

Analisis Kualitas Citra Hasil Reduksi *Noise* Menggunakan *Spatial Median Filter* dan *Adaptive Fuzzy Filter* Terhadap Variasi Kedalaman Citra

I A Pardosi^{*1}, A A Lubis²

^{1,2}Jurusan Teknik Informatika, STMIK Mikroskil, Medan

E-mail: irpan@mikroskil.ac.id¹, ali.akbar@mikroskil.ac.id²

Masuk: 18 Januari 2019, direvisi: 28 Januari 2019, diterima: 30 Januari 2019

Abstrak. Algoritma reduksi *noise salt* pada citra mampu mengurangi sebagian atau keseluruhan *noise*, tapi berdampak pada keragaman informasi dan kualitas citra. Persentase *noise* yang lebih besar juga membuat perubahan yang besar pada citra, namun hasilnya dapat berbeda untuk citra dengan kedalaman bit yang berbeda-beda. Kemampuan algoritma reduksi *noise* mampu bekerja maksimal untuk *noise* di bawah 20%. Penelitian terdahulu mengenai proses reduksi *noise* citra diantaranya menggunakan algoritma *Adaptive Fuzzy Filter* (AFF) dan *Spatial Median Filter* (SMF) yang mampu menghilangkan *noise*. Keduanya mampu mereduksi *noise* dengan hasil maksimal di bawah 45% pada citra 8 bit, namun menyisakan beberapa *noise*. Oleh karena itu, perlu dikaji kinerja algoritma dan dampaknya terhadap citra dengan *noise* yang lebih besar. Penelitian ini khusus mengatasi *noise* jenis *salt and pepper* dengan persentase *noise* di atas 45% pada citra warna *bitmap*. Selain itu, penelitian ini menganalisis citra hasil mulai dari kualitas citra dan keragaman informasi setelah proses reduksi *noise* dengan menggunakan Algoritma SMF dan AFF. Dari hasil pengujian citra untuk persentase *noise salt* 45%, 55%, 65%, dan 75% pada kedalaman citra 8, 16, dan 24 bit, dapat disimpulkan bahwa Algoritma AFF lebih baik dibandingkan SMF mengacu pada nilai *Peak Signal to Noise Ratio* (PSNR), sebaliknya algoritma SMF lebih baik untuk keragaman informasi, mengacu pada nilai *shannon entropy*. Kedua hal ini berlaku untuk semua variasi kedalaman citra warna.

Kata kunci: analisis kualitas citra; reduksi *noise salt*; *spatial median filter*; *adaptive fuzzy filter*

Abstract. The salt noise reduction algorithm in images can reduce some parts or the entire noise, but it has an impact on information diversity and image quality. The greater percentage of noise is, the larger changes to the image will be. However, the results can be different for images with different bit depths. The noise reduction algorithm can work optimally for noise below 20%. Previous research on image noise reduction processes includes using Adaptive Fuzzy Filter (AFF) and Spatial Median Filter (SMF) algorithms that can eliminate noise. Both are capable of reducing noise with a maximum yield of under 45% in 8-bit images, but still leaving some noise. Therefore, it is necessary to study the performance of the algorithm and its impact on images with greater noise. This study specifically addresses salt and pepper type noise with a percentage of noise above 45% in bitmap color images. In addition, this study analyzes the results of images starting from image quality and information diversity after the noise reduction process using the SMF and AFF Algorithms. From the results of image testing

for salt noise percentage 45%, 55%, 65%, and 75% at 8, 16 and 24 bit image depths, it can be concluded that the AFF algorithm is better than SMF referring to the Peak Signal to Noise Ratio (PSNR) value, on the contrary the SMF algorithm is better for information diversity, referring to shannon entropy value. Both of these apply to all depth variations in color images.

Keywords: images quality analysis; salt noise reduction; spatial median filter; adaptive fuzzy filter

1. Pendahuluan

Proses reduksi *noise*, sebagai bagian dari peningkatan kualitas citra mengakibatkan informasi di dalam citra juga berubah, hal tersebut menjadi salah satu tantangan pada pengolahan citra [1][2]. Reduksi *noise* terhadap citra yang mengalami penurunan kualitas yang diakibatkan adanya derau (*noise*) pada citra, mengakibatkan objek di dalam citra menjadi tidak jelas [3]. *Noise* yang paling umum ditemukan dan diperbaiki pada citra adalah jenis *noise gaussian* dan *impulse noise* yang lebih dikenal dengan *salt and pepper* [4][5]. Perubahan informasi pada citra hasil reduksi berdampak pada kombinasi warna *Red*, *Green*, dan *Blue* yang jika terlalu besar mengakibatkan banyak informasi yang hilang dalam citra. Selain itu, kualitas citra hasil reduksi menjadi hal perlu dipertimbangkan oleh citra dengan *noise* besar (di atas 45%) [3].

Pengukuran perubahan pada citra hasil reduksi akan dibandingkan pada citra *original* saat sebelum dan sesudah proses reduksi dilakukan dengan mengacu nilai *Peak Signal to Noise Ratio* (PSNR) pada citra 8, 16, dan 24 bit. Hal itu dilakukan untuk menganalisis dampak reduksi *noise* terhadap kedalaman citra yang bervariasi, sehingga penggunaan algoritma tepat dengan jenis citra yang digunakan. Beberapa penelitian mengenai proses reduksi *noise* citra diantaranya menggunakan algoritma *Adaptive Fuzzy Filter* (AFF) yang mampu menghilangkan *noise gaussian* dan *salt pepper* [3]. Selain itu juga ada yang membandingkan algoritma *Spatial Median Filter* (SMF) dan *Adaptive Noise Reduction* (ANR) untuk mereduksi *noise salt and pepper* dengan kesimpulan bahwa algoritma *Spatial Median Filter* mereduksi *noise* lebih baik untuk *noise* di atas 20% tapi tidak diuji hingga persentase *noise* maksimalnya, sedangkan *Adaptive Noise Reduction* dapat bekerja lebih maksimal dengan jumlah *noise* dibawah 10% [6][7].

Berdasarkan [3], [6], dan [7], metode yang digunakan untuk penghilangan *noise salt and pepper* belum dianalisis dampak reduksi *noisennya* terhadap perubahan informasi pada citra hasil namun sudah teruji menghilangkan *noise salt*, sehingga diperlukan kajian dan analisis yang lebih mendalam untuk mendapatkan dampak reduksi *noise* terhadap kualitas citra dan keragaman informasi di dalamnya. Secara umum, tahapan penelitian ini dimulai dengan penyiapan data pengujian yang meliputi citra warna *bitmap* (BMP) RGB kombinasi 8, 16, dan 24 bit. Kemudian dilakukan proses penambahan *noise salt* menggunakan aplikasi, dilanjutkan proses reduksi *noise* terhadap masing-masing citra menggunakan kedua algoritma. Selanjutnya akan dilakukan analisis hasil reduksi *noise* ditinjau dari kualitas citra paling baik dan selisih keragaman informasi citra paling kecil antara AFF dan SMF. Analisis dilakukan untuk mendapatkan kombinasi algoritma terbaik sesuai dengan jenis kedalaman citra, yang diukur baik dari segi data (informasi) dan kualitas citra hasil reduksi.

2. Kerangka Teoritis

2.1. Jenis Kedalaman Citra

Suatu piksel memiliki nilai dalam rentang tertentu dari nilai minimum sampai maksimum. Jangkauan yang digunakan berbeda-beda tergantung jenis warnanya. Secara umum jangkauannya adalah 0–255. Citra dengan penggambaran seperti ini digolongkan ke dalam citra *integer*. Citra *grayscale* merupakan citra digital yang hanya memiliki satu kanal pada setiap pikselnya, dengan kata lain nilai bagian *RED* = *GREEN* = *BLUE*. Nilai tersebut digunakan untuk menunjukkan tingkat intensitas. Warna yang dimiliki adalah warna dari hitam, keabuan, dan putih.

1. Citra Warna (8 bit)

Setiap piksel dari Citra Warna (8 bit) hanya diwakili oleh 8 bit dengan jumlah warna maksimum yang dapat digunakan adalah 256 warna.

2. Citra Warna (16 bit)

Biasanya disebut sebagai Citra *Highcolor* dengan setiap pikselnya diwakili oleh 2 *byte* memori (16 bit). Warna 16 bit memiliki 65.536 warna. Dalam formasi bitnya, nilai merah dan biru mengambil tempat 5 bit di kanan dan kiri. Komponen hijau memiliki 5 bit ditambah 1 bit ekstra. Pemilihan komponen hijau dengan deret 6 bit dikarenakan penglihatan manusia lebih sensitif terhadap warna hijau.

3. Citra Warna (24 bit)

Setiap piksel dari citra warna 24 bit diwakili dengan 24 bit sehingga terdapat total 16.777.216 variasi warna. Variasi ini sudah lebih dari cukup untuk memvisualkan seluruh warna yang dapat dilihat penglihatan manusia [8].

2.2. *Spatial Median Filter (SMF)*

Algoritma untuk menentukan *spatial* median dari sekumpulan titik x_1, \dots, x_N dapat dijabarkan sebagai berikut:

1. Untuk setiap vektor x , hitung S yang merupakan sekumpulan nilai penjumlahan dari kedalaman spasial dari x ke setiap vektor lainnya.
2. Tentukan nilai kedalaman spasial maksimum dari kumpulan ini yaitu S_{max} . S_{max} adalah *spatial* median dari kumpulan titik. Kedalaman spasial antara sebuah titik dan kumpulan titik dapat didefinisikan seperti pada persamaan (1) [9].

$$S_{depth} = 1 - \frac{1}{N-1} \left\| \sum_{i=1}^N \frac{X - X_i}{\|X - X_i\|} \right\| \quad (1)$$

2.3. *Adaptive Fuzzy Filter (AFF)*

Teknik *fuzzy* memiliki 4 bagian utama yaitu *fuzzy rank selection*, *fuzzy weighted*, *fuzzy network* (FNN) dan *softswitching*. *FNN Filter*, baik *RCFS filter* maupun *switching fuzzy filter* tidak bisa menyesuaikan diri mereka dengan variasi lingkungan atau kurang mampu untuk beradaptasi. Hal ini dapat ditingkatkan melalui kemampuan menyesuaikan diri *filter* dengan menggunakan teknik jaringan saraf.

2.3.1 *Neuro-Fuzzy Filter*. Jendela operasi X dengan 3×3 , dengan $\Delta x_i = x_p - x_i$ ($i = 1, \dots, 8$), $\Delta x'_j = x_p - x'_j$ ($j = 1, \dots, 16$) antara x_p dan piksel yang berdekatan di X , x_p dan 16 piksel luar yang masing-masing berdekatan dari X . Choi dan Krishnapuram dalam [10] memberikan aturan inferensi *fuzzy* memakai $\{\Delta x_i\}$ dan $\{\Delta x'_j\}$. Tujuan utama *neuro-fuzzy filter* adalah untuk menentukan nilai Δy yang disesuaikan untuk pusat tingkat abu – abu x_p . Jadi, *output* y dari *filter* dihitung sebagai berikut: $y = x_p + \Delta y$. Δy adaptif disesuaikan dengan FNN. Inputnya adalah Δx_i ($i = 1, \dots, 8$), $\Delta x'_j$ ($j = 1, \dots, 16$).

Lapisan *fuzzification* terdiri dari empat bagian, bagian pertama dan ketiga dilengkapi dengan fungsi keanggotaan \tilde{L}_p dari himpunan *fuzzy* “*positively large*”. *Output* yang sesuai didefinisikan seperti pada persamaan (2) dan (3).

$$O_{1_i}^{(2)} = \tilde{L}_p(\Delta x_i) (i = 1, \dots, 8) \quad (2)$$

$$O_{3_j}^{(2)} = \tilde{L}_p(\Delta x'_j) (j = 1, \dots, 16) \quad (3)$$

Pada bagian kedua dan keempat dengan fungsi keanggotaan \tilde{L}_n dari himpunan *fuzzy* “*negatively large*” mempunyai *output* masing–masing seperti pada persamaan (4) dan (5).

$$O_{2_i}^{(2)} = \tilde{L}_n(\Delta x_i) (i = 1, \dots, 8) \quad (4)$$

$$O_{4_j}^{(2)} = \tilde{L}_n(\Delta x'_j) (j = 1, \dots, 16) \quad (5)$$

Dengan demikian, dalam kesimpulan lapisan I, *output* dari bagian- k ditunjukkan pada persamaan (6) dan (7).

$$O_{k_i}^{(3)} = \bigvee_{i'=1}^8 (O_{k_i}^{(2)} \cdot w_{i,i'}^k) (k = 1, 2) \quad (6)$$

$$O_{kj}^{(3)} = \bigvee_{j'=1}^{16} (O_{kj}^{(2)} \cdot w_{j,j'}^k) \quad (k = 3,4) \quad (7)$$

Output dari dua unit di inferensi lapisan II ditunjukkan pada persamaan (8) dan (9).

$$O_1^{(4)} = \left(\bigwedge_{i=1}^8 \{O_{1i}^{(3)}\} \right) \wedge \left(\bigwedge_{j=1}^{16} \{O_{3j}^{(3)}\} \right) \quad (8)$$

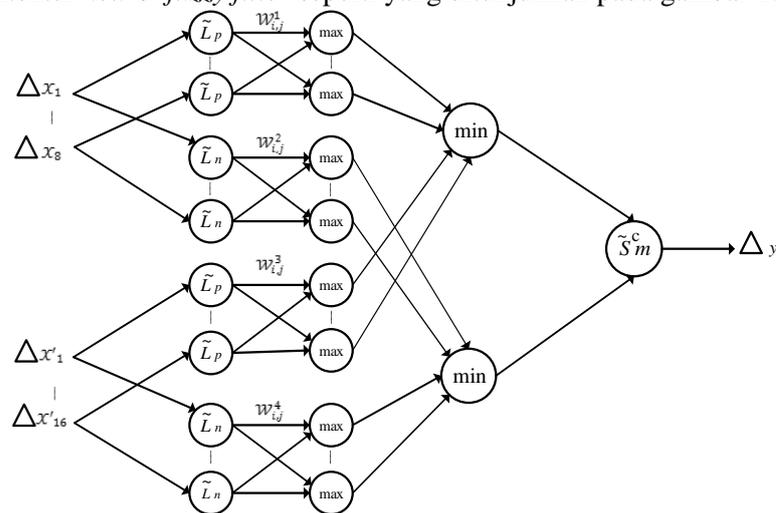
$$O_2^{(4)} = \left(\bigwedge_{i=1}^8 \{O_{2i}^{(3)}\} \right) \wedge \left(\bigwedge_{j=1}^{16} \{O_{4j}^{(3)}\} \right) \quad (9)$$

Output dari FNN sebagai $\Delta y = (L - 1)$.

$$S_m^c \left((L - 1)(O_1^{(4)} - O_2^{(4)}) \right) \cdot \text{sign}(O_1^{(4)} - O_2^{(4)}).$$

Dengan demikian, FNN filter dapat direpresentasikan sebagai $Y_{fn}(X) = x_p + \Delta y$. Dalam penelitian Russo [11], bobot koneksi $w_{i,i}^k$, $w_{j,j}^k$ dan fuzzy set \tilde{S}_m , akibatnya pelengkap ditentukan oleh algoritma genetika. Filter Y_{fn} dapat menghapus intensitas noise yang tinggi. Selanjutnya mungkin juga dapat menyesuaikan dirinya dengan variasi lingkungan. Dapat dijamin kinerja yang baik dari filter pada probabilitas impuls noise yang berbeda. Jika pasangan operator fuzzy (max, min) tidak langsung diberikan, tetapi dipilih secara rasional untuk masalah-masalah seperti pada [12], kita dapat meningkatkan kinerja Y_{fn} lebih lanjut. Namun, dalam aplikasi kinerja penyaringan neuro-fuzzy filter akan menjadi sangat buruk jika noise tidak impuls, tetapi campuran beberapa noise, seperti impuls dan Gaussian noise.

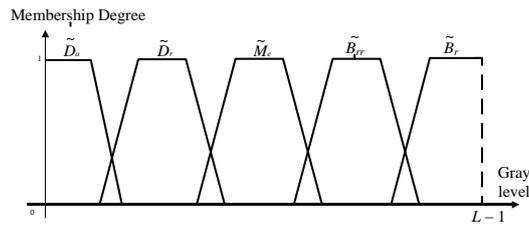
Selanjutnya banyak masalah tentang FNN yang terkait dengan neuro-fuzzy filter yang harus diselesaikan. Permasalahan tersebut antara lain pemilihan pasang operator fuzzy (max atau min) untuk meningkatkan kinerja filter, bagaimana mencari \tilde{L}_n, \tilde{L}_p dan \tilde{S}_m , bagaimana mengembangkan aturan fuzzy yang lebih rasional untuk menentukan arsitektur FNN dan lain sebagainya. Hal-hal tersebut berkaitan dengan arsitektur neuro-fuzzy filter seperti yang ditunjukkan pada gambar 1.



Input layer, fuzzification layer, inference layer I, inference layer II, output layer

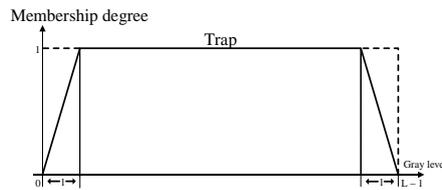
Gambar 1. Arsitektur neuro-fuzzy filter [10]

2.3.2 Selection Type FNN Filter. Asumsikan bahwa nilai-nilai tingkat abu-abu untuk citra dua dimensi $F = \{F(t_1, t_2), 0 \leq t_1, t_2 \leq N\}$ milik $[0, L - 1]$, yaitu $\forall (t_1, t_2) \in \{0, 1, \dots, N\}^2, F(t_1, t_2) \in [0, L - 1]$. Memperkenalkan partisi fuzzy $[0, L - 1]$ untuk mendeskripsikan citra F dengan fuzzy sets $\tilde{G}_1, \dots, \tilde{G}_{k_0}$. Sebagai contoh, fuzzy sets masing-masing menjadi “dark(\tilde{D}_a)”, “darker(\tilde{D}_r)”, “medium(\tilde{M}_e)”, “brighter(\tilde{B}_{rr})”, “bright(\tilde{B}_r)”, yang ditunjukkan pada gambar 2.



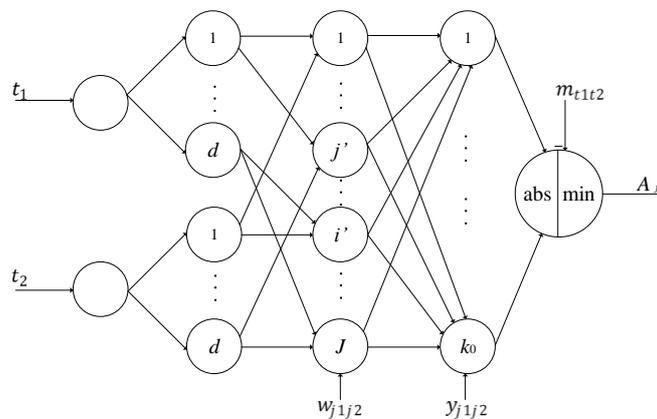
Gambar 2. Tingkat abu – abu fuzzy sets[10]

Dalam rangka untuk menekan impuls *noise*, diperkenalkan *Mean fuzzy set Trap*, yang bentuknya dapat diwakili sebagai bilangan fuzzy trapesium seperti pada gambar 3 dengan tingkat abu-abu fuzzy sets dan *mean fuzzy set selection type FNN* dapat dikembangkan.



Gambar 3. Mean Fuzzy Set[10]

Tetapkan jendela operasi $X = \{x_1, \dots, x_p, \dots, x_d\}$, dimana x_p adalah tingkat abu-abu *pixel* pusat yaitu $x_p = F(t_1, t_2)$. Struktur topologi *selection type FNN* ditunjukkan pada gambar 4, dan hubungan I/O dinyatakan seperti pada persamaan (10).



Gambar 4. Selection type FNN

$$m^k(X) = \frac{\sum_{j=1}^d X_j \tilde{G}_k(X_j)}{\sum_{j=1}^d \tilde{G}_k(X_j)}, m(X) = \frac{\sum_{j=1}^d X_j Trap(X_j)}{\sum_{j=1}^d Trap(X_j)} \quad (10)$$

$$A_f = M^{K'}(X), \text{ di mana}$$

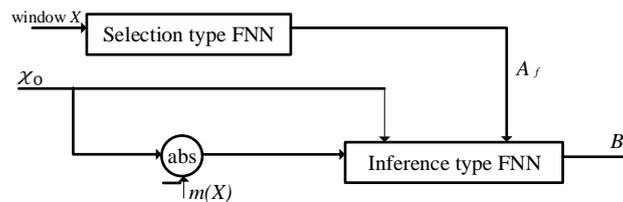
$$K' = \max\{k1: |m(X) - m^{k1}(X)| = \min\{|m(X) - m^k(X)|\}\}$$

Jika F adalah citra bebas dari *noise*, nilai-nilai tingkat abu – abu $x_1, \dots, x_p, \dots, x_d$ di jendela X yaitu $x_1 \approx \dots \approx x_p \approx \dots \approx x_d \approx m(X)$. Jadi jika u adalah variabel *output*, dan $v = |x_p - m(X)|$ adalah variabel *input*, maka kita dapat memperoleh aturan inferensi *Mamdani* sebagai berikut [13]: JIKA v adalah “*small*” MAKA v adalah \tilde{x}_p ; JIKA v adalah “*large*” MAKA v adalah \tilde{A}_f . Di mana \tilde{x}_p, \tilde{A}_f masing-masing berarti *fuzzifications* dari x_p, A_f . dan $x_p \in \ker(\tilde{x}_p), A_f \in \ker(\tilde{A}_f)$. Pada himpunan tingkat abu-abu $[0, L - 1]$ mendefinisikan $\tilde{S} = \text{“small”}$, $\tilde{L} = \text{“large”}$.

$$B_f = \frac{x_p \tilde{S}(v) + A_f \tilde{L}(v)}{\tilde{S}(v) + \tilde{L}(v)} \quad (11)$$

$$= \frac{x_p \tilde{S}(|x_p - m(X)|) + A_f \tilde{L}(|x_p - m(X)|)}{\tilde{S}(|x_p - m(X)|) + \tilde{L}(|x_p - m(X)|)}$$

Persamaan (10) dan (11) membentuk hubungan I/O dari *selection type FNN filter*, yang strukturnya ditampilkan pada gambar 5.



Gambar 5. Selection type FNN filter

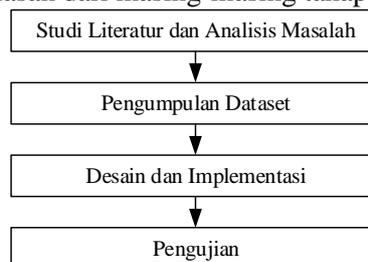
Untuk mencari *selection type FNN filter* yang optimal, sudah cukup untuk merancang algoritma pembelajaran untuk masing-masing parameter dari *selection type FNN* dan *inference type FNN*. Algoritma pembelajaran untuk *selection type FNN* bertujuan untuk menentukan partisi dari himpunan tingkat abu-abu $[0, L - 1]$, yaitu menentukan nilai k_0 , bilangan fuzzy $\tilde{G}_1, \dots, \tilde{G}_{k_0}$ di partisi fuzzy. Algoritmanya dapat dijabarkan sebagai berikut:

1. Untuk n -bit citra digital, variasi yang tidak melebihi 2^{n-4} di tingkat abu-abu citra tidak akan menyebabkan perubahan visual yang jelas [14]. Jadi partisi $[0, L - 1]$ menjadi 2^{n-4} bagian-bagian yang identik.
2. Carilah daerah konsentrasi tinggi dari tingkat abu-abu citra F : Hitung jumlah Γ_k dari F milik $I_k \triangleq [(k - 1)L / (2^{n-4}), (kL / (2^{n-4})) (k = 1, \dots, 2^{n-4})]$ lalu membandingkan apakah $\Gamma_k \geq \eta$? Jika ya, I_k disebut daerah konsentrasi tingkat abu-abu F , di mana η adalah konstan yang diberikan sesuai dengan citra F .
3. Tentukan partisi fuzzy $[0, L - 1]$ menjadi sejumlah daerah konsentrasi tingkat abu-abu $I_{i_1}, \dots, I_{i_{k_0}}$. Sehingga, bilangan fuzzy $\tilde{G}_1, \dots, \tilde{G}_{k_0}$ dapat dibangun.

Dalam definisi *mean fuzzy set* untuk *selection type FNN*, parameter l diasumsikan sekitar 3, dan seleksi nilai standar adalah $m(X)$, kita bisa mendapatkan hasil yang baik untuk menghilangkan impuls *noise*. Berdasarkan algoritma pembelajaran untuk *selection type FNN*, dengan kembali menyebarkan algoritma pembelajaran tentang s_1, s_2 , inferensi FNN dapat dibangun. Dengan demikian, *selection type FNN filter* dapat ditentukan. Pada [15] untuk mendapatkan citra restorasi dengan kualitas tinggi dapat dilakukan dengan *selection type FNN filter* dari citra *noise* terdegradasi oleh probabilitas impuls *noise* yang tinggi atau rendah. Selanjutnya, jika *noise* adalah *hybrid*, kinerja penyaringan yang baik dari *filter* dapat juga dipastikan karena sistem fuzzy umum dapat dilakukan dengan beberapa FNN [13][16].

3. Metodologi Penelitian

Metodologi penelitian yang digunakan dalam penelitian ini digambarkan pada *Diagram Flow* pada gambar 6. Berikut ini adalah penjelasan dari masing-masing tahap.



Gambar 6. Diagram flow metodologi penelitian

3.1. Studi Literatur dan Analisis Masalah

Pada tahapan awal dilakukan analisis terhadap beberapa literatur penelitian sejenis yang sudah pernah ada sebelumnya berupa jurnal termasuk untuk menentukan jurnal utama yang akan dijadikan acuan dalam rangka mendapatkan dan menentukan latar belakang sesuai dengan masalah utama yang akan diselesaikan dalam penelitian.

3.2. Pengumpulan Dataset

Tahapan pengumpulan *dataset* dilakukan setelah masalah utama sudah ditentukan untuk menghilangkan *noise salt* pada citra warna dengan berbagai kedalaman citra. *Dataset* yang tepat untuk digunakan adalah citra warna BMP karena memiliki kedalaman 8, 16, dan 24 bit. Jumlah citra yang akan diuji sebanyak 5 buah dan masing-masing citra akan ditambahkan *noise salt* mulai dari 45%, 55%, 65%, dan 75%.

3.3. Desain dan Implementasi

Tahap desain merupakan tahap untuk merancang pemodelan sistem dan tampilan aplikasi sesuai dengan kebutuhan dan kegunaannya masing-masing. Hasil dari tahap ini akan dijadikan dasar dan fondasi saat implementasi penelitian. Tahap implementasi merupakan tahapan untuk merealisasikan kedua metode ke dalam aplikasi menggunakan pemrograman C# dan menyediakan setiap fungsi sesuai kebutuhan yang sudah dirancang sebelumnya.

3.4. Pengujian

Merupakan tahapan untuk menguji penyelesaian setiap masalah yang sudah disampaikan di awal dan mengumpulkan data hasil pengujian yang akan dijadikan data untuk dianalisis sebelum dijadikan kesimpulan dari penelitian. Tahapan ini dibagi lagi menjadi beberapa bagian, antara lain:

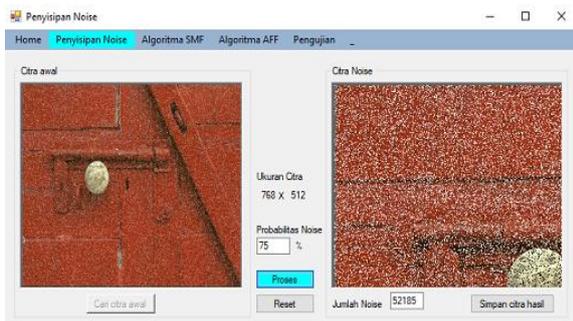
3.4.1. Penambahan Noise. Penambahan *noise salt* dilakukan pada citra *bitmap* untuk masing-masing kedalaman citra yaitu 8, 16, dan 24 bit sesuai dengan persentase yang dibutuhkan [7]. Hal ini dilakukan karena minimnya dataset yang didapatkan sesuai dengan persentase yang akan diuji.

3.4.2. Pengujian Reduksi Noise. Pengujian penelitian ini dilakukan menggunakan Algoritma SMF dan AFF untuk mengurangi *noise* pada citra *bitmap*. Selain itu, pengujian juga mengacu pada nilai PSNR untuk mengukur kualitas citra dan *Shannon Entropy* untuk mengukur selisih keragaman informasi pada citra hasil. Setelah melalui tahap pengolahan data awal dan menjadikan data sesuai dengan persentase *noise* yang akan diuji, maka dataset tersebut telah siap untuk dijalankan pada aplikasi C# yang telah dikembangkan. Tahapan terakhir adalah penarikan kesimpulan dari hasil pengujian yang sudah dianalisis.

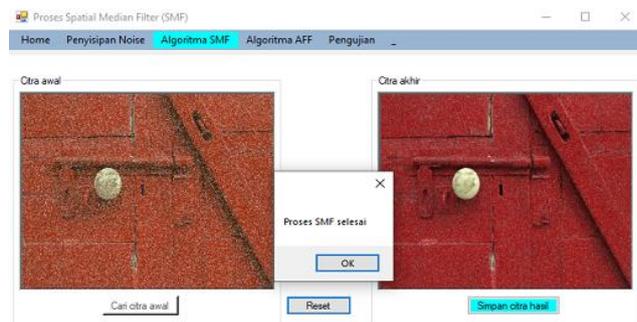
4. Hasil dan Pembahasan

4.1. Hasil

Hasil implementasi dari kedua algoritma ke dalam aplikasi sesuai dengan rancangan yang sudah ditetapkan di awal terlihat pada gambar 7 dan 8. Pada gambar 7 dapat dilihat proses untuk penambahan *noise* pada citra BMP sesuai besaran persentase yang dibutuhkan (45%, 55%, 65%, dan 75%). Proses ini akan dilakukan untuk lima buah citra warna 8, 16, dan 24 bit, sedangkan pada gambar 8 merupakan tampilan proses untuk menghilangkan *noise* menggunakan Algoritma *Spatial Median Filter* (SMF) di mana setiap persentase *noise* akan direduksi dengan cara ini.

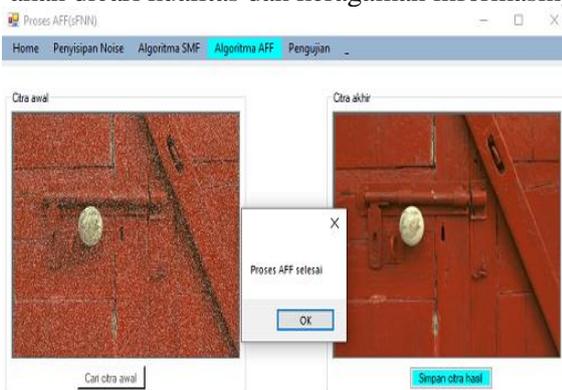


Gambar 7. Proses penambahan *noise* pada citra

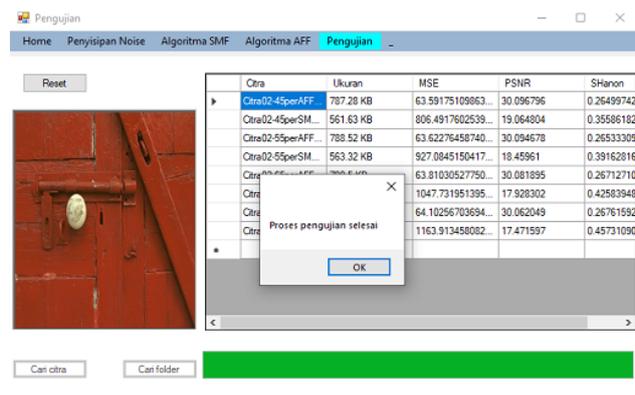


Gambar 8. Proses reduksi *noise* dengan SMF

Pada gambar 9 dapat dilihat proses menghilangkan *noise* dengan Algoritma *Adaptive Fuzzy Filter* (AFF) yang hasilnya terlihat sekilas tidak jauh berbeda dengan SMF, sedangkan pada gambar 10 dapat dilihat cara untuk mengetahui kualitas citra dan keragaman informasi dari citra hasil berdasarkan nilai MSE, PSNR dan *Shannon Entropy*. Prosesnya dimulai dengan menentukan citra asli *original* (tanpa *noise*) sesuai kedalaman citra yang mau diuji, kemudian memilih lokasi folder dari citra hasil yang akan dicari kualitas dan keragaman informasinya.



Gambar 9. Proses reduksi *noise* dengan AFF

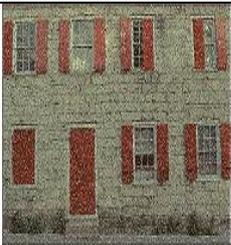


Gambar 10. Proses pengujian citra hasil

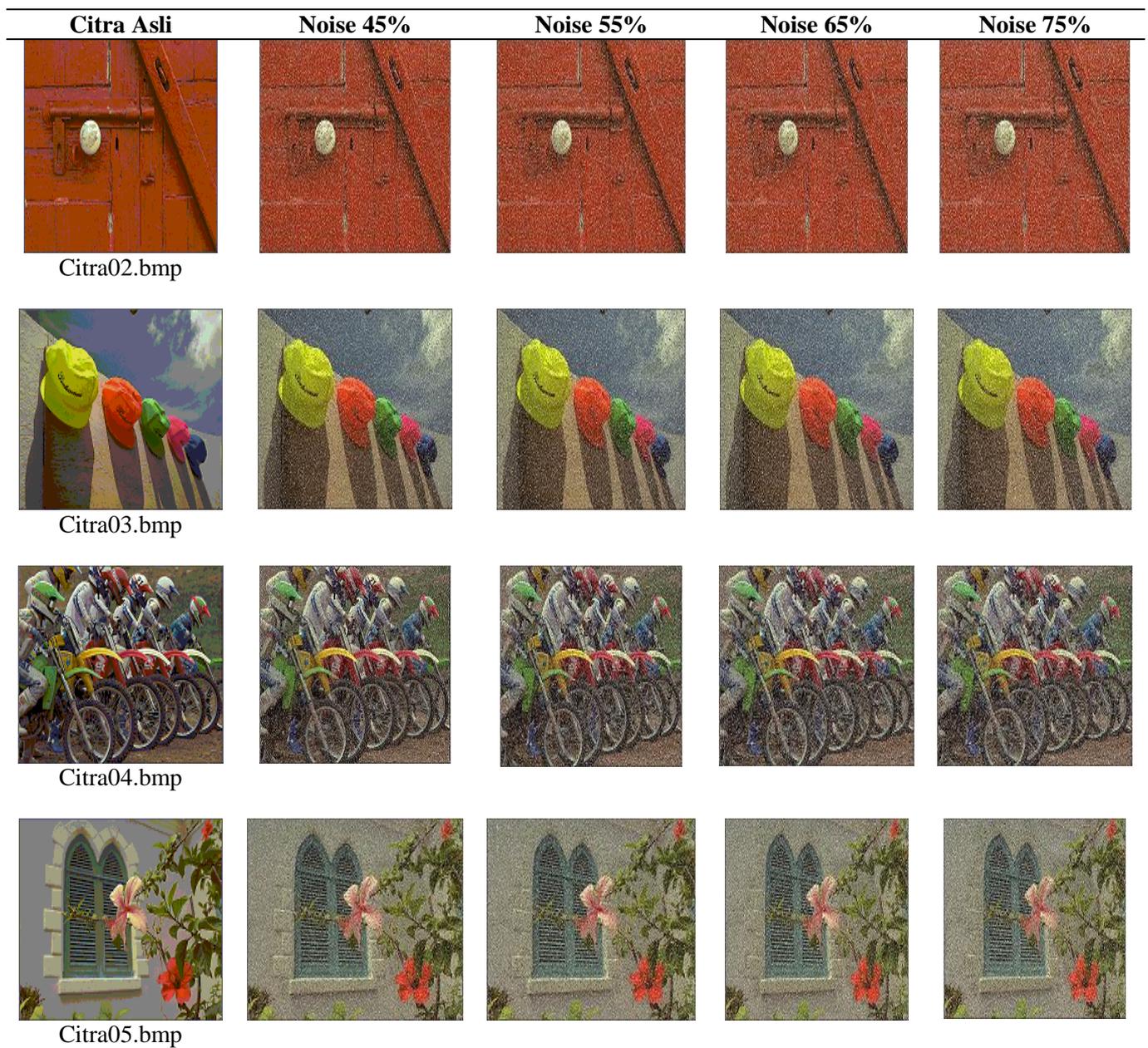
4.2. Pembahasan

Proses pengujian dilakukan terhadap lima citra warna BMP 8, 16, dan 24 bit yang masing-masing ditambahkan *noise* dengan persentase 45%, 55%, 65%, dan 75% seperti terlihat pada tabel 1. Setelah melalui tahapan penambahan *noise*, setiap citra akan direduksi *noise*-nya menggunakan aplikasi.

Tabel 1. Dataset citra BMP warna yang digunakan dalam penelitian

Citra Asli	Noise 45%	Noise 55%	Noise 65%	Noise 75%
				

Citra01.bmp



4.2.1 *Pengujian Citra Warna 24 Bit*. Hasil pengujian citra warna 24 bit ditunjukkan pada tabel 2. Pada pengujian tersebut dapat dilihat untuk kualitas citra berdasarkan nilai PSNR bahwa Algoritma AFF lebih baik dari SMF dengan rata-rata hasil terbaik 28.10 dB pada persentase *noise* 45%. Untuk pengujian keragaman informasi citra pada citra warna 24 bit diperoleh hasil seperti pada tabel 3. Hasil pengujian keragaman informasi menunjukkan bahwa algoritma SMF lebih baik dari AFF dengan selisih nilai perubahan informasi paling kecil 0.08 pada persentase *noise* 45%.

Tabel 2. Citra hasil reduksi *noise* citra warna BMP 24 bit

Citra	Persentase Noise 45%		Persentase Noise 55%		Persentase Noise 65%		Persentase Noise 75%	
	AFF-PSNR	SMF-PSNR	AFF-PSNR	SMF-PSNR	AFF-PSNR	SMF-PSNR	AFF-PSNR	SMF-PSNR
Citra01.bmp	24.79	23.362	24.775	22.985	24.763	22.698	24.742	22.3917

Citra	Persentase Noise 45%		Persentase Noise 55%		Persentase Noise 65%		Persentase Noise 75%	
	AFF- PSNR	SMF- PSNR	AFF- PSNR	SMF- PSNR	AFF- PSNR	SMF- PSNR	AFF- PSNR	SMF- PSNR
	Citra02.bmp	30.452	19.092	30.437	18.4735	30.429	17.946	30.378
Citra03.bmp	30.038	24.129	29.886	23.589	29.732	23.07	29.547	22.6901
Citra04.bmp	24.683	23.275	13.06	24.6004	24.591	22.592	24.518	22.2757
Citra05.bmp	30.573	26.544	30.555	26.0772	30.52	25.576	30.482	25.048
Rata-rata	28.107	23.281	25.743	23.145	28.007	22.376	27.934	21.979

Tabel 3. Keragaman informasi citra warna BMP 24 bit

Citra	Persentase Noise 45%		Persentase Noise 55%		Persentase Noise 65%		Persentase Noise 75%	
	AFF- Shannon	SMF- Shannon	AFF- Shannon	SMF- Shannon	AFF- Shannon	SMF- Shannon	AFF- Shannon	SMF- Shannon
	Citra01.bmp	0.209	0.136	0.209	0.142	0.207	0.142	0.207
Citra02.bmp	0.161	0.174	0.16	0.214	0.16	0.252	0.159	0.287
Citra03.bmp	0.02	0.09	0.02	0.1	0.019	0.109	0.02	0.118
Citra04.bmp	0.104	0.025	0.222	0.103	0.103	0.016	0.103	0.013
Citra05.bmp	0.041	0.019	0.041	0.025	0.041	0.028	0.041	0.033
Rata-rata	0.107	0.089	0.13	0.117	0.106	0.11	0.106	0.12

4.2.2 Pengujian Citra Warna 16 Bit. Hasil pengujian citra warna 16 bit ditunjukkan pada tabel 4. Dari pengujian tersebut terlihat bahwa kualitas citra dengan Algoritma AFF lebih baik dari SMF dengan rata-rata hasil terbaik 27.27 dB pada persentase *noise* 55%. Hasil pengujian keragaman informasi citra pada tabel 5 menunjukkan bahwa algoritma SMF lebih baik dari AFF dengan selisih nilai perubahan informasi paling kecil 1.26 pada persentase *noise* 45%.

Tabel 4. Citra hasil reduksi *noise* citra warna BMP 16 bit

Citra	Persentase Noise 45%		Persentase Noise 55%		Persentase Noise 65%		Persentase Noise 75%	
	AFF- PSNR	SMF- PSNR	AFF- PSNR	SMF- PSNR	AFF- PSNR	SMF- PSNR	AFF- PSNR	SMF- PSNR
	Citra01.bmp	24.505	23.028	24.481	22.7044	24.469	22.393	24.436
Citra02.bmp	29.843	18.708	29.809	18.119	29.768	17.595	29.66	17.1302
Citra03.bmp	27.113	21.748	28.077	23.215	27.853	22.751	27.627	22.3353
Citra04.bmp	24.247	22.934	24.185	22.5897	24.159	22.258	24.077	21.9485
Citra05.bmp	29.927	26.163	29.837	25.6128	29.753	25.086	29.712	24.6345
Rata-rata	27.127	22.516	27.278	22.4482	27.2	22.017	27.102	21.6185

Tabel 5. Keragaman informasi citra warna BMP 16 bit

Citra	Persentase Noise 45%		Persentase Noise 55%		Persentase Noise 65%		Persentase Noise 75%	
	AFF- Shannon	SMF- Shannon	AFF- Shannon	SMF- Shannon	AFF- Shannon	SMF- Shannon	AFF- Shannon	SMF- Shannon
	Cit01.bmp	2.439	1.362	2.44	1.353	2.44	1.349	2.44

Pardosi, Lubis (Analisis Kualitas Citra Hasil Reduksi Noise Menggunakan Spatial Median Filter dan Adaptive Fuzzy Filter Terhadap Variasi Kedalaman Citra)

Citra	Persentase Noise 45%		Persentase Noise 55%		Persentase Noise 65%		Persentase Noise 75%	
	AFF-Shannon	SMF-Shannon	AFF-Shannon	SMF-Shannon	AFF-Shannon	SMF-Shannon	AFF-Shannon	SMF-Shannon
Cit02.bmp	2.402	1.013	2.403	1.041	2.402	1.063	2.402	1.082
Cit03.bmp	2.373	1.243	2.382	1.224	2.38	1.228	2.377	1.237
Cit04.bmp	2.532	1.559	2.531	1.552	2.531	1.548	2.531	1.544
Cit05.bmp	2.456	1.175	2.454	1.178	2.452	1.177	2.45	1.181
Rata-rata	2.44	1.27	2.442	1.27	2.441	1.273	2.44	1.278

4.2.3 Pengujian Citra Warna 8 Bit. Hasil pengujian citra warna 8 bit pada tabel 6 menunjukkan bahwa Algoritma AFF lebih baik dari SMF dengan rata-rata hasil terbaik 27.65 dB pada persentase noise 45%. Dari keragaman informasi citra pada tabel 7 dapat diketahui bahwa Algoritma SMF lebih baik dari AFF dengan selisih nilai perubahan informasi paling kecil 0.27 pada persentase noise 45%.

Tabel 6. Citra hasil reduksi noise citra warna BMP 8 bit

Citra	Persentase Noise 45%		Persentase Noise 55%		Persentase Noise 65%		Persentase Noise 75%	
	AFF-PSNR	SMF-PSNR	AFF-PSNR	SMF-PSNR	AFF-PSNR	SMF-PSNR	AFF-PSNR	SMF-PSNR
Citra01.bmp	24.757	23.165	24.741	22.814	24.708	22.519	24.692	22.185
Citra02.bmp	30.096	19.064	30.094	18.459	30.081	17.928	30.062	17.471
Citra03.bmp	29.177	23.72	29.087	23.256	29.066	22.797	29.013	22.319
Citra04.bmp	24.86	22.918	24.834	22.569	24.799	22.232	24.793	21.962
Citra05.bmp	29.389	26.044	29.366	25.45	29.357	25.119	29.319	24.66
Rata-rata	27.656	22.982	27.624	22.509	27.602	22.119	27.576	21.719

Tabel 7. Keragaman informasi citra warna BMP 8 bit

Citra	Persentase Noise 45%		Persentase Noise 55%		Persentase Noise 65%		Persentase Noise 75%	
	AFF-Shannon	SMF-Shannon	AFF-Shannon	SMF-Shannon	AFF-Shannon	SMF-Shannon	AFF-Shannon	SMF-Shannon
Citra01.bmp	0.4536	0.0969	0.4543	0.0948	0.4546	0.0903	0.4548	0.0872
Citra02.bmp	0.265	0.3559	0.2653	0.3916	0.2671	0.4258	0.2676	0.4573
Citra03.bmp	0.5782	0.2725	0.5783	0.2716	0.5783	0.2773	0.5784	0.2949
Citra04.bmp	0.7059	0.3756	0.7064	0.374	0.7061	0.3768	0.7069	0.3788
Citra05.bmp	0.5872	0.2807	0.5876	0.2831	0.5875	0.2859	0.587	0.2904
Rata-rata	0.518	0.2763	0.5184	0.283	0.5187	0.2912	0.5189	0.3017

Kualitas citra hasil paling baik didapatkan pada citra warna di kedalaman 24 bit dengan nilai rata-rata PSNR 28.10 dB, pada persentase noise 45%, dengan Algoritma AFF. Selisih kehilangan keragaman informasi paling kecil terjadi pada citra warna 24 bit, dengan nilai 0.08 pada persentase noise 45%, dengan Algoritma SMF.

5. Kesimpulan

Berdasarkan hasil pengujian terhadap kedua metode reduksi *noise* terkait dengan kualitas citra hasil dan keragaman informasi citra hasil diperoleh kesimpulan:

1. Hasil paling baik untuk kualitas citra dan keragaman informasi didapatkan pada citra warna dengan kedalaman 24 bit.
2. Hasil rata-rata kualitas citra hasil reduksi *noise* dengan Algoritma AFF memiliki kualitas lebih baik dibandingkan algoritma SMF dengan nilai rata-rata PSNR paling baik 28.10 dB pada persentase *noise* 45% di kedalaman citra warna 24 bit.
3. Perubahan keragaman informasi dengan Algoritma SMF lebih kecil dibandingkan dengan AFF dengan hasil selisih nilai *shannon entropy* terbaik 0.083 pada persentase *noise* 45% di kedalaman citra *grayscale* 24 bit.
4. Perubahan citra hasil dengan Algoritma AFF memiliki dampak perubahan yang paling besar untuk semua warna RGB di semua jenis kedalaman citra hasil.

6. Referensi

- [1] J. R. Tang and N. A. M. Isa, "An Adaptive Fuzzy Contrast Enhancement Algorithm with Details Preserving," *J. ICT Res. Appl.*, vol. 8, no. 2, pp. 126–140, Dec. 2014.
- [2] A. K. Gupta, S. S. Chauhan, and M. Shrivastava, "Low Contrast Image Enhancement Technique By Using Fuzzy Method," *Int. J. Eng. Res. Gen. Sci.*, vol. 4, no. 2, pp. 518–526, 2016.
- [3] M. Pitchammal, S. S. Nisha, and M. M. Sathik, "Noise Reduction in MRI Neck Image Using Adaptive Fuzzy Filter in Contourlet Transform," *Int. J. Eng. Sci. Comput.*, vol. 6, no. 3, pp. 2478–2484, 2016.
- [4] T. Gebreyohannes and D.-Y. Kim, "Adaptive Noise Reduction Scheme for Salt and Pepper," *Signal Image Process. An Int. J.*, vol. 2, no. 4, 2012.
- [5] E. J. Leavline and D. A. A. G. Singh, "Salt and Pepper Noise Detection and Removal in Gray Scale Images: An Experimental Analysis," *Int. J. Signal Process. Image Process. Pattern Recognit.*, vol. 6, no. 5, pp. 343–352, 2013.
- [6] L. Ehsan, "An Adaptive Fuzzy Filter for Gaussian Noise Reduction using Image Histogram Estimation," *Adv. Digit. Multimed.*, vol. 1, no. 4, pp. 190–193, 2013.
- [7] I. Irviantina, Syanti; Pardosi, "Salt and Pepper Noise Removal dengan Spatial Median Filter dan Adaptive Noise Reduction," *JSM (Jurnal SIFO Mikroskil)*, vol. 17, no. 2, pp. 127–136, 2016.
- [8] D. Putra, *Pengolahan Citra Digital*. Yogyakarta: Andi, 2010.
- [9] J. C. Church, Y. Chen, and S. V. Rice, "A Spatial Median Filter for noise removal in digital images," in *Conference Proceedings - IEEE SOUTHEASTCON*, 2008, pp. 618–623.
- [10] Y. S. Choi and R. Krishnapuram, "A robust approach to image enhancement based on fuzzy logic," *IEEE Trans. Image Process.*, vol. 6, no. 6, pp. 808–825, 1997.
- [11] F. Russo, "Hybrid neuro-fuzzy filter for impulse noise removal," *Pattern Recognit.*, vol. 32, no. 11, pp. 1843–1855, 1999.
- [12] F. Russo and G. Ramponi, "Nonlinear fuzzy operators for image processing," *Signal Processing*, vol. 38, no. 3, pp. 429–440, 1994.
- [13] P. Liu and H. Li, "Analyses for $L_p(\mu)$ -norm approximation capability of generalized Mamdani fuzzy systems," *Inf. Sci. (Ny)*, vol. 138, no. 1–4, pp. 195–210, 2001.
- [14] Y. H. Kuo, C. S. Lee, and C. L. Chen, "High-stability AWFM filter for signal restoration and its hardware design," *Fuzzy Sets Syst.*, vol. 114, no. 2, pp. 185–202, 2000.
- [15] P. Liu, "Analysis of approximation of continuous fuzzy functions by multivariate fuzzy polynomials," *Fuzzy Sets Syst.*, vol. 127, no. 3, pp. 299–313, 2002.
- [16] P. & L. X. Liu, "Image restoration techniques based on fuzzy neural networks," *Sci. China Inf. Sci.*, vol. 45, no. 4, 2002.