nama citra asli : Baboon.jpg
Ukuran citra asli : 512×512
Nama watermark : watermark1.jpg
Ukuran watermark : 247×204

Hasil dari citra ter-*watermark* dari uji coba ini dapat dilihat pada Gambar 5.7. Untuk nilai PSNR dapat dilihat pada Tabel 5.5.



Gambar 5.7 Citra Baboon Ter-*watermark* dengan Alpha
(a) 0.001 (b) 0.01 (c) 0.03 dan (d) 0.1 dari Hasil Uji Coba 5.3.1.e

Tabel 5.5 Nilai PSNR dari Citra Asli dan Citra Ter-*watermark* pada Hasil Uji Coba 5.3.1.e

No.	Nilai α	PSNR
1.	0.0010	Inf
2.	0.0100	103.065
3.	0.0300	90.4226
4.	0.1000	78.9648

f. Uji coba keenam

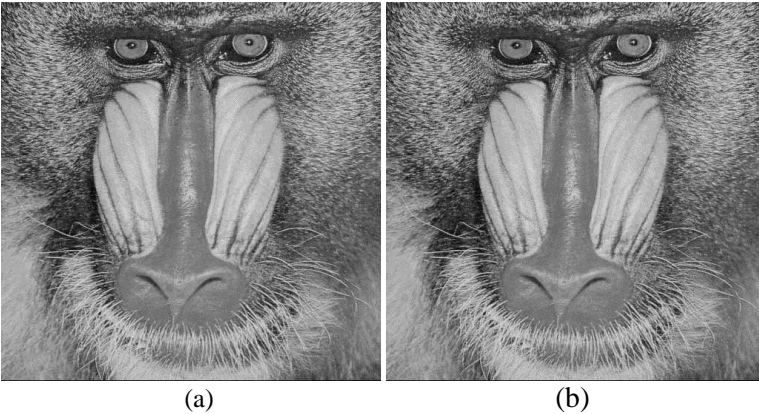
Nama citra asli : Baboon.jpg

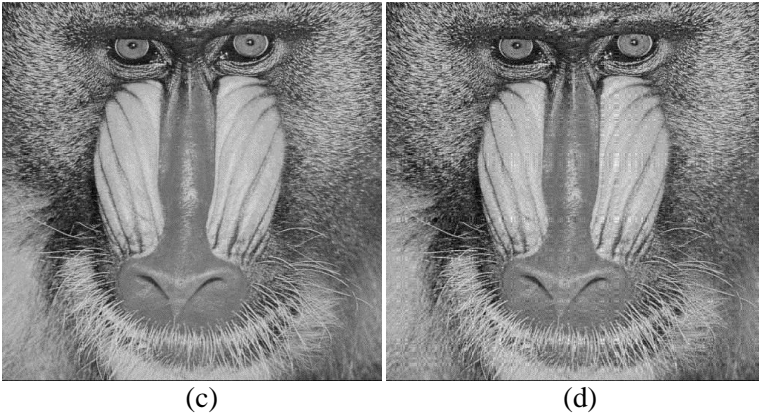
Ukuran citra asli : 512×512

Nama watermark : watermark2.jpg

Ukuran watermark : 247×204

Hasil dari citra ter-*watermark* dari uji coba ini dapat dilihat pada Gambar 5.8. Untuk nilai PSNR dapat dilihat pada Tabel 5.6.





Gambar 5.8 Citra Baboon Ter-*watermark* dengan Alpha
(a) 0.001 (b) 0.01 (c) 0.03 dan (d) 0.1 dari Hasil Uji Coba 5.3.1.f

Tabel 5.6 Nilai PSNR dari Citra Asli dan Citra Ter-*watermark*
pada Hasil Uji Coba 5.3.1.f

No.	Nilai α	PSNR
1.	0.0010	97.7759
2.	0.0100	83.1005
3.	0.0300	79.4245
4.	0.1000	73.5012

Berdasarkan pada percobaan-percobaan yang telah dilakukan, terlihat bahwa sampai pada nilai alpha tertentu secara kasat mata citra ter-*watermark* tidak memiliki perbedaan dengan citra asli. Namun dari perhitungan PSNR, dapat dilihat perbedaan antara keduanya. Dari dua hal tersebut, dapat disimpulkan bahwa faktor-faktor yang mempengaruhi kualitas *watermarking* dengan menggunakan metode SVD adalah :

1. Nilai alpha yang diterapkan

Semakin kecil nilai alpha yang diterapkan maka semakin baik kualitas citra ter-*watermark* yang didapat pada saat proses penyisipan *watermark*.

2. Karakteristik dari citra *watermark* yang digunakan
Setiap citra memiliki tingkat kerumitan atau karakteristik yang berbeda-beda. Pada setiap citra yang disisipi *watermark* seperti gambar logo, citra ter*watermark*nya memiliki kualitas baik dengan nilai PSNR diatas 79 dB. Sedangkan pada setiap citra yang disisipi *watermark* seperti gambar teks atau tulisan, citra ter*watermark*nya memiliki kualitas baik dengan nilai PSNR diatas 76 dB.

5.3.2 Uji Coba dengan Penambahan Noise

Untuk mengetahui apakah teknik *watermarking* dengan menggunakan metode SVD ini tahan terhadap gangguan berupa penambahan *noise*, dilakukan uji ketahanan terhadap beberapa citra ter-*watermark* yang berbeda dengan menambahkan *noise Gaussian, speckle, dan salt & pepper*.

- a. Uji coba pertama
Citra ter-*watermark* : Lena.jpg
Citra *watermark* yang tertanam : watermark1.jpg
Nilai alpha :0.1
Noise : *Gaussian* dengan mean 0

Hasil dari perhitungan nilai koefisien korelasi pada percobaan ini ditunjukkan dalam Tabel 5.7

Tabel 5.7 Kualitas Estimasi *Watermark* dari Hasil Uji Coba 5.3.2.a

No.	Nilai Variansi	Koefisien Korelasi
1.	Tanpa penambahan	0.919723
2.	0.0001	0.882150
3.	0.0005	0.768654
4.	0.0010	0.663673

- b. Uji coba kedua
Citra ter-*watermark* : Lena.jpg

Citra *watermark* yang tertanam : watermark1.jpg
 Nilai alpha : 0.1
 Noise : *Speckle*

Hasil dari perhitungan nilai koefisien korelasi pada percobaan ini ditunjukkan dalam Tabel 5.8

Tabel 5.8 Kualitas Estimasi *Watermark* dari Hasil Uji Coba 5.3.2.b

No.	Nilai Variansi	Koefisien Korelasi
1.	Tanpa penambahan	0.919723
2.	0.0009	0.841708
3.	0.0010	0.836243
4.	0.0050	0.618533

c. Uji coba ketiga

Citra ter-*watermark* : Lena.jpg
 Citra *watermark* yang tertanam : watermark1.jpg
 Nilai alpha : 0.1
 Noise : *Salt & Pepper*

Hasil dari perhitungan nilai koefisien korelasi pada percobaan ini ditunjukkan dalam Tabel 5.9

Tabel 5.9 Kualitas Estimasi *Watermark* dari Hasil Uji Coba 5.3.2.c

No.	Nilai Densitas	Koefisien Korelasi
1.	Tanpa penambahan	0.919723
2.	0.0009	0.835251
3.	0.0010	0.831206
4.	0.0050	0.595347

d. Uji coba keempat

Citra ter-*watermark* : Lena.jpg
 Citra *watermark* yang tertanam : watermark2.jpg
 Nilai alpha : 0.1
 Noise : *Gaussian* dengan mean 0

Hasil dari perhitungan nilai koefisien korelasi pada percobaan ini ditunjukkan dalam Tabel 5.10

Tabel 5.10 Kualitas Estimasi *Watermark* dari Hasil Uji Coba 5.3.2.d

No.	Nilai Variansi	Koefisien Korelasi
1.	Tanpa penambahan	0.791001
2.	0.0001	0.737745
3.	0.0005	0.585720

e. Uji coba kelima

Citra ter-*watermark* : Lena.jpg

Citra *watermark* yang tertanam : watermark2.jpg

Nilai alpha : 0.1

Noise : *Speckle*

Hasil dari perhitungan nilai koefisien korelasi pada percobaan ini ditunjukkan dalam Tabel 5.11

Tabel 5.11 Kualitas Estimasi *Watermark* dari Hasil Uji Coba 5.3.2.e

No.	Nilai Variansi	Koefisien Korelasi
1.	Tanpa penambahan	0.791001
2.	0.0009	0.681973
3.	0.0010	0.668919

f. Uji coba keenam

Citra ter-*watermark* : Lena.jpg

Citra *watermark* yang tertanam : watermark2.jpg

Nilai alpha : 0.1

Noise : *Salt & Pepper*

Hasil dari perhitungan nilai koefisien korelasi pada percobaan ini ditunjukkan dalam Tabel 5.12

Tabel 5.12 Kualitas Estimasi *Watermark* dari Hasil Uji Coba 5.3.2.f

No.	Nilai Densitas	Koefisien Korelasi
1.	Tanpa penambahan	0.791001
2.	0.0009	0.689683
3.	0.0010	0.660676

g. Uji coba ketujuh

Citra ter-*watermark* : Cameraman.jpg

Citra *watermark* yang tertanam : watermark1.jpg

Nilai alpha : 0.1

Noise : *Gaussian* dengan mean 0

Hasil dari perhitungan nilai koefisien korelasi pada percobaan ini ditunjukkan dalam Tabel 5.13

Tabel 5.13 Kualitas Estimasi *Watermark* dari Hasil Uji Coba 5.3.2.g

No.	Nilai Variansi	Koefisien Korelasi
1.	Tanpa penambahan	0.854234
2.	0.0001	0.829530
3.	0.0005	0.745479
4.	0.0010	0.600762

h. Uji coba kedelapan

Citra ter-*watermark* : Cameraman.jpg

Citra *watermark* yang tertanam : watermark1.jpg

Nilai alpha : 0.1

Noise : *Speckle*

Hasil dari perhitungan nilai koefisien korelasi pada percobaan ini ditunjukkan dalam Tabel 5.14

Tabel 5.14 Kualitas Estimasi *Watermark* dari Hasil Uji Coba 5.3.2.h

No.	Nilai Variansi	Koefisien Korelasi
1.	Tanpa penambahan	0.854234
2.	0.0009	0.824564
3.	0.0010	0.815600
4.	0.0050	0.616744

i. Uji coba kesembilan

Citra ter-*watermark* : Cameraman.jpg

Citra *watermark* yang tertanam : watermark1.jpg

Nilai alpha : 0.1

Noise : *Salt & Pepper*

Hasil dari perhitungan nilai koefisien korelasi pada percobaan ini ditunjukkan dalam Tabel 5.15

Tabel 5.15 Kualitas Estimasi *Watermark* dari Hasil Uji Coba 5.3.2.i

No.	Nilai Densitas	Koefisien Korelasi
1.	Tanpa penambahan	0.854234
2.	0.0010	0.787228
3.	0.0050	0.598316

j. Uji coba kesepuluh

Citra ter-*watermark* : Cameraman.jpg

Citra *watermark* yang tertanam : watermark2.jpg

Nilai alpha : 0.1

Noise : *Gaussian* dengan mean 0

Hasil dari perhitungan nilai koefisien korelasi pada percobaan ini ditunjukkan dalam Tabel 5.16

Tabel 5.16 Kualitas Estimasi *Watermark* dari Hasil Uji Coba 5.3.2.j

No.	Nilai Variansi	Koefisien Korelasi
1.	Tanpa penambahan	0.630782
2.	0.0001	0.586739
3.	0.0005	0.501733

k. Uji coba kesebelas

Citra ter-*watermark* : Cameraman.jpgCitra *watermark* yang tertanam : watermark2.jpg

Nilai alpha : 0.1

Noise : *Speckle*

Hasil dari perhitungan nilai koefisien korelasi pada percobaan ini ditunjukkan dalam Tabel 5.17

Tabel 5.17 Kualitas Estimasi *Watermark* dari Hasil Uji Coba 5.3.2.k

No.	Nilai Variansi	Koefisien Korelasi
1.	Tanpa penambahan	0.630782
2.	0.0009	0.555095
3.	0.0010	0.548237

1. Uji coba keduabelas

Citra ter-*watermark* : Cameraman.jpgCitra *watermark* yang tertanam : watermark2.jpg

Nilai alpha : 0.1

Noise : *Salt & Pepper*

Hasil dari perhitungan nilai koefisien korelasi pada percobaan ini ditunjukkan dalam Tabel 5.18

Tabel 5.18 Kualitas Estimasi *Watermark* dari Hasil Uji Coba 5.3.2.l

No.	Nilai Densitas	Koefisien Korelasi
1.	Tanpa penambahan	0.630782
2.	0.0009	0.535095
3.	0.0010	0.525002

m. Uji coba ketigabelas

Citra ter-*watermark* : Baboon.jpg

Citra *watermark* yang tertanam : watermark1.jpg

Nilai alpha : 0.1

Noise : *Gaussian* dengan mean 0

Hasil dari perhitungan nilai koefisien korelasi pada percobaan ini ditunjukkan dalam Tabel 5.19

Tabel 5.19 Kualitas Estimasi *Watermark* dari Hasil Uji Coba 5.3.2.m

No.	Nilai Variansi	Koefisien Korelasi
1.	Tanpa penambahan	0.833262
2.	0.0001	0.795795
3.	0.0005	0.683563
4.	0.0010	0.578189

n. Uji coba keempatbelas

Citra ter-*watermark* : Baboon.jpg

Citra *watermark* yang tertanam : watermark1.jpg

Nilai alpha : 0.1

Noise : *Speckle*

Hasil dari perhitungan nilai koefisien korelasi pada percobaan ini ditunjukkan dalam Tabel 5.20

Tabel 5.20 Kualitas Estimasi *Watermark* dari Hasil Uji Coba 5.3.2.n

No.	Nilai Variansi	Koefisien Korelasi
1.	Tanpa penambahan	0.833262
2.	0.0009	0.752711
3.	0.0010	0.744515
4.	0.0050	0.526709

o. Uji coba kelimabelas

Citra ter-*watermark* : Baboon.jpg

Citra *watermark* yang tertanam : watermark1.jpg

Nilai alpha : 0.1
Noise : Salt & Pepper

Hasil dari perhitungan nilai koefisien korelasi pada percobaan ini ditunjukkan dalam Tabel 5.21

Tabel 5.21 Kualitas Estimasi *Watermark* dari Hasil Uji Coba 5.3.2.o

No.	Nilai Densitas	Koefisien Korelasi
1.	Tanpa penambahan	0.833262
2.	0.0009	0.742958
3.	0.0010	0.744515
4.	0.0050	0.566581

- p. Uji coba keenambelas
 Citra ter-*watermark* : Baboon.jpg
 Citra *watermark* yang tertanam : watermark2.jpg
 Nilai alpha : 0.1
Noise : Gaussian dengan mean 0

Hasil dari perhitungan nilai koefisien korelasi pada percobaan ini ditunjukkan dalam Tabel 5.22

Tabel 5.22 Kualitas Estimasi *Watermark* dari Hasil Uji Coba 5.3.2.p

No.	Nilai Variansi	Koefisien Korelasi
1.	Tanpa penambahan	0.637339
2.	0.0001	0.596217
3.	0.0005	0.512139

- q. Uji coba ketujuhbelas
 Citra ter-*watermark* : Baboon.jpg
 Citra *watermark* yang tertanam : watermark2.jpg
 Nilai alpha : 0.1
Noise : Speckle

Hasil dari perhitungan nilai koefisien korelasi pada percobaan ini ditunjukkan dalam Tabel 5.23

Tabel 5.23 Kualitas Estimasi *Watermark* dari Hasil Uji Coba 5.3.2.q

No.	Nilai Variansi	Koefisien Korelasi
1.	Tanpa penambahan	0.637339
2.	0.0009	0.547871
3.	0.0010	0.538237

- r. Uji coba kedelapanbelas
 Citra ter-*watermark* : Baboon.jpg
 Citra *watermark* yang tertanam : watermark2.jpg
 Nilai alpha : 0.1
 Noise : Salt & Pepper

Hasil dari perhitungan nilai koefisien korelasi pada percobaan ini ditunjukkan dalam Tabel 5.24

Tabel 5.24 Kualitas Estimasi *Watermark* dari Hasil Uji Coba 5.3.2.r

No.	Nilai Densitas	Koefisien Korelasi
1.	Tanpa penambahan	0.637339
2.	0.0009	0.547715
3.	0.0010	0.534454

Berdasarkan pada percobaan yang telah dilakukan, dapat ditarik kesimpulan bahwa :

1. Nilai alpha yang diterapkan mempengaruhi ketahanan dari *watermark* yang tertanam terhadap penambahan *noise*. Semakin besar nilai alpha maka semakin kokoh pula keberadaan *watermark* yang tertanam.
2. Tanpa menghilangkan *noise*, *watermarking* dengan metode SVD ini tahan terhadap penambahan *noise* :
 - a. *Gaussian* dengan mean 0 dan variansi sampai 0.001 untuk setiap citra ter-*watermark* dengan *watermark* yang tertanam adalah watermark1.jpg.

- b. *Gaussian* dengan mean 0 dan variansi sampai 0.0005 untuk setiap citra ter-*watermark* dengan *watermark* yang tertanam adalah watermark2.jpg.
- c. *Speckle* dengan variansi sampai 0.005 untuk setiap citra ter-*watermark* dengan *watermark* yang tertanam adalah watermark1.jpg.
- d. *Speckle* dengan variansi sampai 0.001 untuk setiap citra ter-*watermark* dengan *watermark* yang tertanam adalah watermark2.jpg.
- e. *Salt & Pepper* dengan nilai kerapatan sampai 0.005 untuk setiap citra ter-*watermark* dengan *watermark* yang tertanam adalah watermark1.jpg.
- f. *Salt & Pepper* dengan nilai kerapatan sampai 0.001 untuk setiap citra ter-*watermark* dengan *watermark* yang tertanam adalah watermark2.jpg.

BAB VI PENUTUP

6.1 Kesimpulan

Berdasarkan keseluruhan hasil pembahasan pada bab-bab sebelumnya, dapat diperoleh kesimpulan sebagai berikut :

1. Program *watermarking* menggunakan metode SVD dapat mengekstraksi *watermark* dengan baik dengan menggunakan informasi yang ada pada citra asli.
2. Semakin besar nilai alpha yang digunakan maka semakin kecil error yang didapat pada saat proses ekstraksi *watermark*. Namun hal ini berbanding terbalik dengan kualitas citra ter-*watermark*. Semakin besar nilai alpha yang digunakan, kualitas dari citra ter-*watermark* semakin rendah.
3. Setiap citra memiliki tingkat kerumitan atau karakteristik yang berbeda-beda. Pada setiap citra yang disisipi *watermark* seperti gambar logo, citra ter-*watermark*nya memiliki kualitas baik dengan nilai PSNR diatas 79 dB. Sedangkan pada setiap citra yang disisipi *watermark* seperti gambar teks atau tulisan, citra ter-*watermark*nya memiliki kualitas baik dengan nilai PSNR diatas 76 dB.
4. *Watermark* yang tertanam dalam citra ter-*watermark* bersifat *invisible* dan paling tahan terhadap gangguan berupa penambahan *noise* bertipe *speckle* sampai nilai variansi 0.005

6.2 Saran

Saran yang penulis berikan untuk pengembangan Tugas Akhir berikutnya adalah :

1. Pada Tugas Akhir ini, Citra asli dan *watermark* yang menjadi input adalah citra *grayscale*, diharapkan dalam penelitian selanjutnya dapat menggunakan citra berwarna.

2. Citra asli dan *watermark* dalam program ini menggunakan format bertipe .jpg dan .bmp, diharapkan dalam penelitian selanjutnya dapat menggunakan format bertipe .gif, .png.
3. Algoritma yang digunakan pada Tugas Akhir ini adalah SVD (*Singular Value Decomposition*), dalam penelitian selanjutnya dapat dicoba dengan algoritma yang tanpa menggunakan citra asli untuk proses ekstraksinya.
4. Sebagai pengembangan program, dapat dibuat program *watermarking* pada data digital lainnya misalkan teks, suara, video dan sebagainya.
5. Gangguan yang dilakukan terhadap citra ter-*watermark* berupa *Gaussian noise*, *Speckle*, dan *Salt & Pepper*, pada penelitian selanjutnya diharapkan dapat ditambahkan satu proses lagi yaitu *denoising* sebelum dilakukan proses ekstraksi.

DAFTAR PUSTAKA

- [1]. Putra, D. 2010. *Pengolahan Citra Digital*. Yogyakarta : C.V Andi Offset.
- [2]. Purnomo, M,H dan Muntasa, A. 2010. *Konsep Pengolahan Citra Digital dan Ekstraksi Fitur*. Yogyakarta : Graha Ilmu.
- [3]. Irvan, M. 2011. *Pemberian Tanda Air pada Citra Digital dengan Skema Tanda Air Berdasarkan Kuantisasi Warna*. Surabaya : Institut Teknologi Sepuluh Nopember.
- [4]. Liu, L. *A Survey of Digital Watermarking Technologies*.
- [5]. Leon, S,J. 2001. *Aljabar Linear dan Aplikasinya*. Jakarta : Erlangga
- [6]. Dogan, S, dkk. 2011. *A Robust Color Image Watermarking with Singular Value Decomposition*. **Advances in Engineering Software Vol. 42**, Hal. 336-346.
- [7]. Mohammad, A,A, dkk. 2008. *An Improved SVD-based Watermarking Scheme for Protecting Rightful Ownership*. **Signal Processing Vol. 88**, Hal. 2158-2180.
- [8]. Liu, R dan Tan, T. 2002. *An SVD-based Watermarking Scheme for Protecting Rightful Ownership*. **Transactions on Multimedia Vol. 4**, No. 1.
- [9]. Sepdianto, T. 2011. *Pemberian Tanda Air Menggunakan Teknik Kuantisasi Rata-Rata dengan Domain Transformasi*

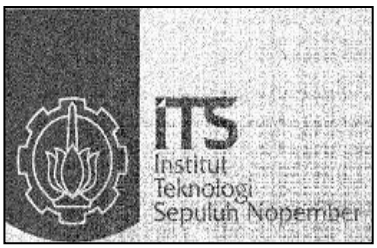

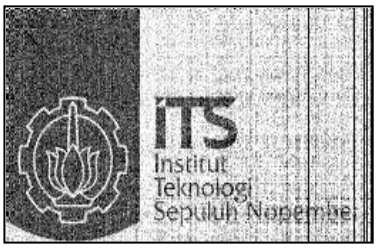

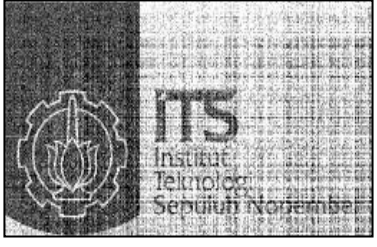

Wavelet Diskrit. Surabaya : Institut Teknologi Sepuluh
Nopember.

BIODATA PENULIS

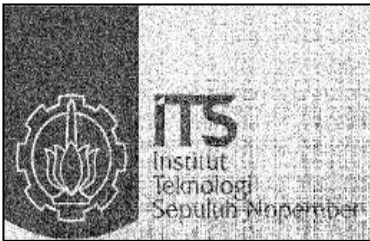


Penulis dilahirkan di Sidoarjo pada tanggal 28 Oktober 1991 ini memiliki nama lengkap Latifatul Machbubah. Saat ini penulis tinggal di daerah Taman-Sidoarjo bersama kedua orang tua dan kakak keduanya. Penulis merupakan anak ketiga dari tiga bersaudara. Penulis telah menempuh pendidikan formal dari MI NUR ROHMAN Sambibulu pada tahun 1997 s.d 2003, SMPN 2 Sukodono pada tahun 2003 s.d 2006, dan SMAN 1 Taman pada tahun 2006 s.d 2009. Pada tahun 2009 penulis mengikuti program PMDK untuk seleksi masuk Perguruan Tinggi Negeri dan Alhamdulillah diterima di Jurusan Matematika FMIPA-ITS terdaftar dengan NRP 1209 100 027. Di Jurusan Matematika ini penulis mengambil Bidang Studi Ilmu Komputer. Penulis juga pernah aktif di beberapa organisasi intra kampus salah satunya adalah Himpunan Mahasiswa Matematika (HIMATIKA) ITS selama dua periode. Selain itu penulis juga aktif melakukan kegiatan di luar kampus seperti menjadi tutor dan pengajar privat serta asisten dosen kalkulus di kampus ITS. Informasi yang berhubungan dengan Tugas Akhir ini dapat ditujukan ke alamat email: m.latifatul@yahoo.co.id

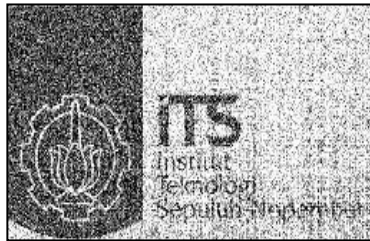
LAMPIRAN

Estimasi <i>Watermark</i> dari Citra Lena	Estimasi <i>Watermark</i> dari Citra Lena
 The image shows the Lena test image with a watermark. The watermark consists of the ITS logo (a stylized flower) and the text "ITS Institut Teknologi Sepuluh Nopember" overlaid on the image.	 The image shows the estimated watermark from the Lena image, which is the text "INI ADALAH WATERMARK" in bold, black, uppercase letters.
Estimasi <i>Watermark</i> dari Citra Cameraman	Estimasi <i>Watermark</i> dari Citra Cameraman
 The image shows the Cameraman test image with a watermark. The watermark consists of the ITS logo and the text "ITS Institut Teknologi Sepuluh Nopember" overlaid on the image.	 The image shows the estimated watermark from the Cameraman image, which is the text "INI ADALAH WATERMARK" in bold, black, uppercase letters.
Estimasi <i>Watermark</i> dari Citra Baboon	Estimasi <i>Watermark</i> dari Citra Baboon
 The image shows the Baboon test image with a watermark. The watermark consists of the ITS logo and the text "ITS Institut Teknologi Sepuluh Nopember" overlaid on the image.	 The image shows the estimated watermark from the Baboon image, which is the text "INI ADALAH WATERMARK" in bold, black, uppercase letters.

Estimasi *Watermark 1* dari Citra Lena dengan Penambahan
Noise Gaussian mean 0



Variansi : 0.0001

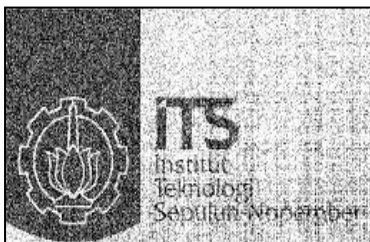


Variansi : 0.0005

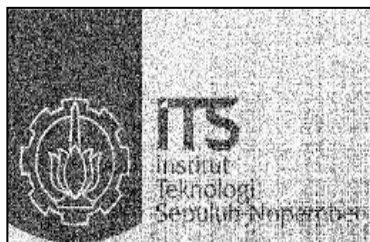


Variansi : 0.001

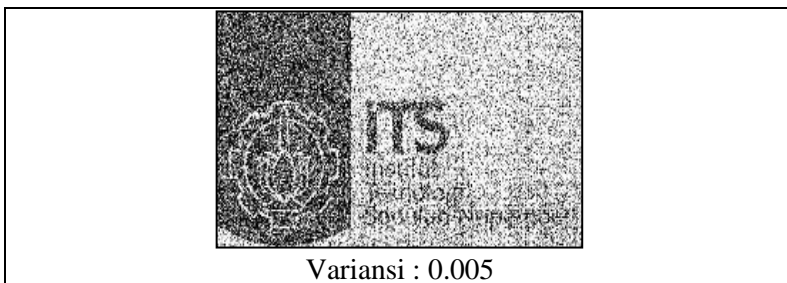
Estimasi *Watermark 1* dari Citra Lena dengan Penambahan
Noise Speckle



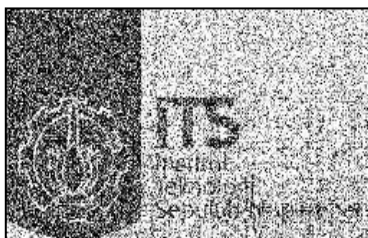
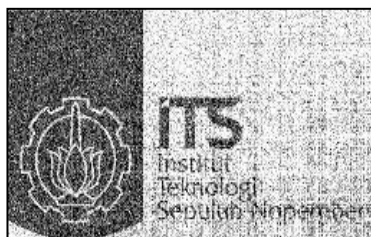
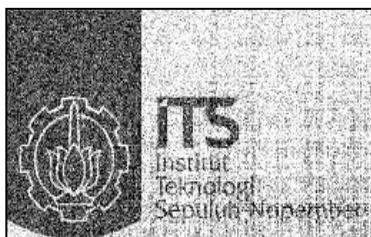
Variansi : 0.0009



Variansi : 0.001

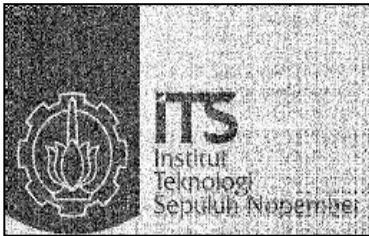


Estimasi *Watermark 1* dari Citra Lena dengan Penambahan
Noise Salt & Pepper



Estimasi <i>Watermark 2</i> dari Citra Lena dengan Penambahan <i>Noise Gaussian</i> mean 0	
<p>INI ADALAH WATERMARK</p>	<p>INI ADALAH WATERMARK</p>
Variansi : 0.0001	Variansi : 0.0005
Estimasi <i>Watermark 2</i> dari Citra Lena dengan Penambahan <i>Noise Speckle</i>	
<p>INI ADALAH WATERMARK</p>	<p>INI ADALAH WATERMARK</p>
Variansi : 0.0009	Variansi : 0.001
Estimasi <i>Watermark 2</i> dari Citra Lena dengan Penambahan <i>Noise Salt & Pepper</i>	
<p>INI ADALAH WATERMARK</p>	<p>INI ADALAH WATERMARK</p>
Densitas : 0.0009	Densitas : 0.001

Estimasi *Watermark 1* dari Citra Cameraman dengan
Penambahan *Noise Gaussian* mean 0



Variansi : 0.0001



Variansi : 0.0005



Variansi : 0.001

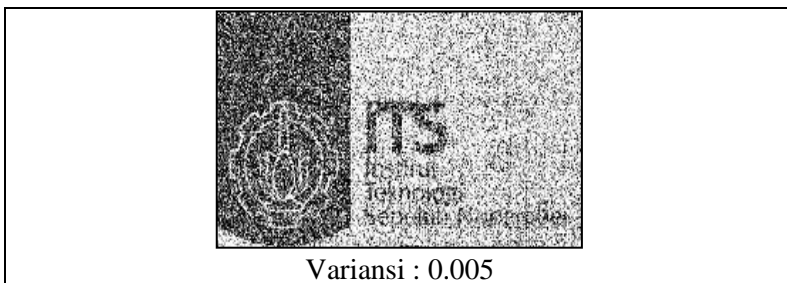
Estimasi *Watermark 1* dari Citra Cameraman dengan
Penambahan *Noise Speckle*



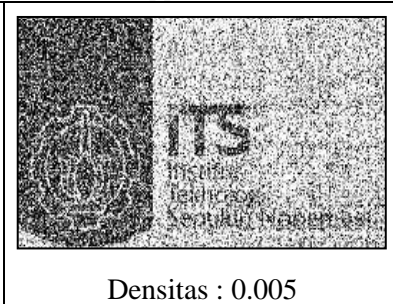
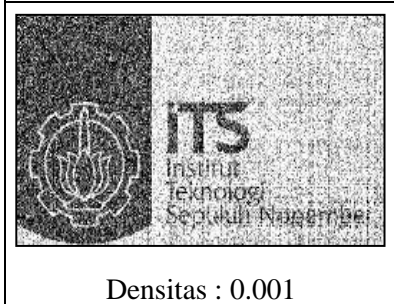
Variansi : 0.0009



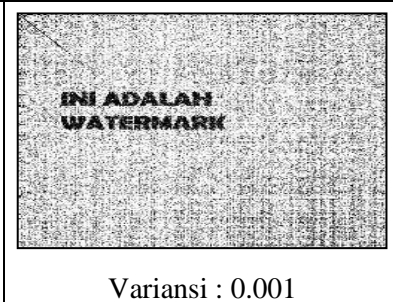
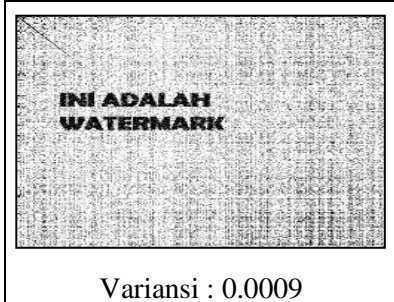
Variansi : 0.001



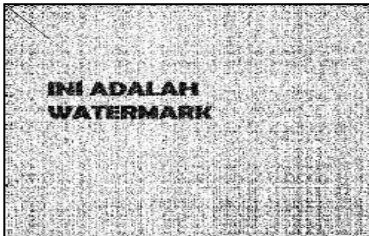
Estimasi *Watermark 1* dari Citra Cameraman dengan Penambahan *Noise Salt & Pepper*



Estimasi *Watermark 2* dari Citra Cameraman dengan Penambahan *Noise Gaussian mean 0*



Estimasi *Watermark 2* dari Citra Cameraman dengan
Penambahan *Noise Speckle*

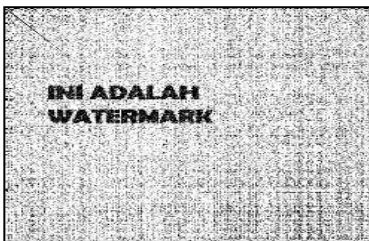


Variansi : 0.0009

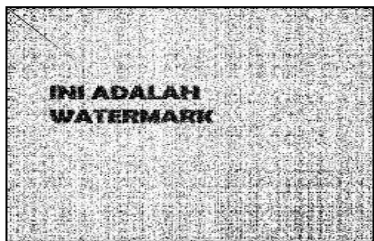


Variansi : 0.001

Estimasi *Watermark 2* dari Citra Cameraman dengan
Penambahan *Noise Salt & Pepper*

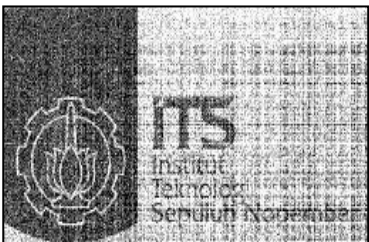


Densitas : 0.0009

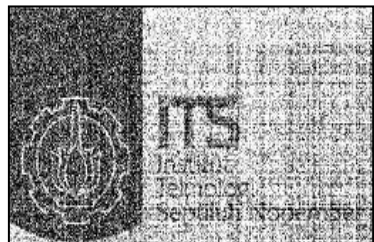


Densitas : 0.001

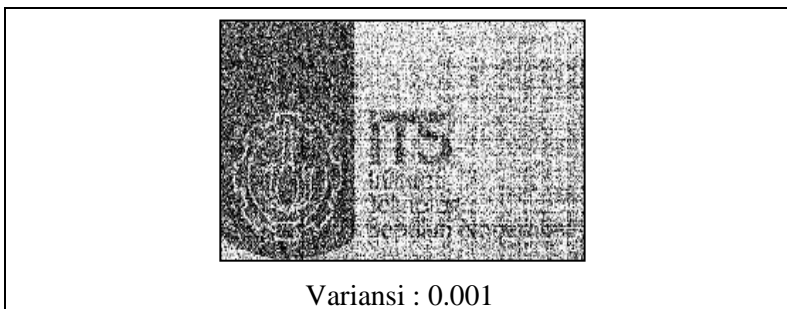
Estimasi *Watermark 1* dari Citra Baboon dengan Penambahan
Noise Gaussian mean 0



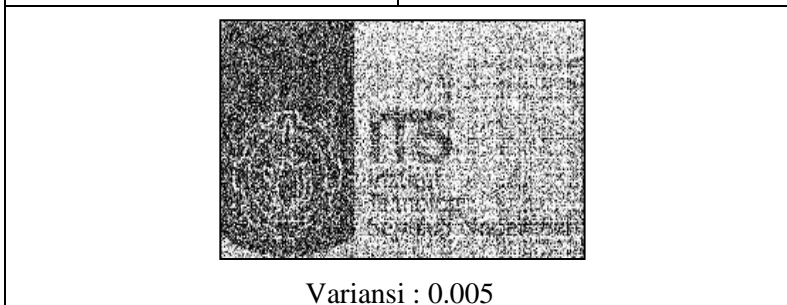
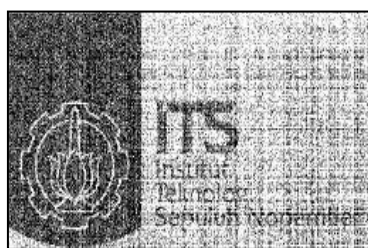
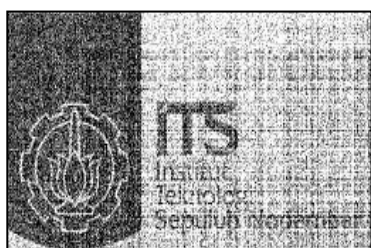
Variansi : 0.0001



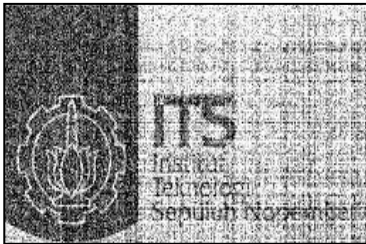
Variansi : 0.0005



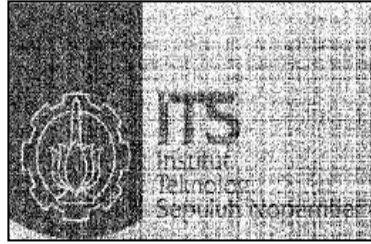
Estimasi *Watermark 1* dari Citra Baboon dengan Penambahan *Noise Speckle*



Estimasi *Watermark 1* dari Citra Baboon dengan Penambahan
Noise Salt & Pepper



Densitas : 0.0009



Densitas : 0.001



Densitas : 0.005

Estimasi *Watermark 2* dari Citra Baboon dengan Penambahan
Noise Gaussian mean 0



Variansi : 0.0001

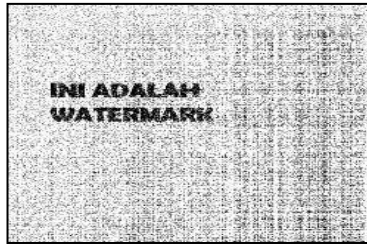


Variansi : 0.0005

Estimasi *Watermark 2* dari Citra Baboon dengan Penambahan
Noise Speckle



Variansi : 0.0009

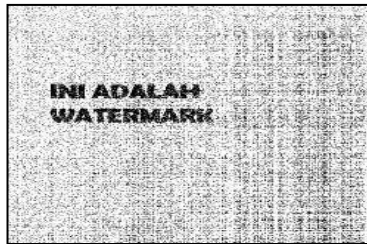


Variansi : 0.001

Estimasi *Watermark 2* dari Citra Baboon dengan Penambahan
Noise Salt & Pepper



Densitas : 0.0009



Densitas : 0.001