

**PENERAPAN OPERASI MORFOLOGI UNTUK PERBANDINGAN
METODE GAUSSIAN FILTER, MEDIAN DAN MEAN DALAM
REDUKSI NOISE CITRA HITAM PUTIH ULTRASONOGRAFI**

SKRIPSI

**Diajukan Untuk Memenuhi Persyaratan Untuk Meraih Gelar Sarjana Komputer
Pada Program Studi Teknik Informatika Fakultas Teknik dan Komputer
Universitas Harapan Medan**



**DENNY ABDUL MALIK
14350156**

**PROGRAM STUDI TEKNIK INFORMATIKA
FAKULTAS TEKNIK DAN KOMPUTER
UNIVERSITAS HARAPAN MEDAN
MEDAN
2021**

**PENERAPAN OPERASI MORFOLOGI UNTUK PERBANDINGAN
METODE GAUSSIAN FILTER, MEDIAN DAN MEAN DALAM
REDUKSI NOISE CITRA HITAM PUTIH ULTRASONOGRAFI**

**DENNY ABDUL MALIK
14350156**

SKRIPSI

**Diajukan untuk Memenuhi Persyaratan Meraih Gelar Sarjana Komputer Pada
Program Studi Teknik Informatika Fakultas Teknik dan Komputer
Universitas Harapan Medan**

**PROGRAM STUDI TEKNIK INFORMATIKA
FAKULTAS TEKNIK DAN KOMPUTER
UNIVERSITAS HARAPAN MEDAN
MEDAN
Juli, 2021**

PERNYATAAN PEMBIMBING

Saya/Kami dengan ini menyatakan bahwa saya/kami telah memeriksa Skripsi Mahasiswa ini dan menurut pendapat saya/kami, Skripsi Mahasiswa ini telah mencukupi untuk ruang lingkup dan kualitas untuk dianugerahkan gelar Sarjana Teknik/Komputer dalam Bidang Teknik Informatika.

Medan, / / 2022

Pembimbing 1

Pembimbing 2

(Tommy, S.T., M.Kom)

(Fera Damayanti, S.T., M.Kom)

PERNYATAAN MAHASISWA

Saya yang bertanda tangan dibawah ini

Nama : Denny Abdul Malik
NPM : 14350156
Program Studi : TEKNIK INFORMATIKA
Fakultas : TEKNIK DAN KOMPUTER
Jenjang Studi : STRATA I

Menyatakan bahwa:

1. Skripsi ini merupakan gagasan, rumusan dan ide saya, tanpa bantuan dari pihak lain kecuali arahan dari Tim Dosen Pembimbing
2. Skripsi ini belum pernah diajukan untuk mendapat gelar sarjana, baik di Prodi Sistem Fakultas Teknik dan Komputer Universitas Harapan Medan maupun di perguruan tinggi lain.
3. Dalam Skripsi ini tidak terdapat karya atau pendapat yang ditulis atau dipublikasikan orang lain, kecuali secara tertulis dengan mencantumkan sebagai acuan dalam naskah dengan menyebutkan nama pengarang dan dicantumkan dalam daftar pustaka.

Demikian pernyataan ini saya perbuat dengan sesungguhnya dan apabila dikemudian hari terbukti pernyataan ini tidak benar, maka saya bersedia menerima sanksi akademik berupa pencabutan gelar yang telah diberikan melalui karya tulis, serta sanksi lainnya sesuai dengan norma yang berlaku di Perguruan Tinggi ini.

Medan, / / 2022
Yang Menyatakan

Denny Abdul Malik
14350156

PERSETUJUAN

JUDUL : **PENERAPAN OPERASI MORFOLOGI
UNTUK PERBANDINGAN METODE
GAUSSIAN FILTER, MEDIAN DAN MEAN
DALAM REDUKSI NOISE CITRA HITAM
PUTIH ULTRA SONOGRAFI**

KATEGORI : **SKRIPSI**

NAMA : **DENNY ABDUL MALIK**

NPM : **14350156**

PROGRAM STUDI : **TEKNIK INFORMATIKA**

FAKULTAS : **TEKNIK DAN KOMPUTER**

TAHUN TAMAT : **2021**

DISETUJUI OLEH
Komisi Pembimbing

Pembimbing 1

Pembimbing 2

Tommy, S.T., M.Kom

Fera Damayanti, S.T., M.Kom

Mengetahui,
Ketua Program Studi Teknik Informatika

Ilham Faisal, S.T., M.Kom

KATA PENGANTAR

Puji dan syukur penulis ucapkan kepada Allah SWT atas limpahan berkat, rahmat, serta kemudahan yang telah diberikan sehingga penulis dapat menyelesaikan skripsi ini yang merupakan syarat untuk mendapatkan gelar Sarjana Teknik/Komputer pada Program Studi Teknik Informatika, Fakultas Teknik dan Komputer, Universitas Harapan Medan. Tak lupa juga shalawat serta salam kepada Nabi Muhammad SAW yang telah memberi petunjuk kepada kita ke jalan yang lurus.

Dalam kurun waktu pengerjaan skripsi ini penulis menyadari bahwa sangat banyak pihak yang berjasa turut membantu penulis dalam penyelesaian skripsi ini. Dalam kesempatan ini penulis mengucapkan terima kasih kepada :

1. Ayahanda dan Ibunda atas doa dan kasih sayang yang tulus dan tak terhingga kepada penulis.
2. Bapak Abdul Jabbar Lubis, S.T., M.Kom. Selaku Dekan Fakultas Teknik dan Komputer.
3. Bapak Ilham Faisal, S.T., M.Kom. Selaku Ketua Program Studi Teknik Informatika Fakultas Teknik dan Komputer.
4. Bapak Tommy, S.T., M.Kom selaku pembimbing 1 dan Ibu Fera Damayanti, S.T., M.Kom selaku pembimbing 2 yang telah meluangkan waktu membimbing penulis selama pengerjaan skripsi ini.
5. Ibu Haida Dafitri, S.T., M.Kom. Selaku Sekretaris Program Studi Teknik Informatika.
6. Keluarga dan teman-teman seperjuangan Program Studi Teknik Informatika Fakultas Teknik dan Komputer yang telah memberikan motivasi dan perhatiannya.
7. Semua pihak yang terlibat baik langsung maupun tidak langsung dalam pengerjaan skripsi ini yang tidak penulis sebutkan satu persatu diucapkan terima kasih.

Penulis menyadari bahwa skripsi ini masih banyak kekurangan dan belum sempurna dari segi bentuk dan isinya, hal tersebut disebabkan karena keterbatasan kemampuan dan pengetahuan penulis. Oleh karena itu untuk menyempurnakan skripsi ini, maka penulis mengharapkan saran dan kritik yang sifatnya membangun pola pikir penulis dalam menyusun skripsi ini. Akhir kata penulis berharap semoga skripsi ini dapat bermanfaat bagi mahasiswa/i Teknik Informatika Universitas Harapan Medan.

Medan, Juli 2021

Denny Abdul Malik
14350156

ABSTRAK

Citra USG diambil dengan menggunakan gelombang suara berfrekuensi tinggi sehingga menghasilkan citra dengan bintik gelap dan terang yang menyebar pada objek. Dalam hal ini citra yang dihasilkan tampak buram dan adanya bintik-bintik yang akan menyebabkan berkurangnya informasi pada citra tersebut. Sebuah citra yang mengalami penurunan kualitas karena adanya *noise*. *Noise* dapat disebabkan oleh gangguan fisik pada alat akuisi maupun secara disengaja akibat proses pengolahan yang tidak sesuai. Berdasarkan masalah tersebut dibutuhkan sebuah sistem yang dapat memperbaiki citra USG dengan cara meminimalisir *noise* yang ada. Akan tetapi banyak metode yang telah tercipta untuk melakukan perbaikan citra dengan cara menghilangkan *noise*, sehingga membingungkan seseorang untuk menggunakan metode yang mana. Oleh sebab itu pada penelitian ini dilakukan perbandingan beberapa metode seperti *Gaussian filter*, *Median* dan *Mean* dalam menghilangkan *noise* pada citra USG agar dapat diketahui metode mana yang lebih baik. Perbandingan tiga metode didasari oleh nilai PSNR dan MSE. Dari hasil penelitian yang telah diterapkan, metode *Median filter* lebih unggul dalam meminimalisir *noise* pada citra USG dengan *noise* hingga 60% daripada metode *Gaussian* dan *Mean*.

Kata Kunci : *Citra, USG, Noise, Gaussian, Median, Mean*

ABSTRACT

Ultrasound images are taken using high-frequency sound waves to produce an image with dark and light spots that spread over the object. In this case the resulting image looks blurry and the presence of spots will cause reduced information in the image. An image that has decreased in quality due to noise. Noise can be caused by physical disturbances to the acquisition device or intentionally due to improper processing. Based on these problems, a system is needed that can improve the ultrasound image by minimizing the existing noise. However, many methods have been created to improve the image by removing noise, so it is confusing for someone to use which method. Therefore, in this study, a comparison of several methods such as Gaussian filter, Median and Mean in removing noise in ultrasound images was carried out in order to know which method is better. The comparison of the three methods is based on the PSNR and MSE values. From the results of the research that has been applied, the Median filter method is superior in minimizing noise in ultrasound images with up to 60% noise than the Gaussian and Mean methods.

Keywords: *Image, USG, Noise, Gaussian, Median, Mean*

DAFTAR ISI

PERNYATAAN PEMBIMBING	i
PERNYATAAN MAHASISWA	ii
PERSETUJUAN	iii
KATA PENGANTAR	iv
ABSTRAK	vi
ABSTRACT	vii
DAFTAR ISI	viii
DAFTAR GAMBAR	x
DAFTAR TABEL	xii
BAB 1 PENDAHULUAN	1
1.1 Latar Belakang Masalah	1
1.2 Rumusan Masalah	3
1.3 Batasan masalah	3
1.4 Tujuan	4
1.5 Manfaat	4
1.6 Metodologi Penelitian	4
1.7 Sistematika Penulisan	5
BAB 2 DASAR TEORI	6
2.1 Pengertian Citra	6
2.1.1 Jenis-Jenis Citra	6
2.1.2 Format File Citra	8
2.2 Jenis Citra Digital	9
2.3 Pengolahan Citra Digital	11
2.3.1 Pemrosesan Citra Digital	12
2.3.2 Noise	13
2.4 Gaussian Filter	14
2.5 Median Filter	15
2.6 Mean Filter	16

2.7	Unified Modeling Language (UML)	17
2.7.1	Use Case Diagram	18
2.8	<i>Flowchart</i>	19
2.9	MATLAB	21
BAB 3 : ANALISIS DAN PERANCANGAN SISTEM		23
3.1	Analisa	23
3.1.1	Analisis Data	23
3.1.2	Penerapan Metode	24
3.1.3	Perancangan Sistem	43
3.1.4	Flowchart	45
BAB 4 : IMPLEMENTASI DAN PENGUJIAN SISTEM		48
4.1	Implementasi dan Perangkat Sistem	48
4.2	Tampilan Sistem Aplikasi	49
4.2.1	Pengujian Sistem	50
4.2.2	Hasil Pengujian Sistem	58
4.2.3	Perbandingan Hasil Pengujian	59
BAB 5 : KESIMPULAN DAN SARAN		64
5.1	Kesimpulan	64
5.2	Saran	64
DAFTAR PUSTAKA		

DAFTAR GAMBAR

Gambar 2.1 Matriks Citra Digital Berdimensi M x N	7
Gambar 2.2 Ilustrasi Digitalisasi Citra	8
Gambar 2.3 Contoh penyimpanan citra warna di dalam memori	10
Gambar 2.4 Citra <i>grayscale</i>	10
Gambar 2.5 Perbandingan gradasi warna 1 <i>bit</i> , 2 <i>bit</i> , 5 <i>bit</i> , 6 <i>bit</i> , 7 <i>bit</i> , 8 <i>bit</i>	11
Gambar 2.6 Citra biner	11
Gambar 2.7 Tampilan <i>State</i> Awal Matlab	22
Gambar 3.1 Sampel Citra Ultrasonography	23
Gambar 3.2 Nilai Piksel Citra Ultrasonography 800 x 600	23
Gambar 3.3 Pebandingan Nilai MSE dan PSNR Citra Hasil Reduksi <i>Noise</i>	42
Gambar 3.4 Rancangan Menu Utama	43
Gambar 3.5 Rancangan Reduksi <i>Noise</i>	44
Gambar 3.6 <i>Flowchart Gaussian Filter</i>	46
Gambar 3.7 <i>Flowchart Mean Filter</i>	46
Gambar 3.8 <i>Flowchart Median Filter</i>	47
Gambar 4.1 Tampilan Menu Utama	49
Gambar 4.2 Tampilan Menu Reduksi <i>Noise</i>	49
Gambar 4.3 Proses Pemilihan Citra USG	50
Gambar 4.4 Memilihan Citra USG	51
Gambar 4.5 Informasi Citra USG10%	51
Gambar 4.6 Informasi Citra USG10% Reduksi <i>Noise Gaussian</i>	52
Gambar 4.7 Informasi Citra USG10% Reduksi <i>Noise Mean</i>	52
Gambar 4.8 Informasi Citra USG10% Reduksi <i>Noise Median</i>	53
Gambar 4.9 Hasil Pebandingan Reduksi <i>Noise</i> Citra USG 10%	53
Gambar 4.10 Citra USG 20% Reduksi <i>Noise Gaussian, Mean, Median</i>	54
Gambar 4.11 Hasil Pebandingan Reduksi <i>Noise</i> Citra USG 20%	54
Gambar 4.12 Citra USG 30% Reduksi <i>Noise Gaussian, Mean, Median</i>	55

Gambar 4.13 Hasil Pebandingan Reduksi <i>Noise</i> Citra USG 30%	55
Gambar 4.14 Citra USG 40% Reduksi <i>Noise Gaussian, Mean, Median</i>	56
Gambar 4.15 Hasil Pebandingan Reduksi <i>Noise</i> Citra USG 40%	56
Gambar 4.16 Citra USG 60% Reduksi <i>Noise Gaussian, Mean, Median</i>	57
Gambar 4.17 Hasil Pebandingan Reduksi <i>Noise</i> Citra USG 60%	57

DAFTAR TABEL

Tabel 2.1 Warna dan Nilai Penyusun Warna	9
Tabel 2.2 Diagram dalam UML	18
Tabel 2.3 Simbol <i>Use Case</i> Diagram	19
Tabel 2.4 Simbol-simbol <i>flowchart</i>	20
Tabel 3.1 Nilai Piksel Citra Ultrasonography 5x5	24
Tabel 3.2 Nilai Piksel Citra Ultrasonography 5x5 Proses Gaussian	24
Tabel 3.3 Nilai Kernel 3x3 <i>Gaussian</i>	25
Tabel 3.4 Nilai Citra <i>Gaussian</i> Kernel 1	25
Tabel 3.5 Nilai Citra <i>Gaussian</i> 1	25
Tabel 3.6 Nilai Citra <i>Gaussian</i> Kernel 2	26
Tabel 3.7 Nilai Citra <i>Gaussian</i> 2	26
Tabel 3.8 Nilai Citra <i>Gaussian</i> Kernel 3	26
Tabel 3.9 Nilai Citra <i>Gaussian</i> Kernel 3	27
Tabel 3.10 Nilai Citra <i>Gaussian</i> Kernel 4	27
Tabel 3.11 Nilai Citra <i>Gaussian</i> 4	27
Tabel 3.12 Nilai Citra <i>Gaussian</i> Kernel 5	27
Tabel 3.13 Nilai Citra <i>Gaussian</i> 5	28
Tabel 3.14 Nilai Citra <i>Gaussian</i> Kernel 6	28
Tabel 3.15 Nilai Citra <i>Gaussian</i> 6	28
Tabel 3.16 Nilai Citra <i>Gaussian</i> Kernel 7	28
Tabel 3.17 Nilai Citra <i>Gaussian</i> 7	29
Tabel 3.18 Nilai Citra <i>Gaussian</i> Kernel 8	29
Tabel 3.19 Nilai Citra <i>Gaussian</i> 8	29
Tabel 3.20 Nilai Citra <i>Gaussian</i> Kernel 9	29
Tabel 3.21 Nilai Citra <i>Gaussian</i> 9	30
Tabel 3.22 Nilai Piksel Citra Ultrasonography <i>Gaussian</i> 5x5	30
Tabel 3.23 Nilai Citra <i>Filter Mean</i> Kernel 1	31
Tabel 3.24 Nilai Citra <i>Filter Mean</i> 1	31
Tabel 3.25 Nilai Citra <i>Filter Mean</i> Kernel 2	31

Tabel 3.26 Nilai Citra <i>Filter Mean</i> 2	31
Tabel 3.27 Nilai Citra <i>Filter Mean</i> Kernel 3	32
Tabel 3.28 Nilai Citra <i>Filter Mean</i> Kernel 3	32
Tabel 3.29 Nilai Citra <i>Filter Mean</i> Kernel 4	32
Tabel 3.30 Nilai Citra <i>Filter Mean</i> 4	32
Tabel 3.31 Nilai Citra <i>Filter Mean</i> Kernel 5	32
Tabel 3.32 Nilai Citra <i>Filter Mean</i> 5	33
Tabel 3.33 Nilai Citra <i>Filter Mean</i> Kernel 6	33
Tabel 3.34 Nilai Citra <i>Filter Mean</i> 6	33
Tabel 3.35 Nilai Citra <i>Filter Mean</i> Kernel 7	33
Tabel 3.36 Nilai Citra <i>Filter Mean</i> 7	34
Tabel 3.37 Nilai Citra <i>Filter Mean</i> Kernel 8	34
Tabel 3.38 Nilai Citra <i>Filter Mean</i> 8	34
Tabel 3.39 Nilai Citra <i>Filter Mean</i> Kernel 9	34
Tabel 3.40 Nilai Citra <i>Filter Mean</i> 9	35
Tabel 3.41 Nilai Piksel Citra Ultrasonography <i>Gaussian</i> 5x5	35
Tabel 3.42 Nilai Citra <i>Filter Median</i> Kernel 1	36
Tabel 3.43 Urutan Nilai Citra <i>Filter Median</i> Kernel 1	36
Tabel 3.44 Nilai Citra <i>Filter Median</i> 1	36
Tabel 3.45 Nilai Citra <i>Filter Median</i> Kernel 2	36
Tabel 3.46 Urutan Nilai Citra <i>Filter Median</i> Kernel 2	37
Tabel 3.47 Nilai Citra <i>Filter Median</i> 2	37
Tabel 3.48 Nilai Citra <i>Filter Median</i> Kernel 3	37
Tabel 3.49 Urutan Nilai Citra <i>Filter Median</i> Kernel 3	37
Tabel 3.50 Nilai Citra <i>Filter Median</i> 3	37
Tabel 3.51 Nilai Citra <i>Filter Median</i> Kernel 4	37
Tabel 3.52 Urutan Nilai Citra <i>Filter Median</i> Kernel 4	38
Tabel 3.53 Nilai Citra <i>Filter Median</i> 4	38
Tabel 3.54 Nilai Citra <i>Filter Median</i> Kernel 5	38
Tabel 3.55 Urutan Nilai Citra <i>Filter Median</i> Kernel 5	38
Tabel 3.56 Nilai Citra <i>Filter Median</i> Kernel 5	39
Tabel 3.57 Nilai Citra <i>Filter Median</i> Kernel 6	39

Tabel 3.58 Urutan Nilai Citra <i>Filter Median</i> Kernel 6	39
Tabel 3.59 Nilai Citra <i>Filter Median Mean</i> 6	39
Tabel 3.60 Nilai Citra <i>Filter Median</i> Kernel 7	39
Tabel 3.61 Urutan Nilai Citra <i>Filter Median</i> Kernel 7	40
Tabel 3.62 Nilai Citra <i>Filter Median</i> 7	40
Tabel 3.63 Nilai Citra <i>Filter Median</i> Kernel 8	40
Tabel 3.64 Urutan Nilai Citra <i>Filter Median</i> Kernel 8	40
Tabel 3.65 Nilai Citra <i>Filter Median</i> 8	40
Tabel 3.66 Nilai Citra <i>Filter Median</i> Kernel 9	41
Tabel 3.67 Urutan Nilai Citra <i>Filter Median</i> Kernel 9	41
Tabel 3.68 Nilai Citra <i>Filter Median</i> 9	41
Tabel 3.69 Nilai Pixel Citra Ultrasonography Median 5x5	41
Tabel 4.1 Kebutuhan Perangkat Keras	48
Tabel 4.2 Kebutuhan Perangkat Lunak	48
Tabel 4.3 Citra USG	50
Tabel 4.4 Hasil Pengujian	58
Tabel 4.5 Perbandingan Hasil <i>Gaussian</i>	59
Tabel 4.6 Perbandingan Hasil <i>Mean</i>	61
Tabel 4.7 Perbandingan Hasil <i>Median</i>	

BAB 1

PENDAHULUAN

1.1 Latar Belakang Masalah

USG (*ultrasonography*) adalah alat diagnostik *noninvasive* yang digunakan untuk mendapatkan hasil gambar berupa struktur organ didalam tubuh. Alat ini menggunakan suatu gelombang suara dengan tinggi frekuensi diatas 20.000 hertz. Gelombang suara dikirim melalui suatu alat yang disebut *transducer* atau *probe*. Objek didalam tubuh akan memantulkan kembali gelombang suara yang kemudian akan ditangkap oleh suatu sensor, gelombang pantulan tersebut akan direkam, dianalisis dan ditayangkan di layar (Andiro dan Ginting 2019). Citra hasil USG umumnya berbentuk citra dengan komposisi warna hitam dan putih. Citra hasil USG umumnya juga terkadang menghasilkan citra yang dengan hasil kualitas yang rendah seperti adanya *noise* yang dapat mengurangi mutu citra. Dengan adanya *noise*, citra yang dihasilkan akan tampak kotor, muncul bintik-bintik, yang akan menyebabkan berkurangnya informasi pada citra tersebut (Furqan, Sriani & Siregar, 2020).

Citra yang mengandung *noise* seringkali membatasi informasi berharga yang dibutuhkan ketika analisis citra dilakukan. Hal tersebut menyebabkan *noise* harus dihilangkan agar memberikan hasil citra yang lebih baik (Afiyat, 2017). Proses menghilangkan *noise* pada citra yang telah dihasilkan dapat dilakukan dengan cara direduksi. Reduksi *noise* adalah salah satu proses dalam perbaikan citra untuk menghasilkan citra/gambar/foto yang jelas (Simangunsong, 2017). Reduksi *noise* pada citra tentu memerlukan metode. Untuk mendapatkan hasil reduksi *noise* yang baik, maka pertimbangan yang paling penting adalah pemilihan metode yang digunakan. Telah banyak cara yang dilakukan untuk memperbaiki *citra noise*, dan telah banyak pula metode-metode yang telah digunakan yang beberapa diantaranya seperti metode *gaussian filter*, *median filter* dan *mean filter*

Gaussian filter merupakan metode perbaikan citra dengan jenis operasi konvolusi. Operasi *gaussian filter* dilakukan dengan mengalikan antara matriks kernel dengan matriks gambar asli. Matriks *kernel gauss* didapat dari fungsi komputasi dari distribusi *gaussian*

(Andiro dan Ginting 2019). *Median filter* juga merupakan suatu metode perbaikan citra dengan jenis operasi konvolusi. Operasi dilakukan secara *nonlinear* dengan cara mengurutkan nilai intensitas sekelompok piksel, kemudian mengganti nilai piksel yang diproses dengan sebuah nilai tengah (*median*), sedangkan *mean filter* adalah metode perbaikan citra dengan proses menggantikan nilai setiap piksel dengan nilai hasil pencarian rata-rata nilai intensitas piksel-piksel tetangga sesuai dengan filter yang dirancang (Wedianto dkk, 2016). Ketiga metode tersebut kerap digunakan dalam proses perbaikan citra RGB yang umumnya memiliki komposisi banyak warna pada penglihatan mata, akan tetapi masih jarang yang membahas tentang reduksi *noise* pada citra hasil USG yang umumnya memiliki warna hitam dan putih pada penglihatan mata. Oleh sebab itu perlu adanya perbandingan metode mana yang lebih baik dalam mereduksi *noise* pada citra hasil USG.

Penelitian yang dilakukan oleh Andiro dan Ginting pada tahun 2019 tentang peningkatan kualitas citra USG dengan metode *gaussian filter* mendapatkan kesimpulan bahwa semakin tinggi nilai *grayscale* citra maka kontras yang dihasilkan juga semakin tinggi dengan hasil perbaikan citra yang lebih baik. Kelemahan dari penelitian Andiro dan Ginting adalah proses reduksi *noise* dilakukan pada citra *graycale* dengan kedalaman 8 bit atau 16 bit tanpa adanya proses reduksi *noise* dengan citra RGB kedalaman 24 bit, sehingga tidak diketahui kinerja dari metode *gaussian filter* dalam mereduksi *noise* citra RGB kedalaman 24 bit. Penelitian yang dilakukan oleh Fadila Wati pada tahun 2018 tentang reduksi *noise* pada citra USG menggunakan metode *median filter* mendapatkan kesimpulan bahwa dengan citra hasil USG yang memiliki bintik-bintik *noise* secara acak, metode *median filter* dapat memfilter *noise* dan membuat citra hasil USG lebih jelas serta bagus. Kelemahan dari penelitian Fadlina Wati adalah tidak adanya parameter hasil dari metode *median filter* berdasarkan proses reduksi *noise*, sehingga hasil hanya dapat dilihat dengan visual mata saja, tanpa ada pembuktian dengan data dan parameter yang ada seperti nilai PSNR dan MSE.

Sedangkan penelitian yang dilakukan oleh Wedianto, Sari dan Suzantri pada tahun 2016 tentang perbandingan reduksi *noise* terhadap citra RGB yang memiliki komposisi warna cerah menggunakan metode *gaussian*, *mean* dan *median filter* menghasilkan kesimpulan metode *mean filter* lebih baik dalam mereduksi *noise*. Kelemahan dari penelitian Wedianto, Sari dan Suzantri adalah pada tahap pengujian dalam perbandingan

yang masih belum akurat. Hal ini didasari oleh proses perbandingan hanya melihat dari bentuk hasil histogram tanpa adanya perbandingan nilai PSNR dan MSE, serta pengujian hanya dilakukan pada citra RGB dengan komposisi banyak warna, tanpa ada pengujian dengan citra komposisi warna dominan gelap atau hanya memiliki dua *visual* warna seperti hitam dan putih sehingga tidak diketahui kinerja metode *gaussian*, *mean* dan *median filter* dalam mereduksi *noise* pada citra yang memiliki komposisi warna sedikit.

Oleh sebab itu penelitian ini akan melakukan perbandingan baru dengan citra hasil USG yang umumnya memiliki komposisi warna hitam dan putih menggunakan metode-metode *gaussian filter*, *median filter* dan *mean filter* dengan parameter hasil perbandingan berupa nilai PSNR dan MSE pada citra RGB kedalaman 24 bit. Perbandingan nilai PSNR dan MSE bertujuan untuk melihat seberapa jauh perbedaan citra USG awal dan citra USG hasil reduksi *noise* menggunakan metode *gaussian filter*, *median filter* dan *mean filter*. MSE tidak memiliki satuan sedangkan satuan dari PSNR adalah *desibel*. Semakin mirip kedua citra maka nilai MSEnya semakin mendekati nilai nol. Sedangkan pada PSNR, dua buah citra dikatakan memiliki tingkat kemiripan yang rendah jika nilai PSNR di bawah 30 dB (Tamtama, 2021). Sehingga parameter PSNR dan MSE akan dijadikan perbandingan dari metode *gaussian filter*, *median filter* dan *mean filter* dengan melihat tingkat kemiripan citra awal *noise* dengan citra hasil reduksi *noise*.

Berdasarkan uraian latar belakang dan masalah tersebut, maka judul penelitian ini adalah **“Penerapan Operasi Morfologi Untuk Perbandingan Metode Gaussian Filter, Median dan Mean Dalam Reduksi Noise Citra Hitam Putih Ultrasonografi”**

1.2 Rumusan Masalah

Adapun rumusan masalah yang diambil berdasarkan latar belakang yang telah diuraikan diatas antara lain adalah :

1. Bagaimana perbandingan kualitas dari metode *gaussian filter*, *median filter* dan *mean filter* dalam rangka mereduksi *noise* pada citra hasil USG ?
2. Bagaimana membangun aplikasi yang dapat mengimplementasikan metode *gaussian filter*, *median filter* dan *mean filter* terhadap reduksi *noise* dari citra hasil USG ?

1.3 Batasan Masalah

Adapun batasan masalah dalam penelitian ini adalah

1. Citra yang digunakan dalam penelitian ini hanya menggunakan citra 24 bit.
2. Format file citra yang digunakan dalam penelitian ini hanya menggunakan JPG, PNG dan BMP.
3. Perbandingan berdasarkan nilai PSNR dan MSE.
4. Kernel yang digunakan 3 x 3.
5. Aplikasi bahasa pemrograman yang digunakan adalah Matlab 2017b.

1.4 Tujuan

Adapun Tujuan dalam penelitian ini sebagai berikut :

1. Untuk perbandingan kualitas dari metode *gaussian filter*, *median filter* dan *mean filter* dalam rangka mereduksi *noise* pada citra hasil USG.
2. Untuk membangun aplikasi yang dapat mengimplementasikan metode *gaussian filter*, *median filter* dan *mean filter* reduksi *noise* dari citra hasil USG.

1.5 Manfaat

Manfaat yang dihasilkan dalam penelitian ini adalah sebagai berikut:

1. Agar diketahui metode mana yang memiliki kinerja yang paling baik dalam mereduksi *noise* pada citra hasil USG.
2. Agar memberikan perbandingan yang lebih akurat dan lebih baik dari penelitian-penelitian terhadulu dalam melakukan reduksi *noise* pada citra hasil USG.
3. Menghasilkan aplikasi perbaikan citra hasil USG menggunakan metode *gaussian filter*, *median filter* dan *mean filter* serta citra yang dihasilkan lebih baik dari citra awal.

1.6 Metodologi Penelitian

Metode pengumpulan data yang digunakan dalam pembahasan penelitian ini adalah sebagai berikut:

1. Studi Literatur

Merupakan tahap pengumpulan data dengan cara mengumpulkan literatur, jurnal, *paper*, dan buku-buku yang berkaitan dengan judul penelitian, serta mencari informasi dari berbagai sumber di internet.

2. Analisa dan perancangan sistem

Tahap ini akan dilaksanakan analisa data, kebutuhan sistem yang digambarkan dalam *flowchart*, dan perancangan antarmuka.

3. Implementasi sistem

Implementasi dilaksanakan berdasarkan hasil analisa dan perancangan yang telah dilakukan sebelumnya. Dalam tahap ini dilakukan pengkodean (*coding*) dalam menggunakan bahasa pemograman *matlab*.

4. Pengujian sistem

Tahap ini dilakukan pengujian terhadap proses reduksi noise pada sistem yang telah dibangun. Meliputi *input* citra USG, proses reduksi *noise gaussian*, proses reduksi *noise median* dan proses reduksi *noise mean*, kemudian membandingkan nilai MSE dan PSNR.

5. Dokumentasi

Membuat dokumentasi sistem dari tahap awal sampai pengujian sistem, kemudian untuk disusun dalam format penelitian.

1.7 Sistematika Penulisan

Agar penulisan menjadi lebih terstruktur, maka adapun sistematika penulisan tugas akhir ini terdiri dari 5 bab, yaitu:

BAB 1 : PENDAHULUAN

Bab 1 berisikan latar belakang masalah, perumusan masalah, batasan masalah, tujuan penelitian, manfaat penelitian, metodologi penelitian dan sistematika penulisan.

BAB 2 : DASAR TEORI

Bab 2 berisi tentang teori-teori yang digunakan sebagai landasan dalam penelitian dan pengertian dari pembahasan dari judul yang digunakan.

BAB 3 : ANALISA DAN PERANCANGAN SISTEM

Bab 3 membahas langkah dari proses pembuatan aplikasi untuk membandingkan metode dalam mereduksi *noise* citra USG dengan menggunakan metode *gaussian*, *median* dan *mean* pada operasi *morfologi*.

BAB 4 : IMPLEMENTASI DAN PENGUJIAN SISTEM

Bab 4 menunjukkan hasil pengujian dari sistem perbandingan reduksi *noise* citra USG yang dibuat disertai dengan analisa sehingga didapatkan bukti kuat dari hipotesis yang dilakukan.

BAB 5 : KESIMPULAN DAN SARAN

Bab 5 menyimpulkan apa yang ada pada bab-bab terdahulu serta memberikan saran atas penulisan skripsi ini.

BAB 2 DASAR TEORI

2.1 Pengertian Citra

Citra adalah suatu *representasi* (gambaran), kemiripan atau imitasi dari suatu objek. Citra sebagai keluaran suatu sistem perekaman data bersifat optik berupa foto, bersifat analog berupa sinyal-sinyal video seperti gambar pada *monitor* atau bersifat digital yang dapat langsung disimpan pada suatu media penyimpanan. Citra sebagai salah satu komponen multimedia memegang satu peranan sangat penting sebagai bentuk informasi visual. Citra memiliki karakteristik yang tidak dimiliki oleh data teks, yaitu citra kaya dengan informasi. Meskipun sebuah citra kaya akan informasi, namun seringkali citra yang kita miliki mengalami penurunan mutu (*degradasi*) citra yaitu penurunan kualitas citra, misalnya karena mengandung cacat atau derau (*noise*), warnanya terlalu kontras, kurang tajam, kabur (*blurring*), dan sebagainya (Sutoyo dkk, 2009).

Kemudian dalam jurnal lain, mengemukakan pendapat bahwa citra merupakan fungsi menerus (*continue*) atas intensitas cahaya pada bidang dua dimensi. Sebuah citra mengandung informasi tentang objek yang direpresentasikan. Citra dapat juga diartikan sebagai gambar 2 dimensi yang dihasilkan dari gambar analog dua dimensi yang kontinu menjadi gambar diskrit melalui proses sampling. (sriani, dkk. 2017).

Sebuah citra terdiri dari M baris dan N kolom, dimana disetiap pasangan indeks baris dan kolom menyatakan suatu titik pada citra. Nilai matriksnya menyatakan nilai kecerahan titik tersebut. Titik – titik tersebut dinamakan sebagai elemen citra, atau *pixel* (*picture elemen*). Dalam buku lain sebuah citra dapat didefinisikan sebagai fungsi dua dimensi $f(x,y)$ dimana x dan y adalah koordinat spasial dan amplitude dari f pada sembarang pasang koordinat (x,y) disebut intensitas atau level keabuan dari citra pada titik tersebut (Prasetyo, 2011).

2.1.1 Jenis-Jenis Citra

Jenis-jenis citra terbagi menjadi dua yaitu citra analog dan citra digital. Citra analog merupakan citra yang bersifat kontinu dan citra yang tidak bisa diolah, sedangkan citra digital merupakan citra yang bisa langsung diolah.

1. Citra Analog

Citra analog adalah citra yang bersifat kontinu, seperti gambar pada *monitor*, foto sinar X, foto yang tercetak di kertas foto, lukisan, pemandangan alam, hasil CT *scan*, gambar-gambar yang terekam pada pita kaset dan lain sebagainya. Citra Analog tidak dapat direpresentasikan dalam komputer sehingga tidak dapat diproses di komputer secara langsung. Oleh sebab itu, agar citra ini dapat diproses di komputer, proses konversi analog ke digital harus dilakukan terlebih dahulu. Citra analog dihasilkan dari alat-alat analog seperti video kamera analog, kamera foto analog, WebCam, CT scan sensor rontgen untuk foto *throat*, sensor gelombang pendek pada sistem radar, sensor ultrasound pada sistem USG dan lain-lain (Sutoyo dkk, 2009).

2. Citra Digital

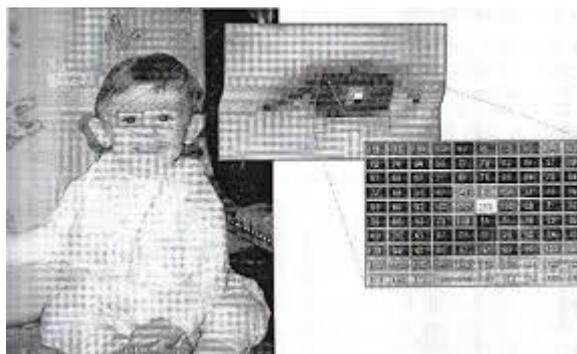
Citra digital adalah citra yang dapat diolah oleh komputer. Citra digital dibentuk oleh kumpulan titik-titik yang dinamakan piksel (*pixel* atau *picture element*). Setiap piksel digambarkan sebagai satu kotak kecil. Setiap piksel mempunyai koordinat posisi. Setiap koordinat yang dipakai untuk menyatakan citra digital (Kadir dan Susanto, 2013). Secara umum, pengolahan citra menunjukkan pada pemrosesan gambar 2 dimensi menggunakan komputer. Dalam konteks yang lebih luas, pengolahan citra digital mengacu pada pemrosesan setiap data 2 dimensi. Citra digital merupakan sebuah larik (*array*) yang berisi nilai-nilai real maupun kompleks yang direpresentasikan dengan deretan bit tertentu. Suatu citra dapat didefinisikan sebagai fungsi $f(x,y)$ berukuran M baris dan N kolom, dengan x dan y adalah koordinat spasial, dan amplitudo f di titik koordinat (x,y) dinamakan intensitas atau tingkat keabuaan dari citra pada titik tersebut. Apabila nilai x,y dan nilai amplitudo f secara keseluruhan berhingga (*finite*) dan bernilai diskrit maka dapat dikatakan bahwa citra tersebut adalah citra digital (Putra, 2009). Sebuah citra digital berdimensi M x N dapat dipresentasikan dalam matriks sebagai berikut.

$$f(x,y) \approx \begin{bmatrix} f(0,0) & f(0,1) & \dots & f(0,M-1) \\ f(1,0) & f(1,1) & \dots & f(1,M-1) \\ \vdots & \vdots & \vdots & \vdots \\ f(N-1,0) & f(N-1,1) & \dots & f(N-1,M-1) \end{bmatrix}$$

Gambar 2.1 Matriks Citra Digital Berdimensi M x N

Sumber: Putra, 2009

Pada proses *digitalisasi* (sampling dan kuantitas) diperoleh besar baris M dan kolom N hingga citra membentuk matriks $M \times N$ dan jumlah tingkat keabuaan piksel G . Ada dua jenis format citra digital yang sering digunakan yaitu citra digital *bitmap* dan citra digital *vektor*. Citra digital *bitmap* disimpan sebagai array yang berisi nilai piksel. Sedangkan citra digital *vektor* disimpan sebagai dekripsi matematis komponen penyusunannya seperti titik, garis, kurva dan bidang. Gambar dibawah ini menunjukkan posisi koordinat citra digital.



Gambar 2.2 Ilustrasi Digitalisasi Citra

Sumber: Putra, 2009

Matriks adalah kumpulan bilangan-bilangan yang disusun dalam larik baris dan kolom. Umumnya matriks diberi notasi huruf kapital. Jika matriks A terdiri dari m baris dan n kolom (sering disebut sebagai ordo $m \times n$), maka dapat ditulis sebagai $A_{m \times n}$.

2.1.2 Format File Citra

Sebuah format file citra harus dapat menyatukan kualitas citra, ukuran file dan kompatibilitas dengan berbagai aplikasi. Format file citra standar yang digunakan saat ini terdiri dari beberapa jenis. Format-format ini digunakan untuk menyimpan citra dalam sebuah file. Setiap format memiliki karakteristik masing-masing. Ini adalah contoh format umum yaitu, *bitmap* (.bmp), *tagged image format* (.tif, .tiff), *Portable Network Graphics* (.png), *JPEG* (.jpg) [30].

Bahkan menurut Sutoyo *et al.* (2009), ada dua jenis format file citra yang sering digunakan dalam pengolahan citra, yaitu citra *bitmap* dan citra *vektor*. Pada citra *bitmap* ini sering disebut juga citra *raster*. Citra *bitmap* ini menyimpan data kode citra secara digital dan lengkap (cara penyimpanannya adalah per piksel). Citra *bitmap* ini dipresentasikan dalam bentuk matriks atau dipetakan dengan menggunakan bilangan biner atau sistem bilangan yang lain. Citra ini memiliki kelebihan untuk memanipulasi warna,

tetapi untuk mengubah objek lebih sulit. Tampilan *bitmap* mampu menunjukkan kehalusan gradasi bayangan dan warna dari sebuah gambar. Tetapi bila tampilan diperbesar maka tampilan di monitor akan tampak pecah-pecah (kualitas citra menurun). Contoh format file citra antara lain adalah *JPG, PNG, BMP, GIFF, TIF, WPG, IMG*.

2.2 Jenis Citra Digital

Ada tiga jenis citra digital yang digunakan dalam memproses citra. Ketiga jenis citra tersebut yaitu citra warna, citra *grayscale* (keabuan) dan citra biner.

1. Citra Warna

Citra berwarna merupakan jenis citra yang menyajikan tiga komponen warna (RGB = *Red, Green, Blue*). Setiap komponen warna menggunakan 8 *bit* (nilainya berkisar antara 0 sampai dengan 255). Dengan demikian, kemungkinan warna yang dapat disajikan mencapai $255 \times 255 \times 255$ atau 16.581.375 warna. Tabel 2.1 menunjukkan contoh warna dan nilai RGB.

Tabel 2.1 Warna dan Nilai Penyusun Warna

Warna	R	G	B
Merah	255	0	0
Hijau	0	255	0
Biru	0	0	255
Hitam	0	0	0
Putih	255	255	255
Kuning	0	255	255

(Sumber : Nafi'iyah, 2015)

Dalam buku lain, setiap piksel pada citra warna mewakili warna yang merupakan kombinasi dari tiga warna dasar (RGB = *Red, Green, Blue*). Setiap warna dasar menggunakan penyimpanan 8 *bit* = 1 *byte*, yang artinya setiap warna mempunyai gradasi sebanyak 255 warna. Berarti setiap piksel mempunyai kombinasi warna sebanyak $2^8 \cdot 2^8 \cdot 2^8 = 2^{24} = 16$ juta warna lebih. Itulah sebabnya format ini dinamakan *true color* karena mempunyai jumlah warna yang cukup besar sehingga bisa dikatakan hampir mencakup semua warna di alam. Penyimpanan citra warna di dalam memori berbeda dengan citra *grayscale*. Setiap piksel dari citra *grayscale* 256 gradasi warna diwakili oleh 1 *byte*. Sedangkan 1 citra *true color* diwakili oleh 3 *byte*, dimana masing-masing *byte* merepresentasikan warna merah (*Red*), hijau (*Green*), dan biru (*Blue*)

(Sutoyo dkk, 2009). Gambar 2.2 adalah contoh penyimpanan citra warna di dalam memori.

Citra warna					Penyimpanan di dalam memori				
					R = 50	R = 40	R = 90	R = 80	R = 70
					G = 65	G = 40	G = 90	G = 50	G = 70
					B = 50	B = 55	B = 90	B = 50	B = 70
					R = 40	R = 50	R = 40	R = 20	R = 50
					G = 80	G = 80	G = 90	G = 20	G = 60
					B = 30	B = 50	B = 80	B = 50	B = 70
					R = 80	R = 70	R = 80	R = 10	R = 90
					G = 60	G = 70	G = 90	G = 70	G = 90
					B = 40	B = 70	B = 70	B = 10	B = 90
					R = 90	R = 40	R = 70	R = 60	R = 50
					G = 60	G = 60	G = 70	G = 20	G = 80
					B = 70	B = 50	B = 70	B = 40	B = 50
					R = 60	R = 40	R = 80	R = 70	R = 90
					G = 60	G = 60	G = 80	G = 60	G = 80
					B = 60	B = 80	B = 80	B = 50	B = 70

Gambar 2.3 Contoh penyimpanan citra warna di dalam memori

Sumber: Sutoyo dkk, 2009

2. Citra *Grayscale*

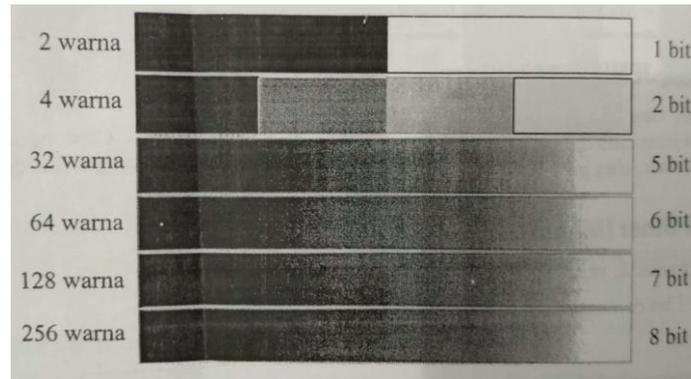
Citra *grayscale* merupakan citra digital yang hanya memiliki satu nilai kanal pada *pixel*nya, dengan kata lain nilai bagian RGB. Nilai tersebut digunakan untuk menunjukkan nilai intensitas. Warna yang dimiliki adalah warna dari hitam, keabuan, dan putih. Citra *grayscale* memiliki kedalaman warna 8 *bit* (256 kombinasi warna keabuan (Putra, 2010). Gambar 2.3 menunjukkan citra *grayscale*.



Gambar 2.4 Citra *grayscale*

Sumber : Putra, 2009

Banyaknya warna dalam citra ini tergantung pada jumlah *bit* yang disediakan di memori untuk menampung kebutuhan warna ini. Semakin besar jumlah *bit* warna yang disediakan di memori, semakin halus gradasi warna yang terbentuk. Gambar 2.4 menunjukkan perbandingan gradasi warna untuk jumlah *bit* tertentu (Sutoyo dkk, 2009).



Gambar 2.5 Perbandingan gradasi warna 1 bit, 2 bit, 5 bit, 6 bit, 7 bit, 8 bit
Sumber : Sutoyo dkk, 2009

3. Citra Biner

Citra biner adalah citra digital yang hanya memiliki dua kemungkinan nilai pixel yaitu hitam dan putih. Hanya dibutuhkan 1 bit untuk mewakili nilai setiap *pixel* dari citra biner. Citra biner sering kali muncul sebagai hasil dari proses pengolahan seperti segmentasi, pengambangan, morfologi, ataupun *dithering* (Putra, 2009). Citra biner adalah citra dengan setiap *pixel* hanya dinyatakan dengan sebuah nilai dari dua kemungkinan yaitu nilai 1 dan 0. Nilai 0 menyatakan warna hitam dan nilai 1 menyatakan warna putih. Gambar 2.5 menunjukkan citra biner.



Gambar 2.6 Citra biner
Sumber : Putra, 2009

2.3 Pengolahan Citra Digital

Pengolahan citra digital dapat dinyatakan sebagai pemrosesan gambar berdimensi-dua melalui komputer digital. Pengolahan citra adalah istilah umum untuk berbagai teknik yang keberadaannya untuk memanipulasi dan memodifikasi citra dengan berbagai cara.

Pengolahan citra merupakan bagian yang mendasari berbagai aplikasi nyata, seperti pengenalan pola, penginderaan jarak jauh melalui satelit atau pesawat udara dan *mechine vision*. (Kadir dan Susanto, 2013).

Pengolahan citra merupakan suatu usaha untuk mengubah citra menjadi citra lain yang lebih sempurna atau *output* yang dihasilkan sesuai dengan keinginan (Sulistiyanti, dkk. 2016). Dalam penelitian lain juga dijelaskan pengolahan citra merupakan pemrosesan citra, khususnya menggunakan komputer untuk menghasilkan citra dari manipulasi citra sebelumnya, sehingga citra tersebut lebih mudah untuk dipresentasikan baik oleh manusia maupun mesin (Sriani, dkk. 2017). Secara umum pengolahan citra digital menunjuk pada pemrosesan gambar dua dimensi menggunakan komputer.

Pengolahan citra digital adalah sebuah disiplin ilmu yang mempelajari hal – hal yang berkaitan dengan perbaikan kualitas gambar (peningkatan kontras, restorasi citra) tranformasi gambar (rotasi, translasi dan skala) melakukan pemilihan citra ciri yang optimal untuk tujuan analisis, melakukan penarikan informasi atau deskripsi objek atau pengenalan objek yang terkandung pada citra, melakukan kompresi atau reduksi data untuk tujuan penyimpanan data, transmisi data, dan waktu proses data. *input* dari pengolahan citra adalah citra, sedangkan *ouput*-nya adalah citra hasil pengolahan (sutoyo dkk, 2009).

Kebanyakan citra belum sesuai dengan hasil yang diharapkan. Hal ini dapat terjadi karena beberapa kemungkinan, seperti adanya noise, adanya kabut yang menghalangi objek yang sedang pengambilan gambar, lensa kamera kotor dan sebagainya. Oleh karena itu pengolahan citra sangat diperlukan. Disiplin ilmu yang melahirkan teknik-teknik untuk pengolahan citra dinamakan pengolaha citra digital.

2.3.1 Pemrosesan Citra Digital

Pemrosesan citra digital melibatkan suatu pengolahan yang disebut *filter*. Citra asli didapatkan dari konvolusi antara asal dengan sebuah kernel filter. Filter adalah alat untuk memproses data yang mempunyai ciri mengambil data asli untuk memproduksi dari hasil sebagaimana yang diinginkan. Dalam pengolahan citra, respon perambatan filter memberikan gambaran bagaimana piksel-piksel pada citra diproses. (Jannah, 2008). Konvolusi adalah suatu proses perkalian dari dua buah fungsi $f(x)$ dan $g(x)$, yang mengalikan sebuah citra dengan sebuah *maks* atau *karnel*. Untuk fungsi diskrit konvolusi didefinisikan sebagai berikut:

$$h(x) = f(x) * g(x) = \sum_{x=-n}^n \sum_{y=-N}^N f(u, v)g(x + u, y + u)$$

Dalam hal ini, $h(x)$ menyatakan *Linier position invariant operator*. $f(x)$ menyatakan citra asli, $g(x)$ menyatakan citra hasil konvolusi dan x, y, u dan v menyatakan posisi titik dalam citra.

2.3.2 Noise

Setiap gangguan pada citra dinamakan *noise*. *Noise* adalah komponen di citra yang tidak dikehendaki karna mengganggu sebuah citra gambar (Khilmawan & Riadi, 2018). Dalam praktik, kehadiran *noise* tidak dapat dihindari. Sebagai contoh, *noise Gaussian* biasa muncul pada sembarang isyarat. *White noise* (derau putih) biasa menyertai pada siaran televise yang berasal dari stasiun pemancar yang lemah. *Noise* butiran biasa muncul dalam film-film fotografi. *Noise* yang dinamakan garam dan merica (*salt and pappers*) sering mewarnai citra. Noise dapat dikelompokkan menjadi 4 kelas (kadir dan susanto, 2013).

1. Additive Noise (Derau Tambahan)

Noise ini biasa muncul karena sensor yang bekerja tidak sempurna dan memberikan isyarat tambahan terdistribusi *Gaussian*, yang tidak bergantung pada isyarat asli. Isyarat yang menghasilkan dapat dinyatakan secara matematis seperti berikut :

$$g(.) = f(.) + d(.)..... (2.1)$$

Dalam hal ini, $g(.)$ menyatakan isyarat telah terkena *noise*, $f(.)$ menyatakan citra asli dan $d(.)$ menyatakan (*Noise*).

2. Multiplicative Noise (Derau Perkalian)

Noise perkalian biasa terjadi pada film fotografi. *Noise* biasanya disebut sebagai *speckle noise* (derau bercak). Secara matematis, isyarat yang terkena *noise* perkalian dapat ditulis sebagai berikut :

$$g(.) = f(.) * d(.).....(2.2)$$

Dalam hal ini, $g(.)$ menyatakan isyarat telah terkena *noise*, $f(.)$ menyatakan citra asli dan $d(.)$ menyatakan (*Noise*).

3. *Impulse Noise* (Derau Implus)

Sensor ataupun saluran data terkadang memberikan *noise* berbentuk biner (0 atau 1). *Noise* seperti itu dimodelkan sebagai berikut :

$$g(.) = (1-p) * f(.) + P * d.....(2.3)$$

Dalam hal ini, $g(.)$ menyatakan isyarat yang telah terkena *noise* $f(.)$ menyatakan citra asli, $d(.)$ menyatakan *noise*, dan p menyatakan parameter biner yang nilainya berupa 0 atau 1. Berdasarkan rumus diatas, isyarat asli akan hilang saat p bernilai 1.

4. *Quantization Noise* (Derau Kuantitas)

Noise kuantitas termasuk sebagai *noise* yang bergantung pada isyarat. Hal ini terjadi saat kuantitas terhadap isyarat dilakukan sebelum dikonversi menjadi isyarat digital. *Noise* ini dapat mengakibatkan detail citra hilang.

2.4 Gaussian Filter

Gaussian Filter adalah salah satu filter linier dengan nilai pembobotan untuk setiap anggotanya dipilih berdasarkan bentuk fungsi *Gaussian* (Afifa, 2016). *Gaussian filter* dipilih sebagai *filter* penghalusan berdasarkan pertimbangan bahwa *filter* ini mempunyai pusat kernel.

Menurut Usman Ahmad *Gaussian filter* sangat baik untuk menghilangkan *noise* yang bersifat sebaran normal, yang banyak dijumpai pada sebaran citra hasil proses digitasi menggunakan kamera karena merupakan fenomena alamiah akibat sifat pantulan cahaya dan kepekaan sensor cahaya pada kamera itu sendiri (Wedianto dkk, 2016). Untuk menghitung atau menentukan nilai-nilai setiap elemen dalam *Gaussian filter* dinyatakan oleh persamaan berikut, (Wedianto dkk, 2016):

$$\text{Pixel } B(i,j) = \frac{1}{K} \cdot \sum_{p=0}^{N-1} \left(\sum_{q=0}^{M-1} G(p,q) \cdot \text{Pixel } A \left(i + p - \frac{(N-1)}{2}, j + q - \frac{(M-1)}{2} \right) \right) \dots\dots\dots(2.4)$$

Berdasarkan pada rumus di atas, dapat dijelaskan bahwa:

K = Total jumlah kernel gaussian

p, q = kordinat citra kernel gaussian

i, j = koordinat citra asli

N = Jumlah kolom matrik kernel gaussian

M = Jumlah baris matrik kernel gaussian

Dapun kernel *gaussian* dengan sigma 1 sebagai berikut, (Wedianto dkk, 2016) :

1	2	1
2	4	2
1	2	1

Adapun sampel hitungan manual proses gaussian dilakukan pada citra bagian dalam tanpa merubah nilai citra bagian luar. Berikut adalah proses gaussian pertama pada citra 5x5 :

3	76	244	215	224
3	76	243	213	220
3	76	242	212	220
2	75	241	211	218
2	74	239	209	217

$$\frac{1}{16} \left(\left(3 + 1 - \frac{(3-1)}{2} \right) + \left(76 + 2 - \frac{(3-1)}{2} \right) + \left(244 + 1 - \frac{(3-1)}{2} \right) + \left(3 + 2 - \frac{(3-1)}{2} \right) + \right. \\ \left. \left(76 + 4 - \frac{(3-1)}{2} \right) + \left(243 + 2 - \frac{(3-1)}{2} \right) + \left(3 + 1 - \frac{(3-1)}{2} \right) + \left(76 + 2 - \frac{(3-1)}{2} \right) + \right. \\ \left. \left(242 + 1 - \frac{(3-1)}{2} \right) \right) = 60,8$$

Berdasarkan pada proses pertama, didapatkan nilai pixel citra baru gaussian yang mengisi bagian dalam citra sebagai berikut :

3	76	244	215	224
3	60,8			220
3				220
2				218
2	74	239	209	217

2.5 Median Filter

Metode *median filter* merupakan filter *no-linear*, dikatakan *non-linear* karena cara bekerja metode ini tidak termasuk dalam kategori operasi kovolusi (Yusro & Sianturi, 2018). Untuk *median filtering* ini, data yang digunakan untuk menghitung median terdiri dari kumpulan data yang ganjil dengan kernel 3x3. Hal ini disebabkan dengan jumlah data yang ganjil maka piksel yang akan diproses dapat berada ditengah.

Pada *median filtering* digunakan matrik berdimensi N X N. Dari matrik tersebut, kemudian data yang ada diurutkan dan dimasukkan dalam sebuah matrik berukuran 1X (N X N). Hal ini berguna untuk mempermudah menemukan *median* dari kumpulan data yang telah urut tersebut. Dapun sebagai contoh perhitungan

$$f(x,y) =$$

5	3	3	0	4	4	0	5	2	2
4	2	1	3	4	0	5	1	3	3
6	3	0	1	6	2	3	0	7	0
7	4	0	1	0	2	3	2	7	0
7	4	5	1	0	6	3	2	7	0
7	4	5	5	7	7	6	2	6	4
6	0	1	4	7	0	7	2	0	2
6	5	1	3	2	4	4	1	0	0

Berdasarkan pada matrik di atas, kemudian nilai matrik kernel 3x3 yang dipilih diurutkan dari terendah hingga tertinggi seperti berikut :

In-0	In-1	In-2	In-3	In-4	In-5	In-6	In-7	In-8
0	1	2	3	3	4	5	5	6

Berdasarkan dari nilai yang telah diurutkan tersebut kemudian ditentukan nilai tengahnya, yaitu indeks ke-4 yang bernilai 3. Maka nilai mediannya adalah 3.

2.6 Mean Filter

Pada filter rata-rata, nilai intensitas setiap piksel diganti dengan rata-rata dari nilai intensitas piksel tersebut dengan piksel-piksel tetangganya. Filter ini biasa disebut sebagai *lowpass filters* (Riandari, 2018). Jumlah tetangga yang dilibatkan tergantung pada *filter* yang dirancang secara matematis *filter* rata-rata berukuran $m \times n$ dinyatakan oleh persamaan berikut :

$$g(x) = \frac{1}{m.n}, 1 \leq x \leq m, 1 \leq y \leq n \dots \dots \dots (2.5)$$

Menurut Usman Ahmad salah satu filter linier adalah filter rata-rata dari intensitas pada beberapa piksel lokal dimana setiap piksel akan digantikan nilainya dengan rata-rata dari intensitas piksel tersebut dengan piksel-piksel tetangganya, dan jumlah piksel tetangga yang dilibatkan tergantung pada filter yang dirancang (Wedianto, 2016).

Cara kerja filter rata-rata sama dengan konvolusi. Filter ini biasanya digunakan untuk mengaburkan citra dan mereduksi *noise*. Sebuah citra keabuan berukuran 10x8 mempunyai 8 skala keabuan dan sebuah filter rata-rata $g(x,y)$ yang berukuran 3x3.

$$f(x,y) = \begin{array}{|c|c|c|c|c|c|c|c|c|c|} \hline 5 & 3 & 3 & 0 & 4 & 4 & 0 & 5 & 2 & 2 \\ \hline 4 & 2 & 1 & 3 & 4 & 0 & 5 & 1 & 3 & 3 \\ \hline 6 & 3 & 0 & 1 & 6 & 2 & 3 & 0 & 7 & 0 \\ \hline 7 & 4 & 0 & 1 & 0 & 2 & 3 & 2 & 7 & 0 \\ \hline 7 & 4 & 5 & 1 & 0 & 6 & 3 & 2 & 7 & 0 \\ \hline 7 & 4 & 5 & 5 & 7 & 7 & 6 & 2 & 6 & 4 \\ \hline 6 & 0 & 1 & 4 & 7 & 0 & 7 & 2 & 0 & 2 \\ \hline 6 & 5 & 1 & 3 & 2 & 4 & 4 & 1 & 0 & 0 \\ \hline \end{array}$$

$$g(x,y) = \frac{1}{9} \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix}$$

Hasil konvolusi $h(x,y)$ dihitung dengan pilih $f(x,y)$ ukuran 3×3 , dimulai dari pojok kiri atas. Kemudian, hitung konvolusinya dengan filter $g(x,y)$.

5	3	3	0	4	4	0	5	2	2
4	2	1	3	4	0	5	1	3	3
6	3	0	1	6	2	3	0	7	0
7	4	0	1	0	2	3	2	7	0
7	4	5	1	0	6	3	2	7	0
7	4	5	5	7	7	6	2	6	4
6	0	1	4	7	0	7	2	0	2
6	5	1	3	2	4	4	1	0	0

Hasil konvolusi $h(x,y)$ adalah :

$$mean = \frac{5 + 3 + 3 + 4 + 2 + 1 + 6 + 3 + 0}{9} = 3$$

diganti dengan 3, tempatkan pada matriks yang baru, hasilnya adalah :

5	3	3	0	4	4	0	5	2	2
4	3								3
6									0
7									0
7									0
7									4
6									2
6	5	1	3	2	4	4	1	0	0

2.7 Unified Modeling Language (UML)

Unified Modeling Language (UML) adalah alah satu standar bahasa yang banyak digunakan didunia industri untuk mendefinisikan requirement, membuat analisis dan

desain, serta menggambarkan arsitektur dalam pemrograman berorientasikan objek (Rosa & Shalahuddin, 2015). Standar spesifikasi UML dijadikan standar defakto oleh OMG (*Object Management Group*) sepanjang tahun 1996. UML yang berorientasikan object mempunyai beberapa notasi standar. Spesifikasi ini menjadi populer dan standar karena sebelum adanya UML, telah ada berbagai macam spesifikasi yang berbeda. Hal ini menyulitkan komunikasi antar pengembang perangkat lunak. Untuk itu beberapa pengembang spesifikasi yang sangat berpengaruh berkumpul untuk membuat standar baru.

Unified Modeling Language (UML) memiliki beberapa tujuan (Rosa & Shalahuddin, 2015), sebagai berikut:

1. Dapat memberikan bahasa pemodelan visual kepada pengguna dari berbagai macam pemrograman maupun proses rekayasa.
2. Dapat menyatukan praktek-praktek terbaik yang ada dalam pemodelan.
3. Dapat menciptakan bahasa pemodelan yang dapat dipergunakan oleh manusia.

Unified Modelling Language (UML) sendiri terdiri atas pengelompokan diagram-diagram sistem menurut aspek sudut pandang tertentu. *Unified Modelling Language* (UML) mempunyai 7 diagram yang umum digunakan (Rosa & Shalahuddin, 2015), yaitu :

Tabel 2.2 Diagram dalam UML

<i>No</i>	<i>Diagram</i>	<i>Main Concepts</i>
1	<i>Use Case</i>	<i>Use case, actor, association, extend, include</i>
2	<i>Class</i>	<i>Class, association, interface</i>
3	<i>Package</i>	<i>Component, realization</i>
4	<i>Sequence</i>	<i>Interaction, object, messege</i>
5	<i>Collaboration</i>	<i>Collaboration role, messege</i>
6	<i>StateChart</i>	<i>State, event, transition</i>
7	<i>Activity</i>	<i>Completion transition, fork, join</i>

(Sumber: Rosa & Shalahuddin, 2015)

2.7.1 Use Case Diagram

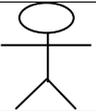
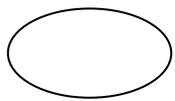
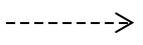
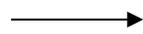
Use case atau diagram *use case* merupakan pemodelan untuk melakukan (*behavior*) sistem informasi yang akan dibuat (Rosa & Shalahuddin, 2015). *Use case* mendeskripsikan sebuah interaksi antara satu atau lebih aktor dengan sistem informasi yang akan dibuat. Secara kasar, *use case* digunakan untuk mengetahui fungsi apa saja yang ada didalam

sebuah sistem informasi dan siapa saja yang berhak menggunakan fungsifungsi itu (Rosa & Shalahuddin, 2015). *Use case* menyediakan cara untuk mendapatkan pemahaman dari apa yang diharapkan dan dibutuhkan oleh pengguna kelak.

Syarat penamaan pada *use case* adalah nama didefinisikan sesimpel mungkin dan dapat dipahami. Ada dua hal utama pada *use case* yaitu pendefinisian apa yang disebut aktor dan *use case* (Rosa & Shalahuddin, 2015).

1. Aktor merupakan orang, proses, atau sistem lain yang berinteraksi dengan sistem informasi yang akan di buat diluar sistem informasi yang akan dibuat itu sendiri, jadi walaupun simbol dari aktor adalah gambar orang, tapi aktor belum tentu orang.
2. *Use case* merupakan fungsionalitas yang disediakan sistem sebagai unit-unit yang saling bertukar pesan antar unit atau aktor.

Tabel 2.3 Simbol *Use Case* Diagram

No	Simbol	Nama	Keterangan
1.		<i>Actor</i>	Orang, proses, atau sistem lain yang berinteraksi dengan sistem
2.		<i>Use case</i>	Fungsionalitas yang disediakan sistem sebagai unit-unit yang saling bertukar pesan antar unit atau factor.
3.		<i>Undirectional Association</i>	Komunikasi antar actor dan <i>use case</i> yang berpartisipasi pada <i>use case</i> atau <i>use case</i> memiliki interaksi dengan aktor.
4.		Ekstensi (<i>Extend</i>)	Relasi <i>use case</i> tambahan ke sebuah <i>use case</i> dimana <i>use case</i> yang ditambahkan dapat berdiri sendiri walau tanpa <i>use case</i> tambahan itu mirip.
5.		Generalisasi/ <i>Generalization</i>	Hubungan generalisasi dan spesialisasi antara dua buah <i>use case</i> .

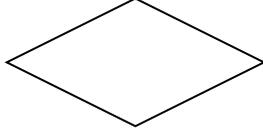
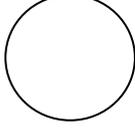
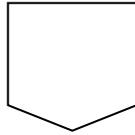
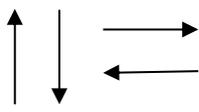
(Sumber: Rosa & Shalahuddin, 2015)

2.8 Flowchart

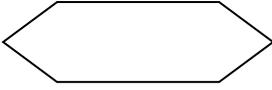
Flowchart dapat diartikan sebagai suatu sarana yang menunjukkan langkah-langkah yang harus dilaksanakan dalam menyelesaikan suatu masalah untuk komputasi dengan cara mengekspresikannya ke dalam serangkaian simbol-simbol grafis khusus.

Dalam jurnal lain, *flowchart* merupakan bagan yang menjelaskan tentang prosedur atau langkah-langkah dalam menyelesaikan suatu masalah dari suatu program (Santoso dan Nurmalina, 2017).

Tabel 2.4 Simbol-simbol *flowchart*

Simbol	Nama	Fungsi
	Proses Terdefenisi	Simbol proses terdefenisi untuk permulaan sub program
	Keputusan	Simbol keputusan untuk perbandingan, pernyataan, penyeleksian data yang memberikan pilihan untuk langkah selanjutnya
	Proses Terdefenisi	Simbol proses terdefenisi untuk permulaan sub program
	Penghubung	Simbol penghubung untuk menghubungkan bagian-bagian <i>flowchart</i> yang berada pada satu halaman
	Penghubung	Simbol penghubung untuk menghubungkan bagian-bagian <i>flowchart</i> yang berada pada halaman yang berbeda
	<i>Terminator</i>	Simbol terminal digunakan untuk menyatakan awal dan akhir suatu program
	Garis Alir	Simbol garis alir untuk menunjukkan arah arus dari sebuah proses

Tabel 2.4 Simbol-simbol *flowchart* (Lanjutan)

Simbol	Nama	Fungsi
	Persiapan	Simbol persiapan untuk proses inisialisasi atau pemberian nilai awal
	Proses	Simbol proses untuk proses pengolahan data
	<i>Input/Output</i>	Simbol <i>Input/Output</i> untuk proses <i>input</i> dan <i>output</i> data

(Sumber : Santoso dan Nurmalina, 2017)

2.9 MATLAB

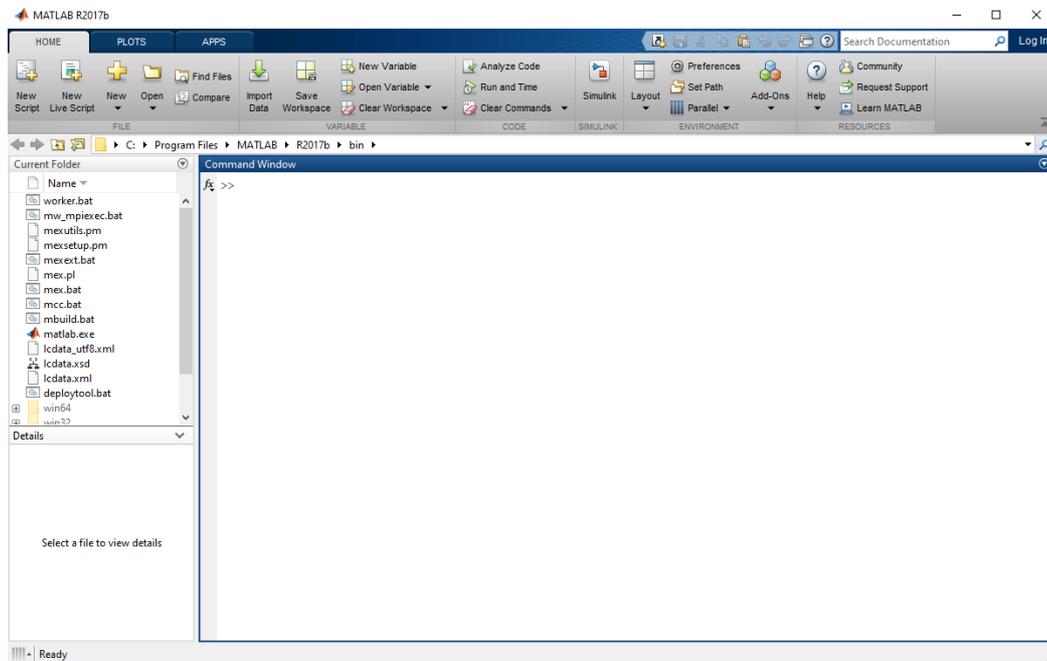
MATLAB adalah singkatan dari *MATrix LABORatory*. Pertama kali dibuat untuk mempermudah penggunaan dua koleksi subrutin pada pustaka FORTRAN yaitu: LINPACK dan EISPACK, dalam menangani komputasi matriks. Sejak itu, MATLAB berkembang menjadi sebuah sistem yang interaktif sekaligus sebagai bahasa pemrograman untuk keperluan-keperluan ilmiah, komputasi teknis, dan visualisasi. Elemen data dasar MATLAB adalah matriks. Perintah-perintah diekspresikan dalam bentuk yang sangat mirip dengan bentuk yang digunakan dalam matematika dan bidang teknik. Contoh persamaan $b=Ax$, dengan A , b , dan x matriks, ditulis: $b=A*x$. Untuk mendapat solusi x dari A dan b , tulis: $x=A\b$. Tidak diperlukan penulisan program khusus untuk operasi-operasi matriks seperti perkalian matriks atau invers matriks. Oleh karena itu bahasa MATLAB menyelesaikan masalah tersebut memerlukan waktu lebih cepat dibanding waktu yang dibutuhkan bahasa pemrograman tingkat tinggi lain (Pahlawan, dkk, 2020).

MATLAB biasanya digunakan dalam :

1. Pengembangan Algoritma matematika dan komputasi
2. Pemodelan, simulasi, dan pembuatan *prototype* dari penerimaan data
3. Analisa, eksplorasi, dan visualisasi data
4. *Scientific* dan *engineering*
5. Pengembangan aplikasi berbasis grafik dan pembuatan *Graphical User*

Fungsi-fungsi yang digunakan MATLAB:

1. Fungsi matematika
2. Fungsi fisika
3. Fungsi statistik
4. Fungsi visualisasi.



Gambar 2.7 Tampilan *State Awal* Matlab

BAB 3 ANALISA DAN PERACANGAN SISTEM

3.1 Analisa

Pada penelitian ini, pembahasan terdiri dari analisis data, representasi data proses penyisipan data citra ronsen yang memiliki noise, kemudian dilakukan perbandingan metode *filter gaussian*, *mean* dan *median*. Perbandingan dilakukan dengan menghitung nilai MSE dan PNSR citra hasil reduksi *noise*.

3.1.1 Analisis Data

Sebelum dilakukanya proses perhitungan manual dalam membandingkan metode *filter gaussian*, *mean* dan *median*, terlebih dahulu menentukan data citra yang akan direduksi noisenya. Adapun citra yang akan direduksi adalah sebuah citra Ultrasonography sampel seperti berikut :



Gambar 3.1 Sampel Citra Ultrasonography

Berdasarkan pada gambar di atas, data citra adalah 800 x 600 piksel. Untuk memudahkan hitungan manual dalam membandingkan proses metode *filter gaussian*, *mean* dan *median*, maka diambil sampel 5x5 citra ultrasonography sebagai berikut :

	1	2	3	4	5	6	7
96	3	76	244	215	224	224	2
97	3	76	243	213	220	219	2
98	3	76	242	212	220	218	2
99	2	75	241	211	218	217	2
100	2	74	239	209	217	216	2
101	2	73	238	208	216	214	2
102	1	72	236	206	214	213	2
103	1	71	235	205	213	212	2
104	1	70	234	204	213	211	2
105	1	75	226	204	207	209	2
106	1	74	225	203	206	208	2
107	1	74	224	201	204	207	2
108	1	73	223	199	203	205	2



Gambar 3.2 Nilai Piksel Citra Ultrasonography 800 x 600

Berdasarkan pada gambar 3.2, diambil nilai matrik citra ultrasonography sebanyak 5x5 piksel yang digunakan sebagai contoh perhitungan manual metode *filtering*. Nilai matrik

piksel ultrasonography didapatkan dari aplikasi matlab. Adapun nilai piksel sampel ultrasonography 5x5 dapat dilihat pada tabel di bawah ini:

Tabel 3.1 Nilai Piksel Citra Ultrasonography 5x5

X,y	0	1	2	3	4
0	3	76	244	215	224
1	3	76	243	213	220
2	3	76	242	212	220
3	2	75	241	211	218
4	2	74	239	209	217

Nilai pada tabel di atas, akan diproses dengan metode *filtering gaussian*, *mean* dan *median* dengan kernel 3x3.

3.1.2 Penerapan Metode

Setelah didapatkan nilai piksel citra sampel 5x5, kemudian dilakukan *filtering* citra dengan metode *gaussian*, *mean* dan *median*. Tahap pertama adalah melakukan *filtering* nilai piksel asli dengan metode *gaussian*, kemudian mencari nilai MSE dan PNSR. Tahap kedua adalah melakukan *filtering* nilai piksel asli dengan metode *mean*, kemudian mencari nilai MSE dan PNSR. Tahap ketiga adalah melakukan *filtering* nilai piksel asli dengan metode *median*, kemudian mencari nilai MSE dan PNSR. Kemudian membanding hasil MSE dan PNSR setiap citra hasil reduksi *noise*.

1. Penerapan Metode *Filter Gaussian*

Langkah awal adalah menentukan nilai citra asli *noise* yang telah didapatkan seperti pada tabel di bawah :

Tabel 3.2 Nilai Piksel Citra Ultrasonography 5x5 Proses Gaussian

X,y	0	1	2	3	4
0	3	76	244	215	224
1	3	76	243	213	220
2	3	76	242	212	220
3	2	75	241	211	218
4	2	74	239	209	217

Langkah selanjutnya adalah menentukan matrik kernel *gaussian*. Pada analisis ini matrik kernel gaussian yang digunakan adalah matriks kernel 3x3 dengan sigma 1 sehingga matriks kernel *gaussian* adalah sebagai berikut:

Tabel 3.3 Nilai Kernel 3x3 *Gaussian*

1	2	1
2	4	2
1	2	1

Selanjutnya adalah melakukan reduksi *noise* dengan perkalian matrik citra 5x5 asli dan matriks kernel 3x3 *gaussian* dengan rumus *gaussian* sebagai berikut:

$$Pixel B(i, j) = \frac{1}{k} \cdot \sum_{p=0}^{N-1} \left(\sum_{q=0}^{M-1} G(p, q) \cdot PixelA \left(i + p - \frac{(N-1)}{2}, j + q - \frac{(M-1)}{2} \right) \right)$$

Berdasarkan pada rumus di atas, dapat dijelaskan bahwa:

K = Total jumlah kernel *gaussian*

p, q = kordinat citra kernel *gaussian*

i, j = koordinat citra asli

N = Jumlah kolom matrik kernel *gaussian*

M = Jumlah baris matrik kernel *gaussian*

Proses gaussian dilakukan pada citra bagian dalam tanpa merubah nilai citra bagian luar. Berikut adalah proses *gaussian* pertama pada citra 5x5 :

Tabel 3.4 Nilai Citra *Gaussian* Kernel 1

3	76	244	215	224
3	76	243	213	220
3	76	242	212	220
2	75	241	211	218
2	74	239	209	217

$$\frac{1}{16} \left(\left(3 + 1 - \frac{(3-1)}{2} \right) + \left(76 + 2 - \frac{(3-1)}{2} \right) + \left(244 + 1 - \frac{(3-1)}{2} \right) + \left(3 + 2 - \frac{(3-1)}{2} \right) + \right. \\ \left. \left(76 + 4 - \frac{(3-1)}{2} \right) + \left(243 + 2 - \frac{(3-1)}{2} \right) + \left(3 + 1 - \frac{(3-1)}{2} \right) + \left(76 + 2 - \frac{(3-1)}{2} \right) + \right. \\ \left. \left(242 + 1 - \frac{(3-1)}{2} \right) \right) = 60,8$$

Berdasarkan pada proses pertama, didapatkan nilai piksel citra baru *gaussian* yang mengisi bagian dalam citra sebagai berikut :

Tabel 3.5 Nilai Citra *Gaussian* 1

3	76	244	215	224
3	60,8			220
3				220
2				218
2	74	239	209	217

Selanjutnya adalah mencari nilai kedua

gaussian sebagai berikut :

Tabel 3.6 Nilai Citra *Gaussian* Kernel 2

3	76	244	215	224
3	76	243	213	220
3	76	242	212	220
2	75	241	211	218
2	74	239	209	217

$$\frac{1}{16} \left(\left(76 + 1 - \frac{(3-1)}{2} \right) + \left(244 - \frac{(3-1)}{2} \right) + \left(215 + 1 - \frac{(3-1)}{2} \right) + \left(76 + 2 - \frac{(3-1)}{2} \right) + \right. \\ \left. \left(243 + 4 - \frac{(3-1)}{2} \right) + \left(213 + 2 - \frac{(3-1)}{2} \right) + \left(76 + 1 - \frac{(3-1)}{2} \right) + \left(242 + 2 - \frac{(3-1)}{2} \right) + \right. \\ \left. \left(212 + 1 - \frac{(3-1)}{2} \right) \right) = 100,3$$

Berdasarkan pada proses kedua, didapatkan nilai piksel citra baru *gaussian* yang mengisi bagian dalam citra sebagai berikut :

Tabel 3.7 Nilai Citra *Gaussian* 2

3	76	244	215	224
3	60,8	100,3		220
3				220
2				218
2	74	239	209	217

Selanjutnya adalah mencari nilai ketiga *gaussian* sebagai berikut :

Tabel 3.8 Nilai Citra *Gaussian* Kernel 3

3	76	244	215	224
3	76	243	213	220
3	76	242	212	220
2	75	241	211	218
2	74	239	209	217

$$\frac{1}{16} \left(\left(244 + 1 - \frac{(3-1)}{2} \right) + \left(215 - \frac{(3-1)}{2} \right) + \left(224 + 1 - \frac{(3-1)}{2} \right) + \left(243 + 2 - \frac{(3-1)}{2} \right) + \right. \\ \left. \left(213 + 4 - \frac{(3-1)}{2} \right) + \left(220 + 2 - \frac{(3-1)}{2} \right) + \left(242 + 1 - \frac{(3-1)}{2} \right) + \left(212 + 2 - \frac{(3-1)}{2} \right) + \right. \\ \left. \left(220 + 1 - \frac{(3-1)}{2} \right) \right) = 127,5$$

Berdasarkan pada proses ketiga, didapatkan nilai piksel citra baru *gaussian* yang mengisi bagian dalam citra sebagai berikut :

Tabel 3.9 Nilai Citra *Gaussian* Kernel 3

3	76	244	215	224
3	60,8	100,3	127,5	220
3				220
2				218
2	74	239	209	217

Selanjutnya adalah mencari nilai keempat *gaussian* sebagai berikut :

Tabel 3.10 Nilai Citra *Gaussian* Kernel 4

3	76	244	215	224
3	76	243	213	220
3	76	242	212	220
2	75	241	211	218
2	74	239	209	217

$$\frac{1}{16} \left(\left(3 + 1 - \frac{(3-1)}{2} \right) + \left(76 + 2 - \frac{(3-1)}{2} \right) + \left(243 + 1 - \frac{(3-1)}{2} \right) + \left(3 + 2 - \frac{(3-1)}{2} \right) + \right. \\ \left. \left(76 + 4 - \frac{(3-1)}{2} \right) + \left(242 + 2 - \frac{(3-1)}{2} \right) + \left(2 + 1 - \frac{(3-1)}{2} \right) + \left(75 + 2 - \frac{(3-1)}{2} \right) + \right. \\ \left. \left(241 + 1 - \frac{(3-1)}{2} \right) \right) = 60,5$$

Berdasarkan pada proses keempat, didapatkan nilai piksel citra baru *gaussian* yang mengisi bagian dalam citra sebagai berikut :

Tabel 3.11 Nilai Citra *Gaussian* 4

3	76	244	215	224
3	60,8	100,3	127,5	220
3	60,5			220
2				218
2	74	239	209	217

Selanjutnya adalah mencari nilai kelima *gaussian* sebagai berikut :

Tabel 3.12 Nilai Citra *Gaussian* Kernel 5

3	76	244	215	224
3	76	243	213	220
3	76	242	212	220
2	75	241	211	218
2	74	239	209	217

$$\frac{1}{16} \left(\left(76 + 1 - \frac{(3-1)}{2} \right) + \left(243 - \frac{(3-1)}{2} \right) + \left(213 + 1 - \frac{(3-1)}{2} \right) + \left(76 + 2 - \frac{(3-1)}{2} \right) + \right. \\ \left. \left(242 + 4 - \frac{(3-1)}{2} \right) + \left(212 + 2 - \frac{(3-1)}{2} \right) + \left(75 + 1 - \frac{(3-1)}{2} \right) + \left(241 + 2 - \frac{(3-1)}{2} \right) + \right. \\ \left. \left(211 + 1 - \frac{(3-1)}{2} \right) \right) = 99,8$$

Berdasarkan pada proses kelima, didapatkan nilai piksel citra baru *gaussian* yang mengisi bagian dalam citra sebagai berikut :

Tabel 3.13 Nilai Citra *Gaussian5*

3	76	244	215	224
3	60,8	100,3	127,5	220
3	60,5	99,8		220
2				218
2	74	239	209	217

Selanjutnya adalah mencari nilai keenam *gaussian* sebagai berikut :

Tabel 3.14 Nilai Citra *Gaussian Kernel 6*

3	76	244	215	224
3	76	243	213	220
3	76	242	212	220
2	75	241	211	218
2	74	239	209	217

$$\frac{1}{16} \left(\left(243 + 1 - \frac{(3-1)}{2} \right) + \left(213 - \frac{(3-1)}{2} \right) + \left(220 + 1 - \frac{(3-1)}{2} \right) + \left(242 + 2 - \frac{(3-1)}{2} \right) + \left(212 + 4 - \frac{(3-1)}{2} \right) + \left(220 + 2 - \frac{(3-1)}{2} \right) + \left(241 + 1 - \frac{(3-1)}{2} \right) + \left(211 + 2 - \frac{(3-1)}{2} \right) + \left(218 + 1 - \frac{(3-1)}{2} \right) \right) = 126,7$$

Berdasarkan pada proses keenam, didapatkan nilai piksel citra baru *gaussian* yang mengisi bagian dalam citra sebagai berikut :

Tabel 3.15 Nilai Citra *Gaussian 6*

3	76	244	215	224
3	60,8	100,3	127,5	220
3	60,5	99,8	126,7	220
2				218
2	74	239	209	217

Selanjutnya adalah mencari nilai ketujuh *gaussian* sebagai berikut :

Tabel 3.16 Nilai Citra *Gaussian Kernel 7*

3	76	244	215	224
3	76	243	213	220
3	76	242	212	220
2	75	241	211	218
2	74	239	209	217

$$\frac{1}{16} \left(\left(3 + 1 - \frac{(3-1)}{2} \right) + \left(76 + 2 - \frac{(3-1)}{2} \right) + \left(242 + 1 - \frac{(3-1)}{2} \right) + \left(2 + 2 - \frac{(3-1)}{2} \right) + \left(75 + 4 - \frac{(3-1)}{2} \right) + \left(241 + 2 - \frac{(3-1)}{2} \right) + \left(2 + 1 - \frac{(3-1)}{2} \right) + \left(74 + 2 - \frac{(3-1)}{2} \right) + \left(239 + 1 - \frac{(3-1)}{2} \right) \right) = 60,1$$

Berdasarkan pada proses ketujuh, didapatkan nilai piksel citra baru *gaussian* yang mengisi bagian dalam citra sebagai berikut :

Tabel 3.17 Nilai Citra *Gaussian* 7

3	76	244	215	224
3	60,8	100,3	127,5	220
3	60,5	99,8	126,7	220
2	60,1			218
2	74	239	209	217

Selanjutnya adalah mencari nilai kedelapan *gaussian* sebagai berikut :

Tabel 3.18 Nilai Citra *Gaussian* Kernel 8

3	76	244	215	224
3	76	243	213	220
3	76	242	212	220
2	75	241	211	218
2	74	239	209	217

$$\frac{1}{16} \left(\left(76 + 1 - \frac{(3-1)}{2} \right) + \left(242 - \frac{(3-1)}{2} \right) + \left(212 + 1 - \frac{(3-1)}{2} \right) + \left(75 + 2 - \frac{(3-1)}{2} \right) + \right. \\ \left. \left(241 + 4 - \frac{(3-1)}{2} \right) + \left(211 + 2 - \frac{(3-1)}{2} \right) + \left(74 + 1 - \frac{(3-1)}{2} \right) + \left(239 + 2 - \frac{(3-1)}{2} \right) + \right. \\ \left. \left(209 + 1 - \frac{(3-1)}{2} \right) \right) = 99,1$$

Berdasarkan pada proses kedelapan, didapatkan nilai piksel citra baru *gaussian* yang mengisi bagian dalam citra sebagai berikut :

Tabel 3.19 Nilai Citra *Gaussian* 8

3	76	244	215	224
3	60,8	100,3	127,5	220
3	60,5	99,8	126,7	220
2	60,1	99,1		218
2	74	239	209	217

Selanjutnya adalah mencari nilai kesembilan *gaussian* sebagai berikut :

Tabel 3.20 Nilai Citra *Gaussian* Kernel 9

3	76	244	215	224
3	76	243	213	220
3	76	242	212	220
2	75	241	211	218
2	74	239	209	217

$$\frac{1}{16} \left(\left(242 + 1 - \frac{(3-1)}{2} \right) + \left(212 - \frac{(3-1)}{2} \right) + \left(220 + 1 - \frac{(3-1)}{2} \right) + \left(241 + 2 - \frac{(3-1)}{2} \right) + \right. \\ \left. \left(211 + 4 - \frac{(3-1)}{2} \right) + \left(218 + 2 - \frac{(3-1)}{2} \right) + \left(239 + 1 - \frac{(3-1)}{2} \right) + \left(209 + 2 - \frac{(3-1)}{2} \right) + \right. \\ \left. \left(217 + 1 - \frac{(3-1)}{2} \right) \right) = 126$$

Berdasarkan pada proses kedelapan, didapatkan nilai piksel citra baru *gaussian* yang mengisi bagian dalam citra sebagai berikut :

Tabel 3.21 Nilai Citra *Gaussian* 9

3	76	244	215	224
3	60,8	100,3	127,5	220
3	60,5	99,8	126,7	220
2	60,1	99,1	126	218
2	74	239	209	217

Berdasarkan pada proses kesembilan, didapatkan nilai piksel citra baru *gaussian* keseluruhan. Sehingga citra ultrasonography mengalami perubahan nilai piksel dengan reduksi *noise* yang ada.

Tabel 3.22 Nilai Piksel Citra Ultrasonography *Gaussian* 5x5

3	76	244	215	224
3	60,8	100,3	127,5	220
3	60,5	99,8	126,7	220
2	60,1	99,1	126	218
2	74	239	209	217

Selanjutnya adalah menghitung nilai MSE dan PNSR. Adapun prosesnya adalah sebagai berikut :

$$\text{MSE} = (3 - 3)^2 + (76 - 76)^2 + (244 - 244)^2 + (215 - 215)^2 + (224 - 224)^2 + \\ (3 - 3)^2 + (76 - 60,8)^2 + (243 - 100,3)^2 + (213 - 127,5)^2 + (220 - 220)^2 + \\ (3 - 3)^2 + (76 - 60,5)^2 + (242 - 99,6)^2 + (212 - 126,7)^2 + (220 - 220)^2 + \\ (2 - 2)^2 + (75 - 60,1)^2 + (241 - 99,1)^2 + (211 - 126)^2 + (218 - 218)^2 + \\ (2 - 2)^2 + (74 - 74)^2 + (239 - 239)^2 + (209 - 209)^2 + (217 - 217)^2 / 25$$

$$\text{MSE} = (0 + 0 + 0 + 0 + 0 + 0 + 231 + 20363,3 + 7310,3 + 0 + 0 + 240,3 + \\ 20220,8 + 7276,1 + 0 + 0 + 222 + 20135,6 + 7225 + 0 + 0 + 0 + 0 + 0 + \\ 0) / 25 = 3328,975$$

$$\text{PSNR} = 10 \log_{10} \left(\frac{244}{\sqrt{3328,975}} \right) = 10 \times 4,229 = 42,29$$

Berdasarkan dari proses perhitungan di atas, didapatkan nilai MSE = 3328,975 dan PSNR = 42,29 untuk proses *filter gaussian*.

2. Penerapan Metode *Filter Mean*

Metode *filter mean* adalah metode reduksi citra dengan melakukan pencarian nilai rata-rata dari matrik kernel kemudian mendapatkan nilai baru. Adapun citra awal yang digunakan masih sama nilainya seperti pada tabel 3.1 yaitu citra ultrasonography dengan 5x5 piksel. Berikut adalah proses pertama metode filter *mean* :

Tabel 3.23 Nilai Citra *Filter Mean* Kernel 1

3	76	244	215	224
3	76	243	213	220
3	76	242	212	220
2	75	241	211	218
2	74	239	209	217

$$mean = \frac{3 + 76 + 244 + 3 + 76 + 243 + 3 + 76 + 242}{9} = 107,3$$

Berdasarkan pada proses pertama, didapatkan nilai piksel citra baru *mean* yang mengisi bagian dalam citra sebagai berikut :

Tabel 3.24 Nilai Citra *Filter Mean* 1

3	76	244	215	224
3	107,3			220
3				220
2				218
2	74	239	209	217

Berikut adalah proses kedua metode *filter mean* :

Tabel 3.25 Nilai Citra *Filter Mean* Kernel 2

3	76	244	215	224
3	76	243	213	220
3	76	242	212	220
2	75	241	211	218
2	74	239	209	217

$$mean = \frac{76 + 244 + 215 + 76 + 243 + 213 + 76 + 242 + 212}{9} = 177,4$$

Berdasarkan pada proses kedua, didapatkan nilai piksel citra baru *mean* yang mengisi bagian dalam citra sebagai berikut :

Tabel 3.26 Nilai Citra *Filter Mean* 2

3	76	244	215	224
3	107,3	177,4		220
3				220
2				218
2	74	239	209	217

Berikut adalah proses ketiga metode *filter mean* :

Tabel 3.27 Nilai Citra *Filter Mean* Kernel 3

3	76	244	215	224
3	76	243	213	220
3	76	242	212	220
2	75	241	211	218
2	74	239	209	217

$$mean = \frac{244 + 215 + 224 + 243 + 213 + 220 + 242 + 212 + 220}{9} = 225,9$$

Berdasarkan pada proses ketiga, didapatkan nilai piksel citra baru *mean* yang mengisi bagian dalam citra sebagai berikut :

Tabel 3.28 Nilai Citra *Filter Mean* 3

3	76	244	215	224
3	107,3	177,4	255,9	220
3				220
2				218
2	74	239	209	217

Berikut adalah proses keempat metode *filter mean* :

Tabel 3.29 Nilai Citra *Filter Mean* Kernel 4

3	76	244	215	224
3	76	243	213	220
3	76	242	212	220
2	75	241	211	218
2	74	239	209	217

$$mean = \frac{3 + 76 + 243 + 3 + 76 + 242 + 2 + 75 + 241}{9} = 106,8$$

Berdasarkan pada proses keempat, didapatkan nilai piksel citra baru *mean* yang mengisi bagian dalam citra sebagai berikut :

Tabel 3.30 Nilai Citra *Filter Mean* 4

3	76	244	215	224
3	107,3	177,4	255,9	220
3	106,8			220
2				218
2	74	239	209	217

Berikut adalah proses kelima metode *filter mean* :

Tabel 3.31 Nilai Citra *Filter Mean* Kernel 5

3	76	244	215	224
3	76	243	213	220
3	76	242	212	220

Tabel 3.31 Nilai

Citra *Filter Mean*

Kernel 5 (Lanjutan)

2	75	241	211	218
2	74	239	211	218

$$mean = \frac{76 + 243 + 213 + 76 + 242 + 212 + 75 + 241 + 211}{9} = 176,6$$

Berdasarkan pada proses kelima, didapatkan nilai piksel citra baru *mean* yang mengisi bagian dalam citra sebagai berikut :

Tabel 3.32 Nilai Citra *Filter Mean 5*

3	76	244	215	224
3	107,3	177,4	255,9	220
3	106,8	176,6		220
2				218
2	74	239	209	217

Berikut adalah proses keenam metode *filter mean* :

Tabel 3.33 Nilai Citra *Filter Mean Kernel 6*

3	76	244	215	224
3	76	243	213	220
3	76	242	212	220
2	75	241	211	218
2	74	239	209	217

$$mean = \frac{243 + 213 + 220 + 242 + 212 + 220 + 241 + 211 + 218}{9} = 224,4$$

Berdasarkan pada proses keenam, didapatkan nilai piksel citra baru *mean* yang mengisi bagian dalam citra sebagai berikut :

Tabel 3.34 Nilai Citra *Filter Mean 6*

3	76	244	215	224
3	107,3	177,4	255,9	220
3	106,8	176,6	224,4	220
2				218
2	74	239	209	217

Berikut adalah proses ketujuh metode *filter mean* :

Tabel 3.35 Nilai Citra *Filter Mean Kernel 7*

3	76	244	215	224
3	76	243	213	220
3	76	242	212	220
2	75	241	211	218
2	74	239	209	217

$$mean = \frac{3 + 76 + 242 + 2 + 75 + 241 + 2 + 74 + 239}{9} = 106$$

Berdasarkan pada proses ketujuh, didapatkan nilai piksel citra baru *mean* yang mengisi bagian dalam citra sebagai berikut :

Tabel 3.36 Nilai Citra *Filter Mean 7*

3	76	244	215	224
3	107,3	177,4	255,9	220
3	106,8	176,6	224,4	220
2	106			218
2	74	239	209	217

Berikut adalah proses kedelapan metode *filter mean* :

Tabel 3.37 Nilai Citra *Filter Mean Kernel 8*

3	76	244	215	224
3	76	243	213	220
3	76	242	212	220
2	75	241	211	218
2	74	239	209	217

$$mean = \frac{76 + 242 + 212 + 75 + 241 + 211 + 74 + 239 + 209}{9} = 175,4$$

Berdasarkan pada proses kedelapan, didapatkan nilai piksel citra baru *mean* yang mengisi bagian dalam citra sebagai berikut :

Tabel 3.38 Nilai Citra *Filter Mean 8*

3	76	244	215	224
3	107,3	177,4	255,9	220
3	106,8	176,6	224,4	220
2	106	175,4		218
2	74	239	209	217

Berikut adalah proses kesembilan metode *filter mean* :

Tabel 3.39 Nilai Citra *Filter Mean Kernel 9*

3	76	244	215	224
3	76	243	213	220
3	76	242	212	220
2	75	241	211	218
2	74	239	209	217

$$mean = \frac{242 + 212 + 220 + 241 + 211 + 218 + 239 + 209 + 217}{9} = 223,2$$

Berdasarkan pada proses kesembilan, didapatkan nilai piksel citra baru *mean* yang mengisi bagian dalam citra sebagai berikut :

Tabel 3.40 Nilai Citra *Filter Mean 9*

3	76	244	215	224
3	107,3	177,4	255,9	220
3	106,8	176,6	224,4	220
2	106	175,4	223,2	218
2	74	239	209	217

Berdasarkan pada proses kesembilan, didapatkan nilai piksel citra baru *filter mean* keseluruhan. Sehingga citra ultrasonography mengalami perubahan nilai piksel dengan reduksi *noise* yang ada.

Tabel 3.41 Nilai Piksel Citra Ultrasonography Mean 5x5

3	76	244	215	224
3	107,3	177,4	255,9	220
3	106,8	176,6	224,4	220
2	106	175,4	223,2	218
2	74	239	209	217

Selanjutnya adalah menghitung nilai MSE dan PNSR. Adapun prosesnya adalah sebagai berikut :

$$\begin{aligned} \text{MSE} = & (3 - 3)^2 + (76 - 76)^2 + (244 - 244)^2 + (215 - 215)^2 + (224 - 224)^2 + \\ & (3 - 3)^2 + (76 - 107,3)^2 + (243 - 177,4)^2 + (213 - 225,9)^2 + (220 - 220)^2 + \\ & (3 - 3)^2 + (76 - 106,8)^2 + (242 - 176,6)^2 + (212 - 224,4)^2 + (220 - 220)^2 + \\ & (2 - 2)^2 + (75 - 106)^2 + (241 - 175,4)^2 + (211 - 223,2)^2 + (218 - 218)^2 + \\ & (2 - 2)^2 + (74 - 74)^2 + (239 - 239)^2 + (209 - 209)^2 + (217 - 217)^2 / 25 \end{aligned}$$

$$\begin{aligned} \text{MSE} & = (0 + 0 + 0 + 0 + 0 + 0 + 979,7 + 4303,4 + 166,4 + 0 + 0 + 948,6 + \\ & 4277,2 + 153,8 + 0 + 0 + 961 + 4303,4 + 148,9 + 0 + 0 + 0 + 0 + 0) / 25 = \\ & 649,689 \end{aligned}$$

$$\text{PSNR} = 10 \log_{10} \left(\frac{244}{\sqrt{649,689}} \right) = 10 \times 9,573 = 95,73$$

Berdasarkan dari proses perhitungan di atas, didapatkan nilai MSE = 649,689 dan PSNR = 95,73 untuk proses *filter mean*.

3. Penerapan Metode *Filter Median*

Metode *filter median* adalah metode yang menghilangkan *noise* pada citra dengan cara mengurutkan citra kernel dari paling kecil hingga terbesar, kemudian mengambil nilai tengah citra hasil pengurutan yang kemudian ditukar pada citra kernel. Adapun proses

filter median dengan citra asli ultrasonography dengan piksel 5x5 berdasarkan nilai tabel 3.1 sebagai berikut:

Tabel 3.42 Nilai Citra *Filter Median* Kernel 1

3	76	244	215	224
3	76	243	213	220
3	76	242	212	220
2	75	241	211	218
2	74	239	209	217

Berdasarkan pada matrik di atas, kemudian nilai matrik kernel 3x3 yang dipilih diurutkan dari terendah hingga tertinggi seperti berikut :

Tabel 3.43 Urutan Nilai Citra *Filter Median* Kernel 1

Nilai Citra								
In-0	In-1	In-2	In-3	In-4	In-5	In-6	In-7	In-8
3	3	3	76	76	76	242	243	244

Berdasarkan dari nilai yang telah diurutkan tersebut kemudian ditentukan nilai tengahnya, yaitu indeks ke-4 yang bernilai 76. Posisi indek k-4 (76) tersebut ditukar dengan posisi nilai tengah citra asli berdasarkan seleksi kernel, sehingga menghasilkan nilai citra dengan susunan baru dari hasil proses *filter median* sebagai berikut :

Tabel 3.44 Nilai Citra *Filter Median* 1

3	76	244	215	224
3	76			220
3				220
2				218
2	74	239	209	217

Berikutnya adalah melakukan median pada kernel kedua:

Tabel 3.45 Nilai Citra *Filter Median* Kernel 2

3	76	244	215	224
3	76	243	213	220
3	76	242	212	220
2	75	241	211	218
2	74	239	209	217

Berdasarkan pada matrik di atas, kemudian nilai matrik kernel 3x3 yang dipilih diurutkan dari terendah hingga tertinggi seperti berikut :

Tabel 3.46 Urutan Nilai Citra *Filter Median* Kernel 2

Nilai Citra								
In-0	In-1	In-2	In-3	In-4	In-5	In-6	In-7	In-8
76	76	76	212	213	215	242	243	244

Berdasarkan dari nilai yang telah diurutkan tersebut kemudian ditentukan nilai tengahnya, yaitu indeks ke-4 yang bernilai 213. Posisi indek k-4 (213) tersebut ditukar dengan posisi nilai tengah citra asli berdasarkan seleksi kernel, sehingga menghasilkan nilai citra dengan susunan baru dari hasil proses *filter median* sebagai berikut :

Tabel 3.47 Nilai Citra *Filter Median 2*

3	76	244	215	224
3	76	213		220
3				220
2				218
2	74	239	209	217

Berikutnya adalah melakukan median pada kernel ketiga:

Tabel 3.48 Nilai Citra *Filter Median Kernel 3*

3	76	244	215	224
3	76	243	213	220
3	76	242	212	220
2	75	241	211	218
2	74	239	209	217

Berdasarkan pada matrik di atas, kemudian nilai matrik kernel 3x3 yang dipilih diurutkan dari terendah hingga tertinggi seperti berikut :

Tabel 3.49 Urutan Nilai Citra *Filter Median Kernel 3*

Nilai Citra								
In-0	In-1	In-2	In-3	In-4	In-5	In-6	In-7	In-8
212	213	215	220	220	224	242	243	244

Berdasarkan dari nilai yang telah diurutkan tersebut kemudian ditentukan nilai tengahnya, yaitu indeks ke-4 yang bernilai 220. Posisi indek k-4 (220) tersebut ditukar dengan posisi nilai tengah citra asli berdasarkan seleksi kernel, sehingga menghasilkan nilai citra dengan susunan baru dari hasil proses *filter median* sebagai berikut :

Tabel 3.50 Nilai Citra *Filter Median 3*

3	76	244	215	224
3	76	213	220	220
3				220
2				218
2	74	239	209	217

Berikutnya adalah melakukan median pada kernel keempat:

Tabel 3.51 Nilai Citra *Filter Median Kernel 4*

3	76	244	215	224
3	76	243	213	220
3	76	242	212	220

Tabel 3.51 Nilai Citra *Filter Median* Kernel 4 (Lanjutan)

2	75	241	211	218
2	74	239	209	217

Berdasarkan pada matrik di atas, kemudian nilai matrik kernel 3x3 yang dipilih diurutkan dari terendah hingga tertinggi seperti berikut :

Tabel 3.52 Urutan Nilai Citra *Filter Median* Kernel 4

Nilai Citra								
In-0	In-1	In-2	In-3	In-4	In-5	In-6	In-7	In-8
2	3	3	75	76	76	242	243	244

Berdasarkan dari nilai yang telah diurutkan tersebut kemudian ditentukan nilai tengahnya, yaitu indeks ke-4 yang bernilai 76. Posisi indek k-4 (76) tersebut ditukar dengan posisi nilai tengah citra asli berdasarkan seleksi kernel, sehingga menghasilkan nilai citra dengan susunan baru dari hasil proses *filter median* sebagai berikut :

Tabel 3.53 Nilai Citra *Filter Median* 4

3	76	244	215	224
3	76	213	220	220
3	76			220
2				218
2	74	239	209	217

Berikutnya adalah melakukan median pada kernel kelima:

Tabel 3.54 Nilai Citra *Filter Median* Kernel 5

3	76	244	215	224
3	76	243	213	220
3	76	242	212	220
2	75	241	211	218
2	74	239	209	217

Berdasarkan pada matrik di atas, kemudian nilai matrik kernel 3x3 yang dipilih diurutkan dari terendah hingga tertinggi seperti berikut :

Tabel 3.55 Urutan Nilai Citra *Filter Median* Kernel 5

Nilai Citra								
In-0	In-1	In-2	In-3	In-4	In-5	In-6	In-7	In-8
75	76	76	211	212	213	241	242	243

Berdasarkan dari nilai yang telah diurutkan tersebut kemudian ditentukan nilai tengahnya, yaitu indeks ke-4 yang bernilai 212. Posisi indek k-4 (212) tersebut ditukar dengan posisi nilai tengah citra asli berdasarkan seleksi kernel, sehingga menghasilkan nilai citra dengan susunan baru dari hasil proses *filter median* sebagai berikut :

Tabel 3.56 Nilai Citra *Filter Median* Kernel 5

3	76	244	215	224
3	76	213	220	220
3	76	212		220
2				218
2	74	239	209	217

Berikutnya adalah melakukan median pada kernel keenam:

Tabel 3.57 Nilai Citra *Filter Median* Kernel 6

3	76	244	215	224
3	76	243	213	220
3	76	242	212	220
2	75	241	211	218
2	74	239	209	217

Berdasarkan pada matrik di atas, kemudian nilai matrik kernel 3x3 yang dipilih diurutkan dari terendah hingga tertinggi seperti berikut :

Tabel 3.58 Urutan Nilai Citra *Filter Median* Kernel 6

Nilai Citra								
In-0	In-1	In-2	In-3	In-4	In-5	In-6	In-7	In-8
211	212	213	218	220	220	241	242	243

Berdasarkan dari nilai yang telah diurutkan tersebut kemudian ditentukan nilai tengahnya, yaitu indeks ke-4 yang bernilai 220. Posisi indeks ke-4 (220) tersebut ditukar dengan posisi nilai tengah citra asli berdasarkan seleksi kernel, sehingga menghasilkan nilai citra dengan susunan baru dari hasil proses *filter median* sebagai berikut :

Tabel 3.59 Nilai Citra *Filter Median Mean* 6

3	76	244	215	224
3	76	213	220	220
3	76	212	220	220
2				218
2	74	239	209	217

Berikutnya adalah melakukan median pada kernel ketujuh:

Tabel 3.60 Nilai Citra *Filter Median* Kernel 7

3	76	244	215	224
3	76	243	213	220
3	76	242	212	220
2	75	241	211	218
2	74	239	209	217

Berdasarkan pada matrik di atas, kemudian nilai matrik kernel 3x3 yang dipilih diurutkan dari terendah hingga tertinggi seperti berikut :

Tabel 3.61 Urutan Nilai Citra *Filter Median* Kernel 7

Nilai Citra								
In-0	In-1	In-2	In-3	In-4	In-5	In-6	In-7	In-8
2	2	3	74	75	76	239	241	242

Berdasarkan dari nilai yang telah diurutkan tersebut kemudian ditentukan nilai tengahnya, yaitu indeks ke-4 yang bernilai 75. Posisi indek k-4 (75) tersebut ditukar dengan posisi nilai tengah citra asli berdasarkan seleksi kernel, sehingga menghasilkan nilai citra dengan susunan baru dari hasil proses *filter median* sebagai berikut :

Tabel 3.62 Nilai Citra *Filter Median* 7

3	76	244	215	224
3	76	213	220	220
3	76	212	220	220
2	75			218
2	74	239	209	217

Berikutnya adalah melakukan median pada kernel kedelapan:

Tabel 3.63 Nilai Citra *Filter Median* Kernel 8

3	76	244	215	224
3	76	243	213	220
3	76	242	212	220
2	75	241	211	218
2	74	239	209	217

Berdasarkan pada matrik di atas, kemudian nilai matrik kernel 3x3 yang dipilih diurutkan dari terendah hingga tertinggi seperti berikut :

Tabel 3.64 Urutan Nilai Citra *Filter Median* Kernel 8

Nilai Citra								
In-0	In-1	In-2	In-3	In-4	In-5	In-6	In-7	In-8
74	75	76	209	211	212	239	241	242

Berdasarkan dari nilai yang telah diurutkan tersebut kemudian ditentukan nilai tengahnya, yaitu indeks ke-4 yang bernilai 211. Posisi indek k-4 (211) tersebut ditukar dengan posisi nilai tengah citra asli berdasarkan seleksi kernel, sehingga menghasilkan nilai citra dengan susunan baru dari hasil proses *filter median* sebagai berikut :

Tabel 3.65 Nilai Citra *Filter Median* 8

3	76	244	215	224
3	76	213	220	220
3	76	212	220	220
2	75	211		218
2	74	239	209	217

Berikutnya adalah melakukan median pada kernel kesembilan:

Tabel 3.66 Nilai Citra *Filter Median* Kernel 9

3	76	244	215	224
3	76	243	213	220
3	76	242	212	220
2	75	241	211	218
2	74	239	209	217

Berdasarkan pada matrik di atas, kemudian nilai matrik kernel 3x3 yang dipilih diurutkan dari terendah hingga tertinggi seperti berikut :

Tabel 3.67 Urutan Nilai Citra *Filter Median* Kernel 9

Nilai Citra								
In-0	In-1	In-2	In-3	In-4	In-5	In-6	In-7	In-8
209	211	212	217	217	220	239	241	242

Berdasarkan dari nilai yang telah diurutkan tersebut kemudian ditentukan nilai tengahnya, yaitu indeks ke-4 yang bernilai 217. Posisi indeks ke-4 (217) tersebut ditukar dengan posisi nilai tengah citra asli berdasarkan seleksi kernel, sehingga menghasilkan nilai citra dengan susunan baru dari hasil proses *filter median* sebagai berikut :

Tabel 3.68 Nilai Citra *Filter Median* 9

3	76	244	215	224
3	76	213	220	220
3	76	212	220	220
2	75	211	217	218
2	74	239	209	217

Berdasarkan pada proses kesembilan, didapatkan nilai piksel citra baru *filter median* keseluruhan. Sehingga citra ultrasonography mengalami perubahan nilai piksel dengan reduksi *noise* yang ada.

Tabel 3.69 Nilai Piksel Citra Ultrasonography Median 5x5

3	76	244	215	224
3	76	213	220	220
3	76	212	220	220
2	75	211	217	218
2	74	239	209	217

Selanjutnya adalah menghitung nilai MSE dan PNSR. Adapun prosesnya adalah sebagai berikut :

$$\begin{aligned} \text{MSE} = & (3 - 3)^2 + (76 - 76)^2 + (244 - 244)^2 + (215 - 215)^2 + (224 - 224)^2 + \\ & (3 - 3)^2 + (76 - 76)^2 + (243 - 213)^2 + (213 - 220)^2 + (220 - 220)^2 + \\ & (3 - 3)^2 + (76 - 76)^2 + (242 - 212)^2 + (212 - 220)^2 + (220 - 220)^2 + \end{aligned}$$

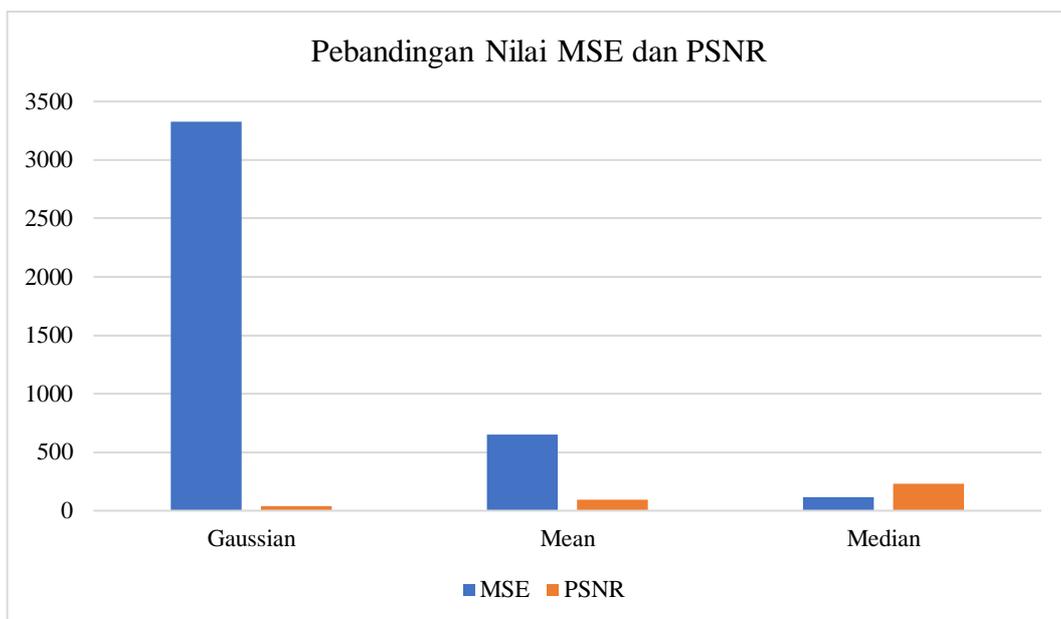
$$(2 - 2)^2 + (75 - 75)^2 + (241 - 211)^2 + (211 - 217)^2 + (218 - 218)^2 + \\ (2 - 2)^2 + (74 - 74)^2 + (239 - 239)^2 + (209 - 209)^2 + (217 - 217)^2 / 25$$

$$\text{MSE} = (0 + 0 + 0 + 0 + 0 + 0 + 0 + 900 + 49 + 0 + 0 + 0 + 900 + 64 + 0 + 0 + \\ 0 + 900 + 36 + 0 + 0 + 0 + 0 + 0 + 0) / 25 = 113,96$$

$$\text{PSNR} = 10 \log_{10} \left(\frac{244}{\sqrt{113,96}} \right) = 10 \times 22,857 = 228,57$$

Berdasarkan dari proses perhitungan di atas, didapatkan nilai MSE = 113,96 dan PSNR = 228,57 untuk proses *filter median*.

Berdasarkan dari proses perhitungan manual dalam membandingkan metode *filter gaussian*, *mean* dan *median* untuk reduksi *noise* citra ultrasonography, didapati hasil perbandingan nilai MSE dan PSNR sebagai berikut :



Gambar 3.3 Pebandingan Nilai MSE dan PSNR Citra Hasil Reduksi *Noise*

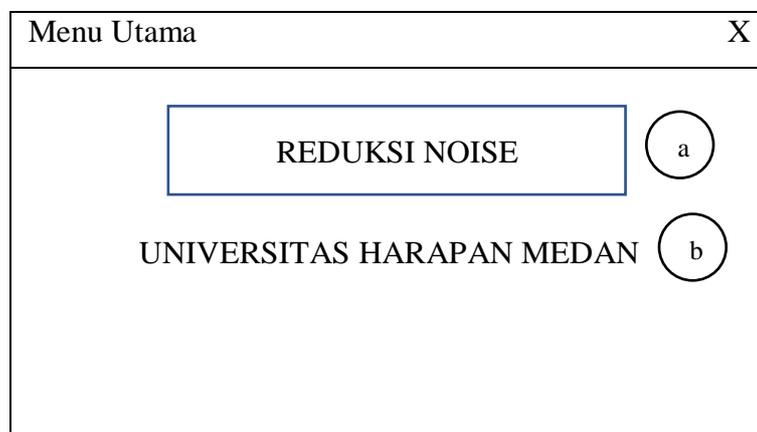
Berdasarkan pada gambar di atas, dapat diambil kesimpulan bahwa metode *median* memiliki tingkat reduksi *noise* yang lebih baik pada proses reduksi citra ultrasonography 5x5. Hal ini ditandai dengan lebih tingginya nilai PSNR dari citra hasil reduksi *noise* menggunakan metode *median filter*.

3.1.3 Perancangan Sistem

Sebelum diimplementasi ke dalam sebuah program aplikasi, terlebih dahulu merancang sistem peningkatan kualitas citra ultrasonography bernoise dengan perbandingan metode *filter gaussian*, *mean* dan *median*. Adapun perancangan sistem terdiri dari rancangan menu utama dan rancangan menu reduksi *noise*. Adapun keseluruhan rancangan *interface* sistem adalah sebagai berikut:

1. Rancangan Menu Utama

Menu utama adalah sebuah menu yang akan tampil ketika pertama kali program dijalankan. Adapun menu utama juga sebagai navigasi untuk kemenu-menu yang lain. Berikut adalah rancangan menu utama dapat dilihat pada gambar di bawah:



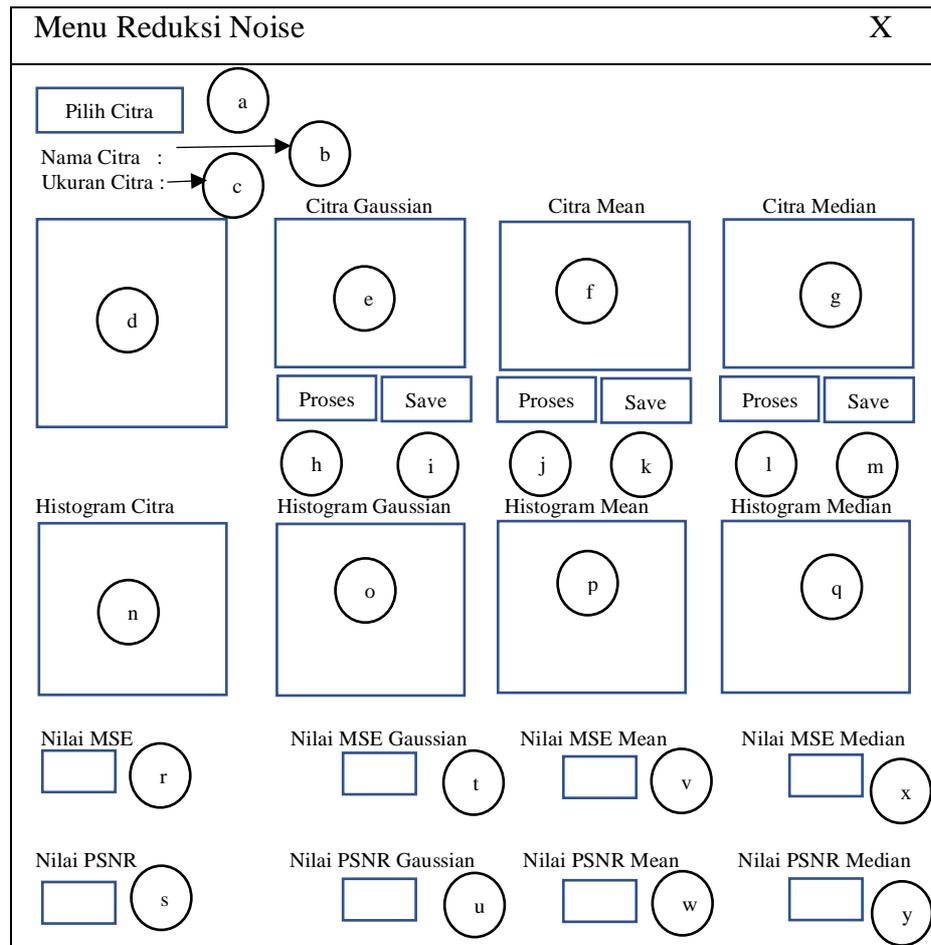
Gambar 3.4 Rancangan Menu Utama

Berdasarkan pada gambar di atas, adapun dapat diberi keterangan sebagai berikut:

- a. *Button* reduksi *noise* yang berfungsi untuk menampilkan menu reduksi citra ultrasonogprahy dengan metode *gaussian*, *mean* dan *median*.
- b. *Statictext* yang berfungsi untuk menampung karakter nama Universitas Harapan Medan.

2. Rancangan Menu Reduksi *Noise*

Menu reduksi *noise* adalah sebuah menu yang akan tampil ketika pengguna akan melakukan peningkatan kualitas citra ultrasonography dengan perbandingan metode *filter gaussian*, *mean* dan *median*. Adapun Berikut adalah rancangan reduksi *noise* dapat dilihat pada gambar di bawah:



Gambar 3.5 Rancangan Reduksi *Noise*

Berdasarkan pada gambar di atas, dapat diberi keterangan sebagai berikut:

- Button* yang berfungsi untuk memilih dan mencari citra ultrasonography.
- Statictext* yang berfungsi menampilkan nama citra ultrasonography yang dipilih.
- Statictext* yang berfungsi menampilkan ukuran citra ultrasonography yang dipilih.
- Axes* yang berfungsi menampung data citra ultrasonography yang telah dipilih.
- Axes* yang berfungsi menampung data citra ultrasonography hasil reduksi *noise gaussian*.
- Axes* yang berfungsi menampung data citra ultrasonography hasil reduksi *noise mean*.
- Axes* yang berfungsi menampung data citra ultrasonography hasil reduksi *noise median*.
- Button* yang berfungsi untuk memproses reduksi *noise* citra ultrasonography menggunakan metode *gaussian*.
- Button* yang berfungsi untuk menyimpan hasil citra reduksi *noise* metode *gaussian*.

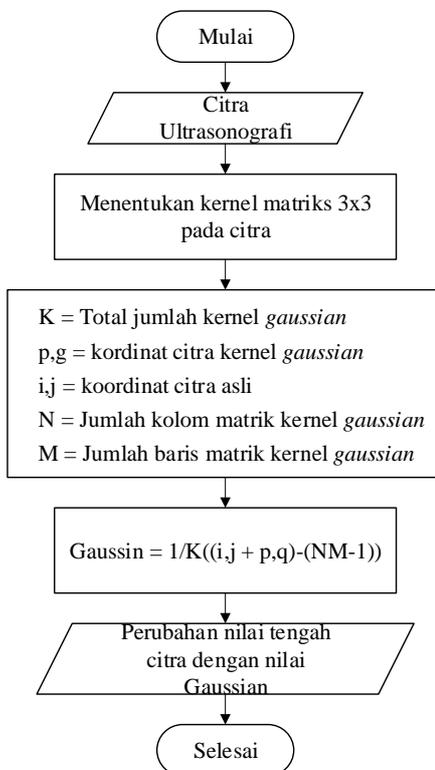
- j. *Button* yang berfungsi untuk memproses reduksi *noise* citra ultrasonography menggunakan metode *mean*.
- k. *Button* yang berfungsi untuk menyimpan hasil citra reduksi *noise* metode *mean*.
- l. *Button* yang berfungsi untuk memproses reduksi *noise* citra ultrasonography menggunakan metode *median*.
- m. *Button* yang berfungsi untuk menyimpan hasil citra reduksi *noise* metode *median*.
- n. *Axes* yang berfungsi menampung data histogram citra ultrasonography.
- o. *Axes* yang berfungsi menampung data histogram ultrasonography hasil reduksi *noise gaussian*.
- p. *Axes* yang berfungsi menampung data histogram ultrasonography hasil reduksi *noise mean*.
- q. *Axes* yang berfungsi menampung data histogram ultrasonography hasil reduksi *noise median*.
- r. *EditText* yang berfungsi menampilkan informasi nilai MSE citra ultrasonography.
- s. *EditText* yang berfungsi menampilkan informasi nilai PSNR citra ultrasonography.
- t. *EditText* yang berfungsi menampilkan informasi nilai MSE citra ultrasonography hasil reduksi *noise gaussian*.
- u. *EditText* yang berfungsi menampilkan informasi nilai PSNR citra ultrasonography hasil reduksi *noise gaussian*.
- v. *EditText* yang berfungsi menampilkan informasi nilai MSE citra ultrasonography hasil reduksi *noise mean*.
- w. *EditText* yang berfungsi menampilkan informasi nilai PSNR citra ultrasonography hasil reduksi *noise mean*.
- x. *EditText* yang berfungsi menampilkan informasi nilai MSE citra ultrasonography hasil reduksi *noise median*.
- y. *EditText* yang berfungsi menampilkan informasi nilai PSNR citra ultrasonography hasil reduksi *noise median*.

3.1.4 Flowchart

Flowchart berfungsi untuk menunjukkan alur proses dari sistem atau metode yang diterapkan. Berikut adalah keseluruhan dari *flowchart* metode yang akan diterapkan:

1. Flowchart Gaussian Filter

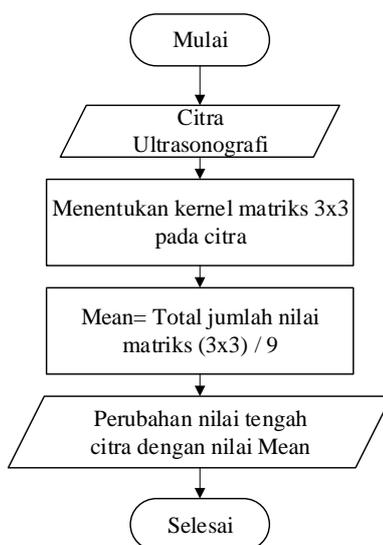
Berikut adalah rancangan dari *flowchart* dari metode *gaussian filter* yang terlihat seperti gambar berikut:



Gambar 3.6 Flowchart Gaussian Filter

2. Flowchart Mean Filter

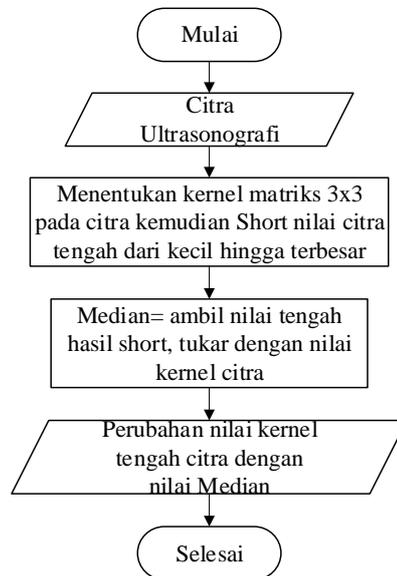
Berikut adalah rancangan dari *flowchart* dari metode *mean filter* yang terlihat seperti gambar berikut:



Gambar 3.7 Flowchart Mean Filter

3. Flowchart Median Filter

Berikut adalah rancangan dari *flowchart* dari metode *median filter* yang terlihat seperti gambar berikut:



Gambar 3.8 *Flowchart Median Filter*

BAB 4

IMPELEMENTASI DAN PENGUJIAN SISTEM

4.1 Implementasi dan Perangkat Sistem

Implementasi sistem adalah tahapan dimana sistem atau aplikasi siap untuk dioperasikan pada keadaan yang sebenarnya sesuai dari hasil analisis dan perancangan yang dilakukan, sehingga akan diketahui apakah sistem atau aplikasi yang dirancang benar-benar dapat menghasilkan tujuan yang dicapai. Aplikasi sistem analisis perbandingan metode *gaussian*, *median* dan *mean* ini dilengkapi dengan tampilan yang bertujuan untuk memudahkan penggunaannya, fungsi dari antarmuka ini adalah untuk memberi kan *input* dan menampilkan *output* dari aplikasi. Adapun analisa kebutuhan perangkat *Hardware* ataupun *Software* dalam pengujian keras, diantara perangkat keras ini dapat diterangkan sebagai berikut.

1. Analisa Kebutuhan Perangkat Keras

Dalam perancangan sistem ini menggunakan dari berbagai perangkat keras, diantara perangkat keras ini dapat diterangkan sebagai berikut :

Tabel 4.1 Kebutuhan Perangkat Keras

No	Hardware	Spesifikasi
1	Processor	Intel Core i3
2	Memory	4GB
3	Hardisk	500GB

Berdasarkan pada tabel di atas perangkat keras yang dibutuhkan adalah sebuah *processor* dengan spesifikasi Intel Core i3, *memory* 4GB dan *harddisk* 500GB.

2. Analisa Kebutuhan Lunak

Selain perangkat keras, perangkat lunak juga sangat dibutuhkan dalam perangkat sistem. Agar sistem dapat berjalan dengan baik maka penulis akan memakai perangkat lunak yang benar-benar mampu atau diperlukan secara menyeluruh dalam perancangan sistem ini. Perangkata lunak yang digunakan diantaranya:

Tabel 4.2 Kebutuhan Perangkat Lunak

No	Software
1	Sistem Operasi Windows 10
2	Matlab 2017b

4.2 Tampilan Sistem Aplikasi

Berdasarkan dari proses tahapan hitungan manual dan rancangan program aplikasi, selanjutnya adalah melakukan tahapan pengujian sistem aplikasi yang telah dibangun menggunakan program Matlab. Berikut adalah tampilan program yang telah dibangun :

1. Tampilan Program Menu Utama

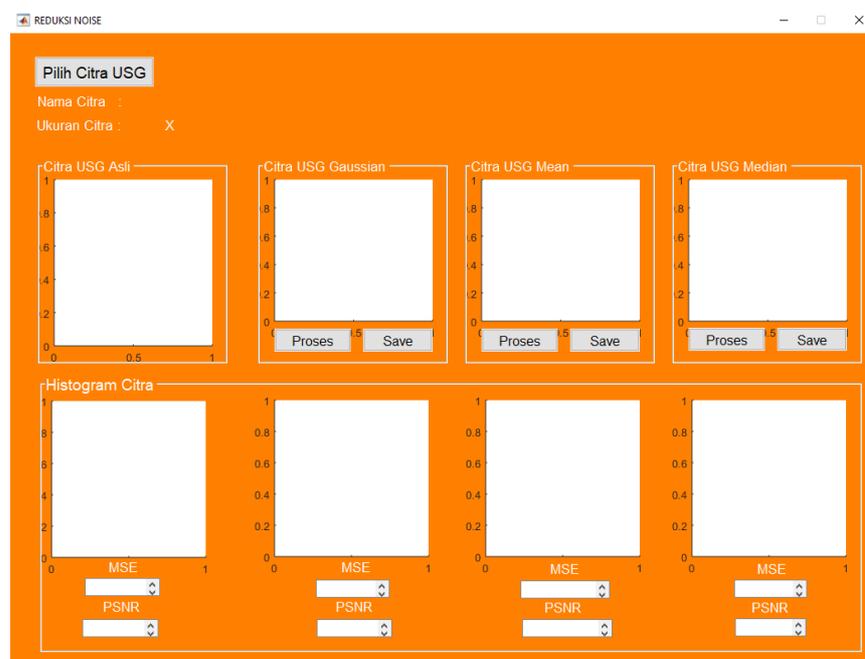
Adapun tampilan program menu utama yang telah dibangun kedalam aplikasi menggunakan matlab dapat dilihat sebagai berikut ini:



Gambar 4.1 Tampilan Menu Utama

2. Tampilan Program Menu Reduksi *Noise*

Adapun tampilan program menu reduksi *noise* yang telah dibangun kedalam aplikasi menggunakan matlab dapat dilihat sebagai berikut ini:



Gambar 4.2 Tampilan Menu Reduksi *Noise*

4.2.1 Pengujian Sistem

Pengujian sistem yang dilakukan adalah mencoba membanding hasil reduksi *noise* citra USG berdasarkan metode filer *gaussian*, *mean* dan *median*. Data yang digunakan adalah 5 citra yang memiliki tingkat *noise* berbeda-beda.

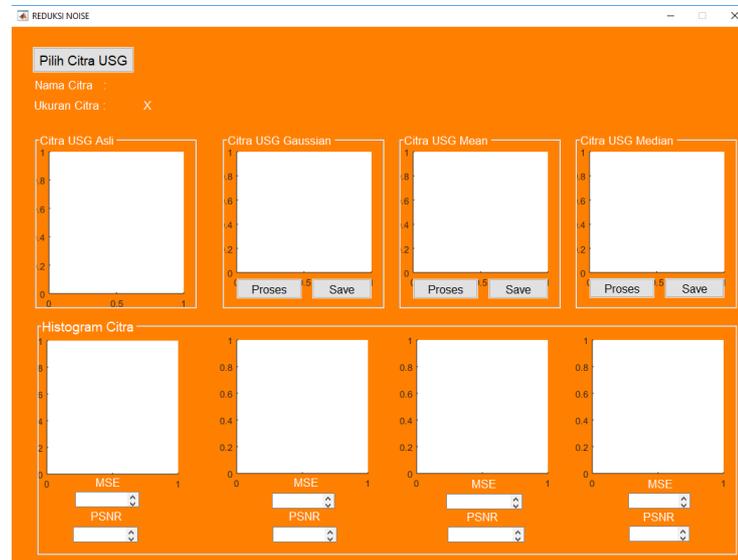
Tabel 4.3 Citra USG

No	Nama Citra	Tingkat Noise	Ukuran
1	USG10%	10%	500x351
2	USG20%	20%	341x301
3	USG30%	30%	400x300
4	USG40%	40%	400x281
5	USG60%	60%	800x602

Berdasarkan pada tabel di atas, berikut adalah proses pengujian metode *filter gaussian*, *mean* dan *median* dalam meningkatkan kualitas citra USG *noise* pada sistem aplikasi yang telah dibangun :

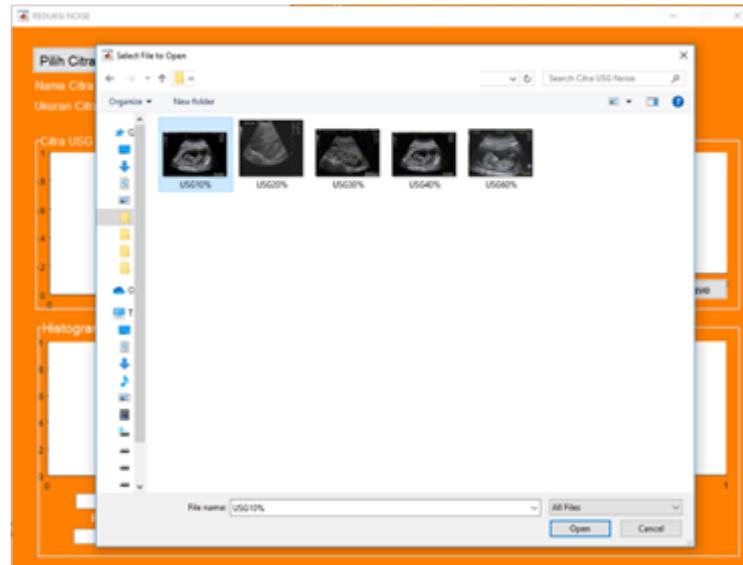
1. Proses Data 1

Adapun tahapanya adalah sebagai berikut :



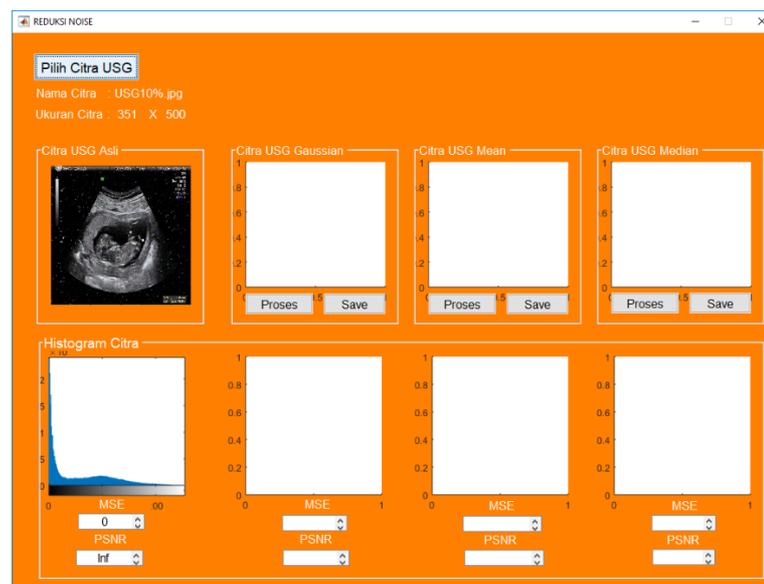
Gambar 4.3 Proses Pemilihan Citra USG

Berdasarkan pada gambar di atas, pemilihan citra USG dapat dilakukan dengan menekan *button* “Pilih Citra USG” sehingga menampilkan menu direktori pencarian citra USG seperti pada gambar di bawah ini:



Gambar 4.4 Memilihkan Citra USG

Berdasarkan pada gambar di atas, citra yang dipilih adalah USG10% kemudian pilih *button open* untuk menampilkannya kedalam aplikasi, sehingga menghasilkan seperti pada gambar di bawah ini:



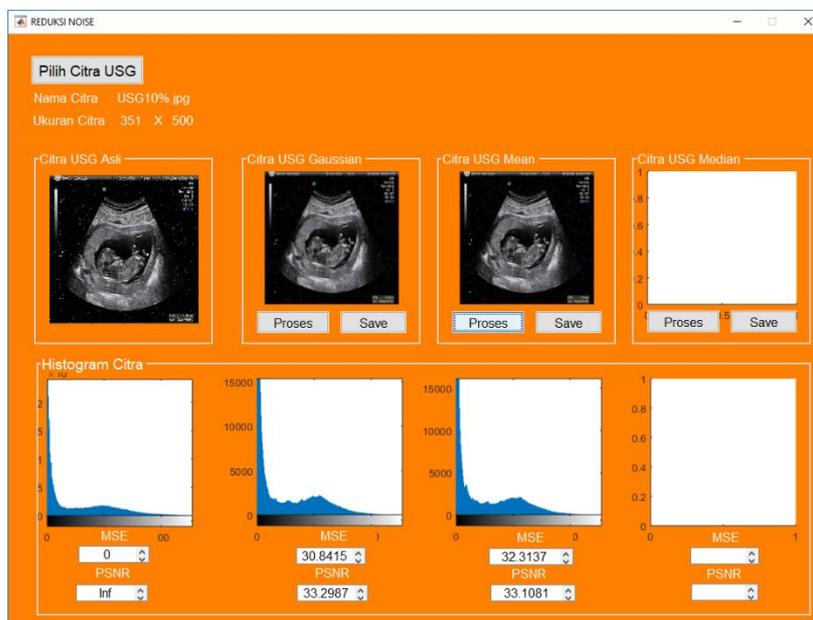
Gambar 4.5 Informasi Citra USG10%

Berdasarkan pada gambar di atas, ketika telah dipilih, sistem akan menampilkan informasi citra seperti nama citra, ukuran citra, bentuk histogram serta nilai MSE dan PSNR. Selanjutnya untuk memulai proses reduksi citra dengan menggunakan metode *gaussian* dengan menekan *button proses* pada panel *gaussian*, sehingga menghasilkan seperti pada gambar di bawah ini:



Gambar 4.6 Informasi Citra USG10% Reduksi *Noise Gaussian*

Berdasarkan pada gambar di atas, sistem menampilkan informasi citra hasil reduksi dengan metode *gaussian* seperti bentuk histogram serta nilai MSE dan PSNR. Adapun nilai MSE yang didapatkan adalah 30.84 sedangkan nilai PSNR adalah 33.29 untuk reduksi citra USG *noise* 10% menggunakan metode *gaussian*. Selanjutnya dilakukan reduksi *noise* dengan metode *mean* menggunakan citra USG *noise* 10% dengan hasil seperti berikut :



Gambar 4.7 Informasi Citra USG10% Reduksi *Noise Mean*

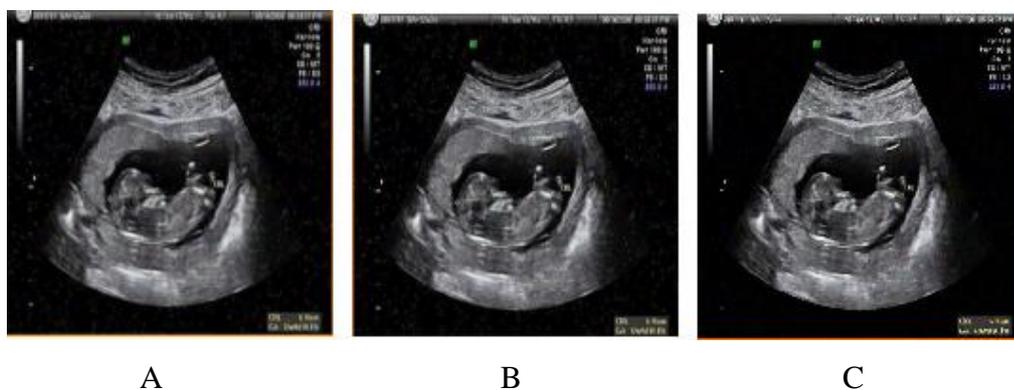
Berdasarkan pada gambar di atas, sistem menampilkan informasi citra hasil reduksi dengan metode *mean* seperti bentuk histogram serta nilai MSE dan PSNR. Adapun nilai

MSE yang didapatkan adalah 32.31 sedangkan nilai PSNR adalah 33.10 untuk reduksi citra USG *noise* 10% menggunakan metode *mean*. Selanjutnya dilakukan reduksi *noise* dengan metode *median* menggunakan citra USG *noise* 10% dengan hasil seperti berikut:



Gambar 4.8 Informasi Citra USG10% Reduksi *Noise Median*

Berdasarkan pada gambar di atas, sistem menampilkan informasi citra hasil reduksi dengan metode *median* seperti bentuk histogram serta nilai MSE dan PSNR. Adapun nilai MSE yang didapatkan adalah 28.04 sedangkan nilai PSNR adalah 36.82 untuk reduksi citra USG *noise* 10% menggunakan metode *median*. Sehingga hasil keseluruhannya reduksi *noise* citra USG *noise* 10% seperti berikut:



Gambar 4.9 Hasil Pebandingan Reduksi *Noise* Citra USG 10%

Berdasarkan pada gambar di atas, A adalah citra hasil reduksi *gaussian*, B adalah citra hasil reduksi *mean* dan C adalah citra hasil reduksi *median*. Secara penghilatan visual,

citra hasil reduksi metode median lebih baik. Akan tetapi secara nilai PSNR, citra hasil reduksi *gaussian* lebih baik.

2. Proses Data 2

Adapun tahapannya yang dilakukan sama seperti sebelumnya, sehingga didapati hasil akhir reduksi citra USG dengan *noise* 20% sebagai berikut :



Gambar 4.10 Citra USG 20% Reduksi *Noise Gaussian, Mean, Median*

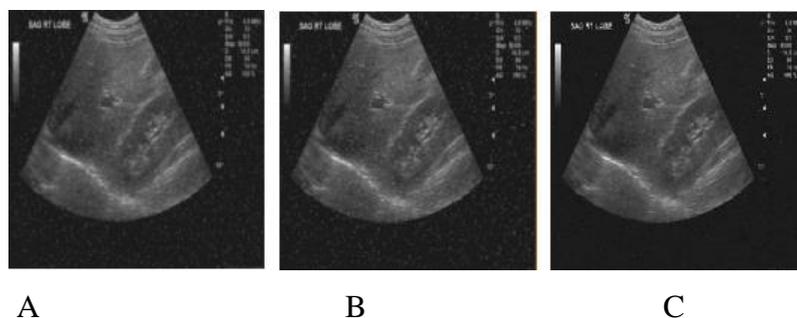
Berdasarkan pada gambar di atas, sistem menampilkan informasi citra hasil reduksi dengan metode *gaussian, mean* dan *median* seperti bentuk histogram serta nilai MSE dan PSNR. Adapun nilai MSE yang didapatkan adalah:

Gaussian = MSE = 25.11, PSNR = 34.16

Mean = MSE = 28.40, PSNR = 33.63

Median = MSE = 24.36, PSNR = 37.30

Sehingga hasil keseluruhannya reduksi *noise* citra USG *noise* 20% seperti berikut:



Gambar 4.11 Hasil Pebandingan Reduksi *Noise* Citra USG 20%

Berdasarkan pada gambar di atas, A adalah citra hasil reduksi *gaussian*, B adalah citra hasil reduksi *mean* dan C adalah citra hasil reduksi *median*. Secara penghilatan visual,

citra hasil reduksi metode median lebih baik, hal ini ditandai dengan nilai PSNR citra hasil reduksi *median* lebih tinggi.

3. Proses Data 3

Adapun tahapanya yang dilakukan sama seperti sebelumnya, sehingga didapati hasil akhir reduksi citra USG dengan *noise* 30% sebagai berikut :



Gambar 4.12 Citra USG 30% Reduksi *Noise Gaussian, Mean, Median*

Berdasarkan pada gambar di atas, sistem menampilkan informasi citra hasil reduksi dengan metode *gaussian*, *mean* dan *median* seperti bentuk histogram serta nilai MSE dan PSNR. Adapun nilai MSE yang didapatkan adalah:

Gaussian = MSE = 41.79, PSNR = 32.30

Mean = MSE = 44.08, PSNR = 33.63

Median = MSE = 45.01, PSNR = 34.69

Sehingga hasil keseluruhanya reduksi *noise* citra USG *noise* 30% seperti berikut:



A

B

C

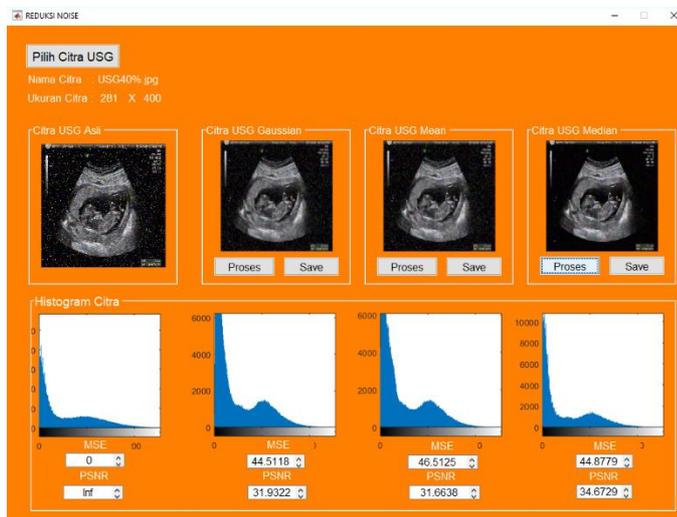
Gambar 4.13 Hasil Pebandingan Reduksi *Noise* Citra USG 30%

Berdasarkan pada gambar di atas, A adalah citra hasil reduksi *gaussian*, B adalah citra hasil reduksi *mean* dan C adalah citra hasil reduksi *median*. Secara penghilatan visual,

citra hasil reduksi metode median lebih baik, hal ini ditandai dengan nilai PSNR citra hasil reduksi *median* lebih tinggi.

4. Proses Data 4

Adapun tahapanya yang dilakukan sama seperti sebelumnya, sehingga didapati hasil akhir reduksi citra USG dengan *noise* 40% sebagai berikut :



Gambar 4.14 Citra USG 40% Reduksi *Noise Gaussian, Mean, Median*

Berdasarkan pada gambar di atas, sistem menampilkan informasi citra hasil reduksi dengan metode *gaussian, mean* dan *median* seperti bentuk histogram serta nilai MSE dan PSNR. Adapun nilai MSE yang didapatkan adalah:

Gaussian = MSE = 44.51, PSNR = 31.93

Mean = MSE = 46.51, PSNR = 31.66

Median = MSE = 44.87, PSNR = 34.67

Sehingga hasil keseluruhanya reduksi *noise* citra USG *noise* 40% seperti berikut:



A

B

C

Gambar 4.15 Hasil Pebandingan Reduksi *Noise* Citra USG 40%

Berdasarkan pada gambar di atas, A adalah citra hasil reduksi *gaussian*, B adalah citra hasil reduksi *mean* dan C adalah citra hasil reduksi *median*. Secara penghilatan visual,

citra hasil reduksi metode median lebih baik, hal ini ditandai dengan nilai PSNR citra hasil reduksi *median* lebih tinggi.

5. Proses Data 5

Adapun tahapanya yang dilakukan sama seperti sebelumnya, sehingga didapati hasil akhir reduksi citra USG dengan *noise* 60% sebagai berikut :



Gambar 4.16 Citra USG 60% Reduksi *Noise Gaussian, Mean, Median*

Berdasarkan pada gambar di atas, sistem menampilkan informasi citra hasil reduksi dengan metode *gaussian, mean* dan *median* seperti bentuk histogram serta nilai MSE dan PSNR. Adapun nilai MSE yang didapatkan adalah:

Gaussian = MSE = 66.02, PSNR = 29.96

Mean = MSE = 69.93, PSNR = 29.72

Median = MSE = 61.21, PSNR = 33.30

Sehingga hasil keseluruhanya reduksi *noise* citra USG *noise* 60% seperti berikut:



Gambar 4.17 Hasil Pebandingan Reduksi *Noise* Citra USG 60%

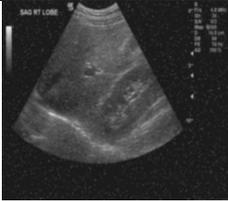
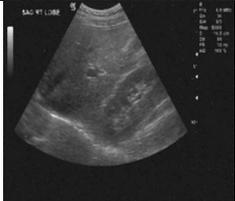
Berdasarkan pada gambar di atas, A adalah citra hasil reduksi *gaussian*, B adalah citra hasil reduksi *mean* dan C adalah citra hasil reduksi *median*. Secara penghilatan visual,

citra hasil reduksi metode median lebih baik, hal ini ditandai dengan nilai PSNR citra hasil reduksi *median* lebih tinggi.

4.2.2 Hasil Pengujian Sistem

Berdasarkan dari hasil pengujian sistem, didapatkan bahwa metode *median* lebih baik dalam mereduksi citra USG dengan *visual* hasil reduksi yang lebih baik ditandai dengan nilai PSNR lebih tinggi. Berikut adalah hasil pengujian sistem yang dapat dilihat pada tabel di bawah ini:

Tabel 4.4 Hasil Pengujian

No	Nama Data	Hasil Reduksi Noise		
		<i>Gaussian</i>	<i>Mean</i>	<i>Median</i>
1	Data 1	 MSE = 30.84 PSNR = 33.29	 MSE = 32.31 PSNR = 33.10	 MSE = 28.04 PSNR = 36.82
2	Data 2	 MSE = 25.11 PSNR = 34.16	 MSE = 28.40 PSNR = 33.63	 MSE = 24.34 PSNR = 37.30
3	Data 3	 MSE = 41.79 PSNR = 32.30	 MSE = 44.08 PSNR = 31.97	 MSE = 45.01 PSNR = 34.69

Tabel 4.4 Hasil Pengujian (Lanjutan)

No	Nama Data	Hasil Reduksi Noise		
		<i>Gaussian</i>	<i>Mean</i>	<i>Median</i>
4	Data 4	 MSE = 44.51 PSNR = 31.93	 MSE = 46.51 PSNR = 31.66	 MSE = 44.87 PSNR = 34.67
5	Data 5	 MSE = 66.02 PSNR = 29.96	 MSE = 69.93 PSNR = 29.72	 MSE = 61.21 PSNR = 33.30

4.2.3 Perbandingan Hasil Pengujian

Untuk melihat perbedaan *noise*, berikut adalah berbanding hasil pengujian berdasarkan metode *gaussian*, *mean* dan *median* dengan citra sebelumnya:

1. Perbandingan Hasil *Gaussian*

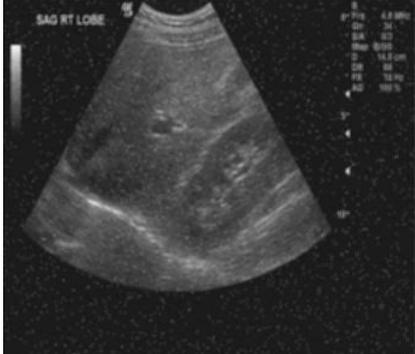
Berikut adalah hasil perbandingan citra USG sebelum dan sesudah diproses dengan metode *gaussian*:

Tabel 4.5 Perbandingan Hasil *Gaussian*

Nama Data	Perbandingan	
	Sebelum	Sesudah
Data 1		

Berikut adalah hasil perbandingan citra USG sebelum dan sesudah diproses dengan metode *Mean*:

Tabel 4.6 Perbandingan Hasil *Mean*

Nama Data	Perbandingan	
	Sebelum	Sesudah
Data 1		
Data 2		
Data 3		
Data 4		

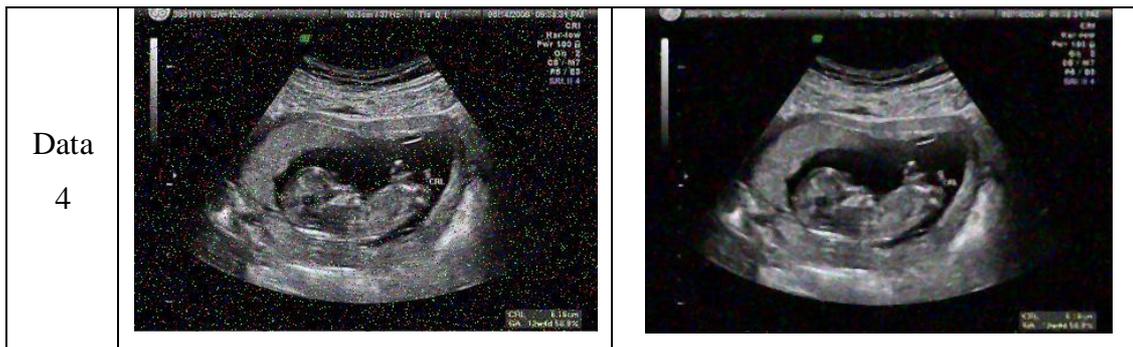


3. Perbandingan Hasil *Median*

Berikut adalah hasil pebandingan citra USG sebelum dan sesudah diproses dengan metode *Median*:

Tabel 4.7 Perbandingan Hasil *Median*

Nama Data	Perbandingan	
	Sebelum	Sesudah
Data 1		
Data 2		
Data 3		



Tabel 4.7 Perbandingan Hasil Median (Lanjutan)

<p>Nama Data</p>	<p>Perbandingan</p>	
	<p>Sebelum</p>	<p>Sesudah</p>
<p>Data 5</p>		

BAB 5

KESIMPULAN DAN SARAN

5.1 Kesimpulan

Dari hasil penelitian yang dilakukan terhadap peningkatan citra USG dengan membandingkan metode *gaussian*, *mean* dan *median*, maka terdapat beberapa kesimpulan berdasarkan uraian yang telah tercantum pada bab-bab sebelumnya. Adapun kesimpulan dari hasil penelitian ini adalah sebagai berikut:

1. Berdasarkan dari hasil pengujian perbandingan kualitas metode *aussian filter*, *median filter* dan *mean filter* yang digunakan dalam mereduksi *noise*, didapatkan metode *median filter* yang paling baik dengan PSNR paling tinggi yaitu 37,30 dan MSE paling rendah yaitu 24,34.
2. Sistem yang dibangun menggunakan program matlab 2017b dapat diimplementasikan untuk membandingkan metode *gaussian filter*, *median filter* dan *mean filter* terhadap citra USG *noise*.
3. Pada citra USG yang memiliki *noise* lebih dari 50%, perbandingan visual citra hasil reduksi masih banyak meninggalkan *noise*, baik metode *gaussian*, *mean* dan *median*, hal ini dapat ditandai dengan nilai MSE yang lebih besar.

5.2 Saran

Adapun saran-saran yang usulkan adalah sebagai berikut:

1. Dengan adanya rancangan aplikasi ini, maka diharapkan dapat memberikan kontribusi bagi peneliti lain serta pengembangan aplikasi yang lebih baik.
2. Diharapkan adanya pengembangan lebih lanjut, dapat ditambahkan metode perbandingan yang lain, atau melakukan reduksi citra dengan penggabungan metode untuk hasil citra USG yang lebih baik.
3. Diharapkan aplikasi dapat dikembangkan menjadi aplikasi yang berbasis *web* atau *android* agar penggunaan lebih fleksibel.

DAFTAR PUSTAKA

- Afifa. Z. 2016. Implementasi Metode Gaussian Filter untuk Menghapus Noise Pada Citra Menggunakan GPU. Skripsi. Malang. Universitas Islam Negeri Malang.
- Afiyat. N. 2017. Analisis Restorasi Citra Kabur Algoritma Wiener Menggunakan Indeks Kualitas Citra. Nusantara Jurnal Of Computers and Its Application. Volume 2 Nomor 1, 24 – 34.
- Andiro. T. & Ginting, G. 2019. Peningkatan Kualitas Citra Ultrasonografi (USG) dengan Menggunakan Metode Gaussian Filter. Jurnal Pelita Informatika. Volume 18 Nomor 1. 121-126.
- Furqon. M., Sriani., & Siregar. Y.K. 2020. Perbandingan Algoritma Contraharmonic Mean Filter Dan Arithmetic Mean Filter untuk Mereduksi Exponential Noise. JISKA. Volume 5 Nomor 2, 107-115.
- Jannah. A. 2008. Analisa perbandingan metode filter gaussian, mean dan median terhadap reduksi noise salt & pappers. Skripsi. Malang. Universitas Islam Negeri Malang.
- Jogiyanto. 2007. Analisis dan Desain Sistem Informasi. Yogyakarta: ANDI
- Kadir. A., & Susanto, A., 2013. Teori dan Aplikasi pengolahan citra, Yogyakarta : penerbit Andi.
- Khilmawan. M.R., & Riadi. A.A. 2018. Implementasi Pengurangan Noise Pada Citra Tulang Menggunakan Metode Median Filter dan Gaussian Filter. Volume 3 Nomor 2, 116-121
- Nafi'iyah. N. 2015. Algoritma Kohonen Dalam Mengubah Citra Graylevel Menjadi Citra Biner. JITIKA. Volume 9 Nomor 2, 49-55.
- Pahlawan. D.A., Wibowo, A.S & Susanto, E. 2020. Perancangan Kendali Pada Bola di Atas Bidang Data Ball On Plate Control Design. Jurnal Engineering. Volume 7 Nomor 3, 1-13.
- Prasetyo. E. 2011. Pengolahan Citra Digital dan Aplikasinya Menggunakan Matlab. Yogyakarta: Penerbit ANDI
- Putra. D. 2009. Pengolahan Citra Digital. Yogyakarta: Penerbit ANDI.
- Tamtama.G.I.W. 2021. Perbandingan dan Analisis Untuk Algoritma Deteksi Tepi Pada Jaringan Saraf Tiruan. CESS. Volume 6 Nomor 1, 67-71.

- Riandari. F. 2018. Implementasi Metode Geometric Mean Filter Untuk Perbaikan Dengan Reduksi Noise Pada Citra Digital. Volume 2 Nomor 2, 175-179.
- Rosa. A.S and Shalahudin, M. 2015. Rekayasa Perangkat Lunak Terstruktur dan Berorientasi Objek. Bandung: Penerbit Informatika.
- Santoso., & Nurmalina, R. 2017. Perencanaan dan Pengembangan Aplikasi Absensi Mahasiswa Menggunakan Smart Card Guna Pengembangan Kampus Cerdas. Jurnal Integrasi, pp.84-91.
- Simangunsong. P.B.N. 2017. Reduksi Noise Pada Citra Digital Menggunakan Metode Arithmetic Mean Filter. Jurnal Teknik Informatika. Volume 2 Nomor 2, 60-63.
- Sriani., Triase, & khairuna. 2017. Pendekomposisian Citra Digital Dengan Algoritma DWT. ALGORITMA : Jurnal Ilmu Computer Dan Informatika. Volume 01 nomor 01, 35-39.
- Sutoyo. T., dkk. 2009. Teori Pengolahan Citra Digital. Penerbit Andi Offset. Yogyakarta*
- Wedianto. A. Sari, H.L. & Suzantri, H.Y. Februari 2016. Analisa perbandingan metode filter gaussian, mean dan median terhadap reduksi noise. Jurnal media infotama Volume 12 Nomor 1.
- Yusro, K.A & Sianturi, R.D. 2018. Penerapan Metode Median Filtering Dan Histogram Equalization Untuk Meningkatkan Kualitas Citra Radiografi. Jurnal Riset Komputer. Volume 5 Nomor 3. Hal 256-260.