Analysis of Medical Image Resizing Using Bicubic Interpolation Algorithm

Bambang Krismono Triwijoyo[a1], Ahmat Adil[a2]

[a]Department of Computer Science, Bumigora University
Jl. Ismail Marzuki Mataram, Indonesia
[1]bkrismono@universitasbumigora.ac.id
[2]ahmat.adil@universitasbumigora.ac.id

Abstract

Image interpolation is the most basic requirement for many image processing tasks such as medical image processing. Image interpolation is a technique used in resizing an image. To change the image size, each pixel in the new image must be remapped to a location in the old image to calculate the new pixel value. There are many algorithms available for determining the new pixel value, most of which involve some form of interpolation between the closest pixels in the old image. In this paper, we use the Bicubic interpolation algorithm to change the size of medical images from the Messidor dataset and then analyze it by measuring it using three parameters Mean Square Error (MSE), Root Mean Squared Error (RMSE), and Peak Signal-to-Noise Ratio (PSNR), and compare the results with Bilinear and Nearest-neighbor algorithms. The results showed that the Bicubic algorithm is better than Bilinear and Nearest-neighbor and the larger the image dimensions are resized, the higher the degree of similarity to the original image, but the level of computation complexity also increases.

Keywords: *Image Interpolation, image resizing, bicubic Interpolation,*

1. Introduction

Image interpolation is a term used for image processing but is often used with different terminology in literature such as image scaling, and image resizing. There are many algorithms currently used to resize digital images. Most reproduce visually appealing replicas of the original. As technology requires a smaller display area to be viewable on multiple devices, the image size is generally lowered to produce thumbnails. Enlarge the image generally for viewing on a large screen monitor or television [1] as well as the face detection system based on facial images from CCTV cameras[2]. When enlarging an image, it is impossible to find more information about the image than there is already, and the image quality is poor. However, there are several methods for increasing the number of pixels contained in an image, which can be created from the original pixels. This method is often referred to as an image interpolation algorithm.

There are various algorithms currently used to interpolate digital images. Some of the common interpolation algorithms are bicubic, bilinear, and nearest neighbor [3]. This algorithm uses a high-order interpolator which takes into account more pixels around it, thus requiring more computation time. The interpolation method retains most of the image details after interpolation and is very useful for correcting distorted images.

In this paper, there are three interpolation algorithms, namely bicubic, bilinear, and nearest-neighbor interpolation to change the size of medical images and analyze the resizing results using parameters Mean Square Error (MSE), Root Mean Squared Error (RMSE), and Peak Signal-to-Noise Ratio (PSNR).

2. Research Methods

2.1. Interpolation Algorithms

Digital image interpolation is the process of generating a continuous intensity surface from a sample of discrete image data. There are many types of interpolation methods, each producing a different look to the final image. So, it is best if the quality, or the difference seen for each pixel, is maintained during the interpolation function [4].

In general, almost every geometric transformation such as translating, rotating, scaling, and warping requires interpolation to be carried out on an image. Such transformations are essential for commercial digital image processing software. The interpolated image quality is affected by several problems such as edge sharpness, freedom from artifacts, and high-frequency detail reconstruction [5].

The interpolation function is performed by convolutional operations which involve many additions and multiplication operations. Therefore, it is necessary to consider the computational complexity and image quality of the interpolated results [6]. The various interpolation algorithms discussed in this paper are Bicubic, Bilinear, and Nearest-neighbor.

2.1.1. Bicubic Interpolation

The Bicubic interpolation method considers the known 4x4 neighboring pixels totaling 16 pixels. Closer pixels are given a higher weight in calculations and are the ideal combination of processing time and output quality. For this reason, this method is standard in many image editing programs including Adobe Photoshop, printer drivers, and in-camera interpolation [7].

2.1.2. Bilinear Interpolation

Bilinear interpolation takes the weighted average of the four nearest neighboring pixels to compute the final interpolation value. The result is an image that is much smoother than the original image. When all known pixel distances are equal, the interpolation value is divided by four. This technique interpolates in both directions, horizontally and vertically [8].

2.1.3. Nearest Neighbour

The nearest neighbor method is the simplest method and requires the least processing time of all interpolation algorithms. The nearest neighbor selects the closest pixel value by rounding the coordinates of the desired interpolation point. Using this method one finds the closest suitable pixel in the source (original) image for each pixel in the destination image [9]. The new pixel is made equal to the other closest pixels. The pixels or dots of color are duplicated to create new pixels as the image grows. It creates pixels or edges that break the curve into steps or jagged edges. This form of interpolation produces the usually unacceptable effect of zooming in and out of an image [10].

2.2. Data Samples

This study, using input data from MESSIDOR (Methods to evaluate segmentation and indexing techniques in the field of retinal ophthalmology) [11]. Messidor is a research program funded by the French Ministry of Research and Defense within a 2004 TECHNO-VISION program. Messidor is an open-access database that can be used for research and educational purposes. Messidor database consists of 1200 eye fundus color digital images of the posterior pole, which were acquired by three ophthalmologic departments, using a color video 3CCD camera on a Topcon TRC NW6 non-mydriatic retina graph with a 45 degrees field of view. The images are saved in uncompressed TIFF format there is three kinds of image sizes at 1440x960, 2240x1488, or 2304x1536 pixels resolution. The fundus is part of the visual system, which is located at the back of the eye. The fundus camera is a device with special abilities that functions for fundus photographs, which can visualize the central part of the retina, macula, and optic disk. Data samples in the form of color retinal images with the RGB format (Red, Green, Blue) form a 3-

dimensional structure of 256 x 256 x 3 pixels. The RGB format was chosen because it retains original image information and in the absence of compression methods, making it easy to process.

2.3. Resizing Images

The original input image is resized to 256 x 256 x 3 size. The image size of 256 x 256 x 3 pixels chosen to reduce the complexity of the original image size. This input data format was obtained through the image resizing process using bicubic interpolation [12]. This method is selected because the result is finer at the edges than the bilinear interpolation. Bicubic is an ideal combination of process time and quality output. The bicubic interpolation uses a distance of 16 neighboring pixels (4x4) as sampling (S) to estimate the pixels in the (i', j') positions, as illustrated in Figure 1.
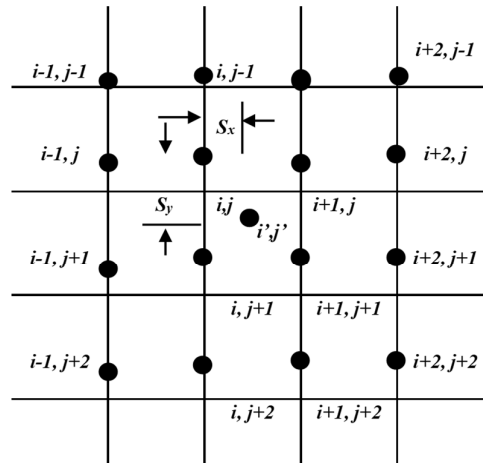

Figure 1. Sampling Interpolasi Bicubic at Position (i', j')

Bicubic Interpolation is calculated using equation 3.1. until 3.5. as follows:

$$f_{i,j} = \begin{bmatrix} W_{-1}(S_y) & W_0(S_y) & W_1(S_y) & W_2(S_y) \end{bmatrix} \begin{bmatrix} f_{i-1,j-1} & f_{i,j-1} & f_{i+1,j-1} & f_{i+2,j-1} \\ f_{i-1,j} & f_{i,j} & f_{i+1,j} & f_{i+2,j} \\ f_{i-1,j+1} & f_{i,j+1} & f_{i+1,j+1} & f_{i+2,j+1} \\ f_{i-1,j+2} & f_{i,j+2} & f_{i+1,j+2} & f_{i+2,j+2} \end{bmatrix} \begin{bmatrix} W_{-1} & (S_x) \\ W_0 & (S_x) \\ W_1 & (S_x) \\ W_2 & (S_x) \end{bmatrix}$$

(1)

Where:
S_y = $j' - j$, S_x $= i' - i$
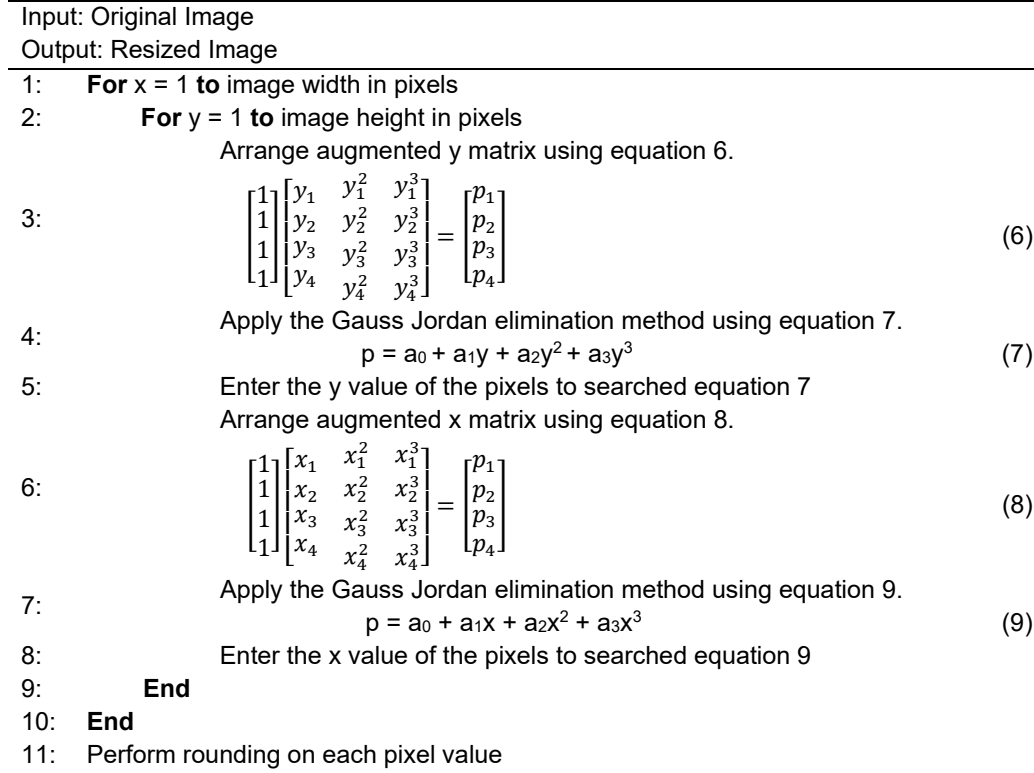$f_{i,j}$ =   pixel value at position (i, j)

$$W_{-1}(S) = \frac{-S^3 + 2S^2 - S}{2}$$ (2)

$$W_0(S) = \frac{3S^3 - 5S^2 + 2}{2}$$ (3)

$$W_1(S) = \frac{-3S^3 + 4S^2 + 2}{2}$$ (4)

$$W_2(S) = \frac{S^3 - S^2}{2}$$ (5)

Figure 2 is an algorithm of image resizing using the bicubic interpolation method:

Input: Original Image	
Output: Resized Image	

1:	**For** x = 1 **to** image width in pixels	
2:	**For** y = 1 **to** image height in pixels	
	Arrange augmented y matrix using equation 6.	
3:	$\begin{bmatrix}1\\1\\1\\1\end{bmatrix}\begin{bmatrix}y_1 & y_1^2 & y_1^3\\y_2 & y_2^2 & y_2^3\\y_3 & y_3^2 & y_3^3\\y_4 & y_4^2 & y_4^3\end{bmatrix} = \begin{bmatrix}p_1\\p_2\\p_3\\p_4\end{bmatrix}$	(6)
4:	Apply the Gauss Jordan elimination method using equation 7. $p = a_0 + a_1y + a_2y^2 + a_3y^3$	(7)
5:	Enter the y value of the pixels to searched equation 7	
	Arrange augmented x matrix using equation 8.	
6:	$\begin{bmatrix}1\\1\\1\\1\end{bmatrix}\begin{bmatrix}x_1 & x_1^2 & x_1^3\\x_2 & x_2^2 & x_2^3\\x_3 & x_3^2 & x_3^3\\x_4 & x_4^2 & x_4^3\end{bmatrix} = \begin{bmatrix}p_1\\p_2\\p_3\\p_4\end{bmatrix}$	(8)
7:	Apply the Gauss Jordan elimination method using equation 9. $p = a_0 + a_1x + a_2x^2 + a_3x^3$	(9)
8:	Enter the x value of the pixels to searched equation 9	
9:	**End**	
10:	**End**	
11:	Perform rounding on each pixel value	

Figure 2. Algorithm 1. Resizing The Image Using Bicubic Interpolation.

2.4. Results analysis

To analyze the results of the resizing process used by three parameters, the Mean Square Error (MSE), Root Mean Squared Error (RMSE), and Peak Signal-to-Noise Ratio (PSNR) to measure the similarity between two images [13]. These parameters are used to compare the results of resizing the image with the initial image or original image. MSE and RMSE do not have unit values while PSNR is had decibels (dB) as a unit value, more similar between two images, then MSE and RMSE values are getting closer to zero. Whereas in PSNR, two images have a low level of similarity if the PSNR value is below 30 dB. To calculate the three parameters using equation 10 until 12 as follows:

$$MSE = \frac{1}{m \times n} \sum_{i=0}^{n-1} \sum_{j=0}^{m-1} [f(i,j) - g(i,j)]^2 \qquad (10)$$

$$RMSE = \sqrt{MSE} \qquad (11)$$

$$PSNR = 10 \, log_{10} \frac{255^2}{MSE} \qquad (12)$$

where m is the image height, n is the image width, f(i, j) is the original image, and g(i, j) is the resizing image. The result of resizing 20 images from 1380x1380 pixels to 1024x1024 pixels using the bicubic interpolation method was measured based on the MSE, RMSE, and PSNR values and compared with the other two bilinear interpolation methods and the nearest neighbor interpolation method.

3. Result and Discussion

There are three kinds of resolution sizes from 20 input images, 1440 x 960 pixels, 2240 x 1488 pixels, and 2304 x 1536 pixels. Resizing input images is needed to reduce the complexity of input data. In this study, all input images were resized to 1024 x 1024 using Bicubic Interpolation. In the following paragraphs, the implementation code of the bicubic interpolation resizing algorithm. First, crop the original image to remove the left and right parts of the background image, more focused on the retinal image, and to reduce the complexity of the image process. The cropping process changes the original image size from 1440x960 pixels to 900x900 pixels, from 2240x1488 pixels to 1380x1380 pixels, and from 2304x1536 pixels to 1452x1452 pixels. After the cropping process, then the three sizes of cropped images are resizing to one dimension of 1024x1024 pixels.

Figure 3 shows an input image from the Messidor dataset in original size, cropped image, and resized image using a bicubic interpolation resizing algorithm. Figure 3 (a) shows the original image from the Messidor dataset. There are three image sizes of 1440x960, 2240x1488, and 2304x1536. Then figure 3 (b) shows the results of the cropping process to remove the black backgrounds on the right and left sides of the image while changing the width and height of the image to be the same so that the size is 900x900, 1380x1380, and 1452x1452. Figure 3 (c) shows the results of resizing the image to equalize three different image sizes into a uniform size of 1024x1024 pixels.
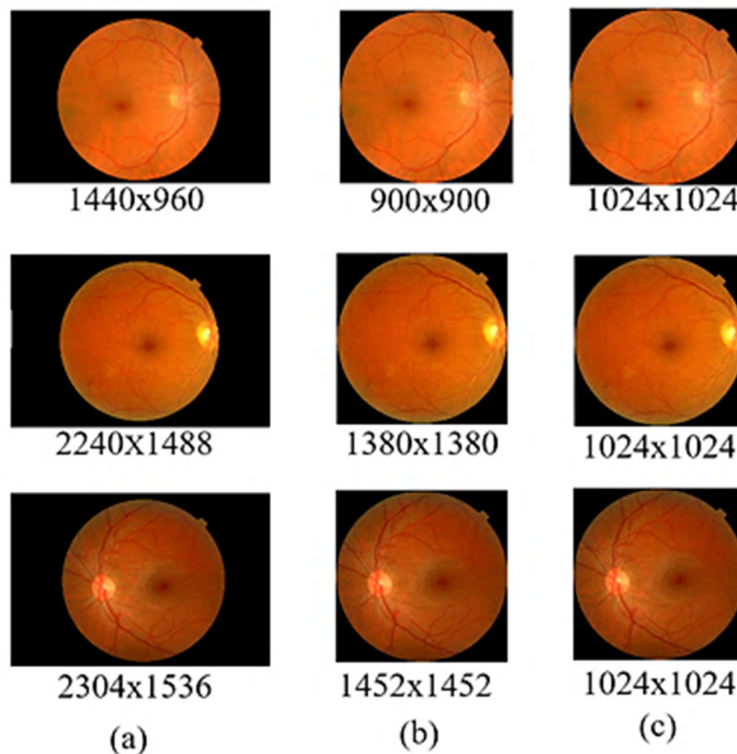

Figure 3. (a) Original Images, (b) Cropped Images, (c) Resized Images

Table 1. Shows the results of measuring the image resizing process using the bicubic interpolation of 20 color images, for each color channel Red (R), Green (G), and Blue (B) calculated MSE, RMSE, and PSNR indicators and averages (AVG) for all channels color. The calculation results of 20 color images from the Messidor database has been obtained. The average MSE value for the red channel is 0.706871, for the green channel is 0.646374, for the blue channel is 0.75686, and the MSE average of all channels is 0.703369.

Table 1. Measurement Results of Image Resizing Uses The Bicubic Interpolation Method

| Img	MSE (R)	MSE (G)	MSE (B)	MSE AVG	RMSE (R)	RMSE (G)	RMSE (B)	RMSE AVG	PSNR (R)	PSNR (G)	PSNR (B)	PSNR AVG
1	1.415375	1.288405	1.507250	**1.403677**	1.189695	1.135079	1.227701	**1.184159**	46.65608	47.06427	46.38295	**46.70110**
2	0.400840	0.359535	0.408816	**0.389730**	0.633119	0.599612	0.639387	**0.624040**	52.13509	52.60739	52.04952	**52.26400**
3	1.011253	0.879849	1.059904	**0.983669**	1.005611	0.938003	1.029516	**0.991043**	48.11620	48.72072	47.91213	**48.24968**
4	0.854892	0.888352	1.045635	**0.929626**	0.924604	0.942524	1.022563	**0.963230**	48.84568	48.67895	47.97100	**48.49854**
5	0.878629	0.902025	1.056194	**0.945616**	0.937352	0.949750	1.027713	**0.971605**	48.72674	48.61261	47.92736	**48.42224**
6	0.417796	0.383007	0.433728	**0.411510**	0.646371	0.618876	0.658580	**0.641276**	51.95516	52.33273	51.79263	**52.02684**
7	0.413395	0.372707	0.425194	**0.403765**	0.642958	0.610497	0.652069	**0.635175**	52.00114	52.45113	51.87893	**52.11040**
8	0.411542	0.375713	0.425749	**0.404335**	0.641516	0.612954	0.652494	**0.635655**	52.02066	52.41624	51.87327	**52.10339**
9	0.419546	0.385057	0.435997	**0.413533**	0.647724	0.620529	0.660301	**0.642851**	51.93701	52.30955	51.76996	**52.00551**
10	0.877098	0.782021	0.930122	**0.863080**	0.936535	0.884319	0.964428	**0.928428**	48.73432	49.23262	48.47940	**48.81545**
11	0.879056	0.784515	0.924932	**0.862834**	0.937580	0.885729	0.961734	**0.928347**	48.72463	49.21878	48.50370	**48.81571**
12	0.405582	0.370573	0.421335	**0.399164**	0.636853	0.608747	0.649103	**0.631568**	52.08401	52.47606	51.91852	**52.15953**
13	0.401645	0.365346	0.414830	**0.393940**	0.633754	0.604438	0.644073	**0.627422**	52.12638	52.53776	51.98610	**52.21675**
14	0.398395	0.358503	0.417813	**0.391570**	0.631185	0.598751	0.646385	**0.62544**	52.16166	52.61987	51.95498	**52.24551**
15	0.414368	0.377524	0.431897	**0.407930**	0.643714	0.61443	0.657188	**0.638444**	51.99093	52.39536	51.81100	**52.06576**
16	1.446954	1.306226	1.546468	**1.433216**	1.202894	1.142902	1.243571	**1.196456**	46.56025	47.00462	46.27139	**46.61209**
17	0.851716	0.758841	0.914649	**0.841735**	0.922885	0.871115	0.956373	**0.916791**	48.86185	49.36329	48.55226	**48.92580**
18	0.890620	0.789465	0.934090	**0.871392**	0.943727	0.888518	0.966483	**0.932909**	48.66787	49.19147	48.46091	**48.77342**
19	0.665396	0.593105	0.693472	**0.650658**	0.815718	0.770133	0.832750	**0.806200**	49.93399	50.43348	49.75451	**50.04066**
20	0.683327	0.606714	0.709133	**0.666391**	0.826636	0.778919	0.842100	**0.815885**	49.81852	50.33496	49.65753	**49.93700**
Avg	**0.706871**	**0.646374**	**0.756860**	**0.703369**	**0.820022**	**0.783791**	**0.846726**	**0.816846**	**50.10291**	**50.50009**	**49.84540**	**50.14947**

Figure 4 shows a graph of (a) Mean Square Error (MSE), (b) Root Mean Squared Error (RMSE), and (c) Peak Signal-to-Noise Ratio (PSNR) of 20 images. The blue line is the value MSE from the resizing process using the bicubic interpolation method, the orange color line is the MSE value from the resizing result using the bilinear interpolation method, and the green color line is the MSE value from the resizing result using the nearest neighbor method. Based on the three graphs in figure 4, the bicubic interpolation method has the smallest error level for image resizing compared to the other two methods, bilinear interpolation, and nearest-neighbor interpolation.
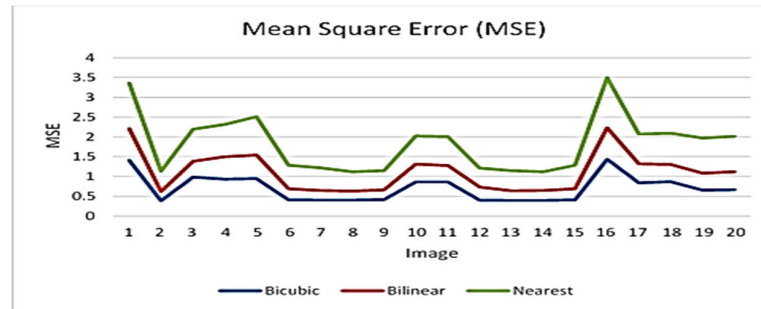
(a)
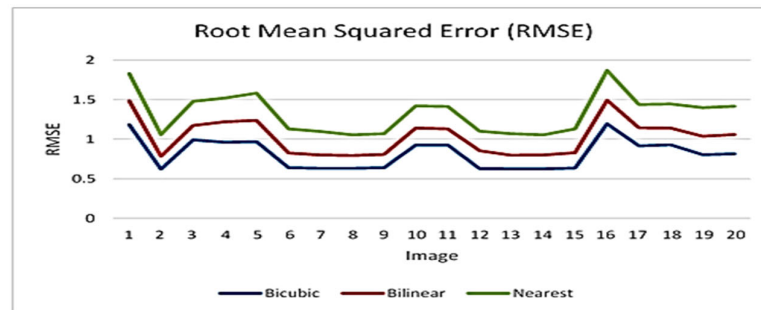
(b)
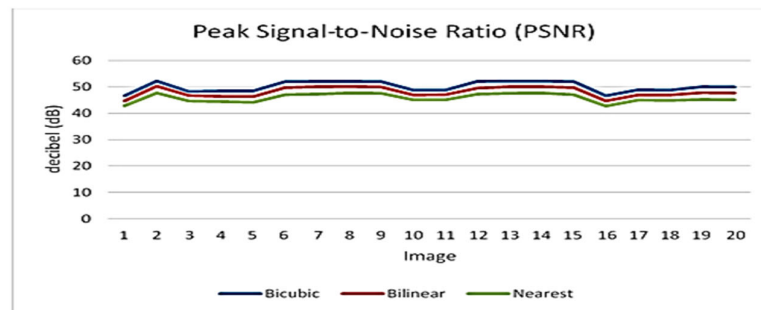
(c)

Figure 4. The Value of (a) MSE, (b) RMSE, and (c) PSNR of 3 Methods of Image

Table 2 shows the average MSE value, the difference between bilinear interpolation and bicubic interpolation method is 0.4079402, the difference between the bicubic interpolation method and the nearest neighbor interpolation is 1.1331529. On the average RMSE value, the difference between the bicubic interpolation method and bilinear interpolation is 0.21283, the difference between the bicubic interpolation method and the nearest neighbor interpolation is 0.3010377. Whereas in the average PSNR value, the difference between the bicubic interpolation method and the bilinear interpolation method is -2.036891 dB and the difference between the bicubic interpolation and the nearest neighbor interpolation method is -4.309458.

Table 2. The Comparison of Three Resizing Methods

| Img	AVG MSE			AVG RMSE			AVG PSNR		
	Bicubic	Bilinear	Nearest	Bicubic	Bilinear	Nearest	Bicubic	Bilinear	Nearest
1	1.403677	0.8059487	1.9547068	1.184159	0.301988	0.3461743	46.70110	-1.975558	-3.795064
2	0.389730	0.2296074	0.7413119	0.624040	0.162343	0.2744503	52.26400	-2.005375	-4.590311
3	0.983669	0.4019058	1.2128898	0.991043	0.185369	0.3049173	48.24968	-1.490873	-3.493563
4	0.929626	0.5654474	1.3868259	0.963230	0.259268	0.2991764	48.49854	-2.077049	-3.978363
5	0.945616	0.5945904	1.5624762	0.971605	0.269234	0.3424178	48.42224	-2.130268	-4.246046
6	0.411510	0.2766394	0.8710738	0.641276	0.187954	0.3018655	52.02684	-2.232241	-4.920887
7	0.403765	0.2396569	0.8123945	0.635175	0.166638	0.2993258	52.11040	-2.023569	-4.769308
8	0.404335	0.2286862	0.7162258	0.635655	0.159547	0.2615503	52.10339	-1.943547	-4.403010
9	0.413533	0.2451334	0.7357309	0.642851	0.168279	0.2592482	52.00551	-2.017735	-4.417902
10	0.863080	0.4466805	1.1574013	0.928428	0.215375	0.2771476	48.81545	-1.812700	-3.699264
11	0.862834	0.4162830	1.1466124	0.928347	0.202138	0.2863876	48.81571	-1.712348	-3.673422
12	0.399164	0.3360404	0.8161617	0.631568	0.225864	0.2446610	52.15953	-2.658636	-4.836504
13	0.393940	0.2459119	0.7568650	0.627422	0.172370	0.2717054	52.21675	-2.110250	-4.641335
14	0.391570	0.2513219	0.7303810	0.625440	0.176180	0.2567653	52.24551	-2.158067	-4.566842
15	0.407930	0.2815349	0.8778774	0.638444	0.191813	0.3024558	52.06576	-2.284359	-4.973836
16	1.433216	0.7965692	2.0646900	1.196456	0.296307	0.3772475	46.61209	-1.924254	-3.882967
17	0.841735	0.4749181	1.2288910	0.916791	0.230226	0.2916622	48.92580	-1.949029	-3.918526
18	0.871392	0.4349321	1.2224564	0.932909	0.209539	0.3040186	48.77342	-1.761525	-3.811545
19	0.650658	0.4311360	1.3194653	0.806200	0.233232	0.3609993	50.04066	-2.206297	-4.781401
20	0.666391	0.4558605	1.3486206	0.815885	0.242931	0.3585785	49.93700	-2.264133	-4.789055
Average		0.4079402	1.1331529		0.212830	0.3010377		-2.036891	-4.309458

From Table 2, it can conclude that the error rate of the bicubic interpolation method is lower than the bilinear interpolation method and the nearest neighbor interpolation for image resizing cases, so this research uses the bicubic interpolation method.

Determination of the input image size of 256 x 256 pixels, based on the measurement results of MSE, RMSE, and PNSR values from 20 color images of size 1380 x 1380 pixels, each resizing into 4 different sizes 28x28, 64x64, 128x128, and 256x256 using a bicubic interpolation method. Figure 5 (a) shows a graph of the Mean Square Error (MSE) results of resizing 20 image samples into 4 different image sizes using the bicubic interpolation method. The yellow line is the MSE value of 20 color images resizing into 256x256 pixels are best because the Mean Square Error (MSE) value is the smallest.

Figure 5 (b) shows the graph of the value of Root Mean Square Error (RMSE) results of resizing 20 color images into 4 different image sizes using the bicubic interpolation method. The yellow line is the RMSE value of 20 color images resizing into 256x256 pixels are best because the Root Mean Square Error (RMSE) value is the smallest. Whereas Figure 5 (c) shows a graph of Peak Signal-to-Noise Ratio (PSNR) results of resizing 20 color images into 4 different image sizes using

the bicubic interpolation method. The yellow line is the PNSR value of 20 image samples resized into a size of 256x256 pixels are best because the Root Mean Square Error (RMSE) value is the largest, meaning that the resized image has the highest level of similarity with the original image.

(a)

(b)

(c)
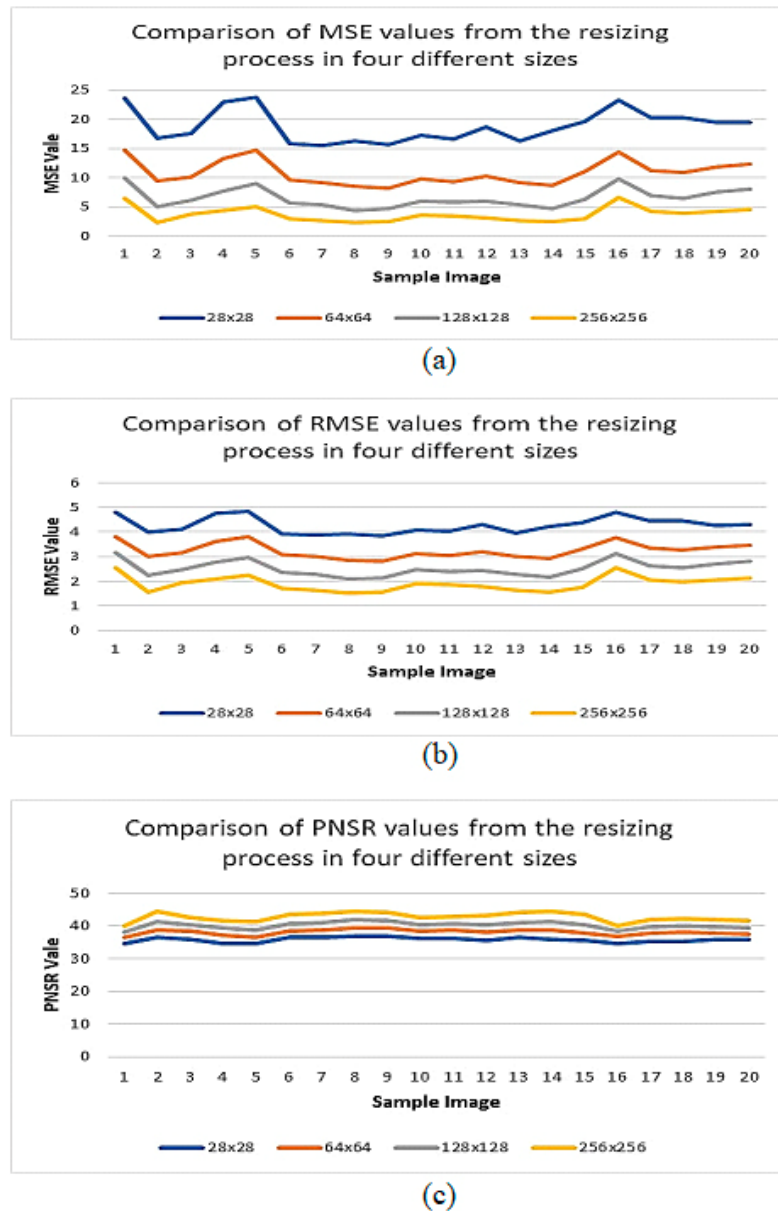
Figure 5. The Value of (a) MSE, (b) RMSE, and (c) PSNR of Image Resizing Into 4 Different Sizes Using Bicubic Interpolation Method

4. Conclusion

The conclusion of the research on image resizing analysis using the Bibubic interpolation method is that the results of the analysis of the experimental results show that first, the Bicubic interpolation method produces a much sharper image than the bilinear interpolation method and the closest neighbors. Second, The bicubic interpolation method has the smallest error rate for image resizing compared to the other two methods, namely bilinear interpolation, and nearest-

neighbor interpolation. Third, the larger the dimensions of the resized image, the higher the level of similarity to the original image, however, the level of computational complexity also increases.

References

[1] D. Vaquero, M. Turk, K. Pulli, M. Tico, and N. Gelfand, "A survey of image retargeting techniques", *SPIE Optical Engineering+ Applications*, pp. 779814-779814, Aug. 2010.

[2] I.N.T.A. Putraa, and E.D. Krisnaa, "Implementasi Sistem Surveillance Berbasis Pengenalan Wajah pada STMIK STIKOM Indonesia", *Jurnal Ilmu Komputer*, Vol. 8, No 2, pp. 65-72, 2015.

[3] R. C. Gonzalez, Digital image processing. Pearson Education India; 2009.

[4] V. Patel and K. Mistree, "A Review on Different Image Interpolation Techniques for Image Enhancement", *International Journal of Emerging Technology and Advanced Engineering*, Vol. 3, pp. 129-13, Dec. 2013.

[5] S.D. Ruikar and D.D. Doye, "Image Denoising using Tri Nonlinear and Nearest Neighbour Interpolation with Wavelet Transform", *International Journal of Information Technology and Computer Science*, Vol.4, pp. 36-44, Aug.2012.

[6] C. Singh, S. Singh, R. Saini, and A. K Saini, "A Comparative Analysis of Image Scaling Algorithms". *International Journal of Image, Graphics and Signal Processing*, pp.55-62, April. 2013.

[7] P. S. Parsania and P. V. Virparia, "A Review: Image Interpolation Techniques for Image Scaling", *International Journal of Innovative Research in Computer and Communication Engineering*, Vol. 2, Issue 12, pp. 7409-7413, Dec. 2014.

[8] A. Prajapati, S. Naik and S. Mehta, "Evaluation of Different Image Interpolation Algorithms", *International Journal of Computer Applications*, Vol.58, Issue.1, pp. 6-12, Jan. 2012.

[9] P. Miklo., "Image interpolation techniques", *In 2nd SiberianHungarian Joint Symposium on Intelligent Systems*. Oct. 2004.

[10]
. Patel and K. Mistree, "A Review on Different Image Interpolation Techniques for Image Enhancement", *International Journal of Emerging Technology and Advanced Engineering*, Vol. 3, pp. 129-13, Dec. 2013.

[11] Messidor, "Methods to evaluate segmentation and indexing techniques in the field of retinal ophthalmology". 2004, http://www.adcis.net/en/third-party/messidor/ , [Access Date: January 14, 2021].

[12] J. Titus and S. Geroge. "A comparison study on different interpolation methods based on satellite images", *International Journal of Engineering Research & Technology*, Vol 2, Issue 6, pp. 82-85, 2013.

[13] H.S. Prashanth, H.L. Shashidhara, and B.M. KN. "Image scaling comparison using universal image quality index". *In 2009 International Conference on Advances in Computing, Control, and Telecommunication Technologies*, Bangalore, India, 2009, pp. 859-863.