

KINERJA PENAPISAN GAUSSIAN DAN MEDIAN DALAM PELEMBUTAN CITRA

Gogor C. Setyawan¹, Maya Putri Nawansari²

¹Informatika, Fiskom, UKRIM, ²Informatika, Fiskom, UKRIM

masgogor@ukrimuniversity.ac.id, mavaputri1309@gmail.com

Abstract – The Gaussian filter method and median filter are two of several methods in image smoothing. Both methods are used to improve image quality. The effect of image softening is that the image becomes blurry. The Gaussian Filter is a blur filter that places significant transition colors in an image, then creates a mid-color to create a soft effect on the edges of an image. For this Median Screener, the data used to calculate the Median consists of an odd set of data. This is due to the odd amount of data expected the pixels to be in the middle. In the Median Filter, the $N \times N$ dimension matrix is used. From the matrix, then the existing data is sorted and entered into a matrix measuring $1 \times (N \times N)$. This is useful for finding the median of the ordered data set. The results of the comparison of the Gaussian Filter and the Median Filter are in accordance with the PSNR (Peak Signal to Noise) value, where the greater the PSNR value, the better the reconstruction results. Of the 30 test images used in this study, it can be said that in these cases the Gaussian Filter method is better than the Media Filter in image softening.

Keywords : Gaussian, Median, Smoothing Image.

Intisari - Metode Penapis Gaussian dan Penapis Median merupakan dua dari beberapa metode dalam pelembutan citra (*smoothing image*). Kedua metode tersebut digunakan untuk memperbaiki kualitas citra. Efek dari pelembutan citra adalah citra menjadi blur. Penapis Gaussian adalah penapis blur yang menempatkan warna transisi yang signifikan dalam sebuah citra, kemudian membuat warna-warna pertengahan untuk menciptakan efek lembut pada sisi-sisi sebuah citra. Untuk Penapis Median ini, data yang digunakan untuk menghitung Median terdiri dari kumpulan data yang ganjil. Hal ini disebabkan oleh jumlah data yang ganjil diharapkan piksel yang akan diproses dapat berada ditengah. Pada Penapis Median digunakan matrik berdimensi $N \times N$. Dari matrik tersebut, kemudian data yang ada diurutkan dan dimasukkan dalam sebuah matrik berukuran $1 \times (N \times N)$. Hal ini berguna untuk mempermudah menemukan median dari kumpulan data yang telah urut tersebut. Hasil perbandingan Penapis Gaussian dan Penapis Median Sesuai nilai PSNR (Peak Signal to Noise), dimana semakin besar nilai PSNR maka semakin baik hasil rekonstruksi gambar. Dari 30 gambar uji yang digunakan dalam penelitian ini, dapat disimpulkan bahwa dalam kasus ini metode Penapis Gaussian lebih baik dari pada Penapis Median dalam pelembutan citra.

Kata Kunci: Gaussian, Median, Pelembutan Citra.

I. PENDAHULUAN

Perkembangan teknologi komputer dalam mengolah citra sudah berkembang pesat saat ini. Dengan perkembangan tersebut, teknologi pengolahan citra sudah mengalami banyak kemajuan dan diterapkan pada beberapa perangkat lunak yang

dipasarkan saat ini. Citra adalah salah satu komponen multimedia dan memegang peranan sangat penting sebagai bentuk informasi visual. Penggunaan citra digital semakin meningkat dikarenakan beberapa kelebihan yang dimiliki oleh citra digital, seperti kemudahan mendapatkan citra, memperbanyak citra dan pengolahan citra. Akan tetapi tidak semua hasil citra digital memiliki tampilan visual yang memuaskan mata manusia. Salah satunya dikarenakan adanya derau (*noise*). Salah satu teknik yang digunakan untuk memperbaiki kualitas citra adalah *Image Smoothing*. Pelembutan citra (*image smoothing*) bertujuan untuk menekan derau (*noise*) pada citra. Efek dari pelembutan citra adalah citra menjadi blur.

Pelembutan citra memiliki beberapa metode yang sering digunakan, dua diantaranya adalah Metode Penapis Gaussian dan Penapis Median. Dalam hal ini, penulis melakukan perbandingan hasil terhadap kedua metode tersebut dalam Pelembutan citra.

II. TINJAUAN PUSTAKA

Menurut Usman [1] penapis Gaussian sangat baik untuk menghilangkan *noise* yang bersifat sebaran normal, yang banyak di jumpai pada sebaran citra hasil proses digitasi menggunakan kamera karena merupakan fenomena alamiah akibat sifat pantulan cahaya dan kepekaan sensor cahaya pada kamera itu sendiri. Gaussian Blur adalah Filter blur yang menempatkan warna transisi yang signifikan dalam sebuah image, kemudian membuat warna-warna pertengahan untuk menciptakan efek lembut pada sisi-sisi sebuah image. Gaussian blur adalah salah satu filter blur yang menggunakan rumus matematika untuk menciptakan efek autofocus untuk mengurangi detail dan menciptakan efek berkabut. Penapis Gaussian didapat dari operasi konvolusi. Operasi perkalian yang dilakukan ialah perkalian antara matriks kernel dengan matriks gambar asli.

2.1 PENAPIS /FILTER

Filter adalah eknik untuk menyaring beberapa komponen yang tidak perlu (*noise*) atau untuk mengekstrak beberapa struktur yang mendasari citra [2].

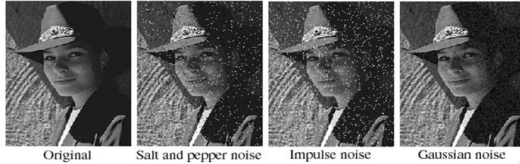


Gambar 1. citra dengan tingkat noise yang berbeda (panel kiri) dan struktur yang diekstraksi (tepi) dari citra asli (panel kanan)

Dari sekian banyak jenis filter, kita akan melihat filter linier dan non-linier. Lebih khusus lagi, fokus kami adalah pada apa itu, bagaimana mereka bekerja, mengapa mereka bermanfaat. Tapi sebelum masuk ke topik ini, pertama-tama kita perlu tahu apa itu noise [3].

2.2 DERAU /NOISE

Noise adalah sinyal yang tidak diinginkan dan tidak terduga dalam citra. Ini mendistorsi citra asli dengan memperkenalkan variasi acak kecerahan, iluminasi atau informasi warna ke dalam citra. Jenis kebisingan yang umum dikenal adalah sebagai berikut: Salt & pepper, Impulse dan Gaussian noise [2].

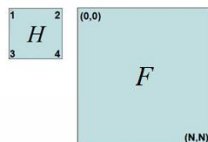


Gambar 2. Citra asli dan citra yang terkena noise

- Salt & pepper noise menunjukkan kemunculan acak piksel hitam (lada) dan putih (garam) di seluruh citra, yang berarti beberapa piksel yang dipilih secara acak ditetapkan sebagai hitam atau putih.
- Sebaliknya, Impulse noise berarti kemunculan piksel putih secara acak saja.
- Terakhir dan yang terpenting, derau Gaussian adalah jenis derau yang paling sering diamati. Ini memperkenalkan variasi intensitas yang diambil dari distribusi Gaussian. Dengan kata lain, setiap intensitas piksel pada citra dimodifikasi oleh variabel acak yang diambil dari distribusi Gaussian.

2.3 KORELASI

Korelasi adalah menggantikan setiap piksel citra **F** dengan jumlah bobot tetangganya, dan bobot tersebut ditentukan oleh filter **H**, yang juga disebut "kernel" atau "masker". Dengan menetapkan $H[u,v]$ untuk semua u dan v sebagai $1/(2k+1)^2$, kita memperoleh Persamaan 1 rata-rata bergerak [1][2]. filter **H** didefinisikan sebagai



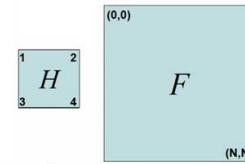
$$G[i, j] = \sum_{u=-k}^k \sum_{v=-k}^k H[u, v]F[i + u, j + v]$$

$$= H \otimes F$$

Gambar 3. Definisi korelasi

2.4 KONVOLUSI

Konvolusi mirip dengan korelasi, kecuali hanya satu tetapi perbedaan yang signifikan. Ini membalik filter di kedua dimensi (kanan ke kiri dan bawah ke atas dalam kasus filter 2D). Dan kemudian itu menerapkan korelasi. Singkatnya, konvolusi adalah kombinasi dari filter flipping dan korelasi[1][2]. Gambar 6 mengilustrasikan hal ini.



$$G[i, j] = \sum_{u=-k}^k \sum_{v=-k}^k H[u, v]F[i - u, j - v]$$

$$G = H \star F$$

Gambar 4. Definisi konvolusi

2.5 PNSR

PSNR atau *peak signal-to-noise ratio* adalah nisbah atau perbandingan antara daya sinyal maksimum terhadap daya derau yang memengaruhi ketepatan data yang dikirimkan. Karena banyak sinyal yang memiliki rentang dinamis yang sangat lebar, nisbah ini biasa dinyatakan dalam nilai logaritma dengan skala desibel.

Nisbah puncak sinyal terhadap derau (PSNR) dapat didefinisikan dalam rata-rata kuadrat galat (MSE) sebagai berikut (dalam dB):

$$PSNR = 10 \cdot 10 \log \left(\frac{maks_I^2}{MSE} \right)$$

$$= 20 \cdot 10 \log \left(\frac{maks_I}{\sqrt{MSE}} \right)$$

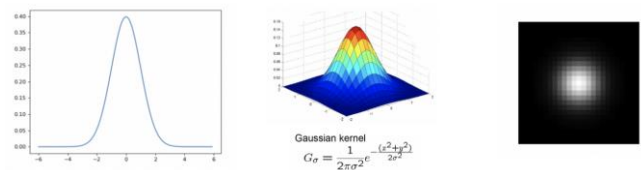
$$= 20 \cdot 10 \log(maks_I) - 10 \cdot 10 \log(MSE)$$

dengan $maks_I$ adalah nilai maksimum sinyal (puncak sinyal).

III. METODOLOGI PENELITIAN

3.1 PENAPIS GAUSIAN

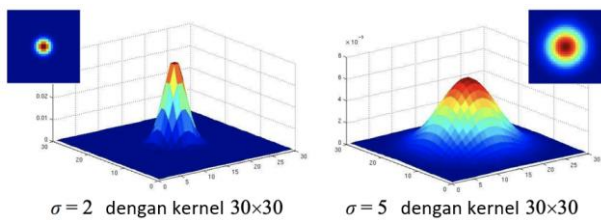
Kernel Gaussian, seperti namanya, memiliki bentuk fungsi 'distribusi Gaussian' untuk menentukan bobot di dalam kernel, yang digunakan untuk menghitung rata-rata tertimbang dari titik-titik tetangga (piksel) dalam sebuah citra [4][5].



Gambar 5. Ilustrasi Gaussian. Panel kiri: Distribusi Gaussian 1D. Panel tengah: Bentuk umum Gaussian 2D dengan rata-rata nol. Panel kanan: penggambaran filter Gaussian dalam skala abu-abu (putih = tinggi, hitam = nilai rendah)

Setiap nilai dalam filter Gaussian berasal dari distribusi Gaussian rata-rata nol. Satu hal yang perlu kita ingat adalah bahwa ukuran kernel bergantung pada standar deviasi σ fungsi Gauss:

Aturan praktis untuk **kernel size** = $2 \cdot \lceil 3\sigma \rceil + 1$



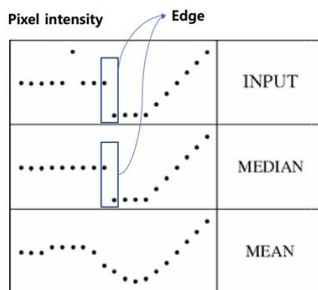
Gambar 6. Penetapan standar deviasi σ

Dengan menetapkan standar deviasi σ , kita dapat mengontrol sampai sejauh mana kita menghaluskan citra. Dengan kata lain, semakin tinggi standar deviasi semakin kuat efek smoothing yang dimiliki citra[2][6].

3.2 PENAPIS MEDIAN

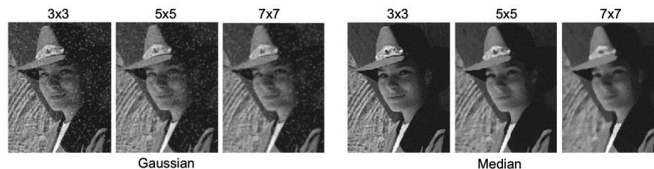
Penapis Median merupakan salah satu Non-linear filter yang juga digunakan untuk smoothing. Ide dasarnya adalah mengganti setiap piksel dengan median piksel tetangganya (piksel di jendela).

Dengan demikian, ia menghilangkan beberapa lonjakan yang disebabkan oleh kebisingan: terutama impuls dan kebisingan garam & merica. Ini karena piksel noise yang berdiri sendiri dengan intensitas ekstrim seperti hitam dan putih tidak dapat bertahan setelah pemfilteran median. Keuntungan lain dari filter median adalah tidak memasukkan nilai piksel baru karena hanya menggunakan kembali nilai piksel yang ada dari jendela. Lebih lanjut, tidak seperti filter rata-rata lainnya, filter ini menghilangkan noise tanpa kehilangan informasi tepi seperti yang diilustrasikan di bawah [7][8].



Gambar 7. Perbandingan visual antara Penapis Median dan Mean filter

Metode penapis Median dapat dipertimbangkan ketika penapis Gaussian tidak dapat menyelesaikan masalah. Gambar 9 menunjukkan perbandingan visual antara filter Gaussian dan Median. Kita dapat mengamati bahwa ketika tingkat derau terlalu tinggi, meskipun jumlah piksel derau berkurang dengan bertambahnya ukuran filter Gaussian, mereka tetap ada dalam citra. Filter median, di sisi lain, sudah menghapus sebagian besar piksel noise dengan ukuran filter 3 x 3. Dengan menerapkan ukuran filter yang lebih besar, filter Median lebih jauh mengecualikan piksel noise tetapi kehilangan banyak informasi struktur citra dan detail citra [8] [9].



Gambar 8. Perbandingan visual antara filter Gaussian dan filter Median

Ukuran filter adalah $2k + 1$, ukuran jendela untuk konvolusi 2D adalah $(2k + 1)^2$. Untuk perbandingan, pertimbangkan dua fase konvolusi 1D. Ukuran filter tetap sama dengan $2k + 1$, tetapi kali ini ukuran jendela juga $2k + 1$ karena konvolusinya adalah 1D dan konvolusi dilakukan dua kali. Jumlah total komputasi untuk setiap langkah adalah $2 \cdot (2k + 1)$ [2][3].

Definisi Konvolusi 2D

$$y[m, n] = x[m, n] * h[m, n] = \sum_{j=-\infty}^{\infty} \sum_{i=-\infty}^{\infty} x[i, j] \cdot h[m - i, n - j]$$

$$= h[m, n] * x[m, n] = \sum_{j=-\infty}^{\infty} \sum_{i=-\infty}^{\infty} h[i, j] \cdot x[m - i, n - j] \quad (\text{Komutatif Konvolusi})$$

(Jika $h_1[m, n]$ dapat dipisahkan terhadap $(M \times 1)$ dan $(1 \times N)$, maka $h[m, n] = h_1[m] \cdot h_2[n]$)

$$= \sum_{j=-\infty}^{\infty} \sum_{i=-\infty}^{\infty} h_1[i] \cdot h_2[j] \cdot x[m - i, n - j]$$

$$= \sum_{j=-\infty}^{\infty} h_2[j] \cdot \left(\sum_{i=-\infty}^{\infty} h_1[i] \cdot x[m - i, n - j] \right)$$

Definisi Konvolusi 1D

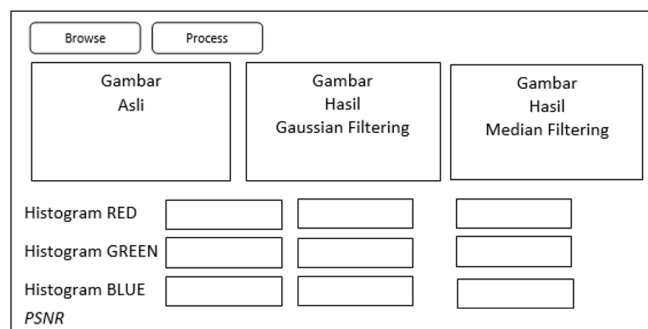
$$= h_2[n] * (h_1[m] * x[m, n])$$

$$= h_1[m] * (h_2[n] * x[m, n])$$

Gambar 9. Pembuktian konvolusi dengan filter terpisah 2D identik dengan konvolusi dengan dua filter 1D.

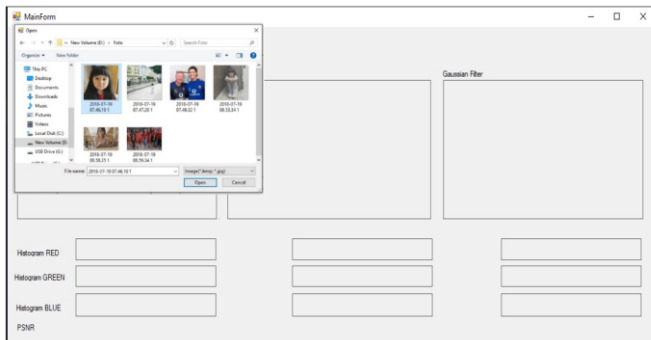
IV. HASIL DAN PEMBAHASAN

Hasil Aplikasi Perbandingan Metode Penapis Gaussian dan Penapis Median dalam Pelembutan Citra ini merupakan aplikasi yang dibuat menggunakan bahasa pemrograman c++ menggunakan Microsoft Visual Studio 2017 yang berjalan pada sistem operasi windows 10 dan berbasis desktop.



Gambar 10. Sketsa Aplikasi pemroses perbandingan penapis Gaussian dan penapis Median

Input Aplikasi adalah gambar yang akan tapis yang akan ditampilkan pada kolom gambar Asli, kolom tengah adalah menampilkan gambar yang sudah ditapis dengan penapis Gaussian, dan yang sebelah kanan adalah kolom untuk menampilkan hasil penapisan Median. Aplikasi juga menampilkan nilai nilai Histogram untuk komponen warna RGB (red, green, blue) berikut nilai PSNRnya



Gambar 11. Aplikasi pemroses perbandingan penapis Gaussian dan penapis Median

Hasil Pelembutan citra dengan penapis Gaussian dan Median ditunjukkan pada tabell. Aplikasi menghasilkan nilai *PSNR* (*Peak Signal to Noise Ratio*). *PSNR* merupakan sebuah parameter yang biasa digunakan dalam proses kompresi image untuk menentukan kualitas hasil rekonstruksi image akhir. *PSNR* digunakan untuk mengukur kualitas gambar setelah rekonstruksi. *PSNR* diukur dalam satuan decibel(db. Semakin besar nilai *PSNR* maka nilai hasil operasi pengurangan derau atau hasil rekonstruksi semakin baik.

Tabel 1 Tabel Nilai Hasil *PSNR*

| No | Input | Nilai <i>PSNR</i> Median | Nilai <i>PSNR</i> Gaussian |
|-----|--------|--------------------------|----------------------------|
| 1. | 1.jpg | 35,4206328851048db | 35,4771339392909db |
| 2. | 2.jpg | 34,5272849786915db | 34,5376996659414db |
| 3. | 3.jpg | 34,5665892929511db | 34,5964912318051db |
| 4. | 4.jpg | 36,2023725791667db | 36,5517740063346db |
| 5. | 5.jpg | 35,0659931210976db | 35,1287539339505db |
| 6. | 6.jpg | 36,9200567421823db | 37,0307201679895db |
| 7. | 7.jpg | 36,7125059246039db | 36,7848361223769db |
| 8. | 8.jpg | 35,6253094002384db | 35,6954370497486db |
| 9. | 9.jpg | 36,6250384445267db | 36,7776637663838db |
| 10. | 10.jpg | 35,5335426067255db | 35,729472107301db |
| 11. | 11.jpg | 34,9926777120343db | 35,1129548206561db |
| 12. | 12.jpg | 36,1997564127562db | 36,3359455415551db |
| 13. | 13.jpg | 34,2050199111922db | 34,3234763326675db |
| 14. | 14.jpg | 34,76902222361712db | 34,8109270508847db |
| 15. | 15.jpg | 35,369789839314db | 35,4654588365681db |
| 16. | 16.jpg | 35,6057874496321db | 35,7818787805808db |
| 17. | 17.jpg | 36,1348515461602db | 36,1811356172715db |
| 18. | 18.jpg | 36,3748593686927db | 36,415992912263db |
| 19. | 19.jpg | 34,6417707487786db | 34,6202740844401db |
| 20. | 20.jpg | 34,3388341649303db | 34,5131001792165db |
| 21. | 21.jpg | 35,3465771601163db | 35,4403165544254db |
| 22. | 22.jpg | 34,9034709325877db | 35,0287439483377db |
| 23. | 23.jpg | 34,1456219315466db | 34,23877896885642db |
| 24. | 24.jpg | 34,398306547929db | 34,4296959599021db |
| 25. | 25.jpg | 35,7723528891704db | 35,9340435855879db |
| 26. | 26.jpg | 34,2840882877884db | 34,3994265935704db |
| 27. | 27.jpg | 34,9746004617543db | 35,0826964639661db |
| 28. | 28.jpg | 36,404425148093db | 36,4492857475908db |
| 29. | 29.jpg | 34,889546317633db | 34,9424006982941db |
| 30. | 30.jpg | 36,091889003555db | 36,1381974428951db |

V. KESIMPULAN DAN SARAN

Dari hasil penelitian yang telah dilakukan penulis, penulis dapat menyimpulkan bahwa:

1. Pelembutan citra dapat diimplementasikan dengan metode Gaussian filter dan Penapis Median pada aplikasi pemroses perbandingan Metode *Penapis Gaussian* dan *Penapis Median* dalam *Pelembutan citra* ini menghasilkan 2 output masing-masing dengan metode

Penapis Gaussian dan metode *Penapis Median*.

2. Sesuai nilai *PSNR* (*Peak Signal to Noise*), dimana semakin besar nilai *PSNR* maka semakin baik hasil rekonstruksi citra, maka dari 30 input citra dapat disimpulkan bahwa dalam kasus ini metode *Penapis Gaussian* lebih baik dari *Penapis Median* dalam Pelembutan citra.

REFERENSI

- [1] U. Ahmad, *Pengolahan citra digital dan teknik pemrogramannya*, Edisi 1. Yogyakarta: Graha Ilmu, 2005. [Online]. Available: <https://opac.perpusnas.go.id/DetailOpac.aspx?id=268553>
- [2] DevBlog, “[CV] 1. Image Processing Basic: Linear Filters,” 2022. <https://medium.com/jun94-devpblog/cv-1-image-processing-basic-filter-noise-moving-average-correlation-and-convolution-c026502f6391>
- [3] P. Li, H. Wang, M. Yu, and Y. Li, “Overview of Image Smoothing Algorithms,” 2021, doi: 10.1088/1742-6596/1883/1/012024.
- [4] Wikipedia, “Gaussian Filter,” *SpringerReference*, 2011. https://en.wikipedia.org/wiki/Gaussian_filter
- [5] A. Wedianto, H. L. Sari, and Y. S. H, “269-Article Text-766-1-10-20160609,” *J. Media Infotama*, vol. 12, no. 1, pp. 21–30, 2016.
- [6] R. Shukla and L. Shrivastava, “Image Restoration of Image with Gaussian Filter,” vol. 07, no. 12, pp. 555–558, 2020.
- [7] J. Micek, “Median Filter Algorithm,” 2015. https://en.wikipedia.org/wiki/Median_filter
- [8] E. Listiyani, “IMPLEMENTASI ADAPTIVE MEDIAN FILTER SEBAGAI”.
- [9] I. Agustina, F. Nasir, and A. Setiawan, “The Implementation of Image Smoothing to Reduce Noise using Gaussian Filter,” *Int. J. Comput. Appl.*, vol. 177, no. 5, pp. 15–19, 2017, doi: 10.5120/ijca2017915755.