Study and Comparison of Image Scaling Algorithms

Fei Yang[1,a,*]

[1]*School of Materials Science and Engineering, School of Tiangong University, 300387, Tianjin, China*
a. F2819875628@outlook.com
**corresponding author*

Abstract: Image scaling is a fundamental task in digital image processing and computer vision, which involves adjusting the size of an image to fit different display devices or storage needs. In this process, there is a trade-off between processing time and the smoothness and sharpness of the target. The purpose of image scaling algorithms is to solve the display of images on different resolutions. Current image scaling algorithms are not only traditional interpolation algorithms, but also algorithms centered on computer vision and content-aware techniques. This article aims at image scaling, and several common interpolation algorithms: nearest neighbor interpolation, bilinear interpolation, bicubic interpolation algorithm, and content-aware Seam Carving algorithm based on content-aware algorithms, and analyzes and compares the performance of these algorithms at multiple levels. Different interpolation formulas are summarised and simulation experiments are performed. Set to the same scaling ratio. Analyse and compare the details, sharpness, speed, and edges of the images of different algorithms after simulation. Discuss the advantages and disadvantages of their operation in different environments

Keywords: Nearest neighbor interpolation, bilinear interpolation algorithm, seam carving algorithm.

1. Introduction

Image scaling is the process of resizing digital images in image processing and computer graphics [1]. Image scaling is a key technology in the field of image processing, which is widely used in computer vision [2], image editing [3], satellite remote sensing, and digital media in a number of fields [4]. With the continuous development of the times, the resolution of modern digital devices is constantly improving, the application of the environment is also more complex, and the requirements for image scaling more complex and diverse. In different environments, it is not only necessary to enlarge the size of the picture, but also to ensure the clarity of the photo and the integrity of the key content. Therefore, it is crucial to choose an appropriate image scaling algorithm in different environments.

Currently, the traditional interpolated image scaling methods include proximity interpolation, bilinear interpolation [5-7], and bicubic interpolation [8][9]. Proximity difference is the simplest to compute, the speed of the operation is faster, but the quality of the generated image is lower, the edges of the image of the bilinear interpolation method will be blurred, and the double cubic interpolation has higher detailing and quality of the image, but it is more complicated to compute. These methods go through the scaling of the image by calculating the values of the pixel points. There is another one is based on content-awareness, represented by Seam Carving algorithm and so on [10][11], this kind of algorithm scales according to the content of the image to ensure the retention of more important information about the image features, suitable for uneven scaling or cropping of the graphics, because of the more complex computation, so the efficiency of a certain limitation, commonly used in intelligent scaling [12][13]. The article explains the principles of several traditional interpolation scaling algorithms as well as seam carving algorithms and pairs the images processed by different algorithms. It illustrates the differences between the algorithms in various aspects such as quality, efficiency and principle in scaling. Thus, the appropriate algorithm can be selected according to different application environments.2 Interpolation algorithm for image scaling

2. Comparison of different image scaling algorithms

2.1. Nearest neighbor interpolation

Nearest Neighbor Interpolation is a simple image interpolation algorithm. It is mainly used to generate high-resolution images from low-resolution images, or to estimate the value of a particular location from the interpolation points of an irregular grid. The basic function is, for the need to interpolate the point to find it around the proximity of the known points, and then according to the value of the known points, to be used as the interpolation point value.

Calculation formula and principle

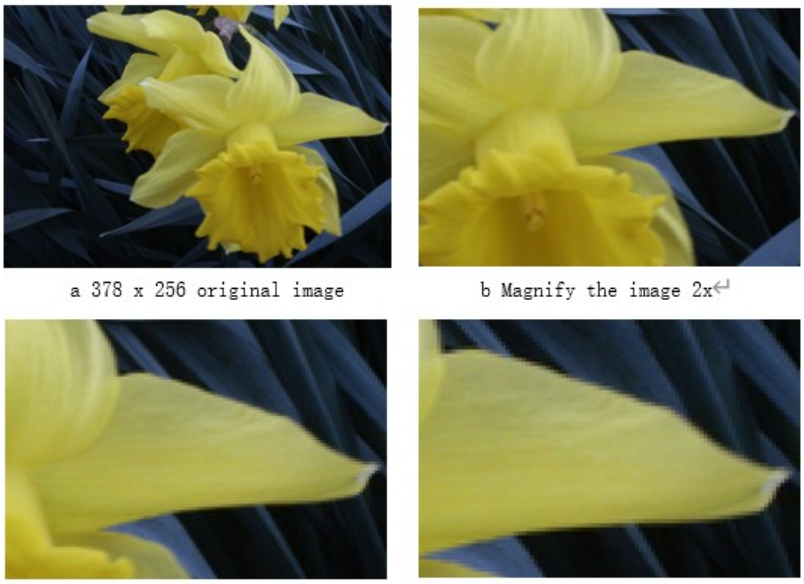
Suppose a two-dimensional grid with interpolated points at coordinates (x,y), and the data points are known to be distributed on the grid.

Then according to the formula $x' = \mathrm{round}(x)$

$$y' = \mathrm{round}(y)$$

The function is a rounding function, it will de-round the data to get the value of the nearest known pixel point (x', y'). Then the value of the known pixel point (x', y') is directly assigned to the desired interpolation point (x, y). The result of the interpolation is $f(x,y)=f(x',y')$.

The following is a simulation of the image in the python language environment



a 378 x 256 original image

b Magnify the image 2x

c Magnify the image 3x

d Magnify the image 4x

Figure 1: Image after processing by nearest neighbor interpolation algorithm.

Figure 1 is the nearest-neighbor interpolation algorithm according to different multiples of the image after the enlargement of the image according to the simulation of the image of the picture found in the picture with the nearest-neighbor interpolation enlargement, the edges of the picture and the details of the part of the picture becomes not clear, in the enlargement of 4 times after the edges of the flower petals appeared in the ‘jagged’ or ‘block’ effect, the edges are not smooth. The edges of the petals appear ‘jagged’ or ‘blocky’ after zooming in 4 times, and the edges are not smooth. The image appears rough when zoomed in. Since nearest-neighbor interpolation does not require complex calculations, it is very efficient in image scaling and is suitable for real-time applications when dealing with large batches of images.

2.2. bilinear interpolation

Bilinear interpolation is a commonly used interpolation method, used in the two-place plane for a given value of four points to find its interpolation, bilinear interpolation method of two directions (x direction and y direction) linear interpolation, to get the value of the interpolated point, he compared to the nearest-neighbor interpolation algorithms, to get a smoother image.

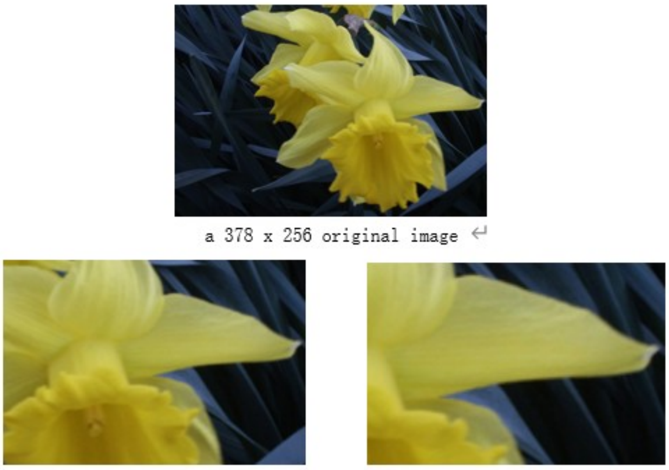
Bilinear interpolation is an extension of unilinear interpolation, it now goes in one dimension and then interpolates in the other dimension. Now suppose there is a rectangular grid with four known points on the grid, and then go through those four points to compute the value of the internal floating point.

Let the values of the four points be (x1,y1), f(x1,y1)=H1,(x2,y1), f(x2,y1)=H2, (x1,y2), f(x1,y2)=H3, (x2,y2), f(x2,y2)=H4

The corresponding pixel values are H1, H2., H3, H4 calculated as

$$H = \frac{(x_2-x)(y_2-y)}{(x_2-x_1)(y_2-y_1)}H1 + \frac{(x-x_1)(y_2-y)}{(x_2-x_1)(y_2-y_1)}H2 + \frac{(x_2-x)(y-y_1)}{(x_2-x_1)(y_2-y_1)}H3 + \frac{(x-x_1)(y-y_1)}{(x_2-x_1)(y_2-y_1)}H4 \tag{1}$$

With this formula you can get the value of the floating point. The physical meaning of this formula is to calculate the weighted average of the four points. The weights are based on the distance of the floating point from these four points. Closer distance contributes more to the interpolation of the floating point, while farther distance contributes less to the floating point.



a 378 x 256 original image ↵

b Magnify the image 2x

c Magnify the image 3x↵

Figure 2: Image after processing by bilinear interpolation method.

Figure 2 shows the image after zooming the original image of 378×256 with bilinear interpolation in python language environment. It can be seen that the bilinear edges are smoothed better compared to the nearest neighbor interpolation algorithm. The nearest neighbor interpolation algorithm causes a jagged effect on the edges of the image during enlargement, whereas bilinear interpolation solves this problem by averaging the weights, making the edges of the image smoother. But when the image is greatly enlarged, it leads to loss of sharpness and loss of details in the image. Compared with other complex algorithms The bilinear interpolation algorithm is less accurate for details and edges. A blurring effect may appear in some areas with complex details. And it can only be applied to 2D plane. The double triple interpolation algorithm is a more complex quadratic interpolation algorithm, which not only uses the values of the four surrounding points when calculating the interpolation point, but also makes use of the values of the 16 points around the interpolation point, which retains more details and makes the image clearer than the quadratic and nearest-neighbor interpolation methods.

2.3. Double Cubic Interpolation

Double Cubic Interpolation is based on going for cubic interpolation in both directions. The main principle is to process the data through a cubic polynomial to find the target interpolation, which uses not only the value of the pixel point, but also the derivative value of the pixel point, which makes the resulting image smoother. The basic idea is to perform the cubic interpolation in the X and Y directions to obtain the weights in the X and Y directions, and then calculate the value of the target interpolated point according to the formula.

The target point is T (x, y) and the values of the 16 known points around it are f (i, j)

Calculate the weights of the points in the X and Y directions.

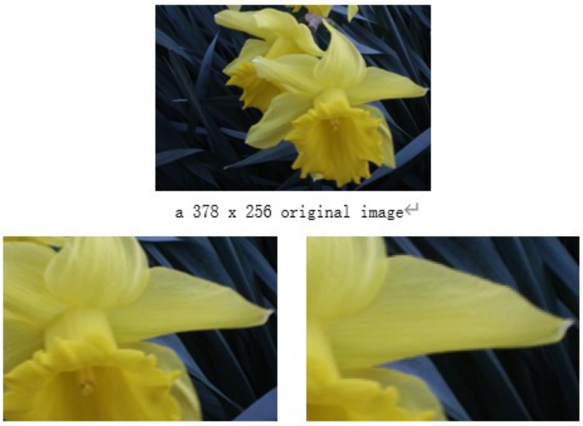
$$W(x) = \begin{cases} (a+2)|x|^3 - (a+3)|x|^2 + 1 & for|x| \leq 1 \\ a|x|^3 - 5a|x|^2 + 8a|x| - 4a & for 1 \leq |x| \leq 2 \\ 0 & others \end{cases} \tag{2}$$

X:Distance of target pixel point from neighboring pixel points a=0.5

The final interpolated value is obtained according to the double cubic interpolation formula

$$f(x,y) = \sum_{i=-1}^{2} \sum_{j=-1}^{2} f(x_i, y_j) \cdot W(x - x_i) \cdot W(y - y_j) \tag{3}$$

Where, W(x-x_i)·W(y-y_j) is the weight function obtained from the cubic interpolation based formula.



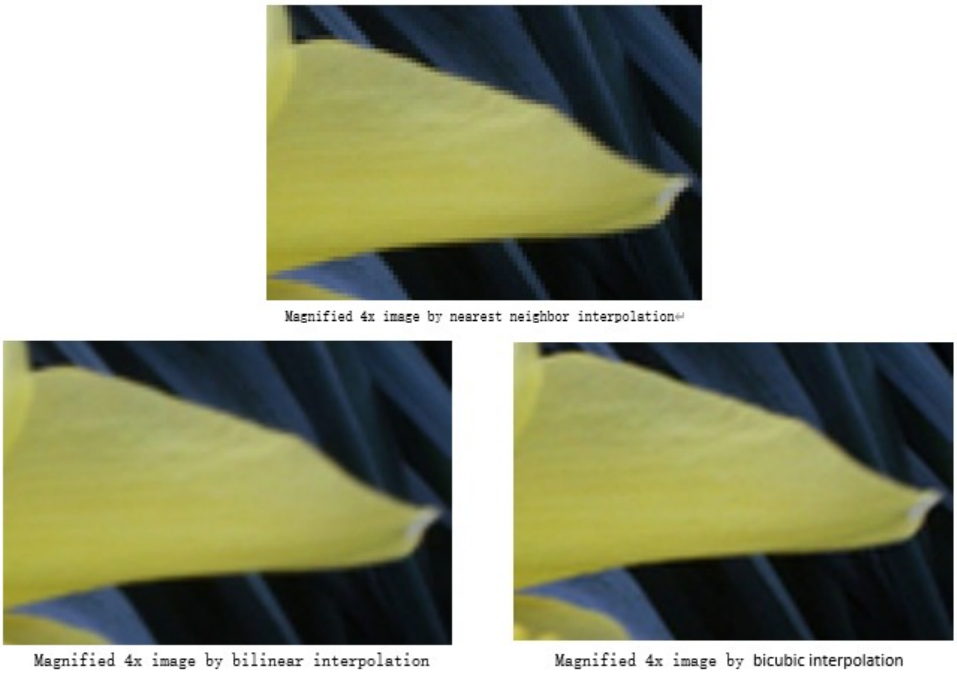
a 378 x 256 original image

b Magnify the image 2x

c Magnify the image 3x

Figure 3: Image after processing using bicubic linear interpolation method.

Figure 3 shows the image after zooming the original image of 378×256 using bicubic linear interpolation in python language environment. Compared to the bilinear interpolation, the bicubic interpolation method estimates the new pixels by considering the surrounding 16 pixels, which produces a smoother and more detailed image than the bilinear interpolation and maintains the details of the image in a better way and reduces the blurred linearity in the interpolation process. It is even better when dealing with high contrast edges and complex structures.



Magnified 4x image by nearest neighbor interpolation

Magnified 4x image by bilinear interpolation

Magnified 4x image by bicubic interpolation

Figure 4: Images are processed by different interpolation algorithms.

Table 1: Running time for each calculation

interpolation algorithm	**CPU usage time(seconds)**
Nearest neighbour interpolation	0.02076560
bilinear new interpolation	0.00217880
double cubic interpolation	0.00363280

Figure 4 shows the image quadrupled using three different interpolation algorithms to magnify the image. Table 1 shows the cpu time taken by each algorithm to scale the image four times. The results produced when scaling the image using different algorithms are different. For Nearest Neighbor Interpolation, the computational complexity is low and the speed of computation is the fastest among the three interpolation algorithms, based on the comparison of the above fig, it can be found that the edges are prone to block and mosaic effects, which makes it suitable for use in places where a quick preview is required for low quality. Bilinear interpolation has a medium computation time, compared to the images processed by the nearest neighbor interpolation algorithm, the edges are smoother and the mosaic effect is reduced, but it is prone to problems such as missing details and blurred edges for processing images with higher magnification. It is suitable for applications that balance efficiency and quality. The image processed by Double Cubic Interpolation has clear edges and retains more details of the image, but the computational complexity is high and is suitable for processing high-quality images with complex details.

2.4. Seam carving algorithm

The seam carving algorithm is a content-aware [14] image algorithm that is able to preserve as much as possible the important parts of the image during image scaling, unlike the traditional interpolation algorithm method, seam carving mainly resizes the image by removing pixels with small energy to avoid the loss of important feature information.

calculation step

Calculate the energy diagram

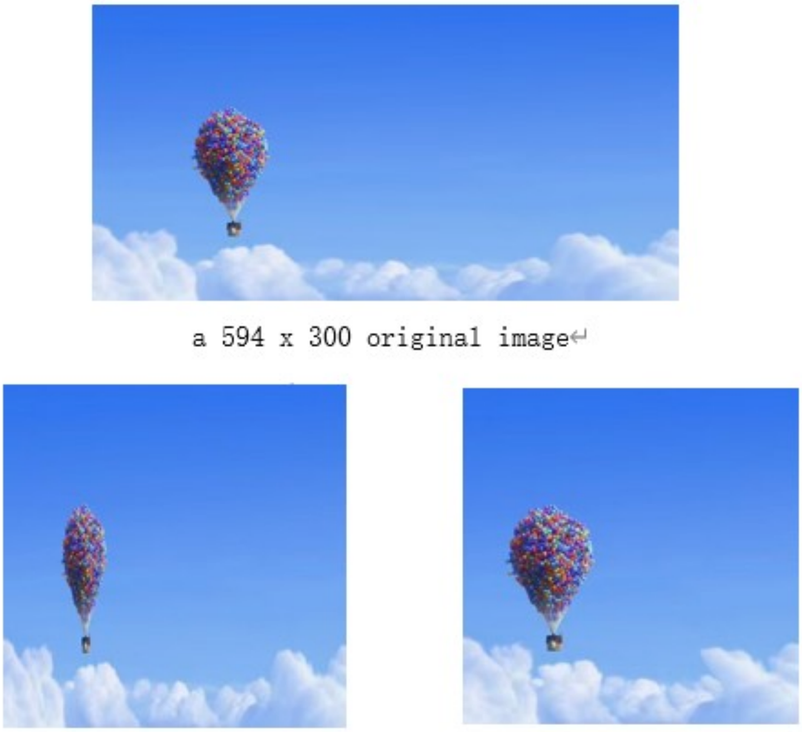
The Seam carving algorithm now calculates an ‘energy value’ for each pixel, which represents the importance of the area, with high energy areas representing important details and features of the image, and low energy areas being unimportant parts.

The energy function is given by Eq.

$$E(x,y) = |\frac{\partial I(x,y)}{\partial x}| + |\frac{\partial I(x,y)}{\partial y}| \tag{4}$$

$E(x,y)$ is the energy value of the pixel point P(x,y)

The resulting energy matrix is then used to calculate the lowest energy path, and then seam is removed or inserted as needed, and the image width is increased or decreased by one pixel.



a 594 x 300 original image

b Images processed by the double cubic linear algorithm

c Images processed by the double cubic linear algorithm

Figure 5: Image after processing by the double cubic linear algorithm and seam carving algorithm

Figure 5 shows the image after scaling the original 594×300 image using the seam carving algorithm and bilinear algorithm, it can be observed that the bilinear processed image just compresses the width of the image and deforms the hot air balloon in the image. Seam carving processed the image retains the main feature in the image (the hot air balloon) and deletes the pixels with low energy values The image was processed by Seam carving to retain the main feature (hot air balloon) in the image and remove pixels with low energy values. Compared to traditional interpolative scaling seam carving is able to intelligently recognise the image retaining more important areas of the image, seam carving resizing is flexible and can be adjusted according to the content of the image without scaling the image proportionally. However, the computational complexity is too high and the time required is too long. If the image is more complex and there is more than one important feature in the image, seam carving algorithm may misjudge and result in poor image processing.

3. Conclusion

Proximity difference calculation is the simplest and fastest, but the quality of the generated image is lower in the image edges will appear jagged and mosaic phenomenon. The bilinear interpolation method image will have blurred edges at high magnification. Bicubic interpolation has higher detail processing and image quality, but the calculation is more complex. These methods do the scaling of the image by calculating the values of the pixel points. Seam Carving algorithm based on content-awareness, this algorithm scales according to the content of the image to ensure that more important image feature information is retained, suitable for uneven scaling or cropping graphics, because the calculation is more complex, so the efficiency has some limitations, and is often used for intelligent scaling. Bilinear difference algorithm compared to other interpolation algorithms image quality is high processing speed, so the algorithm often used in image scaling. the main purpose of Seam Carving algorithm is to retain the main feature information in the image so that the main features in the image are not distorted in the image scaling.

This paper explains the principles of the nearest neighbor interpolation algorithm, bilinear interpolation algorithm, bicubic interpolation algorithm, and seam carving algorithm for content-aware algorithms, and performs simulations, compares the difference between different algorithms for processing images, as well as describes the permutation of each algorithm applicable. In the future, image scaling algorithms will be developed towards artificial intelligence and deep learning, which can have better efficiency and quality in processing complex images.

References

[1] *Lu, R., Jiang, L., & Yang, F. (2011). Research and FPGA design of image scaling. Computer Technology and Development, 12, 205-208.*

[2] *Lai, Q., & Di, L. (2023). Image scaling algorithm based on pixel difference network and edge direction interpolation. Applied Science and Technology, 50(3), 108-115.*

[3] *Hu, M., & Tao, S. (2021). Research on image scaling method based on saliency map and principal component analysis. Journal of Luoyang Institute of Technology (Natural Science Edition), 31(1), 64-70.*

[4] *Huo, D., Peng, L., & Yang, G. (2023). Implementation of FPGA-based real-time video image scaling system. Computer and Information Technology, 31(1), 25-28.*

[5] *Bai, X. (2014). Bilinear interpolation algorithm and its parallel implementation. Computer Knowledge and Technology, 9, 6494-6496.*

[6] *Lu, Y., & Fang, L. (2024). Single target detection algorithm based on bilinear interpolation. Fire and Command and Control, 49(1), 73-79.*

[7] *Gao, C., Zhou, R.-G., & Li, X. (2023). Quantum color image scaling based on bilinear interpolation. Chinese Physics B, 32(5), 235-244.*

[8] *Chen, J., & Ye, S. (2021). Bidirectional seam cropping image retargeting based on Seam Carving. Journal of Fuzhou University (Natural Science Edition), 49(2), 163-169.*

[9] *Cai, S., Zhou, R.-G., Luo, J., et al. (2024). Integer multiple quantum image scaling based on NEQR and bicubic interpolation. Chinese Physics B, 33(4), 259-273.*

[10] *Yang, J. (2019). Seam Carving image scaling algorithm combined with edge detection. Journal of Yichun College, 41(3), 35-38.*

[11] *Zhai, S., Meng, F., & Liu, J. (2018). An improved Seam Carving image redirection algorithm. Journal of Hubei College of Nationalities (Natural Science Edition), 36(2), 194-198.*

[12] *Peng, Y., Hu, D., Liu, Y., et al. (2019). Ferguson double cubic surface interpolation image scaling algorithm based on FPGA implementation. Journal of Guizhou University (Natural Science Edition), 36(6), 68-72.*

[13] Zhu, M., Peng, X., & Chang, J. (2023). Design of nine-point interpolation adaptive image scaling algorithm based on FPGA. Liquid Crystal and Display, 38(8), 1075-1083.

[14] Chen, S. E., & Yang, V. (2017). Research on content-aware image scaling algorithm based on seam carving. Computer Knowledge and Technology, 13(22), 190-192.