



10-22-2018

Image Zooming using Corner Matching

Ronald Marsh

University of North Dakota, ronald.marsh@UND.edu

Md Nurul Amin

m.amin@ndus.edu

Charles Crandall

Raymond Davis

Follow this and additional works at: <https://commons.und.edu/cs-fac>



Part of the [Computer Engineering Commons](#)

Recommended Citation

Marsh, Ronald; Amin, Md Nurul; Crandall, Charles; and Davis, Raymond, "Image Zooming using Corner Matching" (2018).

Computer Science Faculty Publications. 20.

<https://commons.und.edu/cs-fac/20>

This Conference Proceeding is brought to you for free and open access by the Department of Computer Science at UND Scholarly Commons. It has been accepted for inclusion in Computer Science Faculty Publications by an authorized administrator of UND Scholarly Commons. For more information, please contact zeineb.yousif@library.und.edu.

Image Zooming using Corner Matching

Ronald Marsh¹, Md Nurul Amin¹, Charles Crandall², Raymond Davis¹

1. University of North Dakota

2. Minnesota State University Moorhead rmarsh@cs.und.edu, m.amin@ndus.edu,
11ccrandall@gmail.com, raymond.davis@und.edu

Abstract-This work was intended to direct the choice of an image interpolation/zoom algorithm for use in UND's Open Prototype for Educational Nanosats (OPEN) satellite program. Whether intended for a space-borne platform or a balloon-borne platform, we expect to use a low cost camera (Raspberry Pi) and expect to have very limited bandwidth for image transmission. However, the technique developed could be used for any imaging application. The approach developed analyzes overlapping 3x3 blocks of pixels looking for "L" patterns that suggest the center pixel should be changed such that a triangle pattern results. We compare this approach against different types of single-frame image interpolation algorithms, such as zero-order-hold (ZOH), bilinear, bicubic, and the directional cubic convolution interpolation (DCCI) approach. We use the peak signal-to-noise ratio (PSNR) and mean squared error (MSE) as the primary means of comparison. In all but one of the test cases the proposed method resulted in a lower MSE and higher PSNR than the other methods. Meaning this method results in a more accurate image after zooming than the other methods.

I. INTRODUCTION

CubeSats [1] have generated significant interest due to their low cost and ease of launch vehicle integration. However, despite their small size (approximately 10 cm x 10 cm x 10 cm and a mass of 1.33 kg), CubeSats contain numerous systems and subsystems including mechanical, electrical and software components. The Open Prototype for Educational Nanosats (OPEN) [2] program at the University of North Dakota is an interdisciplinary, student-run CubeSat program whose goal is to provide students with the opportunity to get involved in all aspects and levels of a spacecraft development program which is an analog for larger national (and international) space programs. The OPEN program has a second mission and this is to create a CubeSat design that is very affordable (< \$5000.00).

Whether intended for a space-borne platform or a balloonborne platform, to keep costs very low, the OPEN design currently relies on the Raspberry Pi computer for its flight computer. As the Raspberry Pi's bus has limited capacity we are forced to use a limited resolution/size camera, and most likely gray-scale. As such, in order to obtain images with any usable quality we are forced to explore options that include interpolation, such as zero-order hold, bilinear, and bicubic, as well as image zoom techniques or single-frame super-resolution [3]. Multi-frame super-resolution [4] has been ruled out as it would require too many images be captured at too high a frame rate to be feasible given the application. Finally, while we titled this paper "Image Zooming using Corner Matching" we would argue that it is more akin to single-frame super-resolution as we are not overly concerned with the aesthetics of the result, but more so with the accuracy of the resulting image.

Image zooming is a common digital image processing operation and is used to improve image resolution [5-7]. There are a number of reasons one would want to do this. It may be that the imaging

system does not have the desired high resolution, as in our case where the size, weight, power and cost (SWaP-C) limitations dictate the use of a lower resolution camera than desired [8]. It may be that the bandwidth and/or available memory is insufficient to transmit or store the image(s). Regardless of the reason, image zooming/magnification is a common operation and a number of methods are used, or have been proposed. We can categorize these into non-adaptive and adaptive methods. Nonadaptive methods use a fixed pattern of processing for all involved pixels [9]. Non-adaptive methods include: nearest neighbor, or zero-order-hold (ZOH), bilinear, bicubic, and Lanczos. Adaptive methods use the surrounding pixels to estimate the missing data [10]. These methods tend to be more computationally expensive, but generally perform better. Adaptive methods include: fractals [11], neural networks [12], wavelets [13], and smart edge processors [14-17].

II. BACKGROUND

As noted above, this work is concerned with the development of an interpolation method or single-frame super-resolution method that can extrapolate a high resolution image from a low resolution image and do so such that the energy (gray scale) distribution amongst the pixels is as accurate as possible. We are not interested in producing an aesthetically pleasing result if it means a reduced level of accuracy in the energy distribution amongst the pixels. Each of the interpolation methods used are briefly discussed in the following sections. Our approach is discussed in section III.

A. Image Interpolation

We compared the results obtained with the proposed method with four different interpolation methods: zero-order hold, bilinear, bicubic, and directional cubic convolution interpolation (DCCI). DCCI was chosen as we were able to recreate the code and it produces aesthetically pleasing results.

1. Zero-order Hold

Zero-order hold (ZOH) is a model of pixel reconstruction performed by holding each pixel value for one sample interval. In our case, in order to double the size of an image, each pixel value is held, or expanded into 4 pixels. No new information is created. This is also referred to as nearest-neighbor interpolation.

2. Bilinear Interpolation

Bilinear interpolation performs a linear interpolation on functions of two variables on a rectilinear 2D grid. Our implementation is shown in equation 1: $f(a,b) = (A*(1-b)*(1-a)+B*(1-b)*a + C*(b*(1-a) + D*b*a)$ (1) Where A, B, C, and D are the pixel values at the four corners of the immediate region to be interpolated and a and b ($0 \leq a, b \leq 1$) are the horizontal and vertical distances from point A (upper left corner) to the point being interpolated. 3. Bicubic Interpolation Bicubic interpolation is often chosen over bilinear interpolation or zero-order hold when dealing with images, as images resampled with bicubic interpolation are smoother and have fewer interpolation artifacts. The formula for a single linear cubic interpolation is shown in equation 2:

$$\begin{aligned}
f(p_0, p_1, p_2, p_3, x) = & \left(-\frac{1}{2}p_0 + \frac{3}{2}p_1 - \frac{3}{2}p_2 + \frac{1}{2}p_3\right)x^3 + \\
& \left(p_0 - \frac{5}{2}p_1 + 2p_2 - \frac{1}{2}p_3\right)x^2 + \\
& \left(-\frac{1}{2}p_0 + \frac{1}{2}p_2\right)x + p_1
\end{aligned} \quad (2)$$

Where p_0, p_1, p_2 and p_3 are pixel values located at $x=-1, x=0, x=1$, and $x=2$, respectively.

To perform bicubic interpolation on an image we perform cubic interpolation in two dimensions as shown in equation 3:

$$\begin{aligned}
g(x, y) = & f[f(p_{00}, p_{01}, p_{02}, p_{03}, y), \\
& f(p_{10}, p_{11}, p_{12}, p_{13}, y), \\
& f(p_{20}, p_{21}, p_{22}, p_{23}, y), \\
& f(p_{30}, p_{31}, p_{32}, p_{33}, y), x]
\end{aligned} \quad (3)$$

Where p_{ij} are the 16 pixels surrounding the interpolation area, where i and j range from 0 to 3 and p_{ij} located at $(i-1, j-1)$. Thus, we first interpolate the four columns and then interpolate the results in the horizontal direction.

4. Directional Cubic Convolution Interpolation

Cubic convolution (CC) is an image interpolation method which indiscriminately interpolates missing pixels in the horizontal and vertical directions. It's useful for reducing blur, blocking, ringing and other image artifacts. Zhou et al [18] proposed an edge directional cubic convolution interpolation (DCCI) scheme which can adopt to varying edge structures in the images. Edge-directed interpolation first tries to detect edge direction, which can be difficult given certain textures in images. As such, they described an estimation method for finding a strong edge for a missing pixel location which guides the interpolation for the missing pixel. For weak edges in complex regions they use a novel interpolation scheme.

Cubic Convolution (CC) smoothens image pixels in nonhorizontal and non-vertical edges by indiscriminately interpolating the missing pixels. For strong edges, CC interpolation is used. For weak edges, two orthogonal directional CC interpolation (DCCI) results are fused to interpolate the missing pixel.

Zhou et al used 24 color images to determine a directional edge detection threshold value (T). Then the gradient at each missing pixel location needs to be computed based on the four immediately surrounding diagonal or horizontal pixels in the image (I). Using two orthogonal directional gradients ($G1$ and $G2$), the edge direction of a pixel can then be estimated. To find an edge direction for any specific pixel we can use gradient values as determined by the following algorithm (4):

$$\begin{aligned}
& \text{If } (1+G1) / (1+G2) > T \{ \text{The pixel is on a } 135^\circ \text{ or strong vertical edge; } \} \text{ else if } (1+G2) / \\
& (1+G1) > T \{ \text{The pixel is on a } 45^\circ \text{ or strong horizontal edge;} \} \text{ else } \{ \text{The pixel is on a weak} \\
& \text{edge or textured region;} \}
\end{aligned}$$

(4)

Here 1 is added to prevent a division by zero.

Pixel values on a strong edge are diagonally directionally interpolated as determined by algorithm 4. For weak edges one needs to find a weight (W) using equations 5 and 6:

$$W1 = 1 / (1 + G1 K) \quad (5)$$

$$W2 = 1 / (1 + G2 K) \quad (6)$$

Where k is used to adjust the weight effect. To interpolate a pixel value (P) at a location with a weak edge or in a texture region equation 7 is applied:

$$P = (W1P1 + W2P2)/(W1+ W2) \quad (7)$$

Where P1 is 45° diagonal or horizontal directional interpolation value and P2 is 135° diagonal or vertical directional interpolation value.

B. Evaluation Methods

For assessment of image quality the peak signal-to-noise ratio (PSNR) and mean squared error (MSE) were used. PSNR is the ratio between the maximum possible power of a signal and the maximum power of an intermingled noise that corrupts the original signal. PSNR is most commonly used in image processing as a measure of quality for reconstructed images from lossy algorithms. The original signal is the original image data before compression, and the noise being the artifacts caused by the loss of data due to compression. To derive the PSNR we must first define the Mean Squared Error (MSE) as shown in equation 8:

$$MSE = \frac{1}{mn} \sum_{i=0}^{m-1} \sum_{j=0}^{n-1} [I(i, j) - K(i, j)]^2 \quad (8)$$

MSE is an error estimate between the original MxN image (I) and the final processed MxN image (K). MSE measures the average of the squares of the errors or deviations. Using equation 8 PSNR can be calculated as shown in equation 9:

$$PSNR = 10 \log_{10} \left(\frac{MAX_I^2}{MSE} \right) \quad (9)$$

Where, MAX is the maximum pixel value of the original image (I).

III. IMAGE ZOOM USING CORNER MATCHING

In this study we used several readily available “standard” images, such as “Baboon,” “Lena,” “Goldhill,” and “Chiles.” We also used two of our own images referred to as “Crossing” and “Senote.” These images are shown in order in figure 1.





Fig. 1. Test images.

To create the required downsampled images, we used a ZOH approach in that we average non-overlapping blocks of 2x2 pixels into a single pixel. Once we have down sampled the image we use ZOH to expand the image such that it is back to its original size (but with half the resolution). At this point we perform the following steps on 3x3 overlapping blocks of pixels:

1. Create two new arrays of the same size as the image. Let's call these arrays "Value" and "Count." We then populate the Value array with the ZOH pixel values and the Count array with ones.
2. Define four patterns of pixels. In Figure 2 these patterns are depicted as the gray pixels. For each pattern we populate a five element array with the values of the corresponding gray colored pixels. Let's call this array "Pattern". These four patterns are considered for every 3x3 set of overlapping blocks of pixels in the image, very much like how one would perform a spatial convolution using a 3x3 kernel.

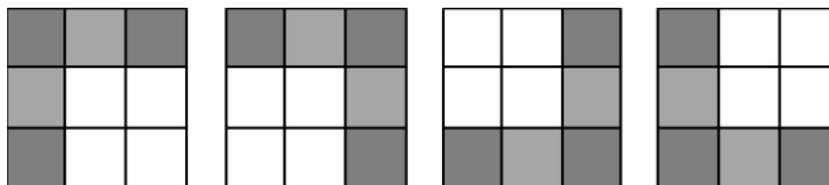


Fig. 2. Pixel patterns.

3. Determine the range, using equation 10, for each pattern n.

$$\text{Rangen} = \text{maximum}(\text{Patternn}) - \text{minimum}(\text{Patternn}) \quad (10)$$

4. Determine the smallest range, using equation 11, of the four patterns.

Smallest = minimum(Rangen) (11)

5. If the smallest range is greater than 64 none of the four patterns suggest that the center pixel should be modified, and we move on to the next 3x3 block of pixels.

6. Otherwise, using the pattern with the smallest range we then determine the average of the associated three dark gray pixels (from Figure 2) and add this value to the appropriate element of the value array. We also increment the appropriate element of the Count array by 1.

7. Once have completed this process for every pixel, we recreate the resulting image by dividing, pixel by pixel, the Value array by the Count array.

IV. RESULTS

As Table 1 shows, for four of the six images, the proposed method (MRMA) results in a higher PNSR and lower MSE than any of the other methods attempted. For one of the two images, where MRMA is not the best, MRMA is nearly tied with ZOH in producing the best results. Only for the Babboon image (which contains a lot of high frequency detail) is ZOH markedly better than MRMA, but MRMA still ranks second best for this image. For each image in Table 1, the best result is the method (type) in boldface.

We extracted thumbnails of three images showing regions where there was a lot of detail: the Lena image around one of the eyes (Figure 3), the zebra Crossing image around the text (Figure 4), and the Senote image through a cave (Figure 5). All figures show these thumbnails in order (clockwise from top left) of original image, reconstructed using ZOH, reconstructed using DCCI, and reconstructed using MRMA.

Examination of the figures show that expected blockiness of the ZOH approach. It also shows that the DCCI method does produce an aesthetically pleasing result. However, careful examination of each figure show that the MRMA method does reconstruct more high frequency details, and is thus more like the original image, than the other three.

TABLE I
TEST RESULTS

Image	Type	PSNR	MSE
Babboon	ZOH	23.8400	268.5842
Babboon	Bilinear	22.8835	334.7600
Babboon	Bicubic	22.9944	326.3186
Babboon	MRMA	23.6809	278.6035
Babboon	DCCI	22.9768	327.6448
Lena	ZOH	32.8736	33.5520
Lena	Bilinear	31.6135	44.8470
Lena	Bicubic	31.9348	41.6487
Lena	MRMA	35.0058	20.5352
Lena	DCCI	31.8227	42.7373
Goldhill	ZOH	30.0454	64.3483
Goldhill	Bilinear	28.9926	82.0021
Goldhill	Bicubic	29.1599	78.9028
Goldhill	MRMA	30.3694	59.7235
Goldhill	DCCI	29.0637	80.6702
Chillies	ZOH	31.7630	43.3289
Chillies	Bilinear	31.0813	50.6936
Chillies	Bicubic	31.3180	48.0040
Chillies	MRMA	33.3572	30.0168
Chillies	DCCI	31.3186	47.9973
Crossing	ZOH	27.4390	117.2674
Crossing	Bilinear	25.9411	165.5660
Crossing	Bicubic	26.1467	157.9107
Crossing	MRMA	27.9725	103.7112
Crossing	DCCI	22.2145	390.5117
Senote	ZOH	28.8398	84.9384
Senote	Bilinear	27.3760	118.9807
Senote	Bicubic	27.5432	114.4891
Senote	MRMA	28.8388	84.9581
Senote	DCCI	26.5175	144.9873



Fig. 3. Lena results: original, ZOH, DCCI, MRMA.

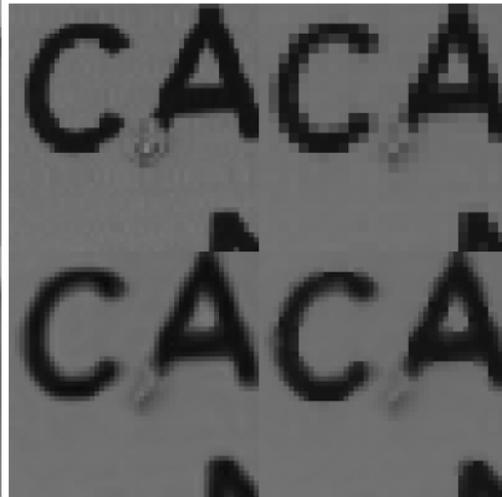


Fig. 4. Crossing results: original, ZOH, DCCI, MRMA.

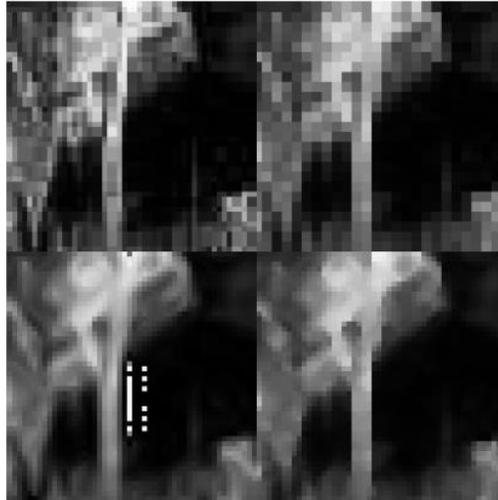


Fig. 5. Senote results: original, ZOH, DCCI, MRMA.

V. CONCLUSIONS

We present a method of zooming images based on the detection and recreation of corners (MRMA). We then compare this approach with three other commonly used methods to zoom/interpolate images as well as the directional cubic convolution interpolation method (DCCI). Using both the MSE and PSNR methods to evaluate the results, and using four commonly used images, as well as two of our own images, we show that the MRMA method produces higher PSNR and lower MSE errors than the other methods. The exception is that on one image with a lot of high frequency detail, the ZOH method returns the best results. For future work, we expect to further improve results by the application of a dithering algorithm to further reduce the error.

ACKNOWLEDGMENT

This material is based upon work supported by the National Science Foundation Research Experiences for Undergraduates under Grant No. (NSF 1359244).

REFERENCES

- [1] R. Deepak and R. Twiggs, "Thinking out of the box: Space science beyond the CubeSat," *Journal of Small Satellites*, 1(1), pp 3–7. 2012.
- [2] J. Straub, "An open prototype for educational NanoSats: Increasing national space engineering productivity via a low-cost platform," 2nd National Academy of Inventors Conference, Tampa, FL. February, 2013.
- [3] T. Nakahara, K. Uruma, T. Takahashi, and T. Furukawa, "Single-frame super-resolution using superpixel based dictionary," *IEEE 59th International Midwest Symposium on Circuits and Systems (MWSCAS)*, Abu Dhabi, United Arab Emirates, ISSN: 1558-3899, 2016.
- [4] R. Marsh, T. Young, T. Johnson, and D. Smith, "Enhancement of small telescope images using super-resolution techniques," *Publications of the Astronomical Society of the Pacific*, May 2004.
- [5] R. C. Gonzalez and R. E. Woods, *Digital Image Processing*, 2nd ed., Prentice Hall.
- [6] Chandan Singh, "Interpolation methods image zooming", *Proceeding of the National Conference FACM*, Thapar University, pp 29-34, March 2005.
- [7] L. Zhiwei, Z. Min, and W. Jiechao, "An image zooming technique based on the relative color difference of pixels," *IEEE Transactions On Image Processing*, Vol. 15, No.2, 2006.
- [8] Curtiss-Wright, "Today's aerospace and defense applications demand high performance in small form factors," <https://www.curtisswrightds.com/technologies/size-weight-andpower.html>. Last accessed 2/8/2018.
- [9] F. Arif and M. Akbar, "Resampling air borne sensed data using bilinear interpolation algorithm," *Proceedings of the 2005 IEEE International Conference on Mechatronics*, Taipei, Taiwan, 2005.
- [10] T. Acharya¹ and P.-S. Tsai, "Computational foundations of image interpolation algorithms," *Ubiquity Information Everywhere*, Vol.8, 2007.
- [11] M. Murrioni, C. Perra, and D. D. Giusto, "Slow motion and zoom in HD digital videos using fractals," *International Journal of Digital Multimedia Broadcasting*, Vol. 2009, Article ID 496934, 12 pages, 2009.
- [12] D. Youssef, B. Mohammed, A. Abdelmalek, H. Tarik, and J. El Miloud, "Zoom and restoring of digital images with artificial neural networks," *Computer Science and Engineering*, Vol. 5, No. 1, pp 14-24, doi: 10.5923/j.computer.20150501.03, 2015.
- [13] N. Kaulgud and U. B. Desai, "Image zooming: use of wavelets," In: Chaudhuri S. (eds) *Super-Resolution Imaging. The International Series in Engineering and Computer Science*, Vol. 632, pp 21-44, Springer, Boston, MA, 2002.
- [14] S. J. Patel and K. Pandey, "Adaptive image resizing using edge contrasting," *International Journal of Advance Higher Education Research & Development*, Vol. 1, No. 9, May 2017.
- [15] K. Nakao, Y. Koya, Y. Kubo, and S. Sugimoto, "Image zooming algorithms using total variation," *Proceedings of the ISCIE International Symposium on Stochastic Systems Theory and its Applications*, Honolulu, Hi, pp 299-306, 2016.

- [16] Y. J. Lee and J. Yoon, "Image zooming method using edge-directed moving least squares interpolation based on exponential polynomials," *Applied Mathematics and Computation*, 269, pp 569-583, 2015.
- [17] N. Nowroozian and H. Hassanpour, "Image zooming using non-linear partial differential equation," *International Journal of Engineering Transactions A: Basics*, Vol. 27, No. 1, pp 15-28, 2014.
- [18] D. Zhou, X. Shen, and W. Dong, "Image zooming using directional cubic convolution interpolation," *IET Image Processing*, Vol. 6, Issue 6, pp 627-634, 2012.