

## PENERAPAN METODE ALGORITMA ARITHMETIC MEAN FILTER UNTUK MEREDUKSI *NOISE SALT AND PEPPER* PADA CITRA

Astriana Mulyani<sup>1</sup>, Gita Apriyanti<sup>2</sup>

<sup>1</sup>Program Studi, Teknik Informatika, STMIK Nusa Mandiri  
Jl. Kramat Raya No. 18, Jakarta Pusat  
[astriana.atm@nusamandiri.ac.id](mailto:astriana.atm@nusamandiri.ac.id)

<sup>2</sup>Program Studi, Teknik Informatika, STMIK Nusa Mandiri  
Jl. Kramat Raya No. 18, Jakarta Pusat  
[ggitariyadi@yahoo.com](mailto:ggitariyadi@yahoo.com)

**Abstract-** *Image recording digital cameras often there are some problems that may occur, such as the lens out of focus, appear spots caused by the capture process is imperfect, uneven lighting resulting intensity is not uniform, image contrast is too low so that objects are difficult to separate background, or disruption caused by dirt attached to the image and so forth. Arithmetic Mean Filter Algorithm Method to Reduce Noise Salt and Pepper in image filtering is one that serves to soften and remove noise in an image that works by replacing the pixel intensity values with the average of the pixel value with the value of neighbor pixels. With this application the algorithm used to produce a program to reduce salt and pepper noise contained in the image, so that the resulting image quality is better and finer than before.*

**Keywords :** *Filter, Image, Noise, Mean, salt, pepper*

**Inti Sari-** Citra hasil rekaman kamera digital sering sekali terdapat beberapa gangguan yang mungkin terjadi, seperti lensa tidak fokus,

muncul bintik-bintik yang disebabkan oleh proses capture yang tidak sempurna, pencahayaan yang tidak merata yang mengakibatkan intensitas tidak seragam, kontras citra terlalu rendah sehingga objek sulit dipisahkan dari latar belakangnya, atau gangguan yang disebabkan oleh kotoran-kotoran yang menempel pada citra dan lain sebagainya. Metode Algoritma *Arithmetic Mean Filter* untuk *Mereduksi Noise Salt and Pepper* pada Citra merupakan salah satu filtering yang berfungsi untuk memperhalus dan menghilangkan noise pada suatu citra yang bekerja dengan menggantikan

intensitas nilai pixel dengan rata-rata dari nilai pixel tersebut dengan nilai pixel-pixel tetangganya. Dengan algoritma yang dipakai aplikasi ini menghasilkan sebuah program yang dapat mereduksi *noise salt and pepper* yang ada pada citra tersebut, sehingga kualitas citra yang dihasilkan lebih baik dan lebih halus dari sebelumnya.

**.Kata kunci :** *Citra, Filter, Noise, Mean, salt, pepper*

### 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.

Pengolahan citra digital merupakan proses yang bertujuan untuk memanipulasi dan menganalisis citra dengan bantuan komputer. "Hal ini dilakukan untuk mendapatkan alternatif solusi sebuah masalah dengan hasil yang lebih efisien dan akurasi yang baik, sebagai contoh untuk deteksi penyakit osteoporosis dari citra X-Rays dan untuk kompresi video" Menurut (Murinto, 2007).

"Pada dasarnya setiap sistem pencitraan dapat menyebabkan terjadinya derau (*noise*) pada citra yang dihasilkan. *Noise* pada citra tersebut pada umumnya terdistribusi secara normal"(Kadir dan Adhi, 2013) . Citra hasil rekaman kamera digital sering sekali terdapat beberapa gangguan yang mungkin

terjadi, seperti lensa tidak fokus, muncul bintik-bintik yang disebabkan oleh proses capture yang tidak sempurna, pencahayaan yang tidak merata yang mengakibatkan intensitas tidak seragam, kontras citra terlalu rendah sehingga objek sulit dipisahkan dari latar belakangnya, atau gangguan yang disebabkan oleh kotoran-kotoran yang menempel pada citra dan lain sebagainya. Meskipun sebuah citra kaya akan informasi, namun sering kali citra yang dimiliki mengalami penurunan mutu, misalnya mengandung cacat atau *noise*. Untuk mengatasi *noise* tersebut perlu dilakukan usaha untuk memperbaiki kualitas citra itu. Salah satunya adalah dengan filtering citra baik secara linear maupun secara non-linear. Mean filter merupakan salah satu filtering linear yang berfungsi untuk memperhalus dan menghilangkan *noise* pada suatu citra yang bekerja dengan menggantikan intensitas nilai pixel dengan rata-rata dari nilai pixel tersebut dengan nilai pixel-pixel tetangganya. Salah satu teknik yang digunakan adalah reduksi *noise* yang melakukan restorasi citra dengan cara peningkatan kualitas. *Noise* yang akan dibahas adalah *noise* yang terjadi karena karakteristik dari derajat keabu-abuan (*gray-level*) atau karena adanya variabel acak yang terjadi karena karakteristik Fungsi Probabilitas Kepadatan (*Probability Density Function* atau *PDF*). Kumpulan *PDF* tersebut sangat berguna untuk memperagakan situasi kerusakan yang diakibatkan oleh *noise*. Teknik reduksi *noise* yang dibahas adalah teknik filter yang melakukan teknik pemisahan *noise* dari objek-objek pada citra. Algoritma yang digunakan pada penelitian reduksi *noise* ini adalah *Arithmetic Mean Filter* (*AMF*), berdasarkan hal yang telah diuraikan di atas maka penulis berniat membuat skripsi dengan judul Analisis Metode Algoritma *Arithmetic Mean Filter* untuk *Mereduksi Noise Salt and Pepper* pada Citra.

#### BAHAN DAN METODE

Suatu citra merupakan fungsi intensitas 2 dimensi  $f(x, y)$ , dimana  $x$  dan  $y$  adalah koordinat spasial dan  $f$  pada titik  $(x, y)$  merupakan tingkat kecerahan suatu citra pada suatu titik. Suatu citra diperoleh dari penangkapan kekuatan sinar yang dipantulkan oleh objek. "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". (Sulistyo, 2009)

Citra Digital adalah citra atau gambar yang dapat didefinisikan sebagai sebuah fungsi dua dimensi  $f(x,y)$  di mana  $x$  dan  $y$  adalah koordinat bidang datar dan harga  $f$  di setiap pasangan koordinat  $(x,y)$  disebut intensitas atau keabuan (*grey level*) dari gambar di titik itu menurut (Hernawati, 2013)

Noise adalah suatu bentuk kerusakan pada image signal yang disebabkan oleh gangguan eksternal. "Gangguan pada citra umumnya berupa variasi intensitas suatu piksel yang tidak berkorelasi dengan piksel-piksel tetangganya" Menurut (Yuwono, 2010).

Dalam penelitian ini menggunakan bahan berupa foto dalam bentuk jpg yang terdapat noise dalam citranya yang nantinya akan dilihat tingkat penurunan noise yang dihasilkan.

#### Metode Pengumpulan Data

1. Observasi  
Pada tahapan ini akan dilakukan uji coba terhadap aplikasi yang telah dibuat dengan melakukan pengujian terhadap permasalahan reduksi *noise* pada citra serta metode yang terbaik untuk mereduksi *noise* Salt and Pepper.
2. Wawancara  
Pada tahap ini dilakukan dengan cara mengadakan kuesioner kepada beberapa orang sebagai tes fungsional program yang di buat penulis.
3. Studi pustaka  
Metode ini dilaksanakan dengan melakukan studi kepustakaan yang relevan serta artikel-artikel atau e-book dan juga journal international yang didapatkan melalui internet.

#### HASIL DAN PEMBAHASAN

Algoritma yang digunakan dalam pembuatan aplikasi tersebut menggunakan algoritma *Arithmetic mean* filter, algoritma dengan metode *mean* merupakan metode yang menitik beratkan pada nilai rata-rata dari jumlah total nilai keseluruhan pixel yang ada disekelilingnya. Dalam metode ini, terdapat alur-alur bagaimana metode

tersebut berjalan. Pada metode *mean*, alur akan dimulai saat aplikasi memilih radio button *mean* dan menekan *button* proses yang maksudnya adalah menghilangkan *noise* dengan menggunakan metode *mean*. Maka dengan menggunakan gambar2 pada *axes2* yaitu citra yang telah dibangkitkan *noise Salt and Pepper*, alur metode *mean* pun dapat dilaksanakan. Berikut adalah algoritma metode *mean*.

Alur berawal dari *input* citra, lalu aplikasi akan menghitung dimensi citra dan di dapat piksel citra, setelah itu citra akan di ubah dari RGB ke dalam *grayscale* citra. Setelah citra diubah ke dalam bentuk *grayscale*, citra akan ditambah atau di bangkitkan *noise*. *Noise* yang dibangkitkan dalam aplikasi ini adalah *noise Salt and Pepper*. Selanjutnya adalah metode *mean* akan mulai bekerja, pertama akan mulai menganalisa tinggi dan lebar dari suatu citra. Setiap pikselnya akan dikunci dan disimpan kedalam memori, kemudian dilihat piksel-piksel tetangga dari piksel yang telah dikunci tersebut. Selanjutnya, piksel-piksel tetangga akan dijumlahkan lalu dibagi dengan banyaknya piksel tetangga yang dijumlahkan. Terakhir, nilai piksel yang terkena *noise* akan dibagi menjadi nilai piksel hasil rata-rata nilai piksel tetangga. Di bawah ini langkah - langkah cara kerja algoritma tersebut:

Langkah 1      *Input* citra ( $f(x,y)$ )

Langkah 2      Mengubah citra berbentuk RGB menjadi citra *grayscale* ( $g(x,y)$ )

Langkah 3      Membangkitkan *noise Salt and Pepper* dengan cara Menentukan  $v=10$

$$f(x,y) = g(x,y)$$

Memberikan *noise Salt and Pepper*  $n(x,y)$  pada  $g(x,y)$

$$g(x,y) = f(x,y) + v * \text{randn}(\text{size}(f(x,y)))$$

*randn* pada pemberian *noise Salt and Pepper* menyatakan bahwa matriks dengan data *random* menggunakan distribusi normal dan hasil dari perhitungan pembangkitan *noise* dengan *Salt and Pepper* akan disimpan pada  $g(x,y)$ .

Langkah 4      Mendekomposisi citra dengan *noise Salt And Pepper* dengan metode *mean*.

Langkah 5      Menampilkan citra yang telah dikurangi *noise* dengan metode *mean* kedalam *axes*.

```
nr_image = imagesc (n(x,y),
'parent',axes3);
```

```
set(axes4,'visible','off');
```

Citra setelah *noise* dihilangkan dengan metode *mean* akan ditampilkan dalam *axes3*.

Langkah 6      Menghitung nilai PSNR pada citra dalam *axes3* dengan potongan kode program sebagai berikut :

```
tmp1=sum(sum(abs(f(x,y)-
(x,y)).^2));
```

```
[M,N]=size(f(x,y));
```

```
RMSE=((1/(M*N))*tmp1)^0.5;
```

```
PSNR=20*log10(255/RMSE
);
```

*Signal to Noise Ratio* untuk membandingkan piksel citra asli dengan nilai piksel citra hasil reduksi *noise*. Hal pertama yang dilakukan adalah deklarasi citra asli dan citra hasil reduksi *noise*. Inisialisasi tinggi dan lebar citra untuk melakukan iterasi pengambilan nilai piksel citra. Lakukan iterasi sebanyak dari tinggi dan lebar citra asli tersebut. Lalu, ambil citra asli dan simpan kedalam variabel gambar1 dan juga ambil citra hasil reduksi *noise* dan simpan kedalam variabel gambar2. Lakukan perhitungan nilai RMSE terlebih dahulu di dalam iterasi. Simpan nilai hasil perhitungan RMSE ke dalam variabel RMSE. Lalu jika semua iterasi telah dilakukan maka hal terakhir adalah menghitung nilai PSNR lalu simpan ke dalam nilai PSNR dan nilai PSNR pun akan ditampilkan. Untuk perhitungan secara keseluruhan nilai PSNR pada matlab adalah sebagai berikut:

```
tmp1=sum(sum(abs(f(x,y) -
g(x,y)).^2));
```

```

[M,N]=size(f(x,y));
RMSE=((1/(M*N))*tmp1)^0.5;
    
```

**1. Testing**

Pengujian black box testing dilakukan penulis untuk mengamati hasil eksekusi input/output dan memeriksa fungsional dari perangkat lunak yang penulis buat.

**A. Pengujian Pengujian Black Box**

Pengujian black box testing dilakukan penulis untuk mengamati hasil eksekusi input/output dan memeriksa fungsional dari perangkat lunak yang penulis buat.

**Tabel 1**  
**Black Box Testing**

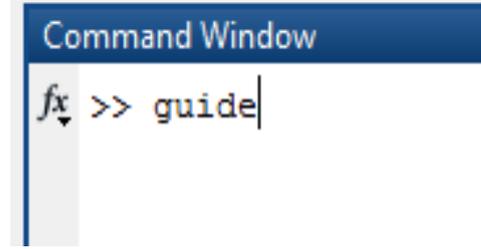
Input	Proses	Output	Hasil
Buka gambar (citra asli)	Pemberian <i>noise</i>	Gambar berubah menjadi cacat/rusak dalam skala rendah dan tinggi	Sesuai
Buka gambar (citra asli)	Pemberian Filter	Gambar yang terkena <i>noise</i> tampak lebih halus	Sesuai
Hitung PNSR	Proses	Hasil Perhitungan	Sesuai

Sumber: Hasil Penelitian (2015)

**2. Implementasi**

Implementasi aplikasi *denoise* ini menggunakan *software* Matlab. Langkah pertama yang dilakukan untuk pembuatan aplikasi ini adalah membuat tampilan halaman awal antarmuka aplikasi dengan pengguna.

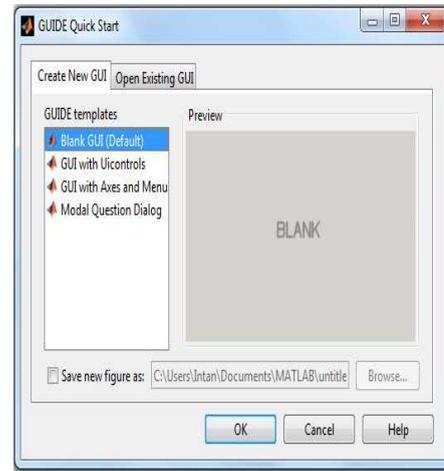
Pada pemrograman Matlab penulis membuat aplikasi dengan *GUI*. untuk masuk kedalam tampilan *GUI* hanya mengetikkan *guide* pada command window.



Sumber: Hasil Penelitian (2015)

**Gambar 3** tampilan command window

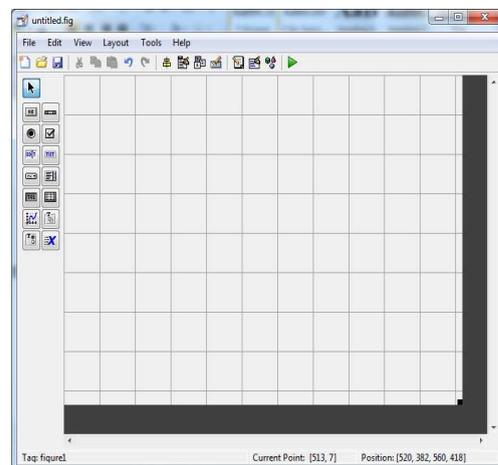
Selanjutnya akan muncul jendela GUIDE Quick Start, tampilan ini untuk memilih tampilan GUI awal yang diinginkan, dan tempat penyimpanan program aplikasi. Tampilannya seperti dibawah ini:



Sumber: Hasil Penelitian (2015)

**Gambar 4** tampilan GUIDE Quick Start

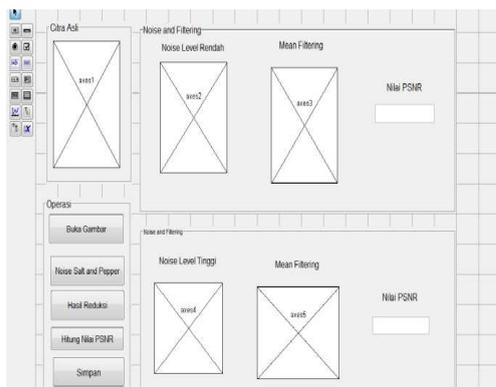
Kemudian klik OK maka akan masuk kedalam tampilan GUI seperti dibawah ini:



Sumber: Hasil Penelitian (2015)

**Gambar 5** tampilan GUI

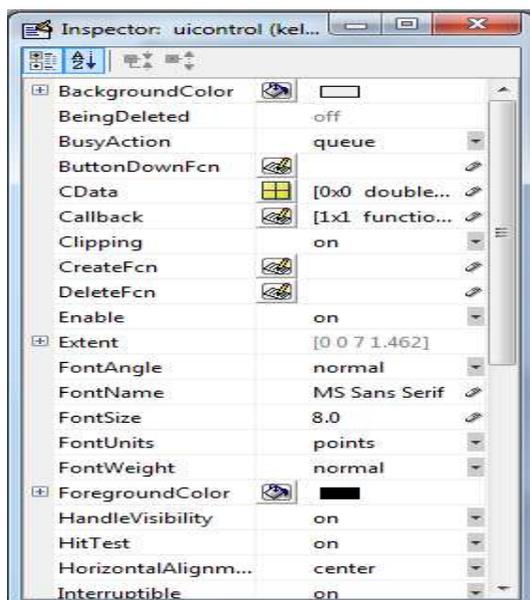
Selanjutnya buat interface sesuai dengan rancangan awal sebelumnya. Pada interface aplikasi ini terdapat 5 axes, 5 pushbutton, 6 text. Seperti gambar dibawah ini:



Sumber: Hasil Penelitian (2015)

**Gambar 6** Rancangan aplikasi interface

Jika interface telah dibuat, tahap selanjutnya adalah pengkodean. Untuk masuk kedalam M-file hanya perlu melakukan klik kanan pada button yang ingin diberi kode. Kemudian pilih view callbacks >> callback. Sedangkan untuk mengganti nama atau tampilan button menggunakan klik kanan dan pilih property inspector. Tampilannya seperti dibawah ini:



Sumber: Hasil Penelitian (2015)

**Gambar 7** tampilan property inspector

Pada bagian ini akan dijelaskan pengkodean program untuk pembuatan aplikasi *denoise* statistika filter. Berikut adalah penggalan pengkodean programnya

## KESIMPULAN

Pada bab ini berisikan tentang kesimpulan dari hasil pembuatan program yang berisikan: Aplikasi "Analisa Metode Algoritma *Arithmetic Mean Filter* Untuk Mereduksi *Noise Salt and Pepper*" dibuat karena citra memegang peranan sangat penting sebagai bentuk informasi visual, dengan seiring perkembangan teknologi dibidang komputerisasi, teknologi pengolahan citra telah banyak dipakai diberbagai bidang termasuk bidang industri. Yang dapat membantu pengerjaan tugas sehingga dapat diperoleh hasil yang efisien dan dengan akurasi yang baik. Dengan algoritma yang dipakai aplikasi ini menghasilkan sebuah program yang dapat mereduksi *noise salt and pepper* yang ada pada citra tersebut, sehingga kualitas citra yang dihasilkan lebih baik dan lebih halus dari sebelumnya. Program ini sangat cocok untuk mereduksi *noise* yaitu dengan Matlab R2013a for windows.

## REFERENSI

- Hernawati, Fajar Astuti. 2013. Pengolahan Citra Digital Konsep dan Teori. Yogyakarta: Andi.
- Kadir, Abdul, Adhi Susanto. 2013. Teori dan Aplikasi Pengolahan Citra. ISSN:978-979-29-3430-4. Yogyakarta. 2013.
- Murinto, Eko Ariwibowo, Risnaldi Syazali. 2007. Analisis Perbandingan Metode *Intensity Filtering* Dengan Metode *Frequency Filtering* Sebagai Reduksi *Noise* Pada Citra Digital. ISSN:1907-5022. Yogyakarta: Seminar Nasional Aplikasi Teknologi Informatika. 16 Juni 2007.
- Sulistyo, Wiwin, Yos Richard Bech dan Filipus Frans. 2009. Analisis Penerapan Metode Median Filter Untuk Mengurangi *Noise* Pada Citra

Digital. Bali: Konferensi Nasional Sistem Dan Informatika, November 2009.

Yuwono, Bambang. 2010. *Image Smoothing Menggunakan Mean Filtering, Modus Filtering dan Gaussian Filtering*. ISSN: 1829-667x. Yogyakarta: Telematika, Vol.1, Juli 2010.

#### BIODATA PENULIS



Astriana Mulyani. Lahir di Jakarta, 11 Agustus 1973. Menempuh S1 pada Fakultas Sains dan Matematika jurusan Matematika program studi Matematika

Industri dan Statistika, S2 Ilmu Komputer Konsentrasi Sistem Informasi Manajemen. Saat ini berkerja sebagai dosen di STMIK Nusa mandiri Jakarta adapun jurnal yang pernah saya hasilkan:

1. Monitoring Traffic Dan Manajemen Bandwidth Jaringan Komputer Pada Badan Sar Nasional Menggunakan Aplikasi PRTG
2. Perancangan Animasi Berkendara Yang Aman Dan Nyaman
3. Perhitungan Komponen Elektronika Dasar Berbasis Android.

