

## PENERAPAN METODE *GAUSSIAN SMOOTHING* UNTUK MEREDUKSI NOISE PADA CITRA DIGITAL

Sastia Hendri Wibowo<sup>1</sup>, Firman Susanto<sup>2</sup>

<sup>1,2</sup>Program Studi Teknik Informatika Universitas Muhammadiyah Bengkulu  
Jl. Bali, Kel. Kampung Bali, Kec. Teluk Segara, Kota Bengkulu, Bengkulu 38119

<sup>1</sup>mastersastia@yahoo.com

### ABSTRACT

Image is one component of multimedia that plays an important role because it contains information in visual form. Imagery has more information that can be conveyed than information in text form. However, not all images have good quality, so the image cannot display clear information. In this study noise reduction is a process to reduce or reduce noise in a digital image in improving image quality (image enhancement). The method used to reduce noise is to do filtering. The purpose of this research is to implement Gaussian-smoothing method to reduce noise on image affected by Gaussian Noise, and Salt Pepper Noise. This study is expected to create a solution in the form of applications and information on how to reduce noise in digital images and provide ease to improve the image quality into a better image. This research is planned not to be done on an institution or institution but done independently, both in the secondary data collection and in data processing process. Data collection method that writer use in this research is by doing literature study and laboratory data is analyzed to be implemented in the form of software. By using Gaussian filter method, the noise contained in the digital image as much as possible can be reduced. An image that has noise (noise) after the resulting image is improved will look smoother and fuzzy.

Keyword: Image Processing, Digital Image, Noise, Gaussian Smothing.

### INTISARI

Citra merupakan salah satu komponen dari multimedia yang memegang peranan penting karena mengandung informasi dalam bentuk visual. Citra memiliki lebih banyak informasi yang dapat disampaikan dibandingkan dengan informasi dalam bentuk teks. Akan tetapi tidak semua citra memiliki kualitas yang bagus, sehingga citra tersebut tidak bisa menampilkan informasi yang jelas. Pada penelitian ini pengurangan *noise* atau *denoise* merupakan suatu proses untuk mereduksi atau mengurangi *noise* pada sebuah citra digital dalam meningkatkan kualitas citra (*image enhancement*). Metode yang digunakan untuk mereduksi *noise* atau *denoise* tersebut adalah dengan melakukan *filtering*. Tujuan penelitian ini adalah mengimplementasikan metode *Gaussian smothing* untuk mereduksi *noise* pada citra yang terkena *Gaussian Noise*, dan *Salt Pepper Noise*. Penelitian ini diharapkan mampu menciptakan suatu solusi berupa aplikasi dan informasi bagaimana cara mereduksi *noise* pada citra digital dan memberikan kemudahan untuk memperbaiki kualitas citra menjadi citra yang lebih bagus. Penelitian ini rencananya tidak akan dilakukan pada suatu lembaga atau pun instansi tetapi dilakukan secara mandiri, baik dalam pengambilan data sekunder maupun proses pengolahan datanya. Metode pengumpulan data yang penulis gunakan dalam penelitian ini adalah dengan melakukan studi pustaka dan laboratorium data tersebut dianalisis untuk diimplementasikan ke dalam bentuk perangkat lunak. Dengan menggunakan metode filter *Gaussian* maka *noise* yang terdapat pada citra digital sedapat mungkin bisa berkurang. Citra yang memiliki *noise* (derau) setelah diperbaiki gambar yang dihasilkan akan terlihat lebih halus dan kabur.

Kata Kunci: Pengolahan Citra, Citra Digital, *Noise*, *Gaussian Smothing*.

### I. PENDAHULUAN

Kecanggihan teknologi pada saat ini memang tidak bisa diragukan lagi. Untuk mendapatkan informasi secara tepat dan praktis tidaklah menjadi hal yang sulit. Hal itu terjadi karena informasi bisa kita peroleh darimana saja. Misalnya informasi berupa citra (gambar), teks, suara, video dan lain sebagainya. Penggabungan komponen di atas disebut multimedia.

Citra sebagai salah satu komponen multimedia yang memegang peranan sangat penting sebagai bentuk informasi visual. Citra mempunyai karakteris-

tik yang tidak dimiliki oleh data teks, yaitu citra kaya dengan informasi (Nurlailah, Siti, 2010). Dalam pengambilan citra secara digital, permasalahan warna sering kita temui. Warna selalu berubah tergantung dengan kondisi pencahayaan pada saat itu.

Dalam pengambilan citra dengan kamera sering terjadi hal-hal yang tidak sesuai dengan yang diharapkan karena adanya gangguan yang terjadi pada citra hasil capture (Ahmad, Usman, 2005). Beberapa kemungkinan gangguan yang terjadi pada hasil pengambilan citra dengan kamera yaitu adanya kabut yang menghalangi objek yang sedang di

capture, lensa kamera kotor, adanya bintik-bintik yang disebabkan oleh proses *capturing* yang tidak sempurna, lensa kamera tidak fokus atau gangguan yang disebabkan oleh kotoran-kotoran yang menempel pada citra, dan lain sebagainya. Setiap gangguan yang terjadi pada citra tersebut dinamakan *Noise*. Secara umum citra yang demikian dapat dikatakan kualitasnya masih rendah, baik oleh karena adanya *noise*, maupun oleh sebab lainnya seperti pencahayaan yang belum sempurna atau karena citra yang sudah tersimpan dengan waktu yang cukup lama. Citra yang memiliki kualitas seperti ini membutuhkan langkah-langkah perbaikan dan kualitasnya perlu ditingkatkan untuk memfasilitasi pengolahan yang akan dilakukan.

Beberapa jenis *noise* yang umum dijumpai adalah *Gaussian Noise*, *Salt and Pepper Noise*, *Speckle Noise*, dan *Exponential Noise*. *Gaussian Noise* merupakan model *noise* yang mengikuti distribusi normal standar dengan rata-rata 0 dan standar deviasi 1. Efek dari *Gaussian Noise* ini pada gambar adalah munculnya titik-titik berwarna yang jumlahnya sama dengan persentase *noise*. *Salt and Pepper Noise* adalah bentuk *noise* yang biasanya terlihat titik-titik hitam dan putih pada citra seperti tebaran garam dan merica. (Kadir, 2013).

Pengurangan *noise* atau *denoise* merupakan suatu proses untuk mereduksi atau mengurangi *noise* pada sebuah citra digital dalam meningkatkan kualitas citra (*image enhancement*) yang merupakan langkah awal dalam *image processing*. Salah satu metode yang digunakan untuk mereduksi *noise* atau *denoise* tersebut adalah dengan melakukan *filtering* pada citra digital contohnya, yaitu dengan menggunakan Mean Filter dan Median Filter. *Mean Filter* biasanya disebut sebagai filter penghalus (*smoothing filters*). Filter ini biasa digunakan untuk mengaburkan (*blurring*) citra untuk mereduksi *noise*. Sementara itu, Median Filter memberikan kemampuan untuk pengurangan *noise* yang sangat bagus dengan memperhatikan *blurring*. Pada bagian tertentu *Median Filter* juga baik untuk menghilangkan *Salt and pepper Noise* karena sifat median yang menjauhi hitam dan putih.

Masalah yang akan dibahas dalam penelitian ini adalah bagaimana mereduksi *noise* pada citra digital dengan menggunakan metode *Gaussian smothing*.

## II. TINJAUAN PUSTAKA

### A) Citra Digital

Secara matematis citra merupakan suatu fungsi intensitas cahaya pada bidang 2-dimensi sehingga dapat disimbolkan dengan  $f(x,y)$ , dengan  $(x,y)$  merupakan titik koordinat pada bidang 2-dimensi dan  $f(x,y)$  merupakan nilai intensitas cahaya pada titik

tersebut. Simbol  $x$ ,  $y$  dan  $f(x,y)$  bernilai diskrit. (Nurlailah, 2010).

### B) Jenis-jenis Citra Digital

1) *Citra Warna*. Setiap titik (piksel) pada citra warna mewakili warna yang merupakan kombinasi dari tiga warna dasar yaitu merah, hijau dan biru yang dikenal sebagai citra RGB (*Red, Green, Blue*). Setiap warna dasar mempunyai intensitas sendiri dengan nilai maksimum 255 (8 bit). Setiap titik pada citra warna membutuhkan data 3 byte. Jumlah kemungkinan kombinasi warna untuk citra warna adalah  $224 =$  lebih dari 16 juta warna, disebut *true color* karena dianggap mencakup semua warna yang ada.

2) *Citra Grayscale*. Citra *grayscale* merupakan citra yang skala keabuannya menggunakan 8-bit, setiap pixel-nya mempunyai derajat keabuan antara 0 untuk warna hitam dan 255 untuk warna putih. Nilai tersebut dihasilkan dari 28 yaitu 256 nilai keabuan. Angka 8 merupakan jumlah bit yang digunakan. Gambar 2.1.3 berikut merupakan skala keabuan menggunakan 8-bit.

3) *Citra Biner*. Citra biner merupakan citra yang hanya memiliki dua warna, yaitu hitam dan putih. Citra biner membutuhkan satu bit di memori untuk menyimpan kedua warna tersebut.

### C) Resolusi Citra

Masing-masing elemen dari matriks yang tidak lain adalah elemen dari citra digital yang merupakan bagian terkecil dari suatu citra disebut dengan *picture element* atau *pixel*. Jadi jika sebuah citra mempunyai ukuran 125 x 96 *pixel* artinya citra tersebut mempunyai 125 baris dan 96 kolom dalam sebuah matriks. Sedangkan banyaknya *pixel* yang digunakan untuk membentuk suatu citra digital disebut *resolusi*. Semakin tinggi *resolusi* maka citra yang terbentuk akan semakin baik.

### D) Pengolahan Citra

Pengolahan citra merupakan proses untuk menghasilkan citra sesuai dengan keinginan atau kualitasnya menjadi lebih baik. Inputannya adalah citra dan keluarannya juga citra tapi dengan kualitas lebih baik daripada citra masukan.

Umumnya, operasi-operasi pengolahan citra diterapkan pada citra bila : 1) Perbaikan atau modifikasi citra untuk meningkatkan kualitas visual atau menonjolkan beberapa aspek informasi yang terkandung dalam citra. 2) Elemen di dalam citra perlu di kelompokkan, dicocokkan atau diukur. 3) Sebagian citra perlu digabung dengan bagian citra yang lain.

### E) Metode Gaussian

Titik-titik tepi yang dilacak dengan cara menemukan perpotongan dengan sumbu  $x$  oleh

fungsi turunan kedua dari intensitas citra sangat sensitif terhadap noise. Oleh karena itu, diperlukan suatu filter yang dapat melemahkan noise sebelum penguatan tepi dilakukan. (Ahmad, Usman, 2005). Hal ini dapat dilakukan sekaligus dengan Laplacian of Gaussian, yang merupakan kombinasi dari filter Gaussian dan pelacak tepi Laplacian.

Karakteristik mendasar dari pelacak tepi Laplacian of Gaussian adalah : 1) Filterisasi pengaburnya adalah filter Gaussian, 2) Penguatan tepi adalah fungsi turunan kedua, 3) Kriteria pelacakan adalah dengan menemukan titik perpotongan dengan sumbu x dalam fungsi turunan kedua yang bersesuaian dengan puncak dalam suatu fungsi turunan pertama, dan 4) Lokasi dari tepi dapat diduga dengan resolusi subpixel menggunakan interpolasi linear.

Dengan metode ini, citra sebelumnya harus dikonvolusi menggunakan filter Gaussian. Langkah ini akan mengaburkan citra dan melemahkan noise. Titik-titik noise yang terisolasi dan susunan kecil titik noise akan dihilangkan atau dilemahkan. Karena pengaburan berakibat pada pelebaran tepi obyek, pelacak tepi hanya akan menganggap suatu titik adalah milik tepi jika titik tersebut merupakan puncak lokal dalam gradien. Hal ini dapat dicapai dengan menemukan titik perpotongan dari fungsi turunan kedua. Laplacian digunakan sebagai perkiraan dari fungsi turunan kedua dalam dua dimensi karena ia merupakan operator isotropik (Ahmad, Usman, 2005).

Untuk menghindari pelacakan tepi yang tidak berbeda nyata, hanya titik perpotongan dengan sumbu x yang bersesuaian dengan turunan pertama dan bernilai di atas nilai tertentu saja yang dipilih sebagai titik-titik tepi. Hasil keluaran dari operator Laplacian of Gaussian,  $h(x, y)$ , didapatkan dari operasi konvolusi.

$$h(x,y) = \nabla^2[g(x,y)*f(x,y)]$$

Menggunakan aturan turunan untuk konvolusi, didapat :

$$h(x,y) = [\nabla^2 g(x,y)] * f(x,y)$$

Bentuk persamaan di atas biasa disebut operator tepi Meksiko karena bila nilai di sebelah kiri tanda sama dengan diplotkan terhadap koordinat x dan y akan membentuk lekukan seperti tepi Meksiko.

Filter Gaussian adalah salah satu filter linear dengan nilai pembobotan untuk setiap anggotanya dipilih berdasarkan bentuk fungsi Gaussian. Filter ini digunakan untuk menghilangkan noise yang bersifat sebaran normal. (Ahmad, Usman, 2005).

Untuk pengolahan citra digital yang merupakan bidang dua dimensi, dinyatakan dalam persamaan dua variabel bebas yang bersifat diskret sebagai berikut:

Salah satu filter Gaussian berukuran 7 X 7 yang umum digunakan untuk proses menghilangkan noise adalah sebagai berikut:

$$\begin{pmatrix} 1 & 1 & 2 & 2 & 2 & 1 & 1 \\ 1 & 2 & 2 & 4 & 2 & 2 & 1 \\ 2 & 2 & 4 & 8 & 4 & 4 & 2 \\ 2 & 4 & 8 & 16 & 8 & 4 & 2 \\ 2 & 2 & 4 & 8 & 4 & 4 & 2 \\ 1 & 2 & 2 & 4 & 2 & 2 & 1 \\ 1 & 1 & 2 & 2 & 2 & 1 & 1 \end{pmatrix}$$

F) Jenis-jenis Noise

Noise pada citra tidak hanya terjadi karena ketidaksempurnaan dalam proses *capturing*, tetapi bisa juga disebabkan oleh kotoran-kotoran yang terjadi pada citra. Noise muncul biasanya sebagai akibat dari pembelokan pixel yang tidak baik. Gangguan tersebut umumnya berupa variasi intensitas suatu pixel yang tidak berkorelasi dengan pixel-pixel tetangganya. Secara visual, gangguan mudah dilihat oleh mata karena tampak berbeda dengan pixel tetangganya. Pixel yang mengalami gangguan umumnya memiliki frekuensi tinggi. Komponen citra yang berfrekuensi rendah umumnya mempunyai nilai pixel konstan atau berubah sangat lambat. (Munir, Rinaldi, 2004)

1) *Gaussian Noise*. Gaussian Noise merupakan jenis noise yang mengikuti distribusi normal standar dengan rata-rata 0 dan standard deviasi 1. Gaussian Noise dapat dibangkitkan dengan cara membangkitkan bilangan acak dengan nilai berkisar antara 0 dan 1. Kemudian pada titik-titik yang terkena noise, nilai fungsi citra ditambahkan dengan noise yang ada. Efek dari Gaussian Noise pada citra adalah munculnya titik-titik berwarna yang jumlahnya sama dengan persentase noise. (Prihatini, Tuti Adi, 2010).

2) *Salt and Pepper Noise*. Salt and Pepper Noise biasa dinamakan sebagai noise impuls positif dan negatif. Noise ini disebabkan karena terjadinya error bit dalam pengiriman data, pixel-pixel yang tidak berfungsi dan kerusakan pada lokasi memori. Bentuk noise ini berupa titik-titik hitam atau putih di dalam citra. (Kadir, A. & Susanto, A. 2013).

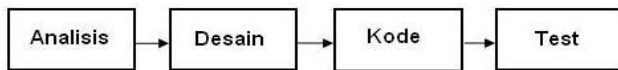
3) *Speckle Noise*. *Speckle Noise* merupakan model noise yang memberikan warna hitam pada titik yang terkena noise. *Speckle Noise* dapat dibangkitkan dengan membangkitkan bilangan 0 (warna hitam) pada titik-titik yang secara probabilitas lebih kecil dari nilai probabilitas noise. (Prihatini, Tuti Adi, 2010).

4) *Exponential Noise*. *Exponential Noise* sering disebut juga dengan noise eksponensial negatif yang merupakan jenis *noise* yang dihasilkan oleh laser yang koheren ketika citra diperoleh, sehingga noise ini sering disebut sebagai bercak laser. (Kadir, A. & Susanto, A. 2013).

### III. METODOLOGI PENELITIAN

#### A) Metode Pengembangan Sistem

Berikut ini merupakan metode yang penulis gunakan dalam penelitian ini yaitu metode sekuensial linier.



Gambar 1. Model Sekuensial Linear

1) *Analisis kebutuhan perangkat lunak.* Pengumpulan kebutuhan dengan fokus pada perangkat lunak, yang meliputi: Domain informasi, fungsi yang dibutuhkan, unjuk kerja/performansi dan antarmuka. Hasilnya harus didokumentasi dan direview ke pelanggan.

2) *Desain.* Ada 4 atribut untuk program yaitu : Struktur Data, Arsitektur perangkat lunak, Prosedur detil dan Karakteristik Antarmuka. Proses desain mengubah kebutuhan-kebutuhan menjadi bentuk karakteristik yang dimengerti perangkat lunak sebelum dimulai penulisan program. Desain ini harus terdokumentasi dengan baik dan menjadi bagian konfigurasi perangkat lunak.

3) *Generasi kode.* Penterjemahan perancangan ke bentuk yang dapat dimengerti oleh mesin, dengan menggunakan bahasa pemrograman.

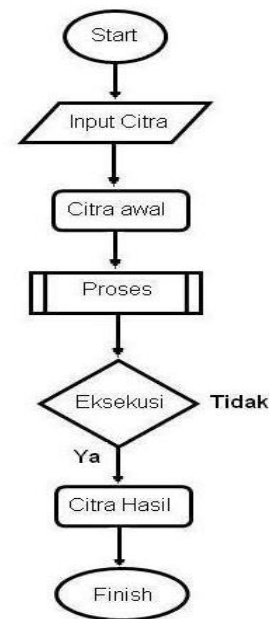
4) *Pengujian.* Setelah kode program selesai testing dapat dilakukan. Testing memfokuskan pada logika internal dari perangkat lunak, fungsi eksternal dan mencari segala kemungkinan kesalahan dan memeriksa apakah sesuai dengan hasil yang diinginkan.

#### B) Analisa Sistem

Dalam penelitian ini akan dilakukan proses perbaikan kualitas citra, adapun citra yang digunakan dalam penelitian ini adalah citra digital dengan format citra adalah bitmap (\*.bmp), JPEG (\*.jpg). adapun alasan pemakaian format citra yang disebutkan diatas adalah untuk memudahkan pengguna dalam menjalankan atau menggunakan aplikasi ini tanpa terbatas pada format citra tertentu.

#### C) Diagram Alir ( Flowchart )

Diagram alir adalah bagan yang menggambarkan arus logika dari data yang akan diproses dalam suatu program dari awal sampai akhir (Jogiyanto, 1999). Diagram alir berikut ini Menggambarkan secara keseluruhan urutan proses/logika, dalam perbaikan kualitas citra (reduksi noisi pada citra) yang akan di selesaikan, dan menunjukkan tingkat dari detail penyelesaian menggunakan metode *Gaussian smothing*.



Gambar 2. Diagram Alir Sistem

### IV. PEMBAHASAN

#### A) Pembahasan

Filter Gaussian secara meluas telah digunakan dalam bidang analisis citra terutama untuk proses penghalusan (*smoothing*), pengaburan (*Bluring*), menghilangkan detil, dan menghilangkan Noise. Pada filter Gaussian, nilai intensitas setiap pixel diganti dengan rata-rata dari nilai pembobotan untuk setiap pixel-pixel tetangganya dan pixel itu sendiri. Pixel tetangga adalah pixel yang berada disekeliling pixel yang dimaksud. Jumlah tetangga yang dilibatkan tergantung pada filter yang dirancang. Berikut pembahasan pelembutan citra (*image smoothing*) dan penapisan derau (*noise filtering*).



Gambar 4.1 Citra Ady yang mengalami gangguan berupa *spike*

Dalam penelitian ini digunakan operasi pelembutan dengan menggunakan metode Gaussian smothing yang digunakan adalah Operasi perataan di atas dapat dipandang sebagai konvolusi

antara citra  $f(x,y)$  dengan penapis  $h(x,y)$ :  $g(x,y) = f(x,y) * h(x,y)$

Berikut adalah gambaran matriks citra dengan ukuran 6 x 6 pixel. Scanline dari kiri ke kanan disebut dengan scanline horizontal, dimana ditandai dengan simbol X. Scanline dari atas ke bawah disebut dengan scanline vertikal, dimana ditandai dengan simbol Y.

Pada citra digital, index terkecil dari titik X adalah 0 dan index terbesar dari titik X adalah (lebar citra - 1). Index terkecil dari titik Y adalah 0 dan index terbesar dari titik Y adalah (tinggi citra - 1). Tiap kotak memiliki titik koordinat (x,y) disebut dengan pixel. Bila pada citra dengan ukuran 6 x 6, maka index terkecil adalah 0 dan berturut-turut index terbesar pada x dan y adalah 5. Nilai pada kotak ada 2 dimana nilai atas merupakan titik koordinat pixel dan nilai bawah adalah nilai pixel tersebut.

		x					
		0,0	1,0	2,0	3,0	4,0	5,0
		10	20	30	40	50	60
		0,1	1,1	2,1	3,1	4,1	5,1
		15	25	35	45	55	65
		0,2	1,2	2,2	3,2	4,2	5,2
		20	30	40	50	60	70
		0,3	1,3	2,3	3,3	4,3	5,3
		25	35	45	55	65	75
		0,4	1,4	2,4	3,4	4,4	5,4
		30	40	50	60	70	80
		0,5	1,5	2,5	3,5	4,5	5,5
		35	45	55	65	75	85
y							

Berikut adalah matriks dengan ukuran 3 x 3 yang pada umumnya dipakai untuk deteksi tepi. Pemahaman perhitungan perkalian matriks dengan citra digital akan dipaparkan dalam bentuk simbol dan diarahkan ke alur pemrograman. Matriks tersebut disebut juga dengan filter dan tiap kotak akan berisi sebuah angka dengan tipe double (dapat berupa angka non desimal dan angka desimal) yang didapat dari sebuah perhitungan.

<b>A [1]</b>	<b>B[0]</b>	<b>C[-1]</b>
X-1,Y-1	X,Y-1	X+1,Y-1
<b>D[0]</b>	<b>E[0]</b>	<b>F[0]</b>
X-1,Y	X,Y	X+1,Y
<b>G[1]</b>	<b>H[0]</b>	<b>I[-1]</b>
X-1,Y+1	X,Y+1	X+1,Y+1

Konsep perhitungan matriks sebelum diterapkan dengan bahasa pemrograman adalah sebagai berikut :

- 1) Pada tiap kotak pasti memiliki sebuah nilai, sebagaimana disimbolkan diatas dengan huruf dari A - I. Nilai didalam kurung siku adalah nilai yang akan digunakan untuk perhitungan.
- 2) Ukuran matriks pada umumnya bernilai ganjil karena memiliki titik tengah. Pada matriks diatas nilai tengahnya adalah E. Titik selain E adalah titik perluasan tetangga.
- 3) Karena titik E adalah titik pusat matriks maka perluasan tetangga pada citra akan selalu menyesuaikan dengan ukuran matriks.
- 4) Seluruh pixel pada citra akan dilakukan perhitungan perkalian matriks dengan perluasan tetangga sejumlah ukuran matriks.

Berikut tahap perhitungan citra dengan matriks dengan *scanline* secara horizontal dimana pola perulangan yang digunakan adalah *nested looping*. Karena titik pixel didefinisikan dengan titik (x,y) maka perulangan X akan berada di dalam perulangan Y :

```
For ( int y = 0 ; y < tinggi_citra ; y++ ) { For
( int x = 0 ; x < lebar_citra ; x++ ) {
// kode program titik pixel (x,y)
}
}
```

Dimulai dari pixel pada titik (x,y) = (0,0). Ikuti aturan matriks sebelum mendaftarkan pixel tetangga ke dalam suatu list. Perluasan matriks digambarkan sebagai berikut :

<b>A[1]</b>	<b>B[0]</b>	<b>C[-1]</b>
-1,-1	0,-1	1,-1
<b>D[0]</b>	<b>E[0]</b>	<b>F[0]</b>
-1,0	0,0	1,0
<b>G[1]</b>	<b>H[0]</b>	<b>I[-1]</b>
-1,1	0,1	1,1

Seluruh pixel diatas akan didaftarkan dalam sebuah list atau array dan pendaftaran pixel tetangga secara scanline horizontal. Perlu diperhatikan tidak seluruh pixel tetangga yang didaftarkan memiliki koordinat yang valid. Valid disini adalah nilai x dan y harus lebih dari atau sama dengan 0 dan untuk koordinat x harus kurang dari lebar citra sedangkan untuk koordinat y harus kurang dari tinggi citra.

Untuk pixel pada titik (0,0), list tetangga adalah sebagai berikut :

- { (-1,-1), (0,-1), (1,-1), (-1,0), (0,0), (1,0), (-1,1), (0,1), (1,1) }

Seperti digambarkan diatas pada kotak dengan warna abu-abu adalah pixel yang tidak valid karena tidak ada nilai pada pixel dengan titik tersebut.

Sehingga pixel yang tidak valid tersebut di set nilainya sama dengan 0. Sedangkan dari pixel valid diambil nilainya dari citra yang diolah. Sehingga list pixel tetangga yang akan digunakan perhitungan dengan matriks pada titik (0,0) adalah sebagai berikut:

- (-1,-1) : 0 \* 1 = 0
- (0,-1) : 0 \* 0 = 0
- (1,-1) : 0 \* -1 = 0
- (-1,0) : 0 \* 0 = 0
- (0,0) : 10 \* 0 = 0
- (1,0) : 20 \* 0 = 0
- (-1,1) : 0 \* 1 = 0
- (0,1) : 15 \* 0 = 0
- (1,1) : 25 \* -1 = -25

Dari perhitungan tersebut, nilai perkalian dengan matriks akan dijumlahkan dan mendapatkan nilai baru dari pixel di titik (0,0) :

$$\text{pixel}(0,0) = 0 + 0 + 0 + 0 + 0 + 0 + 0 + 0 + (-25) = -25$$

Nilai baru tersebut harus di dalam range 0 - 255 karena merupakan nilai batas pada citra, sehingga bila nilai kurang dari 0 maka nilai tersebut diubah menjadi 0, dan jika nilai tersebut lebih dari 255 maka nilai tersebut diubah menjadi 255.

Sesuai dengan dijelaskan, maka nilai baru dari pixel (0,0) adalah 0. Maka citra akan menjadi seperti gambar di bawah ini. Kolom citra yang telah berubah nilainya diwarnai dengan warna biru. Lakukan langkah yang sama untuk pixel berikutnya tetapi pendaftaran nilai tetangga tetap menggunakan citra awal, bukan citra yang telah diubah.

	x					
	0,0	1,0	2,0	3,0	4,0	5,0
0	20	30	40	50	60	
0,1	1,1	2,1	3,1	4,1	5,1	
15	25	35	45	55	65	
0,2	1,2	2,2	3,2	4,2	4,2	
20	30	40	50	60	70	
0,3	1,3	2,3	3,3	4,3	4,3	
25	35	45	55	65	75	
0,4	1,4	2,4	3,4	4,4	5,4	
30	40	50	60	70	80	
0,5	1,5	2,5	3,5	4,5	5,5	
35	45	55	65	75	85	

Dengan melakukan konvolusi citra Image dengan penapis merata yang berukuran 6 x 6. Hasil pelembutan disimpan di dalam hasil Image. Jika citra hasil dikurangi dari citra semula (yang mengandung derau), maka yang dihasilkan adalah peningkatan relatif komponen citra yang berfrekuensi tinggi tanpa peningkatan komponen derau. Operasi ini mempu-

nyai efek pemerataan derajat keabuan, sehingga gambar yang diperoleh tampak lebih kabur kontrasnya.



(a) (b)

Gambar 2. Pelembutan (smoth) dan penapisan derau (noise) (a) Citra Ady mengalami gangguan noise, (b) Citra Ady setelah penapisan derau dengan pelembutan (smooth)

**B) Pengujian**

Dari hasil pengujian menggunakan model klasifikasi di dasarkan pada pengujian untuk memperkirakan obyek benar dan salah, uraian pengujian di tabulasikan dalam confusion matrix, setiap sel berisi angka yang menunjukkan beberapa kasus yang sebenarnya dari kelas yang di amati untuk mendapatkan tingkat akurasi. Dengan menggunakan 50 buah citra sebagai bahan uji dalam penelitian ini maka di dapat hasil sebagai berikut :

- 1) Jumlah gambar yang bersih dari noise = 38
- 2) Jumlah gambar yang tidak bersih dari noise = 7
- 3) Jumlah yang tidak gambar hilang noise = 2
- 4) Jumlah yang tidak gambar tidak hilang noise = 3

Tabel hasil confusion matrix

	True	False
Positif	38	2
Negatif	7	3

Berdasarkan tabel *confusion matrix* di atas, maka tingkat akurasi di dapat berdasarkan rumuskan sebagai berikut :

$$\begin{aligned}
 hcc &= \frac{38 + 2}{38 + 3 + 7 + 2} \\
 &= \frac{40}{50} \\
 &= 0,8 \times 100\% \\
 &= 80\%
 \end{aligned}$$

Dari tabel hasil pengujian dapat dilihat bahwa noise dapat dihilangkan. Penghitungan tingkat keberhasilan ini berdasarkan nilai yang didapat dari pengujian aplikasi yang berhasil di lembutkan atau di haluskan. Tingkat keberhasilan untuk noise ini masih 80% dan itu merupakan hasil yang memuaskan.



## V. KESIMPULAN

Dari hasil penelitian yang dilakukan maka dapat disimpulkan hal-hal sebagai berikut:

- 1) Pada proses pengurangan noise ukuran kernel sangat mempengaruhi dalam memperoleh hasil kualitas citra.
- 2) Nilai intensitas pixel pada citra baru dari hasil pengurangan noise diperoleh dengan mengganti nilai intensitas citra asal dengan rata-rata nilai pembobotan matriks kernel untuk setiap pixel dan pixel tetangganya dan pixel itu sendiri.
- 3) Dengan menggunakan metode filter Gaussian maka noise yang terdapat pada citra digital sedapat mungkin bisa berkurang.
- 4) Citra yang memiliki *noise* (derau) setelah diperbaiki gambar yang dihasilkan akan terlihat lebih halus dan kabur.

Salt and Pepper Berdasarkan Sifat Ketetanggaan Pixel,

Sri Kusumadewi. 2003. *Teori dan konsep kecerdasan Buatan*. PT. Elex Media Komputindo. Jakarta.

## DAFTAR PUSTAKA

- Ahmad, Usman. *Pengolahan Citra Digital dan teknik pemrogramannya*. Graha Ilmu. Yogyakarta. 2005.
- Darmayuda, Ketut, 2010. "Pemograman Aplikasi Database dengan Microsoft Visual Basic. NET 2008", Penerbit Informatika Bandung,
- Gonzales, Rafael C, Woods, Richard E. 2002, *Digital Image Processing*, Prentice Hall Inc, New Jersey.
- Jogianto H.M. *Analisa dan Disain Sistem Informasi*, Andi Offset, Yokyakarta, 2009
- Kadir, A. & Susanto, A. 2013. *Teori dan Aplikasi Pengolahan Citra*. Yogyakarta: Penerbit ANDI.
- Munir, 2006. *Pengolahan Citra Digital*. Institut Teknologi Bandung. Teknik Informatika
- Nurlailah, Siti. 2010. *Aplikasi Fragile Watermarking untuk Melindungi Keaslian Foto*. Jakarta: UIN.
- Pressman, Roger S. 2002. *Rekayasa Perangkat Lunak. Pendekatan Praktisi (Buku Satu)*. (Yogyakarta: Andi. 2002).
- Sutoyo, T, 2009. "Teori Pengolahan Citra Digital", Penerbit Andi, Yogyakarta,
- Purwiyanti, Sri. & FX. Arinto Setyawan.. 2008. Penentuan Letak Derau pada Citra Berderau