

Pengujian Modifikasi Kernel Konvolusi Untuk Penajaman dan Penghalusan Citra Berwarna

Suriski Sitinjak

Badan Kepegawaian Daerah Provinsi Kalimantan Tengah

Keywords:

*convolution
kernel
digital image*

ABSTRACT

A digital image sometimes has poor quality, for example there is noise, too sharp, not sharp, blurry, too dark, too bright, and so on. For this reason, it is necessary to do image enhancement as a preprocessing step in image processing so that images can be used for further processes, such as pattern recognition. Convolution is an image enhancement technique that is used to manipulate pixels in the image, in order to obtain images that have the desired properties and information. This study aims to conduct experiments on the convolution kernel/filter with several tests for sharpening and blurring true color image. Experimental testing results show that the sharpness of a image depends on the value of each element and the size of the kernel matrix. The greater value and size of the matrix used, the higher level of sharpness produced. The level of blurring depends on the size of the kernel matrix. The greater size of matrix, the greater level of smoothness (more blurred). This research is expected to help to find out the right kernel to do image enhancement, so that it can be used for further processing.

*Copyright © 2020 Universitas Indraprasta PGRI.
All rights reserved.*

Corresponding Author:

Suriski Sitinjak,

Badan Kepegawaian Daerah Provinsi Kalimantan Tengah,

Jl. Willem AS No.11, Menteng, Kec. Jekan Raya, Kota Palangka Raya, Kalimantan Tengah 74874

Email: suriskisitinjak@gmail.com

1. PENDAHULUAN

Citra digital merepresentasikan suatu objek dalam bentuk diskrit sehingga dapat diolah komputer. Suatu citra yang ditangkap oleh digitizer terkadang tidak sesuai dengan yang diharapkan, karena bisa saja mengandung derau (noise), tidak dapat ditangkap penglihatan manusia atau diakibatkan lensa pada digitizer kotor [1], sehingga citra tersebut akan sulit diinterpretasikan karena informasi yang disampaikan berkurang [2]. Dalam pengolahan citra, konvolusi digunakan untuk memanipulasi piksel-piksel dalam citra, sehingga diperoleh citra yang memiliki sifat dan informasi yang diinginkan.

Penelitian ini bertujuan untuk melakukan pengujian konvolusi dengan memodifikasi kernel konvolusi pada citra berwarna. Pengujian menggunakan perangkat lunak Matlab yang dilakukan beberapa kali untuk penajaman dan penghalusan citra untuk melihat pengaruh perubahan nilai kernel terhadap ketajaman dan kehalusan citra. Pengujian penajaman dan penghalusan citra berwarna dilakukan dengan memodifikasi nilai-nilai matriks kernel pada ukuran 3x3 dan 5x5. Dari penelitian ini diharapkan dapat membantu mengetahui kernel yang tepat untuk memperbaiki kualitas suatu citra, sehingga dapat digunakan untuk proses lebih lanjut. Penelitian ini terdiri dari pendahuluan, metode, hasil yang berisi pembahasan hasil penajaman dan penghalusan citra berwarna dan kesimpulan.

2. METODE

Pada penelitian ini pengujian menggunakan perangkat lunak Matlab. Citra berwarna yang digunakan untuk semua pengujian adalah sama. Pengujian untuk penajaman citra dilakukan sebanyak 5 kali sedangkan

untuk penghalusan citra dilakukan sebanyak 3 kali dengan menggunakan kernel konvolusi yang berbeda, baik nilai elemen tiap matrik dan ukuran matrik kernel untuk membandingkan citra yang dihasilkan.

2.1. Konvolusi Citra

Suatu citra digital didefinisikan sebagai suatu fungsi $f(x,y)$ terdiri dari sejumlah baris dan sejumlah kolom dengan x dan y adalah koordinat spasial, dan amplitude f di titik koordinat (x, y) dinamakan intensitas dari citra pada titik tersebut [1] [3]. Konvolusi dilakukan dengan mengalikan elemen-elemen pada dua buah matriks dan menjumlahkan hasilnya, sehingga terdapat keluaran nilai dari sebuah piksel yang merupakan hasil konvolusi. Satu dari matriks tersebut merepresentasikan citra dan yang lain merepresentasikan filter/tapis/kernel [4].

Notasi yang umum untuk konvolusi adalah [5]:

$$y = h \otimes x, \text{ dimana:}$$

h = matriks kernel

x = matriks citra

y = matriks hasil konvolusi

Proses konvolusi dilakukan dengan cara berikut ini [5]:

- 1) Rotasikan matriks kernel konvolusi sebesar 180°
- 2) Geser matriks kernel konvolusi sebanyak (m,n) , sehingga terjadi overlap dengan matriks input
Jumlah hasil perkalian dari daerah yang overlap akan menghasilkan konvolusi pada titik (m,n) .

2.2. Penajaman (Sharpening) pada Citra Berwarna

Pada penajaman citra, intensitas suatu citra akan diperkuat dengan melewati pada suatu tapis lolos atas (high pass filter). Ada berbagai macam kernel konvolusi yang membentuk high-pass filter, ada yang menonjolkan arah-arrah tertentu (vertikal, horisontal, diagonal, atau juga utara, selatan, barat dan timur) [4]. Penajaman citra bisa terjadi karena citra tersebut didiferensiasi (dikurangi) dengan matriks kernel (h). Hasil penjumlahan tiap elemen pada matriks kernel adalah 1.

2.3. Penghalusan (Blurring) pada Citra Berwarna

Proses penghalusan citra umumnya dilakukan dengan melewati citra asli pada suatu low pass filter. Pada penghalusan citra, nilai antar piksel direrata. Nilai hasil kali antara matriks kernel dan nilai pembagi haruslah 1.

3. HASIL DAN PEMBAHASAN

Berikut ini dilakukan pengujian terhadap kernel yang berbeda-beda, baik elemen maupun ukuran dari matriks kernelnya. Citra berwarna yang diuji adalah "bird.bmp" ukuran 256 x 256.

3.1. Penajaman (Sharpening) pada Citra Berwarna

3.1.1. Pengujian 1 Penajaman Citra

Menggunakan kernel:

$$h = \begin{bmatrix} -1 & -1 & -1 \\ -1 & 9 & -1 \\ -1 & -1 & -1 \end{bmatrix}$$

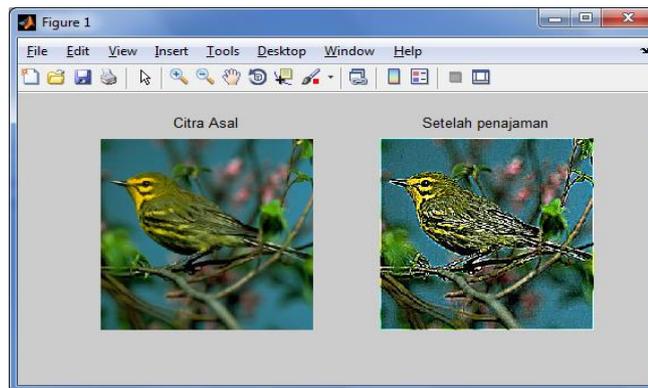
Kode program seperti ditunjukkan pada gambar 1 berikut.

```
%konvolusi : penajaman citra RGB
x = imread('bird.BMP');
x = double(x);
h = [-1 -1 -1;
     -1 9 -1;
     -1 -1 -1];
hasilKonvolusi=convn(x,h,'same');
hasilKonvolusi=uint8(hasilKonvolusi);
subplot(1,2,1);
imshow('bird.bmp');title('Citra Asal');
```

```
subplot(1,2,2);
imshow(hasilKonvolusi);title('Setelah penajaman');
imwrite(hasilKonvolusi,'birdRGB_Sharp.bmp','bmp');
```

Gambar 1. Kode Program Pengujian 1 Penajaman Citra

Pada pengujian 1 ini digunakan kernel (h) ukuran 3x3. Nilai elemen matriks kernel bagian tengah haruslah lebih besar. Selain itu nilai penjumlahan semua elemen-elemen matriks kernel haruslah 1 (-1 + -1 + -1 + -1 + 9 + -1 + -1 + -1 + -1 = 1). Karena jika nilai penjumlahan semua elemen matriks lebih dari satu, maka akan menghasilkan penajaman citra dengan tingkat kecerahan citra yang berbeda dari citra asal (jika lebih dari 1 hasilnya akan menjadi lebih cerah, sedangkan jika 0 atau kurang dari 1 citra akan menjadi lebih gelap). Hasil untuk konvolusi penajaman citra berdasarkan kernel yang digunakan terlihat pada Gambar 2.



Gambar 2. Hasil Pengujian 1 penajaman citra

3.1.2. Pengujian 2 Penajaman Citra

Menggunakan kernel:

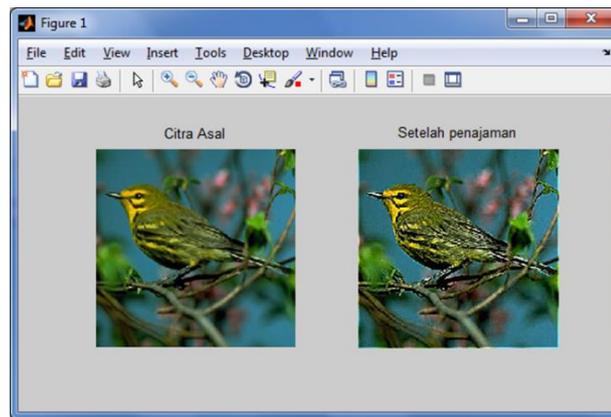
$$h = \begin{bmatrix} 0 & -1 & 0 \\ -1 & 5 & -1 \\ 0 & -1 & 0 \end{bmatrix}$$

Kode program seperti ditunjukkan pada gambar 3 berikut.

```
%konvolusi : penajaman citra RGB
x = imread('bird.BMP');
x = double(x);
h = [0 -1 0;
     -1 5 -1;
     0 -1 0];
hasilKonvolusi=convn(x,h,'same');
hasilKonvolusi=uint8(hasilKonvolusi);
subplot(1,2,1);
imshow('bird.bmp');title('Citra Asal');
subplot(1,2,2);
imshow(hasilKonvolusi);title('Setelah penajaman');
imwrite(hasilKonvolusi,'birdRGB_Sharp_kernelModif2.bmp','bmp');
```

Gambar 3. Kode Program Pengujian 2 Penajaman Citra

Pada pengujian kedua ini, nilai elemen-elemen matriks kernel dimodifikasi untuk penajaman citra bagian vertikal dan horizontal. Nilai elemen-elemen matriks kernel jika dijumlahkan menghasilkan nilai 1 (0 + -1 + 0 + -1 + 5 + -1 + 0 + -1 + 0 = 1), nilai elemen pada bagian tengah matriks haruslah bernilai lebih besar. Seperti penjelasan pada pengujian 1, jika hasil penjumlahan elemen-elemen matriks melebihi 1 akan terjadi perubahan tingkat kecerahan. Pada Gambar 4 terlihat tingkat ketajaman pada pengujian kedua ini lebih rendah dibandingkan pada pengujian pertama.



Gambar 4. Hasil Pengujian 2 penajaman citra

3.1.3. Pengujian 3 Penajaman Citra

Menggunakan kernel:

$$h = \begin{bmatrix} -1 & -1 & -1 & -1 & -1 \\ -1 & -1 & -1 & -1 & -1 \\ -1 & -1 & 25 & -1 & -1 \\ -1 & -1 & -1 & -1 & -1 \\ -1 & -1 & -1 & -1 & -1 \end{bmatrix}$$

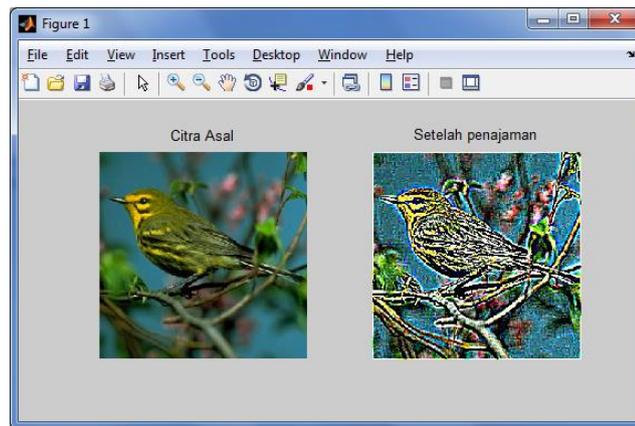
Kode program seperti ditunjukkan pada gambar 5 berikut.

```
%konvolusi : penajaman citra RGB
x = imread('bird.BMP');
x = double(x);
h = [-1 -1 -1 -1 -1 ;
     -1 -1 -1 -1 -1 ;
     -1 -1 25 -1 -1 ;
     -1 -1 -1 -1 -1 ;
     -1 -1 -1 -1 -1 ];
hasilKonvolusi=convn(x,h,'same');
hasilKonvolusi=uint8(hasilKonvolusi);

subplot(1,2,1);
imshow('bird.bmp');title('Citra Asal');
subplot(1,2,2);
imshow(hasilKonvolusi);title('Setelah penajaman');
imwrite(hasilKonvolusi,'birdRGB_Sharp_kernelModif3.bmp','bmp');
```

Gambar 5. Kode Program Pengujian 3 Penajaman Citra

Pada pengujian ketiga ini dilakukan dengan memodifikasi elemen-elemen dan ukuran matriks kernel menjadi 5x5 dengan penyebaran yang merata. Nilai tengah elemen matriks lebih besar daripada sekelilingnya dan jika dijumlahkan semua elemen-elemen matriks tersebut akan menghasilkan nilai 1. Pada Gambar 6 hasil konvolusi terlihat bahwa dengan menggunakan kernel dengan ukuran dan nilai elemen matriksnya lebih besar akan menghasilkan tingkat penajaman yang lebih tinggi dibandingkan dengan menggunakan kernel dengan ukuran matriks kecil.



Gambar 6. Hasil Pengujian 3 penajaman citra

3.1.4. Pengujian 4 Penajaman Citra

Menggunakan kernel:

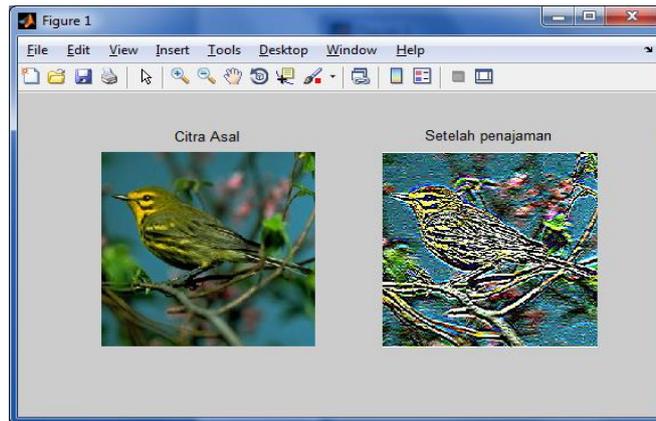
$$h = \begin{bmatrix} 0 & 0 & -9 & 0 & 0 \\ 0 & 0 & -7 & 0 & 0 \\ 0 & 0 & 33 & 0 & 0 \\ 0 & 0 & -7 & 0 & 0 \\ 0 & 0 & -9 & 0 & 0 \end{bmatrix}$$

Kode program ditunjukkan pada gambar 7 berikut.

```
%konvolusi : penajaman citra RGB
x = imread('bird.BMP');
x = double(x);
h = [0 0 -9 0 0 ;
     0 0 -7 0 0 ;
     0 0 33 0 0 ;
     0 0 -7 0 0 ;
     0 0 -9 0 0 ];
hasilKonvolusi=convn(x,h,'same');
hasilKonvolusi=uint8(hasilKonvolusi);
subplot(1,2,1);
imshow('bird.bmp');title('Citra Asal');
subplot(1,2,2);
imshow(hasilKonvolusi);title('Setelah penajaman');
imwrite(hasilKonvolusi,'birdRGB_Sharp_kernelModif4.bmp','bmp');
```

Gambar 7. Kode Program Pengujian 4 Penajaman Citra

Pada pengujian keempat digunakan kernel dengan ukuran matriks 5x5 dan nilai-nilai elemen hanya pada bagian tengah arah vertikal pada matriks. Terlihat pada Gambar 8 hasil penajaman citra nampak seperti membentuk penajaman arah horizontal, hal ini dikarenakan pada matriks kernel elemen-elemen yang ada hanya pada arah vertikal saja.



Gambar 8. Pengujian 4 penajaman citra

Pada pengujian ke-5 selanjutnya ditunjukkan bagaimana perbedaannya jika menggunakan kernel yang nilai elemen-elemen matriksnya hanya pada arah horizontal bagian tengah.

3.1.5. Pengujian 5 Penajaman Citra

Menggunakan kernel:

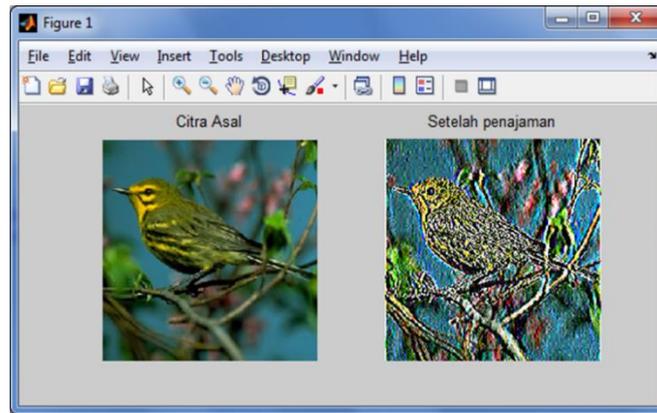
$$h = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ -9 & -7 & 33 & -7 & -9 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

Kode program seperti ditunjukkan gambar 9 berikut.

```
%konvolusi : penajaman citra RGB
x = imread('bird.BMP');
x = double(x);
h = [ 0 0 0 0 0 ;
      0 0 0 0 0 ;
      -9 -7 33 -7 -9 ;
      0 0 0 0 0 ;
      0 0 0 0 0 ];
hasilKonvolusi=convn(x,h,'same');
hasilKonvolusi=uint8(hasilKonvolusi);
subplot(1,2,1);
imshow('bird.bmp');title('Citra Asal');
subplot(1,2,2);
imshow(hasilKonvolusi);title('Setelah penajaman');
imwrite(hasilKonvolusi,'birdRGB_Sharp_kernelModif5.bmp','bmp');
```

Gambar 9. Kode Program Pengujian 5 Penajaman Citra

Matriks kernel diberi nilai yang sama dengan nilai matriks pada pengujian ke-4 tapi dengan angka -9, -7, 33, -7 dan -9 yang membentuk arah horizontal di bagian tengah matriks. Terlihat pada gambar 10 bahwa citra hasil penajaman seperti membentuk penajaman arah vertikal. Sangat terlihat perbedaan arahnya dibandingkan dengan pengujian ke-4.



Gambar 10. Pengujian 5 penajaman citra

3.2. Pengujian Penghalusan (Blurring) pada Citra Berwarna

Berikut contoh penghalusan citra dengan beberapa pengujian pada kernel yang berbeda baik elemen maupun ukuran matriksnya.

3.2.1 Pengujian 1 Penghalusan Citra

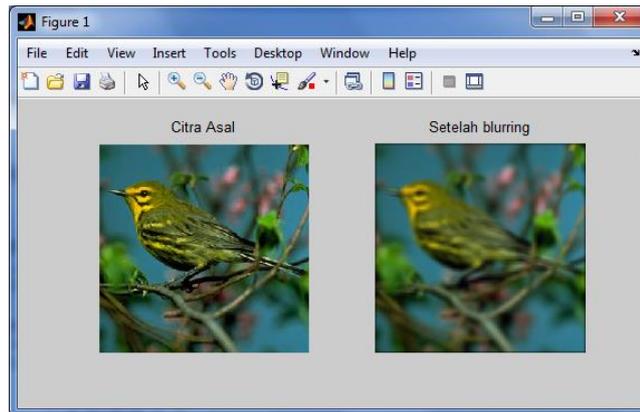
Menggunakan kernel:

```
h = (1/49)*[1 1 1 1 1 1 1;
            1 1 1 1 1 1 1;
            1 1 1 1 1 1 1;
            1 1 1 1 1 1 1;
            1 1 1 1 1 1 1;
            1 1 1 1 1 1 1;
            1 1 1 1 1 1 1];
```

Kode program seperti ditunjukkan pada gambar 11 berikut.

```
%konvolusi : blurring citra RGB
x = imread('bird.bmp');
x = double(x);
h = (1/49)*[1 1 1 1 1 1 1;
            1 1 1 1 1 1 1;
            1 1 1 1 1 1 1;
            1 1 1 1 1 1 1;
            1 1 1 1 1 1 1;
            1 1 1 1 1 1 1;
            1 1 1 1 1 1 1];
hasilKonvolusi=convn(x,h,'same');
hasilKonvolusi=uint8(hasilKonvolusi);
subplot(1,2,1);
imshow('bird.bmp');title('Citra Asal');
subplot(1,2,2);
imshow(hasilKonvolusi);title('Setelah blurring');
imwrite(hasilKonvolusi,'birdRGB_Blur.bmp','bmp');
```

Gambar 11. Kode Program Pengujian 1 Penghalusan Citra



Gambar 12. Hasil Pengujian 1 *blurring* citra

Pada penghalusan citra, kernel yang digunakan berbeda, karena dikalikan dengan perata-rata. Pada pengujian pertama untuk penghalusan citra ini menggunakan matriks kernel ukuran 7×7 yang memiliki masing-masing nilai elemennya 1. Nilai perata-rata adalah $1/49$ dan jumlah tiap elemen adalah 49. Sehingga hasil kali antara nilai perata-rata dengan nilai penjumlahan tiap elemen adalah 1 ($1/49 * 49 = 1$). Gambar 12 menunjukkan hasil penghalusan citra bird.bmp untuk pengujian pertama penghalusan citra. Pada pengujian berikutnya menggunakan ukuran matriks kernel yang lebih kecil untuk melihat pengaruh ukuran matriks dalam penghalusan citra.

3.2.2. Pengujian 2 Penghalusan Citra

Menggunakan kernel:

$$h = (1/9) * \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix};$$

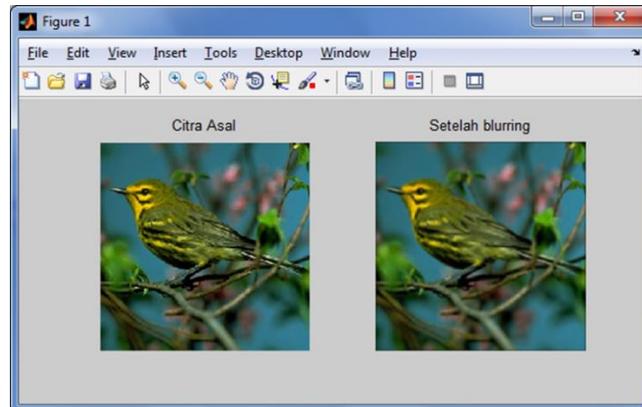
Kode program seperti ditunjukkan gambar 13 berikut.

```
%konvolusi : blurring citra RGB
x = imread('bird.bmp');
x = double(x);
h = (1/9) * [1 1 1;
            1 1 1;
            1 1 1];
hasilKonvolusi=convn(x,h,'same');
hasilKonvolusi=uint8(hasilKonvolusi);
subplot(1,2,1);
imshow('bird.bmp');title('Citra Asal');
subplot(1,2,2);
imshow(hasilKonvolusi);title('Setelah blurring');
imwrite(hasilKonvolusi,'birdRGB_Blur_kernelModif2.bmp','bmp');
```

Gambar 13. Kode Program Pengujian 2 Penghalusan Citra

Kernel yang digunakan menggunakan matriks ukuran 3×3 dengan nilai tiap elemen adalah 1. Nilai pembagi adalah $1/9$ dan nilai penjumlahan tiap elemen dalam matriks adalah 9, sehingga hasil kali nilai perata-rata dengan jumlah tiap elemen adalah 1 ($1/9 * 9 = 1$). Nilai hasil kali pembagi atau perata-rata dengan nilai penjumlahan tiap elemen matriks haruslah bernilai 1, karena jika tidak maka akan menghasilkan citra dengan tingkat kecerahan yang berbeda dengan aslinya. Jika > 1 akan menghasilkan citra dengan tingkat kecerahan yang lebih tinggi/terang, sedangkan jika < 1 akan menghasilkan citra dengan tingkat kecerahan yang lebih rendah/gelap. Gambar 14 menunjukkan hasil penghalusan citra yang lebih rendah dibandingkan dengan pengujian 1. Hal ini membuktikan bahwa ukuran matriks pada kernel juga mempengaruhi tingkat penghalusan citra. Semakin besar citra semakin tinggi penghalusan

citra. Berikutnya dilakukan pengujian dengan mengubah nilai tiap elemen dalam matriks kernel untuk melihat bagaimana pengaruh perbesaran nilai elemen dalam penghalusan citra.



Gambar 14. Hasil Pengujian 2 *blurring* citra

3.2.3. Pengujian 3 Penghalusan Citra

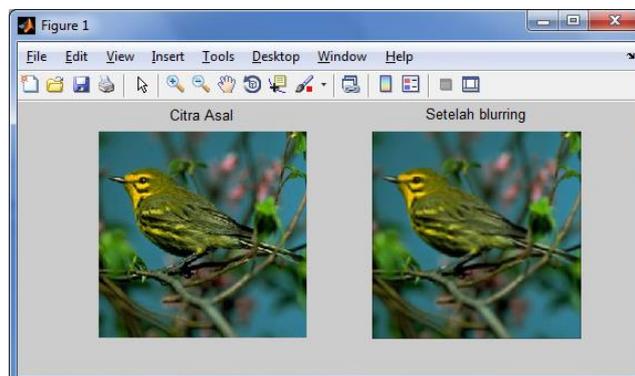
Menggunakan kernel:

$$h = (1/69) * \begin{bmatrix} 7 & 7 & 7; \\ 9 & 9 & 9; \\ 7 & 7 & 7 \end{bmatrix};$$

Kode program ditunjukkan pada gambar 15 berikut.

```
%konvolusi : blurring citra RGB
x = imread('bird.bmp');
x = double(x);
h = (1/69) * [7 7 7;
              9 9 9;
              7 7 7];
hasilKonvolusi=convn(x,h,'same');
hasilKonvolusi=uint8(hasilKonvolusi);
subplot(1,2,1);
imshow('bird.bmp');title('Citra Asal');
subplot(1,2,2);
imshow(hasilKonvolusi);title('Setelah blurring');
imwrite(hasilKonvolusi,'birdRGB_Blur_kernelModif3.bmp','bmp');
```

Gambar 15. Kode Program Pengujian 3 Penghalusan Citra



Gambar 16. Hasil Pengujian 3 *blurring* citra

Pada pengujian 3 untuk penghalusan citra ini digunakan nilai tiap elemen yang lebih besar dibandingkan pengujian 2. Karena nilai penjumlahan tiap elemen adalah 69, maka digunakan perata-rata adalah 1/69. Dapat

dilihat pada gambar 16 hasil penghalusan citra dengan kernel tersebut. Ternyata dengan menaikkan nilai tiap elemen matriks kernel tidak membuat penghalusan citra semakin meningkat. Hasilnya tidak berbeda dengan penghalusan citra dengan pengujian 2 dengan nilai tiap elemennya 1 dan ukuran matriks yang sama.

4. PENUTUP

Konvolusi pada penajaman citra melibatkan kernel yang nilai tiap elemen matriks jika dijumlahkan haruslah 1, sedangkan pada penghalusan terjadi karena nilai antara piksel direrata dan nilainya juga harus 1. Pada penajaman citra, nilai elemen matriks kernel bagian tengah harus lebih besar. Penajaman citra bisa dilakukan pada arah vertikal, horizontal atau keduanya. Tingkat ketajaman bergantung pada nilai tiap elemen dan ukuran matriks. Semakin besar angka dan ukuran matriks yang digunakan, maka semakin tinggi pula tingkat ketajaman yang dihasilkan. Pada penghalusan citra, tingkat kehalusan bergantung pada ukuran matriks kernel. Semakin besar ukuran matriks kernel maka semakin besar tingkat kehalusannya (semakin blur).

DAFTAR PUSTAKA

- [1] D. Putra, *Pengolahan Citra Digital*. Yogyakarta: Penerbit Andi, 2009.
- [2] W. Gazali, H. Soeparno, and J. Ohliati, "PENERAPAN METODE KONVOLUSI DALAM PENGOLAHAN CITRA DIGITAL," *J. Mat Stat*, vol. 12, pp. 103–113, 2012.
- [3] V. Sutoyo, T., S. Si., M. Kom., Mulyanto, Edy, S. Si., Suhartono and M. K. Nurhayati, Oky Dwy, M. T., Wijanarto, *Teori Pengolahan Citra Digital*. Yogyakarta: Andi Publisher, 2009.
- [4] A. Wijaya, Marvin Ch. dan Prijono, *Pengolahan Citra Digital Menggunakan Matlab Image Processing Toolbox*. Bandung: Informatika Bandung, 2007.
- [5] B. Y. Dwiandiyanta, *Konvolusi dan Transformasi, Bahan Ajar Pengolahan Citra, Magister Teknik Informatika*. 2011.