

ANALISIS PERBANDINGAN METODE INTENSITY FILTERING DENGAN METODE FREQUENCY FILTERING SEBAGAI REDUKSI NOISE PADA CITRA DIGITAL

Murinto, Eko Aribowo, Risnadi Syazali

Program Studi Teknik Informatika, Universitas Ahmad Dahlan Jogjakarta

Jln. Prof. Dr. Soepomo, Janturan, Jogjakarta

Telp. (0274) 379418; e-mail: rintokusno@yahoo.com

ABSTRAKSI

Citra memegang peranan sangat penting sebagai bentuk informasi. Namun kadang pada citra terdapat gangguan (biasa disebut noise) yang menyebabkan citra tersebut tidak dapat dilihat dengan jelas karena adanya noise. Metode yang biasa digunakan untuk mereduksi noise adalah intensity filtering dan frequency filtering. Pada penelitian ini akan dijelaskan karakteristik dari kedua metode tersebut, serta analisis kelebihan dan kekurangannya. Hasil penelitian menunjukkan reduksi noise menggunakan metode high pass frequency filtering lebih mengurangi noise pada citra dibandingkan dengan metode lain namun memerlukan timing-run yang lama.

Kata kunci: frequency filtering, intensity filtering, reduksi noise.

1. PENDAHULUAN

Citra (*image*), merupakan istilah lain untuk gambar. Sebagai salah satu komponen multimedia, citra memegang peranan sangat penting sebagai bentuk informasi visual. Seiring dengan perkembangan teknologi di bidang komputerisasi, teknologi pengolahan citra (*image processing*) telah banyak dipakai di berbagai bidang termasuk di bidang industri, yang dapat membantu pengerjaan tugas sehingga dapat diperoleh hasil yang lebih efisien dan dengan akurasi yang baik.

Pada dasarnya setiap sistem pencitraan dapat menyebabkan terjadinya derau (*noise*) pada citra yang dihasilkan. *Noise* pada citra tersebut pada umumnya terdistribusi secara normal (*Gaussian*) (Balza, Kartika, 2004). Pengurangan *noise* atau *denoise* adalah salah satu proses dalam peningkatan kualitas citra (*image enhancement*) yang termasuk langkah awal dalam *image processing*. Peningkatan kualitas citra adalah proses mendapatkan citra yang lebih mudah diinterpretasikan oleh mata manusia. Pada proses ini, ciri-ciri tertentu yang terdapat di dalam citra lebih diperjelas kemunculannya.

Proses-proses yang termasuk dalam peningkatan kualitas citra antara lain pengubahan kecerahan gambar (*image brightness*), perenggangan kontras (*image stretching*), pengubahan histogram citra, pengurangan derau (*denoising*), penajaman tepi (*edge sharpening*), pewarnaan semu (*pseudocolouring*) dan pengubahan geometrik. Salah satu cara yang digunakan untuk *image processing* adalah dengan mereduksi derau (*noise reduction*) yang terdapat pada citra sehingga menjadi citra lain yang kualitasnya lebih baik dibandingkan dengan kondisi awal, sehingga mudah diinterpretasi baik itu oleh manusia maupun mesin.

Banyak cara dan metode yang digunakan dalam pengurangan derau. Metode tersebut antara lain operasi aritmatik (aljabar), transformasi *wavelet*,

metode *contour*, metode *intensity filtering* dan metode *frequency filtering*. Untuk metode *intensity filtering* dan metode *frequency filtering* sama-sama bekerja pada level piksel citra. Selama ini belum ada penelitian yang menjelaskan karakteristik dari kedua metode tersebut dalam lingkup pengurangan *noise*. Perlu dilakukan analisis kelebihan dan kekurangan dari kedua metode tersebut. Metode *intensity filtering* dapat mendeteksi piksel dengan *high pass intensity* dan *low pass intensity*, begitu pula dengan metode *frequency filtering* mendeteksi piksel citra dengan *high pass frequency* dan *low pass frequency*. Dari keempat metode tersebut akan memberikan hasil pengurangan *noise* yang berbeda.

Dari penelitian sebelumnya yang dilakukan (Perwitasari, E. W, 2006) Penelitian ini membahas tentang kelebihan dan kekurangan dari metode transformasi *wavelet* dengan metode *contour*, dua metode tersebut memberikan hasil pengurangan *noise* yang berbeda. Operasi pengurangan *noise* bekerja dengan cara menekan komponen yang berfrekuensi tinggi dan meloloskan komponen yang berfrekuensi rendah. Aplikasi ini menggunakan Matlab 7.0.1 dan hanya dipakai untuk jenis citra *grayscale* berekstensi .png berukuran 512 x 512 piksel dan 256 x 256 piksel.

Dalam penelitian lain yang dilakukan oleh (Nalwan, A., 1997) sistem yang dibuat membahas mengenai cara pembersihan *noise* dengan mengetahui jenis *noise* yang menggabung dalam citra menggunakan metode *intensity filtering*. Setelah mengetahui jenis *noise* ini barulah dilakukan proses pembersihan derau yang cocok dengan *noise* tersebut. Salah satu pembersihan *noise* adalah dengan mendeteksi intensitas dari setiap titik di layar. Cara lain pembersihan *noise* adalah dengan menganalisis jumlah *noise* yang ada pada gambar. Dalam hal ini kita asumsikan bahwa titik-titik *noise*

yang berintensitas sama memiliki jumlah yang hampir sama.

Pada penelitian kali ini akan dibahas tentang pengurangan *noise* untuk meningkatkan mutu citra menggunakan metode *intensity filtering* dan *frequency filtering*, yang kemudian hasil akhirnya dianalisis. Menganalisis metode yang digunakan secara spesifik diharapkan diperoleh perbandingan yang menyatakan kekurangan dan kelebihan masing-masing metode. Penelitian ini menggunakan bahasa pemrograman Borland Delphi 7.0.

2. LANDASAN TEORI

2.1 Noise dan Pengurangan Noise (Denoise)

Noise adalah titik-titik pada citra yang sebenarnya bukan merupakan bagian dari citra, melainkan ikut tercampur pada citra karena suatu sebab. Ada tiga macam *noise*, yaitu:

a. Noise Aditif

Noise aditif adalah *noise* yang bersifat menambahkan secara seragam pada sebuah bidang citra dengan varian tertentu. Contoh *noise* ini adalah *noise salt-and-peppers* yang menambahkan aras gelap dan terang pada citra.

b. Noise Gaussian

Noise ini memiliki intensitas yang sesuai dengan distribusi normal yang memiliki rerata (*mean*) dan varian tertentu.

c. Noise Speckle

Noise ini muncul pada saat pengambilan citra tidak sempurna karena alasan cuaca, perangkat pengambil citra dan sebagainya. Sifat *noise* ini *multipikatif*, artinya semakin besar intensitas citra atau semakin cerah citra, semakin jelas juga *noise*.

Noise muncul biasanya sebagai akibat dari pembelokkan yang tidak bagus (*sensor noise*, *photographic gain noise*). Gangguan tersebut umumnya berupa variasi intensitas suatu piksel yang tidak berkorelasi dengan piksel-piksel tetangganya. Secara visual, gangguan mudah dilihat oleh mata karena tampak berbeda dengan piksel tetangganya. Piksel yang mengalami gangguan umumnya memiliki frekuensi tinggi. Komponen citra yang berfrekuensi rendah umumnya mempunyai nilai piksel konstan atau berubah sangat lambat. Operasi *denoise* dilakukan untuk menekan komponen yang berfrekuensi tinggi dan meloloskan komponen yang berfrekuensi rendah (Munir, R., 2004).

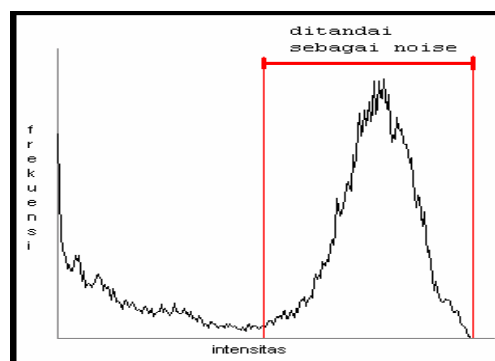
2.2 Intensity filtering

Dengan metode ini, *noise* akan di reduksi dengan mendeteksi intensitas dari setiap titik di layar. Cara ini akan efektif apabila banyak titik *noise* yang memiliki warna sama pada titik-titik pada gambar asli. *Intensity filtering* akan bagus digunakan apabila terdapat sedikit titik-titik *noise* yang memiliki warna sama dengan gambar asli. Terdapat dua macam *noise reduction* pada *intensity filtering*,

yaitu *high pass filtering* dan *low pass filtering* (Nalwan, A., 1997).

2.3 High pass intensity filtering

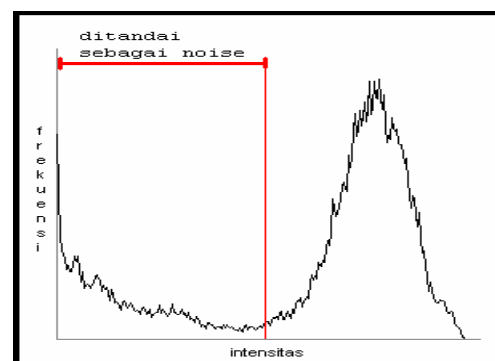
High pass intensity filtering digunakan jika *noise* diketahui memiliki intensitas warna tinggi. Misalnya *noise* berwarna 220-255, maka dilakukan pendeteksian untuk setiap titik yang memiliki warna antara 220 hingga 255 akan ditandai sebagai *noise*.



Gambar 1. High intensity noise

2.4 Low pass intensity filtering

Low pass intensity filtering digunakan pada gambar yang memiliki intensitas warna yang rendah. Karena letak *noise* berada di intensitas rendah, maka dilakukan pencarian pada titik-titik gambar dan kemudian akan ditandai sebagai *noise*. Selanjutnya titik tersebut akan diganti dengan mencari warna rata-rata di sekitar titik tersebut.



Gambar 2. Low intensity noise

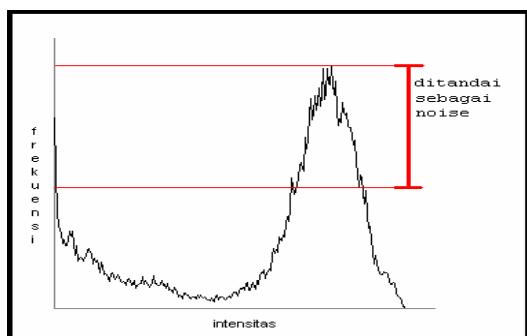
2.5 Frequency filtering

Apabila *noise* bercampur dengan gambar asli secara merata, diperlukan metode ini untuk menganalisis jumlah *noise* yang ada pada gambar. Dengan metode ini kita mengetahui jumlah titik yang memiliki intensitas warna yang sama. Apabila terdapat titik yang berada pada range tertentu akan dianggap sebagai *noise*. Namun kadang intensitas warna titik pada gambar asli berada pada range tersebut yang menyebabkan titik tersebut dapat ikut terfilter karena dihitung sebagai *noise* (Nalwan, A., 1997).

2.6 High pass frequency filtering

Filter ini dilakukan apabila titik-titik *noise* menggunakan sebuah intensitas warna dengan jumlah yang cenderung lebih tinggi, maka titik-titik yang mempergunakan intensitas tersebut adalah *noise*. Proses pendeteksian *noise* dilakukan dengan memasukkan jumlah titik yang menggunakan suatu intensitas warna (frekuensi intensitas) ke dalam sebuah array, kemudian akan diketahui titik-titik *noise* yang memiliki frekuensi penggunaan warna tertentu lebih banyak daripada gambar asli.

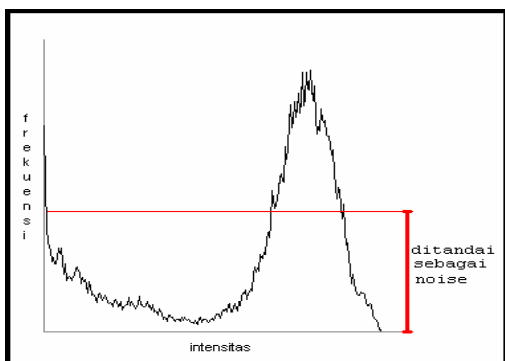
Teknik akan lebih efektif digunakan pada titik-titik gambar yang memiliki frekuensi intensitas warna tertentu tidak lebih sebanyak intensitas warna pada titik *noise*. Apabila warna pada gambar memiliki ketinggian frekuensi yang sama dengan *noise* maka titik tersebut akan ikut tersaring. Akan lebih baik bila digunakan pada gambar/citra yang memiliki banyak variasi warna.



Gambar 3. High frequency noise

2.7 Low pass frequency filtering

Pada kondisi ini titik-titik *noise* menggunakan intensitas warna dengan frekuensi lebih rendah daripada intensitas warna pada gambar asli. Pendeteksian *noise* kebalikan dari teknik *high pass frequency filtering*, dimana intensitas warna dengan frekuensi lebih rendah daripada *range* akan dianggap sebagai *noise*. Hasil filter akan baik apabila gambar tidak memiliki variasi warna yang banyak, karena gambar yang memiliki warna yang banyak akan cenderung menggunakan banyak intensitas warna dengan frekuensi masing-masing yang rendah.



Gambar 4. Low frequency noise

2.8 SNR (Signal to-Noise Ratio)

SNR digunakan untuk menentukan kualitas citra setelah dilakukan operasi pengurangan *noise*. Citra hasil dibandingkan dengan citra asli untuk memberi perkiraan kasar kualitas citra hasil. Semakin besar nilai SNR berarti pengurangan *noise* dapat meningkatkan kualitas citra, sebaliknya jika nilai SNR semakin kecil maka pada citra hasil hanya sedikit juga peningkatan kualitasnya.

Nilai SNR yang tinggi adalah lebih baik karena berarti rasio sinyal terhadap *noise* juga tinggi, dimana sinyal adalah citra asli. SNR biasanya diukur dengan satuan *decibels* (dB).

Rumus untuk menghitung SNR dapat dilihat dalam persamaan berikut:

$$SNR = 10 \cdot \text{Log}_{10} \frac{\sum_{m,n} I_{m,n}^2}{\sum_{m,n} (I_{m,n} - \bar{I}_{m,n})^2}$$

dimana:

$I_{m,n}$ adalah citra asli, $\bar{I}_{m,n}$ adalah citra filter, sedangkan m,n adalah ukuran citra

3. METODE

Perancangan dan pembuatan aplikasi dilakukan dengan langkah-langkah:

1. Perancangan Diagram Alir Sistem
Perancangan ini akan menggambarkan diagram alir (*Flow chart*) perangkat lunak (*software*) pengolahan citra dengan operasi *noise reduction* menggunakan metode *intensity filtering* dan metode *frequency filtering*.
2. Perancangan Interface
Perancangan *interface* meliputi perancangan masukan (*input*) dan perancangan keluaran(*output*).

4. PENGUJIAN

Pada penelitian menggunakan 20 citra uji. Citra tersebut di-load dan ditampilkan pada program. Kemudian dilakukan proses reduksi *noise* menggunakan metode *intensity filtering* dan metode *frequency filtering* baik secara *high pass filtering* maupun *low pass filtering*, serta menampilkan histogramnya. Parameter yang dicatat adalah hasil reduksi *noise* secara visual mata, *timing-run* dan *signal to-noise ratio* (SNR).

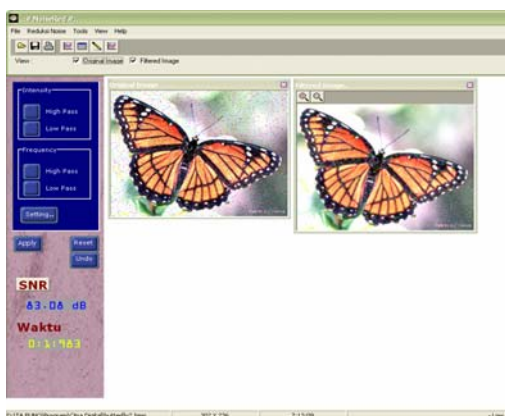
Uji coba yang dilakukan memberikan data-data berdasar parameter yang telah ditetapkan. Parameter yang digunakan untuk perbandingan pada penelitian ini adalah pengurangan derau, *timing run*, histogram dan SNR (*Signal to-Noise Reduction*). Parameter-parameter yang digunakan akan diterapkan pada beberapa citra yang berbeda dan akan ditampilkan hasilnya, kemudian data yang dihasilkan tersebut dianalisis sehingga didapat

kelebihan dan kekurangan dari masing-masing metode berdasar parameter yang digunakan.

5. HASIL DAN PEMBAHASAN

Jenis citra *input* adalah citra *bitmap* berekstensi *.bmp dengan ukuran 8 bit dan 24 bit, serta skala maksimal 640 X 480 piksel. *Output* yang dihasilkan adalah citra hasil pengurangan *noise* dengan metode *High Pass Intensity Filtering*, *Low Pass Intensity Filtering*, *High Pass Frequency Filtering* dan *Low Pass Frequency Filtering*, serta *timing-run*, histogram citra dan SNR.

Pada aplikasi reduksi *noise* (Gambar 5) ini menggunakan Metode *Intensity Filtering* dan *Frequency Filtering* dibuat dengan bahasa pemrograman Borland Delphi7.0. Tahap implementasi merupakan tahap yang akan membangun sebuah sistem berdasarkan atas analisis kebutuhan sistem yang telah dirancang sehingga akan dihasilkan sistem yang dapat menghasilkan tujuan yang akan dicapai. Aplikasi hanya untuk tipe data berupa tipe *bitmap* dan ukuran citra maksimum adalah 640 X 480 piksel. Selain itu kemungkinan akan terjadi *error*.



Gambar 6. Tampilan program reduksi noise

5.1 Pengurangan Noise

Secara visual dengan mata kita, hasil yang diperoleh dari pengurangan *noise* menggunakan *Intensity Filtering* dan *Frequency Filtering* pada citra sampel tidak lah berbeda jauh, karena masih terdapat *noise*. Namun jika diamati, hasil dari pengurangan *noise* menggunakan *High Pass Intensity Filtering* masih terdapat *noise* berupa titik-titik biru yang kabur dan tersebar di seluruh bagian citra, tetapi tepian dari citra masih terlihat. Hasil dari pengurangan *noise* menggunakan *Low Pass Intensity Filtering* juga masih terdapat *noise* berupa titik-titik kuning yang masih nampak serta titik-titik biru yang sudah kabur dan tersebar di seluruh bagian citra, namun tepian citra sudah agak kabur. Hasil pengurangan *noise* menggunakan *High Pass Frequency Filtering* sudah menghasilkan citra yang menghilangkan *noise* hampir seluruhnya, tapi tepian dari citra masih nampak. Hasil pengurangan *noise* menggunakan *Low Pass Frequency Filtering*

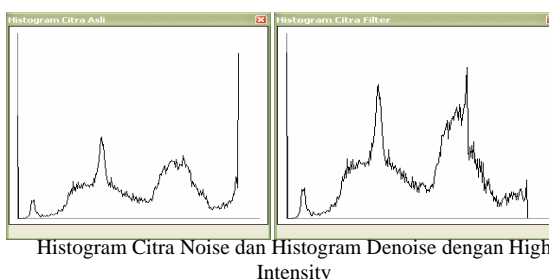
menghasilkan citra yang masih banyak *noise* nya, serta menghasilkan gambar yang kabur.



Gambar 7. Hasil pengurangan noise dengan highpass Intensity, Frecuency filtering, low pass filtering, low pass intensity filtering

5.2 Histogram

Aplikasi ini menampilkan histogram citra asli dan citra *filter*. Dari ke empat metode mempunyai banyak kesamaan, yaitu dari lebar histogram yang terdistribusi secara merata keseluruhan daerah, baik dari derajat keabuan maupun derajat *true-color*. Perbedaan terdapat pada daerah tumpukan histogram (gunung dan lembah).



Histogram Citra Noise dan Histogram Denoise dengan High Intensity

Gambar 8. Histogram Citra Cat_Fish.bmp

Tumpukan histogram citra hasil pengurangan *noise* menggunakan metode *intensity filtering* maupun metode *frequency filtering* terhadap citra *noise* tidak terdapat perbedaan yang mencolok. Hal ini ditunjukkan dengan tumpukan histogram yang terdistribusi merata keseluruhan daerah, dengan distribusi yang merata pada setiap nilai intensitas piksel yang berarti tingkat kecerahan gambar tetap terjaga. Perbedaan pada puncak histogram yang nampak terjadi terdapat sewaktu menggunakan metode *high pass intensity filtering*. Pada metode ini mengakibatkan puncak histogram meningkat lebih tinggi daripada puncak histogram citra asli, sehingga batas tepi citra menjadi lebih jelas namun titik-titik *noise* juga semakin membesar. Pada

Penggunaan metode *low pass intensity filtering* dan metode *high pass frequency filtering* tidak terjadi perubahan puncak histogram yang terlalu signifikan dibandingkan dengan citra aslinya. Sehingga citra yang didapat juga hampir sama dengan citra aslinya, namun metode *low pass frequency filtering*, meskipun tidak menonjolkan dengan *noise* yang lebih sedikit. Lain halnya dengan penggunaan puncak pada histogramnya, namun

histogram secara hampir keseluruhan menjadi tidak rapi, hal ini menyebabkan citra hasil menjadi lebih kabur daripada citra asli.

5.3 Timing-run

Timing-run ini adalah lama waktu proses reduksi *noise* pada suatu citra. Dari data *timing-run* diatas, kita dapat menghitung nilai rata-rata dari tiap metode. Dengan menjumlahkan nilai tiap citra dari suatu metode lalu membaginya dengan jumlah sampel citra yang ada, sehingga didapat nilai rata-rata nya yaitu:

Tabel 1. Nilai Rata-Rata *Timing-run Denoise*

<i>Metode Denoise</i>	<i>Waktu Proses Rata-Rata (detik)</i>
High Pass Intensity Filtering	2.612
Low Pass Intensity Filtering	2.576
High Pass Frequency Filtering	3.881
Low Pass Frequency Filtering	3.856

Berdasar pada tabel di atas maka diperoleh *timing-run* terbaik dari ke empat metode adalah *timing-run* dengan menggunakan metode *low pass intensity filtering*.

Secara garis besar terdapat perbedaan waktu proses yang besar antara pengurangan *noise* menggunakan metode *intensity filtering* dengan metode *frequency filtering*. Hal ini terjadi pada semua citra yang digunakan dalam uji coba. Perhitungan waktu proses pengurangan *noise* dilakukan ketika citra *noise* dilakukan proses pengurangan *noise*.

Perbedaan waktu dari keempat proses metode pengurangan *noise* sangat besar. Data *timing-run* menunjukkan, pengurangan *noise* menggunakan metode *intensity filtering* baik dengan *high pass intensity* maupun *low pass intensity* membutuhkan waktu jauh lebih cepat daripada menggunakan metode *high pass frequency filtering* dan *low pass frequency filtering*. Hal ini terjadi karena metode *frequency filtering* bekerja dengan mendeteksi dan menghitung terlebih dahulu jumlah frekuensi pemakaian tiap nilai warna yang digunakan pada tiap pikselnya, sehingga diketahui jumlah titik yang memiliki intensitas warna yang sama. Apabila terdapat titik yang berada pada range tersebut akan dianggap sebagai *noise*, yang menyebabkan proses bekerja dua kali lebih lama.

5.4 SNR

Sama halnya dengan *timing-run* di atas, SNR juga menggunakan 20 sampel yang digunakan dalam uji coba program ini, agar memperoleh hasil yang lebih akurat untuk kepentingan analisis yang lebih baik. Dari data SNR diatas, kita dapat menghitung nilai rata-rata dari tiap metode. Dengan menjumlahkan nilai tiap citra dari suatu metode lalu membaginya dengan jumlah sampel citra yang ada, sehingga didapat nilai rata-rata nya yaitu:

Tabel 2. Nilai Rata-Rata *SNR Denoise*

<i>Metode Denoise</i>	<i>SNR Rata-Rata (db)</i>
High Pass Intensity Filtering	80,414
Low Pass Intensity Filtering	80,25211
High Pass Frequency Filtering	79,631
Low Pass Frequency Filtering	79,6895

Ada saatnya reduksi *noise* menggunakan metode *intensity filtering* menghasilkan nilai SNR yang lebih kecil, hal ini disebabkan karena kemungkinan banyak titik *noise* yang memiliki warna sama pada titik-titik pada gambar asli, sehingga *noise* dapat dengan mudah disaring. Sedangkan penggunaan reduksi *noise* dengan metode *frequency filtering* yang menghasilkan nilai SNR yang lebih kecil dimungkinkan karena citra mempunyai banyak variasi warna. Karena *frequency filtering* menghitung banyaknya intensitas warna pada seluruh piksel citra, maka akan diketahui titik-titik *noise* yang memiliki frekuensi warna tertentu sehingga akan mudah disaring.

6. KESIMPULAN

Dari analisis yang dilakukan secara menyeluruh pada aplikasi *noise reduction* dapat diambil suatu kesimpulan bahwa untuk menghilangkan *noise* pada citra hasil, perlu dilakukan beberapa kali proses *noise reduction* dengan kombinasi dari beberapa metode. Dari keempat metode yang digunakan, tiap metode memiliki kelebihan maupun kekurangan berdasarkan macam *noise*.

SNR yang dihasilkan berbeda-beda tergantung pada piksel dan kompleksitas citra yang diuji. Nilai SNR yang lebih kecil adalah nilai SNR yang lebih baik dalam penggunaan metode reduksi *noise*, karena nilai SNR didapat dari nilai kualitas citra filter dibandingkan nilai kualitas citra asli yang ada *noise* nya. Sehingga semakin kecil nilai SNR maka semakin rendah ratio sinyal terhadap *noise*, dimana sinyal adalah citra asli yang ada *noise* nya.

PUSTAKA

- Balza, A. dan Firdausy, Kartika, 2004, *Teknik Pengolahan Citra Digital Menggunakan Delphi*, Ardi Publishing, Yogyakarta.
- Munir, Rinaldi, 2004, *Pengolahan Citra Digital dengan pendekatan Algoritmik*, Informatika, Bandung.
- Nalwan, Agustinus, 1997, *Pengolahan Gambar Secara Digital*, Elex Media Komputindo, Jakarta.
- Perwitasari, Endah. W, 2006, *Analisis Perbandingan Metode Transformasi Wavelet Dengan Metode Contour Untuk Pengurangan Derau Pada Suatu Citra*, Universitas Ahmad Dahlan, Yogyakarta.
- Pranata, Antony, 2001, *Pemrograman Borland Delphi*. Andi, Yogyakarta.