

# Noise Cancellation Using Non-LocalMeans SVD

Muhammad Eka Suryana  
Universitas Negeri Jakarta  
[eka-suryana@unj.ac.id](mailto:eka-suryana@unj.ac.id)

Abstrak – *image denoising* merupakan upaya untuk menghilangkan *noise* dari citra digital. Sejumlah teknik untuk *denoising* menggunakan prinsip pemisahan sinyal asli dan sinyal *noise*. Tidak seperti sejumlah teknik yang melakukan pendekatan ini. *Non local means* merupakan teknik *denoising* untuk mendapatkan citra asli menggunakan *weighted similarities* antara piksel. Tulisan ini merupakan laporan investigasi mengenai NLM algoritma tersebut dan NLM yang telah terimprovisasi menggunakan SVD.

**Keywords-component; Image Denoising, NLM**

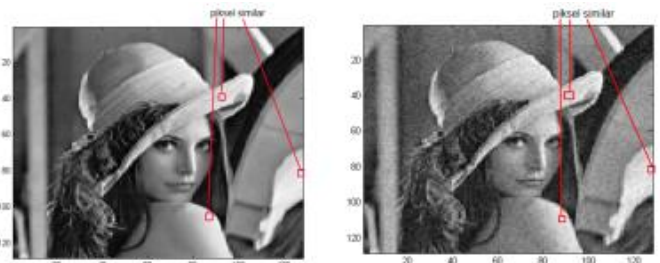
## I. PENDAHULUAN

Citra digital dapat terkorupsi akibat keterbatasan perangkat pencitraan(Kamera) dalam menangkap detail citra dalam format digital, atau kesengajaan. Salah satu bentuk citra terkorupsi adalah munculnya artifak pada sebagian atau seluruh piksel yang menyebabkan nilai piksel yang terkorupsi terdeviasi dari nilai piksel asli sehingga membuat citra yang terkena *noise* mengalami penurunan kualitas visual. Misalkan  $N_o$  merupakan citra asli,  $N_f$  merupakan *noise*, dan  $N_i$  merupakan citra asli yang terkena *noise*. Relasi antara citra ketiga elemen tersebut dapat dinotasikan dengan,

$$N_i = N_o + N_f \tag{p1}$$

Citra asli dapat dikembalikan dengan memisahkan  $N_f$  dari  $N_i$ , namun dalam praktik *denoising* hanya informasi  $N_i$  yang diketahui. Seandainya *noise* dapat diestimasi dengan baik maka, *denoising* akan memberikan citra yang mendekati citra asli. Salah satu bentuk estimasi *noise* adalah dengan menganggap *noise* memiliki properti berfrekuensi tinggi, sehingga  $N_o$  didapatkan dengan mengurangi frekuensi tinggi dari  $N_i$ . Metode yang menerapkan prinsip ini dinamakan dengan *frequency domain filter*, derivatifnya antara lain *wiener filter*, *wavelet thresholding*, *total variant minimation*. Citra gambar yang dikembalikan menggunakan pendekatan pemisahan frekuensi rendah dari frekuensi tinggi, memiliki keterbatasan detail tekstur dari gambar juga hilang karena tekstur merupakan pola yang berfrekuensi tinggi.

*Non-local Means*(NLM) merupakan metode *denoising* yang cukup efektif untuk menghilangkan *white noise/gaussian noise* dengan *mean* 0 dan *variance* tertentu. NLM memiliki asumsi bahwa setiap citra gambar bersifat redundan/repetitif, lebih detailnya: nilai piksel pada koordinat spasial citra sangat mungkin memiliki nilai piksel yang berdekatan pada koordinat spasial lainnya pada citra yang sama. Memanfaatkan informasi ini, NLM merestorasi citra asli dengan melakukan *averaging* nilai piksel yang berdekatan tersebut. Nilai piksel yang similar ini sangat mungkin ditemui pada daerah tetangga piksel tersebut, namun tidak menutup kemungkinan ditemukan juga pada piksel yang berjarak spasial cukup jauh dari piksel referensi.



Gambar 1a. Citra asli

Gambar 1b. Citra terkorupsi white noise

Karena informasi kelompok piksel yang memiliki nilai berdekatan(relevan) tidak diketahui. NLM yang diperkenalkan Buades et al(2005), melakukan *denoising* dengan melakukan *weighted averaging* piksel referensi(POI – *pixel of interest*) dengan seluruh piksel selain piksel tersebut. Piksel yang memiliki kedekatan nilai akan diberikan *weight* maksimum, jika tidak *weight* diset minimum. Setelah semua *weight* dihitung untuk POI, dilakukan *averaging* untuk mendapatkan nilai piksel terestorasi untuk POI.

Piksel-piksel yang relevan diukur dari citra gambar asli. Misalkan sebelum terkorupsi POI A memiliki nilai 100, untuk suatu waktu ditemukan piksel yang relevan sebut B & C masing-masing memiliki nilai 98 & 103. Setelah diberikan *noise*, nilai A := 110, B := 88, C := 92. Oleh sebab ini proses *averaging* per pixel tidak akan memberikan hasil yang optimal. NLM melakukan *averaging* terhadap *neighborhood* piksel POI dengan *neighborhood* piksel yang relevan, sebab bisa jadi masih terdapat piksel tetangga B & C yang tidak banyak terkorupsi, dan informasi itulah yang dilibatkan dalam perhitungan *averaging* antara dua buah *neighborhood* yang relevan. Atas dasar yang sama, dapat dianggap NLM tidak berpengaruh besar, untuk citra yang banyak terkorupsi *noise*. Sebab akan piksel yang benar-benar relevan terhadap POI akan berjumlah minoritas.

Notasi matematis dari metode NLM dapat direpresentasikan dengan formula berikut:

$$w_{ij} = \exp\left\{-\frac{|G_a \cdot (x_i - x_j)|^2}{h^2}\right\} \tag{p2}$$

$$y_i = \frac{\sum w_{ij} x_j}{\sum w_{ij}} \quad (p3)$$

Dimana:

$W_{ij}$  : *Weight* similaritas dua *neighborhood* piksel. • berarti perkalian *element-wise*.

$h$  : Estimasi *noise variance* yang terkandung dalam seluruh citra terkorupsi *noise*.

$Ga$  : Kernel gaussian 2-D yang berukuran sama dengan ukuran *neighborhood* piksel.

$x_j$  : Nilai piksel dengan indeks ke- $j$  yang diperbandingkan.

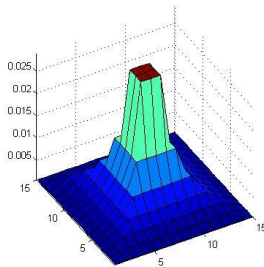
$x_i$  : Nilai piksel POI. .

$y_i$  : Piksel POI yang telah di-*denoising*

Gaussian kernel dapat dihitung menggunakan formula berikut,

$$Ga = \frac{1}{R} \sum_{i=m}^R \frac{1}{(2i + 1)^2} \quad (p3)$$

Nilai  $2.i+1$  pada formula  $p3$  merupakan derajat elevasi kurva gaussian, nilai ini dapat dirubah sesuai kebutuhan. Kernel gaussian memiliki properti unik, dimana pada array(1-D)/matriks(2-D) yang dihasilkan dari fungsi gaussian nilai dominan berada pada indeks tengah, dan semakin menurun untuk indeks yang menjauhi indeks tengah.



Gambar 1. Plot 2-D kernel gaussian ukuran 7x7

## II. FONDASI

### A. Lingkungan Eksperimen

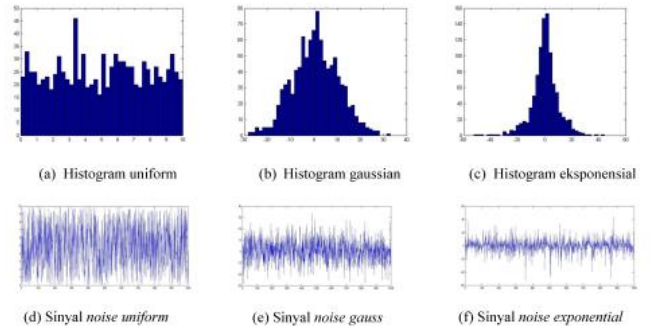
NLM dibuat pertama kali menggunakan *Java* menggunakan framework *Marvin*. kemudian diporting ke kode *Matlab*, hal ini dilakukan sebab *Matlab* menyediakan banyak *Application Programming Interface/API/fungsi* untuk dukungan komputasi numerik.SVD merupakan salah satu fitur yang dipergunakan untuk pengembangan NLM dengan SVD.

### B. Noise

Prinsip pemisahan *noise* dari citra gambar yang terkorupsi *noise* merupakan cara yang ideal untuk *denoising*, meskipun *perfect estimation of noise values* sama sulitnya dengan *denoising*. Namun pemahaman karakteristik berbagai jenis *noise* merupakan dasar kerangka berpikir untuk *denoising*. Menganalisa karakteristik *noise* akan membantu dalam

menentukan metode yang tepat untuk *denoising*, dan memahami mengapa NLM efektif untuk *white noise*.

*Noise* merupakan nilai deviasi dari nilai piksel sebenarnya, nilai intensitas piksel bernilai 0-255 sehingga *noise* juga berada pada interval yang sama. *Noise* diukur berdasarkan nilai *variance*, hal ini berlaku untuk *white noise*.



Gambar 3. Histogram *noise*(atas) & Plot sinyal *noise* (bawah), 1000 elemen, sigma=10.

*White noise/gaussian noise* memiliki karakteristik: bilangan *random* yang dihasilkan berada dalam rentang nilai bilangan *signed integer* dan memiliki histogram gaussian dengan *mean* 0, dan kerampingan(skewness) tertentu yang dihitung berdasarkan sigma/standar deviasi(gambar 3.a). Untuk *noise uniform*, bilangan *random* yang dihasilkan berada dalam rentang positif *integer* dan memiliki distribusi histogram uniform(gambar 3.b). Untuk *noise exponential*, rentang nilai bilangan *random* yang dihasilkan berupa *signed integer* dan memiliki *distribusi* histogram *exponential*(gambar 3.c).

### C. Non-local Means SVD

NLM melakukan *averaging* terhadap seluruh piksel. Proses ini memiliki kompleksitas waktu yang mahal,  $O(N^4)$ . Untuk mengurangi waktu kompleksitas kinerja NLM dapat dibatasi dengan melakukan *averaging* pada *search window* yang ditentukan. Karena piksel tetangga seringkali memiliki nilai piksel yang berdekatan, pembatasan *search window* tidak terlalu mempengaruhi kinerja *denoising*.

*Denoising* akan optimal jika jumlah piksel yang relevan dominan. Karena informasi piksel mana saja yang relevan mula-mula tidak diketahui, implementasi NLM original memanfaatkan *weighted averaging*. Jika piksel mana saja yang relevan dapat diestimasi dengan baik hasil *denoising* yang didapatkan akan lebih optimal. Orchard, 2008 memperkenalkan metode *find most relevant patch*(MRP) yang dihitung menggunakan SVD. Pencarian MRP menggunakan SVD bersifat fundamental sehingga diadaptasi dan diperbaiki oleh penelitian yang lain, seperti *Thaipanich*(2010) yang memanfaatkan SVD untuk menghitung gradient sehingga tekstur dapat lebih dipertahankan setelah *denoising* dan *neighborhood rotated matching* untuk mencari MRP. Dua *neighborhood window* piksel bisa jadi tidak memiliki nilai similaritas yang sama, namun setelah dilakukan *neighborhood window rotation* pada piksel yang diperbandingkan bisa jadi nilai similaritas meningkat. MRP yang ditentukan menggunakan SVD memiliki kelemahan, *signal subspace* yang

didapatkan *over estimate* menggunakan *local basis training* Yanmin He mendapatkan *patch* yang lebih relevan[6].

Prinsip kerja SVD memiliki kemiripan dengan pemisahan *noise* dan *signal* pada citra yang terkorupsi. Asumsi yang dipergunakan oleh metode SVD, jika citra terkorupsi *noise* maka *signal subspace* akan terletak pada *lower dimensional moneyfold*, sementara *noise subspace* akan terdapat pada *remainder dimensional moneyfold*. Misalkan terdapat citra sebuah gambar sebut sebagai  $M$ . SVD mendekomposisi  $M$  menjadi eigen vektor kiri  $U$ , nilai eigen  $S$ , dan eigen vektor kanan  $V$ . Hubungan tersebut dapat direpresentasikan berdasarkan formula berikut.

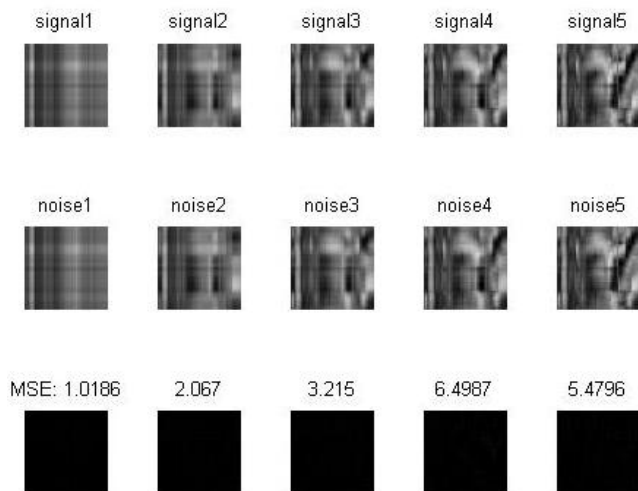
$$USV^t = M \tag{p4}$$

Nilai eigen yang paling utama terdapat pada rank 1 dari matriks  $S$ . Semakin meningkat rank nilai eigen pada matriks  $S$ , semakin menurun keutamaannya. Hal inilah yang disebut *lower dimensional moneyfold*. Aproksimasi ruang sinyal dapat dilakukan dengan memilih rank  $k(1 \leq k \leq \sim)$  tertentu,  $k$  dipilih sedemikian rupa sehingga didapatkan aproksimasi terbaik.

**Hipotesa 1:**

“*Signal subspace* pada citra terkorupsi terdapat pada aproksimasi rank ke  $k$  dari *lower dimensional monefold*.”

Indikasi kebenaran *signal subspace* berada pada *lower dimensional monefold* dapat dilakukan dengan melakukan svd pada dua citra: citra asli dan citra terkorupsi *noise*. Jika dekomposisi rank 1 *lower dimensional monefold* dari kedua citra memberikan hasil yang berdekatan, hal tersebut mengindikasikan hipotesa 1 memiliki nilai kebenaran. Citra asli dan citra terkorupsi yang dipergunakan untuk mendapatkan *signal subspace* menggunakan gambar 1, namun citra lebih banyak dikorupsi dengan  $\sigma=20$ .



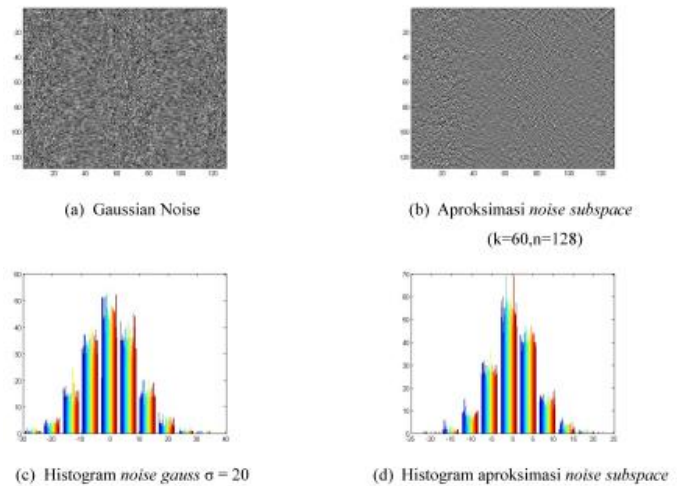
**Gambar 2. Aproksimasi rank-  $k$  dari Signal Subspace. Signal subspace dari citra asli(baris 1), dan citra terkorupsi(baris 2).**

*Signal space* diaproksimasi terhadap kedua citra, menggunakan aproksimasi SVD rank 1 sampai 5. Perhitungan *mean square error*(MSE) dilakukan untuk mengetahui *difference* dari kedua citra aproksimasi. Hingga maksimum rank-5 yang diujikan diketahui MSE masih bernilai minimal. Informasi ini mengindikasikan untuk citra terkorupsi beberapa tingkat rank aproksimasi memberikan *signal subspace* yang sangat berdekatan dengan citra sebelum terkorupsi.

**Hipotesa 2:**

“*Noise subspace* pada citra terkorupsi akan berada pada aproksimasi remainder rank ke  $k$  hingga rank  $n$ . dimana  $n$  merupakan rank maksimum dari citra  $M$ .”

Pembuktian hipotesa ini tidak dapat dilakukan dengan eksperimen. Misalkan mula-mula terdapat citra asli  $M$ , dan citra terkorupsi  $M_k$ . *Difference* dari  $M_k - M$  kemudian disimpan dalam  $D$ . Aproksimasi dilakukan dengan mengambil rank  $k=50$ , dengan rank  $n=128$ . Histogram dan *difference* antara  $D$  dan aproksimasi rank-50. Dapat dilihat pada gambar berikut.



**Gambar 3. Noise Subspace**

Berdasarkan informasi yang terdapat pada gambar 5, dapat disimpulkan hal-hal berikut:

- Aproksimasi *remainder rank* menggunakan SVD memberikan citra gambar yang jauh dari aslinya, citra aproksimasi lebih memiliki karakteristik *noise*.
- Meskipun karakteristik *remainder rank approximation* memiliki karakteristik *noise*. Namun perbedaan tekstur *noise* asli dengan aproksimasi *noise* terlalu jauh. Aproksimasi *noise* masih menyimpan sedikit informasi tekstur (tekstur topi pada gambar 5.b). Pemisahan langsung *noise* dari citra terkorupsi bukan cara yang tepat. **Aproksimasi noise signal tidak akan pernah mendekati noise signal sebenarnya.**
- Percobaan dilakukan berulang kali hingga mendapatkan histogram distribusi yang sangat mendekati distribusi histogram sinyal *noise*. Selain

menggunakan aproksimasi  $k=60$ , dan  $n$ =maksimal, konfigurasi yang lain memiliki interval intensitas yang jauh dari *noise signal* sebenarnya dan histogram distribusi yang terbentuk memiliki *skewness* yang tidak mirip. Permasalahan menentukan *histogram* distribusi yang mendekati *histogram signal noise* merupakan kesulitan tersendiri.

Salah satu perbedaan ide yang diperkenalkan Orchard(2008)[3] dengan adaptif SVD yang digagas oleh Yanmin He(2011)[6] adalah Orchard hanya mempertimbangkan hipotesa 1 dalam algoritmanya sementara Yanmin He turut memasukkan hipotesa 2 dalam pengembangan algoritma. Re-eksperimentasi yang dilakukan penulis hanya membuktikan kebenaran gagasan yang diperkenalkan oleh Orchard.

Metode yang diperkenalkan oleh Orchard dapat diimplementasikan dengan langkah-langkah berikut:

- Untuk setiap piksel  $i$  yang terdapat pada citra, buat matriks *neighborhood* window-nya. Setiap matriks *neighborhood window* yang terbentuk kemudian di-*reshape* menjadi sebuah vektor kolom.
- *Merge* setiap vektor *neighborhood* kolom menjadi sebuah matriks yang berukuran  $m \times n$  ( $n > m$ /kolom berjumlah lebih banyak dari baris). Sebut matriks yang terbentuk sebagai  $Z$ .
- Lakukan dekomposisi SVD dari matriks  $Z$  sehingga terdekomposisi menjadi matriks  $U, S$ , dan  $V$ .
- Menggunakan *right singular vector*  $V$ , hitung koordinat  $C$  menggunakan formula  $C = Z \times V$ . Koordinat  $C$  akan memiliki ukuran yang sama dengan  $Z$ , dan representasi koordinat piksel ke- $i$  dapat diakses melalui indeks kolom dari  $C$ .

MRP dapat dicari dengan menghitung  $L2$  norm antara POI ke- $x$  dengan piksel selainya. Misalkan  $\alpha_1$  merupakan koordinat pertama dari  $x$  dan  $\beta_1$  merupakan koordinat pertama dari piksel selain  $x$ . Aproksimasi rank-1  $L2$  norm dapat dihitung dengan  $(\alpha_1 - \beta_1)^2$ . MRP akan memiliki *norm* yang kecil sehingga eliminasi piksel yang memberikan perhitungan *norm* besar, hal ini dapat dilakukan salah satunya dengan mensort nilai *norm*, kemudian simpan sejumlah *norm* terkecil menggunakan rasio tertentu. Pada percobaan yang dilakukan rasio ditentukan sebesar 5%. Untuk citra berukuran 128 x 128.  $C$  yang terbentuk akan memiliki panjang 16.384, menggunakan treshold 5% akan menyisakan MRP sejumlah 819. Untuk mengerucutkan MRP, misalkan rank-1  $L2$  norm yang telah dihitung dinotasikan dengan *norm-1*, aproksimasi rank norm berikutnya dihitung menggunakan  $norm-2 = norm-1 + norm-2$ . Rasio untuk aproksimasi norm subsekuen rank berikutnya diset sebesar 80%-90% tentatif. Total rank aproksimasi yang dipergunakan oleh Orchard  $k=20$ .

Implementasi NLM MRP yang dilakukan sedikit berbeda karena terdapat perbedaan persepsi. Perbedaan tersebut terletak pada dua hal: Koordinat yang penulis implementasikan didapat dengan menyelesaikan sistem persamaan linier,

$$z_i = c_1(v_1) + c_2(v_2) + \dots + c_n(v_n)$$

Setelah didapatkan MRP, denoising dilakukan menggunakan MRP pada *signal subspace* hal ini berlaku pada implementasi yang dilakukan Orchard, namun *denoising* yang penulis lakukan tetap dilakukan pada *full space* menggunakan MRP yang didapat.

### III. EKSPERIMEN

#### A. Evaluasi Hasil Denoising

Diperlukan tolak ukur penilaian kualitas *denoising* menggunakan data statistik selain memperhatikan kualitas citra *denoise* menggunakan pengamatan fisik. Metode pengukuran yang dipergunakan disini menggunakan standar penilaian yang telah umum disepakati: MSE, PSNR, dan SSIM.

*Means Square Error*(MSE) merupakan pengukuran selisih rata-rata kuadrat error dua citra. Untuk data 1-d MSE dapat didefinisikan dengan,

$$MSE = \frac{1}{n} \sum (X_i - X_j)^2 \tag{p5}$$

Pendekatan yang sama dilakukan untuk MSE 2-d. *Peak Signal To Noise Ratio*(PSNR) merupakan rasio citra sinyal terkorupsi terhadap citra asli. PSNR dapat dihitung menggunakan formula berikut,

$$PSNR = 20 * \log_{10} \left( \frac{\text{MAX}_i}{\sqrt{\text{MSE}}} \right) \tag{p6}$$

PSNR maupun MSE terkadang tidak konsisten dengan persepsi mata manusia[7], oleh sebab itu SSIM dikembangkan mengukur kemiripan antara kedua citra yang diperbandingkan. SSIM *metric* dihitung dari sejumlah ukuran sub-window yang berbeda antara dua citra. SSIM memiliki nilai 0 ~ 1. Nilai 1 diberikan untuk dua citra yang identik. Penjelasan lebih lanjut untuk mengetahui formula SSIM dapat dilihat di [7]

#### B. Hasil Eksperimen dan Analisa

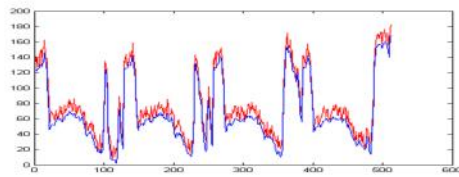
Eksperimen dilakukan dengan sejumlah skenario pengujian.

##### Skenario 1

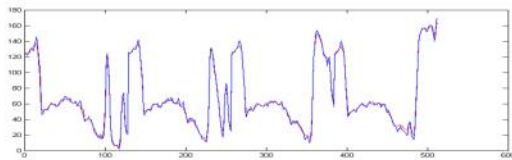
Skenario pertama dilakukan untuk mengetahui keefektifan NLM dalam *denoising white noise* dan juga jenis *noise* lainnya. Untuk percobaan pertama ini, dikonfigurasi nilai  $\sigma=20$ . Kemudian parameter *denoising* menggunakan  $ks(\text{neighborhood window})= 3$ ,  $sw(\text{search window})=2$ ,  $h(\text{estimasi } \sigma)=20$ . Eksperimen *denoising* untuk citra sinyal terkorupsi *noise uniform* dapat dilihat pada gambar 6.



(a) Citra asli (b) Citra terkorupsi *uniform noise*,  $\sigma = 20$  (c) Hasil NLM *denoising*



(c) Plot 1-d sampel nilai piksel, mengambil 4 baris pertama nilai piksel dari citra. Garis biru menunjukkan plot citra asli, garis merah menunjukkan plot citra terkorupsi. PSNR = 26.89 db, SSIM = 0.98



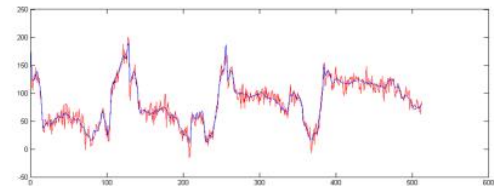
(d) Plot 1-d *denoise signal* terhadap citra asli. Biru menandakan sinyal asli, merah menandakan sinyal *denoise*. PSNR = 31.89 db, SSIM=0.92.

Gambar 4. Hasil restorasi citra terkorupsi *uniform noise*

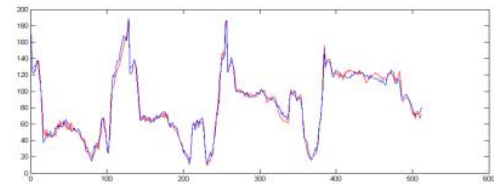
Pengamatan secara fisik terhadap citra terestorasi memberikan hasil yang cukup baik kecuali terdapat detail tekstur yang ter-*smooth* (tekstur topi terutama), dan terdapatnya peningkatan *brightness* pada sejumlah wilayah (kulit lena). NLM tidak dapat merestorasi citra asli dengan sempurna hal ini terlihat pada gambar 6.d terdapat region dimana terjadi *smoothness* terutama pada region yang memiliki resonansi tinggi/kaya akan tekstur. Eksperimen berikutnya dilakukan terhadap citra yang terkorupsi *white noise* dengan nilai  $\sigma = 10$ .



(a) Citra asli (b) Citra terkorupsi *uniform noise*,  $\sigma = 10$  (c) Hasil NLM *denoising*



(c) Plot 1-d sampel nilai piksel, mengambil 4 baris berurut dari indeks *random* nilai piksel citra. Garis biru menunjukkan plot citra asli, garis merah menunjukkan plot citra terkorupsi *gaussian*. PSNR=28.15 db, SSIM=0.80

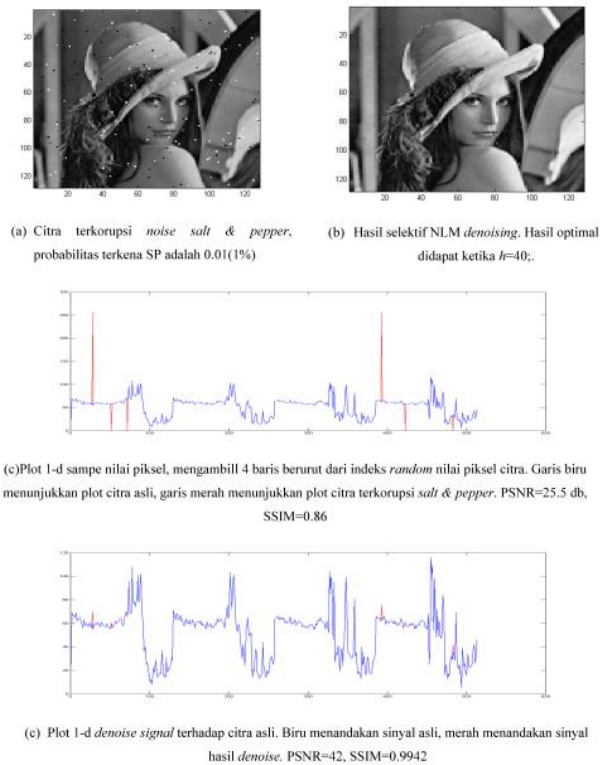


(d) Plot 1-d *denoise signal* terhadap citra asli. Biru menandakan sinyal asli, merah menandakan sinyal hasil *denoise*. PSNR=31.68 db, SSIM=0.93

Gambar 5. Hasil restorasi citra terkorupsi *gaussian noise*

Berdasarkan gambar 7.d, NLM tidak optimal dalam mengestimasi sinyal lena asli dari sinyal yang terkorupsi *gaussian noise*, hal ini terlihat setidaknya jika diperbandingkan dengan gambar 6.d dimana deviasi yang terjadi pada sinyal *denoise* dari sinyal asli cukup bervariasi. Hal ini bisa dimengerti sebab karakteristik dari *gaussian noise* yang turut memperhitungkan nilai negatif dalam menghasilkan sinyal *noise* sehingga estimasi sinyal asli lebih sulit untuk dilakukan. Yang menarik informasi SSIM terhadap sinyal terestorasi menunjukkan kualitas yang lebih baik dari sinyal terestorasi dari *uniform noise* (0.93 vs 0.92), hal ini sejalan dengan pengamatan fisik dimana masih terdapat tekstur yang masih terjaga setelah dilakukan *denoising*.

Eksperimen berikutnya dilakukan terhadap *noise salt & pepper* (SP), dimana setiap piksel memiliki kemungkinan tertentu untuk berubah nilainya menjadi 0 atau 255. Efek dari *noise* ini pada citra adalah munculnya bintik hitam dan putih pada citra terkorupsi. NLM melakukan *denoising* terhadap seluruh piksel, karena SP merubah nilai piksel pada indeks tertentu saja, NLM perlu dimodifikasi terlebih dahulu agar hanya mengolah piksel yang terkena *salt & pepper*.

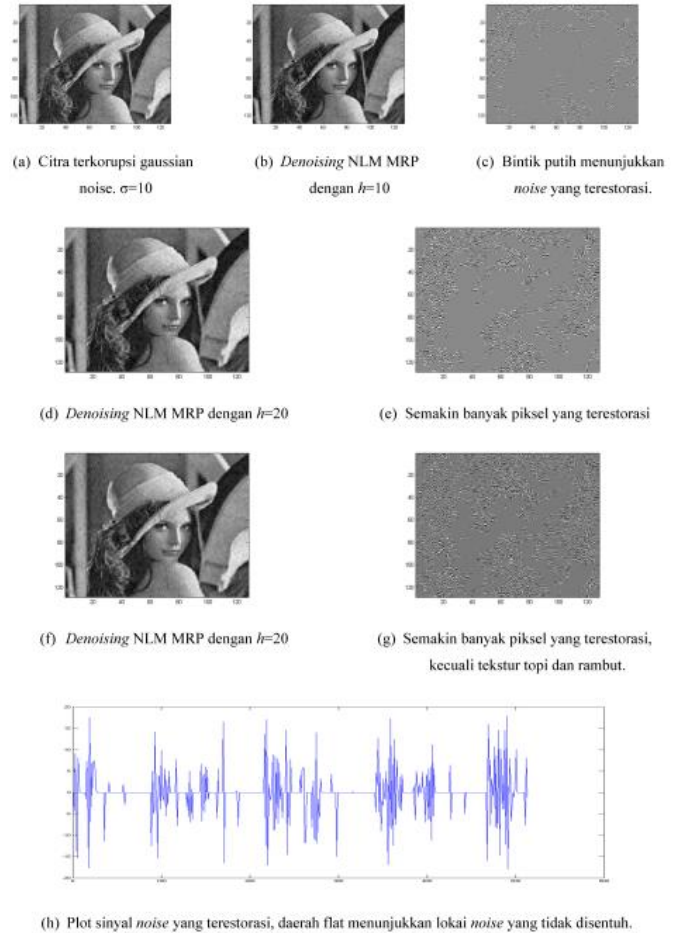


Gambar 6. Hasil restorasi citra terkorupsi *salt & pepper*

Selektif NLM mampu merestorasi citra yang mendekati asli. Namun hasil ini didapat dengan sejumlah catatan: Parameter *search window* atau *neighborhood window* sangat tidak berpengaruh terhadap hasil *denoising*, bahkan mengatur *full search window* hampir tidak berpengaruh. Parameter  $h$  (*decay parameter*) memiliki efek yang signifikan terhadap hasil *denoising*, ketika nilai  $h > 40$  atau  $h < 40$  kualitas citra terestorasi sedikit menurun. Sedikitnya pengaruh *search window* atau *neighborhood window* disebabkan karena kebanyakan piksel tetangga akan tidak relevan dengan POI sehingga dialokasikan *weight* yang sangat kecil, dalam hal ini POI merupakan *outlier* terhadap piksel tetangga.

### Skenario 2

Tujuan dari eksperimen kedua adalah untuk membuktikan NLM MRP mampu memberikan hasil *denoising* yang lebih baik dari NLM standar. Metode NLM MRP menggunakan implementasi Orchard, hal ini disebabkan *denoising* menggunakan MRP pada *signal subspace* lebih masuk akal, dan pendekatan inilah yang dipergunakan juga pada [6].



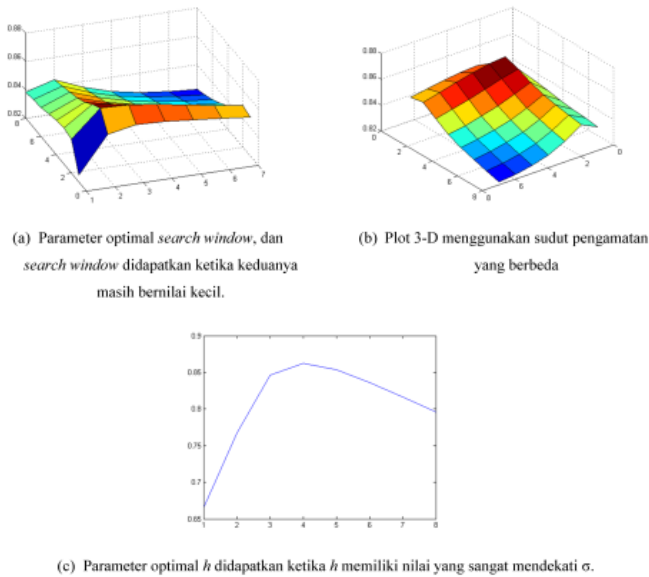
Gambar 7. Hasil *denoising* menggunakan NLM MRP

Percobaan *denoising* menggunakan NLM MRP menunjukkan, NLM MRP tidak merestorasi seluruh piksel yang terkena *noise*. Dari informasi yang disajikan pada gambar 9, mayoritas hanya tekstur *smooth* yang dihilangkan *noise*-nya. Melihat data ini, NLM masih memiliki kinerja yang lebih baik namun sifat dari NLM MRP yang melakukan *denoising* daerah tertentu saja merupakan sebuah fakta yang menarik, sangat mungkin *noise* yang terestorasi merupakan estimasi *noise* yang sangat baik pada daerah yang terestorasi. Berdasarkan karakteristik ini NLM MRP dapat dijadikan sebagai *pre-processing* dalam tahap *denoising* menggunakan NLM.

### Skenario 3

Eksperimen ketiga bertujuan untuk mencari parameter optimal NLM untuk *denoising white noise Benchmark* dilakukan menggunakan *test script* Matlab yang dirancang untuk memaksimalkan nilai SSIM dengan mencoba semua kemungkinan *search window*, *neighborhood window*, dan  $h$  dengan interval tertentu. Karena *benchmark* terhadap ketiga ketiga tersebut memakan waktu yang lama, *benchmark* disederhanakan menggunakan dua fase: fase pertama bertujuan untuk mengoptimasi parameter *search window* dan *neighborhood window*(2-D), dan fase kedua bertujuan untuk

mengoptimasi parameter  $h$ . Pada eksperimen ini citra terkorupsi *gaussian noise* dengan  $\sigma=15$ .



Gambar 8. Benchmark Parameter NLM untuk mencapai SSIM terbaik.

Parameter  $h$  (*decay parameter*) merupakan estimasi dari  $\sigma$ , sehingga wajar jika SSIM bernilai optimal saat  $h=\sigma$ . Analisa lebih lanjut diperlukan untuk mengetahui kenapa *search window* dan *neighborhood window* bernilai optimal ketika keduanya bernilai kecil. Hal ini kontradiktif dengan yang karakteristik NLM yang seharusnya bernilai optimal ketika menjalankan *full search space*. Setidaknya terdapat dua hal yang menyebabkan hal ini terjadi: Konfigurasi pembentukan *gaussian kernel* untuk menghitung  $L2$  norm dari dua *neighborhood window*, pembentukan *gaussian kernel* pada implementasi mempergunakan rumus ( $p3$ ) dimana setiap indeks pada *gaussian kernel* dihitung dengan  $\frac{1}{2f+1}$ . Kedua, nilai similaritas dua *neighborhood window* sangat ditentukan oleh *weight* yang dihitung menggunakan ( $p2$ ).

*Weight* merupakan distribusi eksponensial:  $w = e^{\frac{-d}{h}}$ , perhatikan  $d$  pasti bernilai positif karena merupakan bentuk kuadrat (*euclidean distance* antara dua *patch*), sehingga nilai  $-d$  bernilai negatif. Untuk *patch/neighborhood window* yang tidak similar, nilai  $d$  akan sangat besar, sehingga ketika dinegatifkan melalui  $e$  akan bernilai mendekati 0. Permasalahannya melalui konfigurasi *gaussian kernel* yang memiliki elevasi tidak terlalu curam (gambar 2), *weighted similarities* antara dua *patches* ketika memiliki *gaussian kernel* yang relatif besar akan

memberikan hasil yang buruk sebab informasi yang relevan sangat mungkin ditemukan pada tetangga piksel *neighborhood window* yang diperbandingkan. Atas dasar ini *patch* sebaiknya diatur kecil, kecuali formula pembentukan *gaussian kernel* dimodifikasi.

*Averaging* dilakukan terhadap penjumlahan dari *weighted similarities* untuk POI, over estimasi akan dihasilkan ketika jumlah piksel yang di-*averaging* melebihi rasio yang dibutuhkan. Atas dasar ini, *search window* sebaiknya diset tidak terlalu besar, 7 piksel sekitar POI merupakan nilai yang ideal.

#### IV. PENUTUP

Re-eksperimentasi ini merupakan upaya untuk mengimprovisasi algoritma NLM. Improvisasi tidak mungkin dilakukan tanpa pemahaman yang kokoh terhadap NLM, untuk itulah dilakukan analisa yang laporannya dijabarkan pada tulisan ini. Upaya mengimprovisasi NLM sudah dilakukan menggunakan NLM MRP, implementasi yang penulis lakukan menggunakan persepsi yang dijelaskan pada bagian IV tidak memberikan hasil yang lebih baik dari NLM original, untuk itulah fokus analisa dilakukan menggunakan NLM MRP yang diperkenalkan Orchard.

#### REFERENSI

- [1] Antoni Buades, Bartomeu Coll, Jean-Michel Morel. *A non-local algorithm for image denoising*. Tracked on CiteSeerX with total 60 citations. 2005.
- [2] Antoni Buades, Bartomeu Coll, Jean-Michel Morel. *A Review Of Image Denoising Algorithms With a New One*. Society for Industrial and Applied Mathematics Journal vol 4, no 2, pp 490-530. 2005.
- [3] Jeff Orchard, Mehran Ebrahimi, Alexander Wong. *Efficient Nonlocal-Means Using The SVD*. 15th IEEE International Conference on Image Processing, 2008. ICIP 2008.
- [4] Antoni Buades, Bartomeu Coll, Jean-Michel Morel. *Image and Movie Denoising by Non Local Means*. International Journal of Computer Vision vol 76, Issue 2. 2008.
- [5] Thanaphol Thaipanich, et al. *Improved Image Denoising with Adaptive Nonlocal Means Algorithm*. International Conference on Consumer Electronics(ICCE), Las Vegas, 2010.
- [6] Yanmin He et al. *Adaptive Denoising by Singular Value Decomposition*. IEEE Signal Processing Letter vol 18, no 4, April 2010.
- [7] [http://en.wikipedia.org/wiki/Structural\\_similarity](http://en.wikipedia.org/wiki/Structural_similarity), *Structural Similarity*, diakses pada Juni 2011.