



Implementasi Metode Alpha-Trimmed Mean Filter dan Adaptive Median Filter Untuk Mereduksi Noise Poisson Pada Citra Digital

Restima

Prodi Teknik Informatika, Universitas Budi Darma, Medan, Indonesia

Email: restima@gmail.com

Abstrak—Pada dasarnya, setiap pengambilan citra dapat menyebabkan terdapatnya noise pada citra yang dihasilkan. Noise Poisson, Uniform, Salt and Pepper, Speckle noise merupakan model noise yang banyak terdapat pada citra. Citra digital yang memiliki noise dapat menimbulkan masalah berupa gambar yang tidak dapat diinterpretasikan dengan baik oleh manusia. Namun noise dapat dikurangi melalui perbaikan citra yaitu filtering. Metode filter yang dapat digunakan untuk mereduksi noise diantaranya adalah AlphaTrimmed Mean Filter dan Adaptive Median Filter. Hasil dari penelitian ini diperoleh bahwa Adaptive Median Filter (AMF) baik digunakan untuk mereduksi Noise Poisson dibandingkan dengan Alpha-Trimmed Mean Filter (ATMF). Perancangan yang dilakukan nantinya dapat memperbaiki kualitas citra digital dan mempermudah ahli radiologi dalam menganalisa gambar yang terkena noise dan menghasilkan citra yang lebih baik lagi.

Kata Kunci: Alpha-Trimmed Mean Filter; Adaptive Median Filter; Citra Digital; Pengolahan Citra; Reduksi Noise

Abstract—Basically, every image taking can cause noise in the resulting image. Noise Poisson, Uniform, Salt and Pepper, Speckle noise are noise models that are common in images. Digital images that have noise can cause problems in the form of images that cannot be interpreted properly by humans. However, noise can be reduced through image enhancement, namely filtering. Filter methods that can be used to reduce noise include AlphaTrimmed Mean Filter and Adaptive Median Filter. The results of this study indicate that the Adaptive Median Filter (AMF) is good for reducing Noise Poisson compared to Alpha-Trimmed Mean Filter (ATMF). The design can later improve the quality of digital images and make it easier for radiologists to analyze images exposed to noise and produce even better image.

Keywords: Alpha-Trimmed Mean Filter; Adaptive Median Filter; Digital Image; Image Processing; Noise Reduction

1. PENDAHULUAN

Citra adalah gambar dua dimensi yang dihasilkan dari gambar analog dua dimensi yang kontiniu menjadi gambar diskrit melalui proses sampling. Dengan adanya noise juga dapat mengakibatkan kualitas citra menjadi lebih buruk oleh karena itu diperlukan metode dalam memperbaiki kualitas citra tersebut agar tampilan citra menjadi lebih baik. Istilah citra digital menyatakan "pemrosesan gambar berdimensi dua melalui komputer digital" (Jain, 1989). Pengolahan citra adalah istilah umum untuk berbagai teknik yang keberadaannya untuk memanipulasi dan memodifikasi citra dengan berbagai cara, foto merupakan gambar berdimensi dua yang dapat diolah dengan mudah, setiap foto dalam bentuk citra digital (misalnya berasal dari kamera digital) dapat diolah melalui perangkat lunak tertentu.

Terdapat beberapa noise pada citra digital, beberapa diantaranya adalah Gaussian noise, Speckle noise, Salt and Pepper noise, Poisson Noise. Efek gaussian filter pada gambar ditandai titik-titik berwarna yang jumlahnya sama dengan persentase noise. Speckle noise adalah model noise yang memberikan warna hitam pada titik yang terkena noise. Salt and papper adalah berbentuk seperti bintang hitam dan putih pada citra sedangkan poisson noise adalah bukan merupakan noise buatan, poisson merupakan noise yang ditambahkan langsung pada citra tanpa kita menambahkan parameter apapun, sehingga efeknya pada citra pun tetap. Pada penelitian ini akan merancang aplikasi untuk mereduksi noise poisson dengan menerapkan kedua metode diantaranya Alpha-Trimmed Mean Filter Dan Adaptive Median Filter. Dalam implementasi kedua metode dari perhitungan Peak Signal to Noise Ratio (PSNR) dan MSE.

Metode Alpha trimmed mean merupakan filter jenis non-linear, dengan mempertahankan tepi dan mengurangi kepadatan noise. Alpha trimmed mean filter adalah metode menghitung nilai rata-rata yang menghilangkan persentase tertentu dari nilai terbesar dan terkecil sebelum menghitung mean. Adanya beberapa tahapan proses pencarian nilai alpha trimmed mean filter dan preprocessing gambar masukan akan membutuhkan waktu yang banyak terutama untuk gambar masukan dengan ukuran besar atau memiliki kernel konvolusi besar[1].

Sedangkan metode Adaptive median filter melakukan pengolahan spasial untuk menentukan piksel dalam gambar yang telah dipengaruhi oleh impuls noise. Adaptive median filter mengklasifikasikan piksel sebagai kebisingan dengan membandingkan setiap piksel pada gambar dengan piksel tetangga disekitarnya. Ukuran lingkungan disesuaikan, serta ambang batas untuk perbandingan. Sebuah piksel yang berbeda dari mayoritas tetangganya, maupun yang tidak selaras secara struktural dengan piksel mereka yang sama, akan diberi label sebagai impuls noise[2].

Sebelumnya pernah dilakukan penelitian oleh Ivan Maulana dan Pulung Nurtanto Andono dengan judul "Analisis perbandingan Adaptive Median Filter Dan Median Filter Dalam Reduksi Noise Salt & Pepper" dan memberikan hasil bahwa Perbedaan mendasar antara kedua metode ini yaitu pada besarnya windows pada adaptif median filter adalah variabel. Dari hasil penelitian, citra yang menggunakan metode adaptif median filter lebih baik dari pada median filter. Dari perhitungan Peak Signal to Noise Ratio (PSNR) citra yang menggunakan adaptif median filter mendapatkan 29,2495 dB sedangkan median filter mendapatkan 23,8181 dB[2].



Dalam penelitian ini mengimplementasikan metode Alpha-trimmed mean filter dan adaptif median filter untuk Mereduksi Noise poisson Pada Citra Digital” Aplikasi ini dibuat dengan perangkat lunak visual basic 2008 dan proses aplikasi menggunakan reduksi noise poisson dengan filtering dilakukan pada gambar berformat BMP. Parameter yang akan diukur untuk mengetahui hasil implementasi adalah Mean Square Error (MSE) dan Peak Signal to Noise Ratio (PSNR)

2. METODOLOGI PENELITIAN

2.1 Noise

Noise adalah bintik-bintik atau kotoran yang menempel pada citra. *Noise* merupakan suatu gangguan yang disebabkan oleh penyimpanan data digital yang diterima oleh alat penerima data gambar yang dapat mengganggu kualitas citra. Noise dapat disebabkan oleh gangguan fisik (optik) pada alat penangkap citra misalnya kotoran debu yang menempel pada lensa foto maupun akibat proses pengolahan yang tidak sesuai.[3].

Ada tiga jenis *noise* yaitu *gaussian noise*, *speckle noise*, dan *salt and pepper noise* [3].

1. *Noise gaussian* : model *noise* yang mengikuti distribusi normal standar dengan rata-rata nol dan standard deviasi 1. Efek dari *gaussian noise* ini pada gambar adalah munculnya titik-titik berwarna yang jumlahnya sama dengan persentase *noise*. Dibuat dengan cara membangkitkan bilangan random [0,1] dengan distribusi *Gaussian* Untuk piksel yang terkena *noise*, nilai fungsi citra ditambahkan dengan *noise* yang ada.
2. *Noise speckle* : model *noise* yang memberikan warna hitam pada titik yang terkena *noise*. *Noise speckle* adalah bentuk *noise* yang biasanya terlihat titik-titik hitam dan putih pada citra seperti tebaran garam dan merica. *Noise* ini dapat dibangkitkan dengan cara membangkitkan bilangan 0 (warna hitam) pada titik titik yang secara probabilitas lebih kecil dari nilai probabilitas *noise*.
3. *Noise salt and pepper* : disebabkan karena terjadinya *error bit* dalam pengiriman data, *pixel-pixel* yang tidak berfungsi dan rusak pada lokasi memori. *Noise* ini dapat dibangkitkan dengan cara membangkitkan bilangan 255 (warna putih) pada titik-titik yang secara probabilitas lebih kecil dari nilai probabilitas *noise*.

2.2 Noise Poisson

Kebisingan *Poisson* dihasilkan dari data bukannya menambahkan *noise* buatan ke data. Jika itu gambar adalah presisi ganda, maka masukan nilai-nilai piksel diartikan sebagai sarana *Poisson* distribusi ditingkatkan oleh $1e12$. Misalnya, jika piksel masukan memiliki nilai $5.5e-12$, maka piksel output yang sesuai akan dihasilkan dari distribusi *Poisson* dengan rata-rata 5,5 dan kemudian diperkecil lagi oleh $1e12$. Jika gambar itu presisi tunggal, faktor skala yang digunakan adalah $1e6$. Jika gambarnya uint8 atau uint16, maka nilai masukan piksel digunakan secara langsung tanpa penskalaan. Misalnya, jika piksel dalam *input uint8* memiliki nilai 10, maka yang sesuai piksel output akan dihasilkan dari distribusi *Poisson* dengan *mean* 10.

2.3 Mean Filters

a. Arithmetic Mean Filter

Ini adalah metode paling mudah dari *mean filters*. Misalkan S_{xy} mewakili himpunan koordinat dalam sebuah window kotak sub image berukuran $m \times n$, berpusat di titik (x, y) . Proses *Arithmetic Mean Filtering* menghitung rata – rata nilai dari gambar yang rusak $g(x, y)$ pada area yang didefinisikan oleh S_{xy} . Nilai dari gambar yang diperbaiki \hat{f} pada tiap titik (x, y) adalah *arithmetic mean* mudah yang dihitung dengan menggunakan piksel dalam daerah yang didefinisikan oleh S_{xy} . Dengan kata lain:

$$\hat{f} = \frac{1}{mn} \sum_{(s,t) \in S_{xy}} g(s,t) \quad (1)$$

Keterangan :

x,y adalah koordinat piksel pada citra
 mn adalah lebar citra, s,t adalah nilai intensitas

Operasi ini dapat diimplementasikan dengan menggunakan *convolution mask* dimana semua koefisien mempunyai nilai $1/mn$, sebuah *mean filter* secara mudah memperhalus variasi *local* dalam sebuah *image*. *Noise* dikurangi sebuah hasil dari pengaburan.

b. Geometric Mean Filter

Sebuah *image* diperbaiki dengan menggunakan *geometric mean filter* yang diberikan oleh persamaan ini:

$$\hat{f}(x,y) = \left[\prod_{(s,t) \in S_{xy}} g(s,t) \right]^{\frac{1}{mn}} \quad (2)$$

Keterangan :

x,y adalah koordinat piksel pada citra



mn adalah dimensi citra

s,t adalah nilai intensitas

π adalah perkalian nilai piksel yang terkena *filter*

Di sini, setiap piksel yang diperbaiki diberikan oleh hasil piksel dalam sub *image window*, dinaikkan dengan factor $1/mn$. *Geometric Mean Filter* menghasilkan smoothing yang dapat dibandingkan dengan *arithmetic mean filter*, tetapi cenderung menghilangkan detail *image* lebih sedikit dalam prosesnya.

c. *Harmonic Mean Filter*

Operasi *Harmonic Mean Filtering* diberikan oleh persamaan :

$$\hat{f} = \frac{mn}{\sum_{(s,t) \in S_{xy}} \frac{1}{g(s,t)}} \quad (3)$$

Keterangan :

$m \times n$ adalah mbaris dan nkolom kernel *harmonic mean filter*

$g(s,t)$ adalah baris dan kolom piksel yang akan diproses

x,y adalah koordinat piksel pada citra

Harmonic Mean Filter bekerja dengan baik untuk salt noise, tetapi gagal untuk *noise pepper*. Ini juga bekerja baik dengan tipe noise yang lain seperti Gaussian noise.

d. *Contraharmonic Mean Filter*

Operasi *Contraharmonic Mean Filtering* menghasilkan sebuah perbaikan image berdasarkan persamaan berikut :

$$\hat{f}(x,y) = \frac{\sum_{(s,t) \in S_{xy}} g(s,t)^{Q+1}}{\sum_{(s,t) \in S_{xy}} g(s,t)^Q} \quad (4)$$

Keterangan:

$m \times n$ adalah m baris dan n kolom kernel *harmonic mean filter*

$g(s,t)$ adalah baris dan kolom piksel yang akan diproses

x,y adalah koordinat piksel pada citra

q adalah order dari *filter*

Filter ini sangat cocok untuk mengurangi atau secara virtual mengeliminasi efek *salt-and-pepper noise*. Untuk nilai Q positif, *filter* mengeliminasi *noise pepper*. Untuk nilai Q negatif ini mengeliminasi *salt noise*. Ini tidak dapat melakukan keduanya secara bersamaan. Perhatikan bahwa *contraharmonic filter* mengurangi pada arithmetic mean filter jika $Q = 0$, dan ke *harmonic mean filter* jika $Q = -1$.

2.4 Alpha-Trimmed Mean Filter (ATMF)

Alpha-trimmed mean filter pada prinsipnya adalah menghilangkan $/2$ buah piksel dengan nilai gray-level terendah dan $/2$ buah piksel dengan nilai gray-level tertinggi dari $g(s,t)$ dalam $S_{x,y}$. Range nilai d berada di antara 0 sampai $mn - 1$. Ketika $d = 0$, maka *alpha-trimmed mean filter* menjadi *arithmetic mean filter*, dan jika $d = mn - 1$, maka *alpha-trimmed mean filter* menjadi *median filter*[2].

Formula yang digunakan (Prasetyo, 2011) :

$$F(x,y) = \frac{1}{mn-d} \sum_{(s,t) \in S_{xy}} g_r(s,t) \quad (5)$$

Keterangan :

$F(x,y)$: hasil filtering

mn : ukuran panjang dan lebar *window*

$g(s,t)$: sub-image S_{xy}

S_{xy} : *window* daerah yang diliputi oleh *filter*

2.5 Adaptif Median Filter

Adaptif Median Filter adalah metode pengembangan dari *median filter* biasa. Perbedaan yang menonjol antara dua metode ini adalah bahwa besarnya *window* (jendela) yang ada pada *adaptif median filter* setiap piksel adalah variabel. Variasi ini tergantung pada nilai median dari piksel dalam *window* saat ini. Ukuran jendela akan diperluas jika nilai rata – rata adalah *impuls*. Pada proses pengolahan citra diperlukan piksel pusat (*median*) dari *window* (jendela) untuk memastikan apakah piksel tersebut merupakan suatu *impuls* atau tidak. Jika merupakan *impuls*, maka nilai piksel baru pada citra yang telah difilter akan diganti dengan nilai median dari pusat piksel dalam jendela itu. Jika tidak merupakan *impuls*, maka piksel tersebut akan dipertahankan dalam citra yang difilter [2].

Adaptif median filter mempunyai 2 tujuan yaitu mengurangi distorsi pada gambar dan menghapus *impuls noise* pada gambar. Metode ini melakukan pengolahan spasial untuk menentukan nilai mana dalam citra yang terkena *noise*



dengan membandingkan setiap piksel dengan piksel tetangganya.

Berikut ini adalah algoritma dari metode *adaptif median filter* :

Zmin = Nilai tingkat keabuan minimum pada Sxy

Zmax = Nilai tingkat keabuan maksimum pada Sxy.

Zmed = Nilai tengah keabuan pada Sxy.

Zxy = tingkat keabuan pada koordinat (x,y).

Smax = Ukuran maksimum Sxy.

Ada dua tahap pada metode ini yaitu tahap A dan tahap B :

Tahap A : A1 = Zmed – Zmin

A2 = Zmed – Zmax

If A1 > 0 dan A2 < 0,

lanjut ke tahap B

Else tambah ukuran *window*

If ukuran *window* <= Smax, ulangi tahap A

Else output Zxy.

Tahap B : B1 = Zxy – Zmin

B2 = Zxy – Zmax

If B1 > 0 dan B2 < 0 output Zxy

Else output Zmed

2.6 Mean Square Error (MSE)

Mean Square Error (MSE) adalah rata-rata kuadrat nilai kesalahan antara citra asli sebelum mengalami proses *filtering* dengan citra hasil yang telah mengalami proses *filtering*. Secara matematis *MSE* dapat dirumuskan (Sutoyo, 2009) :

$$MSE = \frac{1}{M \times N} \sum_{i=1}^M \sum_{j=1}^M (f_a(i, j) - f_b(i, j))^2 \quad (6)$$

Keterangan :

MSE = *Mean Square Error*

M x N = ukuran panjang dan lebar citra

$f_a(i, j)$ = nilai intensitas citra pada titik (i,j) sebelum dilakukan *filtering*

$f_b(i, j)$ = nilai intensitas citra pada titik (i,j) setelah dilakukan *filtering*

Nilai *MSE* yang semakin kecil menandakan prosedur *filtering* yang digunakan semakin baik. Hal ini dikarenakan kualitas citra sebelum mengalami proses *filtering* hampir sama dengan kualitas citra setelah mengalami proses *filtering* (Sutoyo, 2009). Dalam pengertian yang lebih singkat, nilai *MSE* digunakan untuk mengevaluasi perbaikan kesalahan pada suatu metode.

2.7 Peak Signal to Noise Ratio (PSNR)

Peak Signal to Noise Ratio (PSNR) sering digunakan sebagai parameter pembandingan antara citra yang telah dikonstruksi (diberi *noise* atau diberi *filter*) dengan citra aslinya. *PSNR* merupakan nilai perbandingan antara nilai maksimum warna pada hasil citra *filtering* dengan kuantitas gangguan (*noise*) yang merupakan akar rata-rata kuadrat nilai kesalahan (\sqrt{MSE}). *PSNR* dinyatakan dengan satuan desibel (dB). Secara matematis, nilai *PSNR* dapat dirumuskan (Santoso, et al. 2013) :

$$PSNR = 20 \log_{10} \left(\frac{255}{\sqrt{MSE}} \right) \quad (7)$$

Keterangan :

PSNR = *Peak Signal to Noise Ratio*

255 = nilai maksimum intensitas citra

MSE = nilai *Mean Square Error*

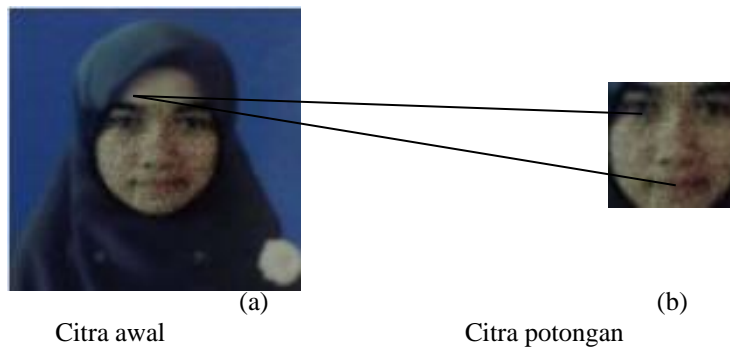
Semakin kecil nilai kuantitas gangguan (*noise*), citra hasil proses *filtering* juga akan semakin baik. Karena, kualitas citra setelah *filtering* akan mendekati kualitas citra asli. Nilai kuantitas gangguan (\sqrt{MSE}) yang semakin kecil, akan menyebabkan nilai *PSNR* semakin besar. Oleh karena itu, semakin besar nilai *PSNR*, semakin bagus kualitas citra hasil *filtering* [3].

3. HASIL DAN PEMBAHASAN

3.1 Penerapan Metode *Alpha-Trimmed Mean Filter*

Proses yang dilakukan adalah citra awal yang diambil dari pencahayaan dari citra berwarna. Kemudian melakukan pemilteran untuk mereduksi *noise poisson* dengan menggunakan metode *Alpha Trimmed Mean Filter*. Adapun tujuan *Alpha trimmed filter* adalah menghilangkan piksel dengan nilai gray level terendah dengan gray level tertinggi pada *noise*. Dengan menerapkan metode *Alpha Trimmed Mean Filter* ini berguna dalam mengurangi *noise poisson* pada citra. Citra

yang diambil memiliki variasi pencahayaan dari citra berwarna. persentase noise *poisson* menggunakan metode Alpha Trimmed Mean Filter, MSE dan PSNR. Berikut contoh citra 24 bit pada noise *poisson* dengan format *bmp* dengan resolusi 5x5 piksel dapat dilihat pada gambar 1 sebagai berikut.



Gambar 1. Analisa Citra noise poisson

Dari gambar diatas sebuah *noise poisson* dengan resolusi 5x5 piksel. Adapun memperlihatkan matriks pada nilai-nilai piksel tersebut dapat dilihat dibawah ini :

F (x,y)

232	221	209	189	162
239	219	200	176	154
246	216	186	160	146
232	198	161	136	129
205	170	135	113	112

Kernel

2	5	1
2	6	5
3	3	1

Selanjutnya matriks tersebut dihitung dengan menggunakan rumus *Alpha Trimmed Mean Filter* yaitu :

$$F(x, y) = \frac{1}{mn-d} \sum (s, t) \in s_{xy} \text{ } 9r(s, t)$$

Perhitungan *trim mean* dimulai dari koordinat f(0,0) dan dilakukan dengan mengambil matriks 3x3 dari matriks. Contoh penggunaan rumus *Alpha Trimmed Mean Filter* dapat dilihat seperti bawah ini :

$$f(1,1) = \frac{1}{9-2} \times (186 + 200 + 209 + 216 + 219 + 221 + 232 + 239 + 246)$$

$$= \frac{1}{7} \times 1968 = 281$$

$$f(1,2) = \frac{1}{9-5} \times (160 + 176 + 186 + 189 + 200 + 209 + 216 + 219 + 221)$$

$$= \frac{1}{4} \times 1776 = 444$$

$$f(1,3) = \frac{1}{9-1} \times (146 + 154 + 160 + 162 + 176 + 186 + 189 + 200 + 209)$$

$$= \frac{1}{8} \times 1582 = 197$$

$$f(2,1) = \frac{1}{9-2} \times (161 + 186 + 198 + 200 + 216 + 219 + 232 + 239 + 246)$$

$$= \frac{1}{7} \times 1897 = 271$$

$$f(2,2) = \frac{1}{9-6} \times (136 + 160 + 161 + 176 + 186 + 198 + 200 + 216 + 219)$$

$$= \frac{1}{3} \times 1652 = 550$$

$$f(2,3) = \frac{1}{9-5} \times (129 + 136 + 146 + 154 + 160 + 161 + 176 + 186 + 200)$$

$$= \frac{1}{4} \times 1448 = 362$$

$$f(3,1) = \frac{1}{9-3} \times (135 + 161 + 170 + 186 + 198 + 205 + 216 + 232 + 246)$$

$$= \frac{1}{6} \times 1749 = 291$$

$$f(3,2) = \frac{1}{9-3} \times (113 + 135 + 136 + 160 + 161 + 170 + 186 + 198 + 216)$$

$$= \frac{1}{6} \times 1475 = 245$$

$$f(3,3) = \frac{1}{9-1} \times (112 + 113 + 129 + 135 + 136 + 146 + 160 + 161 + 186)$$

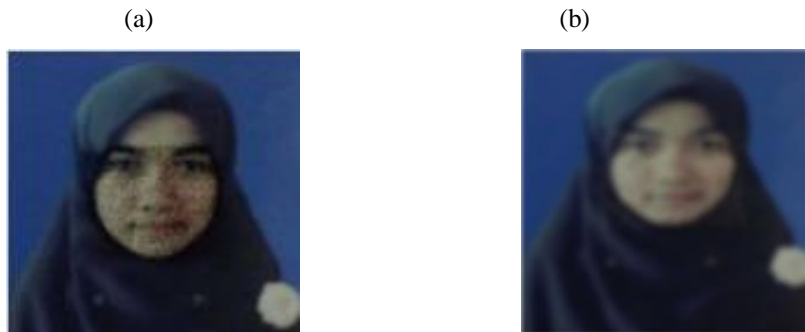
$$= \frac{1}{8} \times 1278 = 159$$

Hasil perhitungan alpha trimmed mean filter diatas adalah $g(1,1)$ adalah $f(1,1) = 281$, sehingga nilai 119 diganti dengan nilai 281 dan proses selanjutnya sampai selesai. Seperti yang dapat dilihat pada pada matriks berikut ini :

232	221	209	189	162
239	219	200	176	154
246	216	186	160	146
232	198	161	136	129
205	170	135	113	112

➔

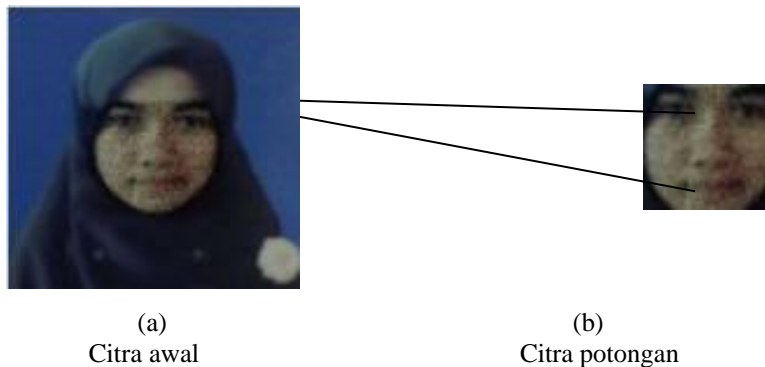
232	221	209	189	162
239	281	444	197	154
246	271	550	362	146
232	291	245	159	129
205	170	135	113	112



Gambar 2. (a) Matriks citra 5x5 sebelum di *filtering* dan (b) matriks citra sesudah di *filtering* metode *Alpha Trimmed Mean Filter*

3.2 Penerapan Metode Adaptive Median Filter

Proses yang dilakukan adalah citra awal yang diambil dari pencahayaan dari citra berwarna. Kemudian melakukan pemilteran untuk mereduksi *noise poisson* dengan menggunakan metode *Adaptif Median filter*. Adapun tujuan *adaptif median filter* mengurangi distorsi pada gambar dan menghapus impuls *noise* pada gambar. Dengan menerapkan metode *Adaptif Median filter* ini berguna dalam mengurangi *noise poisson* pada citra. Citra yang diambil memiliki variasi pencahayaan dari citra grayscale. persentase *noise poisson* menggunakan metode Median filter, MSE dan PSNR. Berikut contoh citra 24 bit pada *noise poisson* dengan format *bmp* dengan resolusi 5x5 piksel dapat dilihat pada gambar 3.3 sebagai berikut.



Gambar 3. Analisa Citra *noise poisson*

Dari gambar diatas sebuah *noise poisson* dengan resolusi 5x5 piksel. Adapun memperlihatkan hasil perhitungan pada *Alpha Trimmed Mean Filter* selanjutnya dihitung menggunakan *adaptif median filter*. Berikut matriks pada nilai-nilai piksel tersebut dapat dilihat dibawah ini :

$F(x,y)$

232	221	209	189	162
239	281	444	197	154
246	271	550	362	146
232	291	245	159	129
205	170	135	113	112

Perhitungan *adaptif median filter* dimulai dari koordinat $f(0,0)$ dan dilakukan dengan mengambil matriks 3x3 dari matriks. Contoh penggunaan metode Adaptive Median Filter dapat dilihat seperti bawah ini :

$G(1,1) = 232, 221, 209, 239, 281, 444, 246, 271, 550$

$$\begin{aligned}
 &= 209, 221, 232, 239, \mathbf{246}, 271, 281, 444, 550 \\
 &= 246 \\
 G(1,2) &= 221, 209, 189, 281, 444, 197, 271, 550, 362 \\
 &= 189, 197, 20, 221, \mathbf{271}, 281, 362, 444, 550 \\
 &= 271 \\
 G(1,3) &= 209, 189, 162, 444, 197, 154, 550, 362, 146 \\
 &= 146, 154, 162, 189, \mathbf{197}, 209, 362, 444, 550 \\
 &= 197 \\
 G(2,1) &= 239, 281, 444, 246, 271, 550, 232, 291, 245 \\
 &= 232, 239, 245, 246, \mathbf{271}, 281, 191, 444, 550 \\
 &= 271 \\
 G(2,2) &= 281, 444, 197, 271, 550, 362, 291, 245, 159 \\
 &= 159, 197, 2,45, 271, \mathbf{281}, 291, 362, 444, 550 \\
 &= 281 \\
 G(2,3) &= 444, 197, 154, 550, 362, 146, 245, 159, 129 \\
 &= 129, 146, 154, 159, \mathbf{197}, 245, 362, 444, 550 \\
 &= 197 \\
 G(3,1) &= 246, 271, 550, 232, 291, 245, 205, 170, 135 \\
 &= 135, 170, 232, 205, \mathbf{245}, 246, 271, 291, , 550 \\
 &= 245 \\
 G(3,2) &= 271, 550, 362, 291, 245, 159, 170, 135, 113 \\
 &= 113, 135, 159, 170, \mathbf{245}, 271, 291, 362, 550 \\
 &= 245 \\
 G(3,3) &= 550, 362, 146, 245, 159, 129, 135, 113, 112 \\
 &= 112, 113, 129, 135, \mathbf{146}, 159, 245, 362, 550 \\
 &= 146
 \end{aligned}$$

Hasil perhitungan jendela adaptive median filter diatas adalah $g(1,1)$ adalah $f(1,1)$ bila dijadikan baris : 246, 271, 197, 271, 281, 197, 245, 245, 146 dan bila diurutkan menjadi : 146, 197, 197, 245, **245**, 246, 271, 271, 281. Sehingga median/titik tengahnya adalah 245 dan lakukan proses selanjutnya sampai selesai. Seperti yang dapat dilihat pada pada matriks berikut ini :

232	221	209	189	162
239	146	197	197	154
246	245	245	246	146
232	271	271	281	129
205	170	135	113	112

➔

232	221	209	189	162
239	146	197	197	154
246	245	245	246	146
232	271	271	281	129
205	170	135	113	112

(a)



(b)



Gambar 4. (a) Matriks citra 5x5 sebelum di *filtering* dan (b) matriks citra sesudah di *filtering* metode adaptive mean filter

Selanjutnya menghitung MSE, PSNR untuk mengetahui hasil perbandingan dari metode alpha Trimmed mean filter dan Adaptive Mean Filter.

Perhitungan MSE dan PSNR dari hasil Alpha trimmed mean filter :

232	221	209	189	162
239	146	197	197	154
246	245	245	246	146
232	271	271	281	129
205	170	135	113	112

2	5	1
2	6	5
3	3	1

$$\text{MSE} = \frac{(232 - 2)^2 + (221 - 5)^2 + (209 - 1)^2 + (239 - 2)^2 + (281 - 6)^2 + (444 - 5)^2 + (246 - 3)^2 + (271 - 3)^2 + (550 - 1)^2}{3 \times 3}$$

$$\text{MSE} = \frac{529 + 466 + 432 + 561 + 756 + 192 + 590 + 718 + 301}{9} = 505$$

$$\text{PSNR} = 20 \log_{10} \frac{7}{\sqrt{505}} = 20,311$$

Perhitungan MSE dan PSNR dari hasil Adaptive media filter filter :

232	221	209	189	162
239	146	197	197	154
246	245	245	246	146
232	271	271	281	129
205	170	135	113	112



2	5	1
2	6	5
3	3	1

$$\text{MSE} = \frac{(232 - 2)^2 + (221 - 5)^2 + (209 - 1)^2 + (239 - 2)^2 + (146 - 6)^2 + (197 - 5)^2 + (246 - 3)^2 + (245 - 3)^2 + (245 - 1)^2}{3 \times 3}$$

$$\text{MSE} = \frac{529 + 466 + 432 + 561 + 196 + 368 + 590 + 585 + 595}{9} = 480$$

$$\text{PSNR} = 20 \log_{10} \frac{7}{\sqrt{480}} = 20,319$$

Tabel 1. Hasil implementasi Metode *Alpha-Trimmed Mean Filter* dan *Adaptive Median Filter*

Metode	Hasil Citra	MSE	PSNR
<i>Alpha-Trimmed Mean Filter</i>		505	20,311 dB
<i>Adaptive Median Filter</i>		480	20,319 dB

Keterangan :

Berdasarkan rumus MSE dan PSNR dari metode *Alpha-Trimmed Mean Filter* dan *Adaptive Median Filter* yang dalam perhitungan di atas, terlihat bahwa perbandingan kedua metode tersebut berbeda dalam pemilteran pada noise *poisson* nilai PSNR ditentukan oleh nilai dari MSE, semakin kecil nilai MSE maka akan menyebabkan semakin besarnya nilai dari PSNR.



4. KESIMPULAN

Berdasarkan hasil yang di dapat dalam penelitian dan penyusunan skripsi ini serta disesuaikan dengan tujuan, maka diperoleh kesimpulan metode Filtering yang digunakan adalah Alpha Trimmed Mean Filter dan Adaptive Median Filter. Citra yang memiliki Noise poisson pada citra digital akan memiliki bintik-bintik disebabkan karena terjadinya error bit dari pixel-pixel, tetapi setelah dilakukan proses pemfilteran pada noise dengan Alpha Trimmed Mean Filter dan Adaptive Median Filter maka noise-nya akan hilang sehingga citra akan menjadi lebih jelas. Proses pemilteran noise poisson pada citra digital melalui aplikasi yang telah dibangun maka dapat dibedakan bahwa citra pemilteran dengan Alpha Trimmed Mean Filter hasilnya bagus dan setelah diperbaiki dengan Adaptive Median Filter hasilnya menjadi lebih bagus lagi.

REFERENCES

- [1] Kurnia Rizky Amalia. 2016. Implementasi Metode Alpha Trimmed Mean Filter Menggunakan Opecl Untuk Penghapusan Noise Pada Citra Berwarna. Skripsi. Malang: Universitas Islam Negeri Malang.
- [2] Ivan Maulana dan Pulung Nurtanto Andono. Analisis perbandingan Adaptvie Media Filter Dan Median Filter Dalam Reduksi Noise Salt & Pepper. Cogito Smart Journal. Vol. 2, No. 2. 2016.
- [3] Anhar Ismail, 2017. Analisis perbandingan kombinasi metode alpha- trimmed mean filter dan Geometric Mean Filter Untuk Mereduksi Noise Pada Citra Digital. Skripsi USU Medan.
- [4] Usman, Nurdin, 2002, Konteks Implementasi Berbasis Kurikulum Jakarta,PT. Raja Grafindo Persada
- [5] Aniasi Murni, 2007, Pengantar Pengolahan Citra, PT Elex Media Komputindo, Jakarta
- [6] Wijaya, M.C dan A.Prijono, 2007. Pengolahan citra digital menggunakan matlab procesing toolbox. Bandung : informatika. Publishing.
- [7] Abdul kadir & Adhi susanto, 2013. Teori dan Aplikasi Pengolahan Citra, Penerbit Andi, Yogyakarta.
- [8] Munir, Rinaldi. 2007. Pengolahan Citra Digital Dengan Pendekatan Algoritmik, Bandung: Program Studi Teknik Informatika STEI ITB
- [9] Darma putra, 2010, Pengolahan Citra Digital, Penerbit Andi, Yogyakarta
- [10] Manjunath Savadatti. Analysis and Comparison of Filtering Techniques for Image Restoration. International journal of multidisciplinary approach and studies , Vol.2 No.2, 2015. ISSN : 2348-537X. India.
- [11] Indrajani, S.Kom, MM. 2011. Perancangan Basis Data Dalam Allin1.