

CUBIC SURFACE FITTING TO IMAGE WITH EDGES AS CONSTRAINTS

ZHANG CaiMing^{1,2}, ZHANG Xin¹, Li XueMei¹ and Cheng Fuhua³

¹ School of Computer Science and Technology, Shandong University, Jinan, China;

² Shandong Key Lab. of Digital Media Technology,

Shandong University of Finance and Economics, Jinan, China

³Department of Computer Science, University of Kentucky, Lexington, KY, USA

ABSTRACT

Conventional polynomial interpolation methods produce interpolated images with blurred edges, while edge-directed interpolation methods make enlarged images with good quality edges but with detail distortion in the non-edge portion for some cases. A new method for constructing a fitting surface to image data is presented. Unlike existing methods which produce enlarged images using image data as interpolation data, the new method constructs the fitting surface using the image data as constraints to reverse the sampling process for improving the fitting precision. To remove the zigzagging artifact, for each pixel and its nearby region, the edge information is used to determine the quadratic polynomial which approximates the original scene with a quadratic polynomial precision. Comparison results of the new method with other methods are included.

Index Terms— Surface fitting, quadratic polynomial, image resizing, reversing

1. INTRODUCTION

Resizing an image is of fundamental importance in fields such as CG and image processing[1]. This paper addresses the problem of constructing a surface to fit the image data so that the resized image has better precision and quality.

The simplest interpolation method for resizing image is pixel replication, but usually results in the undesirable blocking effect. Bi-cubic interpolations[2, 3] use cubic polynomial to make the resized image more visually pleasing. Paper[4] uses non-separable cubic-convolution kernels for image interpolation, the test results show, however, that it has no obvious advantage over the traditional separable cubic convolution methods. Using both oblique and orthogonal projections, an optimal spline-based method for resizing digital images is discussed in paper[5]. The experiments show that this method outperforms the standard interpolation techniques. An edge-directed interpolation method[6][7] is proposed to provide a solution to enlarged image based on their geometric duality.

Recently, Zhang and Wu[8] developed a soft-decision interpolation method which is able to estimate missing pixels by groups instead of by pixels.

Edges are crucial to image perception, blurred edges and annoying artifacts are the most annoying visual artifacts[6]-[8]. Conventional polynomial interpolation methods fail to capture the fast evolving statistics around edges and consequently produce enlarged images with blurred edges and annoying artifacts. Edge-directed methods make enlarged image with good quality around edges while with the detail distortion of the portion surrounded by edges for some cases. Moreover, the methods[1]-[8] are based on the sampling data points, each of which is taken from a region of the original scene, so, some details of the image will be lost.

Our study shows that it is not polynomial interpolation that produces the blurred edges, but the interpolation conditions used in the interpolation procedure. Based on the fact that image data are sampled from an original scene that can be approximated by piecewise polynomials, a new method for constructing surface to fit image data is proposed. The fitting surface is constructed by reversing the process of image sampling. The edge information is used as constraints to construct the fitting surface, which makes the surface have good shape and approximate the original scene with a quadratic polynomial precision. The fitting surface is formed by the combination of the quadratic polynomial patches.

2. DESCRIPTION OF NEW METHOD

Suppose that \mathbf{P} is an image composed of $n \times n$ image elements, $P_{i,j}$ (the position marked by ‘•’ in Figure 1), $i, j = 1, 2, \dots, n$. These elements are generally sampled from an original scene $F(x, y)$ on the region $[1/2, n+1/2] \times [1/2, n+1/2]$. For brevity, suppose that each element $P_{i,j}$ is sampled from a unit square, i.e.,

$$P_{i,j} = \int_{j-\frac{1}{2}}^{j+\frac{1}{2}} \int_{i-\frac{1}{2}}^{i+\frac{1}{2}} w(x, y) F(x, y) dx dy \quad (1)$$

where $w(x, y)$ is a weight function with $w(x, y) = 1$, the case of $w(x, y)$ being a function will be studied in the future.

Note: $P_{i,j}$, $i, j = 1, 2, \dots, n$ are integers, hence (1) does not hold in general, but it holds approximately.

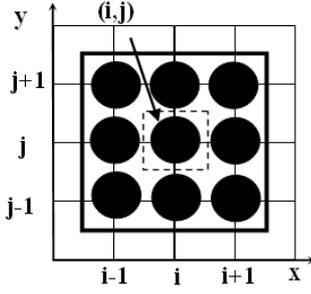


Fig. 1. Image region

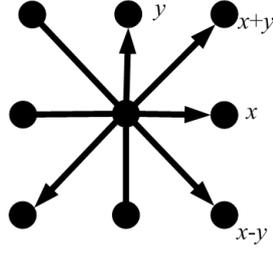


Fig. 2. Four directions

2.1. Basic idea

If $F(x, y)$ is known, we can resize the image by (1) easily. Hence, resizing \mathbf{P} becomes a problem of reconstructing $F(x, y)$. The goal here is to construct a fitting surface $f(x, y)$ which approximates $F(x, y)$ with a quadratic polynomial precision. The construction of $f(x, y)$ is described as below. On each sub-region $[i-1.5, i+1.5] \times [j-1.5, j+1.5]$ (the square bounded by solid line), $i, j = 2, 3, \dots, n-1$, as shown in Figure 1, a quadratic polynomial patch $f_{i,j}(x, y)$ is constructed, which satisfies the condition that if $F(x, y)$ in (1) is a quadratic polynomial, $f_{i,j}(x, y)$ should reproduce $F(x, y)$ exactly. In this case, $f_{i,j}(x, y)$ is known as having a quadratic polynomial precision. $f(x, y)$ is constructed by the weighted combination of $f_{i,j}(x, y)$, $i, j = 2, 3, \dots, n-1$.

Let $u = x - i, v = y - j$, then on $[-1.5, 1.5] \times [-1.5, 1.5]$ in uv plane, $f_{i,j}(x, y)$ can be represented as

$$f_{i,j}(x, y) = au^2 + buv + cv^2 + du + ev + f \quad (2)$$

where a, b, c, d, e, f are unknowns to be determined.

2.2. Constructing patch $f_{i,j}(x, y)$

Following, the determination of the unknowns in (2) is discussed. Since the quality around the edges plays an important role in the visual effect of an image, $f_{i,j}(x, y)$ should reflect the characteristics around the edges as well as possible. In Figure 2, the center pixel is supposed to be $P_{i,j}$, there are four directions formed by $P_{i,j}$ and its neighbor pixels, denoted as $y, x+y, x$ and $x-y$, respectively. Similarly, for the 8 neighbor pixels of $P_{i,j}$, there are also 4 directions for each one. If the image varies linearly along a direction \vec{d} , then the images along \vec{d} form an edge of the image, $f_{i,j}(x, y)$ should be a linear function along \vec{d} . The unknowns in (2) should reflect the variations along the edges at the nine pixels. We group the unknowns in (2) into 3 sets: $f, \{d, e\}$ and $\{a, b, c\}$, where f is used to make $f_{i,j}(x, y)$ satisfy (1), the second set is used

to reflect the variation along four directions of $P_{i,j}$, while the last set reflects the variation at 9 pixels of $P_{i,j}$ in Figure 2. We first discuss how to determine the unknowns d and e .

Theorem 1 If $F(x, y)$ is defined by (2), $P_{i,j}$ satisfies (1), then, the following conditions hold

$$\begin{aligned} e &= e_1 & d + e &= e_2, \\ d &= e_3 & d - e &= e_4 \end{aligned} \quad (3)$$

where

$$\begin{aligned} e_1 &= (P_{i,j+1} - P_{i,j-1})/2, \\ e_2 &= (P_{i+1,j+1} - P_{i-1,j-1})/2, \\ e_3 &= (P_{i+1,j} - P_{i-1,j})/2, \\ e_4 &= (P_{i+1,j-1} - P_{i-1,j+1})/2 \end{aligned}$$

Prof: Substituting (2) into (1) and integrating gets

$$\begin{aligned} P_{i-1,j} &= \frac{13}{12}a + \frac{1}{12}c - d + f \\ P_{i+1,j} &= \frac{13}{12}a + \frac{1}{12}c + d + f \end{aligned}$$

Thus, $d = e_3$ in (3) holds. Similarly, the rest of the cases can be proved.

In (3), there are 4 equations with unknowns d and e . To make $f_{i,j}(x, y)$ reflect the characteristics of the image edge passing $P_{i,j}$, d and e will be determined by constrained least squares, i.e, by minimizing the following function

$$\begin{aligned} G(d, e) &= w_1(e - e_1)^2 + w_2(d + e - \frac{\sqrt{2}}{2}e_2)^2 \\ &+ w_3(d - e_3)^2 + w_4(d - e - \frac{\sqrt{2}}{2}e_4)^2 \end{aligned} \quad (4)$$

where $w_i, i = 1, 2, 3, 4$ are weight functions.

We discuss how to determine $w_i, i = 1, 2, 3, 4$ in (4). Substituting (2) into (1) and integrating gets

$$\begin{aligned} \Delta_1 &= (P_{i,j+1} + P_{i,j-1})/2 - P_{i,j} = c \\ \Delta_2 &= (P_{i+1,j+1} + P_{i-1,j-1})/2 - P_{i,j} = a + b + c \\ \Delta_3 &= (P_{i+1,j} + P_{i-1,j})/2 - P_{i,j} = a \\ \Delta_4 &= (P_{i-1,j+1} + P_{i+1,j-1})/2 - P_{i,j} = a - b + c \end{aligned} \quad (5)$$

In Figure 2, if $f_{i,j}(x, y)$ (2) is a linear function along the direction y , the determination of w_1 should make e_1 (3) play a primary role on the determination of e , so, w_1 (4) should be assigned a bigger value. When the variation of the image along direction y closes a linear function, then $\Delta_1 = 2c$ closes 0. Hence, w_1 should be inversely proportional to Δ_1 . Similarly, we can define w_2, w_3 and w_4 . They are defined as

$$w_i = \frac{1}{1 + \Delta_i^2}, \quad i = 1, 2, 3, 4 \quad (6)$$

Now we discuss how to determine f in (2). Substituting (2) into (1) and integrating gets

$$f = P_{i,j} - \frac{1}{12}a - \frac{1}{12}c$$

Now, $f_{i,j}(u, v)$ can be written as

$$f_{i,j}(u, v) = au^2 + buv + cv^2 + du + ev - \frac{a+c}{12} + P_{i,j} \quad (7)$$

Next, we will discuss how to determine a, b, c . Firstly, we determine a, b and c by making $f_{i,j}(u, v)$ approximate 8 pixels around $P_{i,j}$ by constrained least square method. Let

$$g_{k,l}(a, b, c) = \int_{l-\frac{1}{2}}^{l+\frac{1}{2}} \int_{k-\frac{1}{2}}^{k+\frac{1}{2}} f_{i,j}(u, v) dudv = P_{i+k,j+l}$$

Then, a, b and c are defined by minimizing the function

$$G(a, b, c) = \sum_{\substack{k,l=-1,0,1 \\ k \neq l=0}} w_{k,l} (g_{k,l}(a, b, c) - P_{i+k,j+l})^2 \quad (8)$$

where $w_{k,l}, k, l = -1, 0, 1, k \neq l = 0$ are weight functions.

To make $f_{i,j}(x, y)$ (2) have good quality, the information of edges are used to determine $w_{k,l}$. Near $P_{i+1,j}$, if $f_{i,j}(x, y)$ (2) is a linear function along the direction x , then $P_{i+1,j}$ should play a primary role on the determination of a, b and c . In this case, $\Delta_{1,0} = (P_{i+2,j} + P_{i,j})/2 - P_{i+1,j}$ is close to zero, so $w_{1,0}$ can be assigned a bigger value by

$$w_{1,0} = \frac{1}{1 + \Delta_{1,0}^2} \quad (9)$$

Similarly, $w_{1,1}, w_{0,1}, w_{-1,1}, w_{1,1}, w_{-1,0}, w_{-1,-1}$ and $w_{0,-1}$ are determined.

Second, considering equation (5), the information of the edges at $P_{i,j}$ can also be used to determine a, b, c by the following function.

$$E(a, b, c) = w_1(c - \Delta_1)^2 + w_2(a + b + c - \Delta_2)^2 + w_3(a - \Delta_3)^2 + w_4(a - b + c - \Delta_4)^2 \quad (10)$$

where w_1, w_2, w_3 and w_4 are defined by (6).

Now, a, b, c are determined by the following function

$$H(a, b, c) = G(a, b, c) + \lambda E(a, b, c) \quad (11)$$

with λ being a parameter.

The discussion above showed that if $F(x, y)$ is a quadratic polynomial, and $P_{i,j}, i, j = 1, 2, \dots, n$, are defined by (1), then $f_{i,j}(x, y)$ will be determined uniquely. So, there is the following Theorem.

Theorem 2. For image \mathbf{P} which is composed of $n \times n$ elements, $P_{i,j}, i, j = 1, 2, \dots, n$, if $P_{i,j}$ is defined by (1), $f_{i,j}(x, y)$ has a quadratic polynomial precision.

On each sub-region $[i-1, j-1] \times [i+1, j+1]$, $f_{i,j}(x, y)$ in (2) should satisfy $0 \leq f_{i,j}(x, y) \leq 255$, so that $f(x, y)$ which will be constructed in Section 3 satisfies $0 \leq f(x, y) \leq 255$. If $f_{i,j}(x, y)$ in (2) does not satisfy the required condition, it will be modified to satisfy the condition.

3. CONSTRUCTING FITTING SURFACE

On each sub-region $[i, i+1] \times [j, j+1], i, j = 1, 2, \dots, n-1$, a bi-cubic patch $B_{i,j}(x, y)$ is constructed, all $B_{i,j}(x, y)$'s are put together to form the fitting surface $f(x, y)$.

On $[i, i+1] \times [j, j+1], i, j = 2, 3, \dots, n-2$, $B_{i,j}(x, y)$ is constructed by $f_{i,j}(x, y), f_{i+1,j}(x, y), f_{i,j+1}(x, y)$ and $f_{i+1,j+1}(x, y)$, i.e., $B_{i,j}(x, y)$ is defined by

$$B_{i,j}(x, y) = w_{i,j}(x, y)f_{i,j}(x, y) + w_{i+1,j}(x, y)f_{i+1,j}(x, y) + w_{i,j+1}(x, y)f_{i,j+1}(x, y) + w_{i+1,j+1}(x, y)f_{i+1,j+1}(x, y) \quad (12)$$

where

$$w_{i,j}(x, y) = (1-v)(1-w), \quad w_{i+1,j+1}(x, y) = vw$$

$$w_{i,j+1}(x, y) = (1-v)w, \quad w_{i+1,j}(x, y) = v(1-w)$$

are weight functions with $v = x - i, w = y - j$.

The patches on the boundary of \mathbf{P} are $B_{1,j}(x, y)$ and $B_{n-1,j}(x, y), j = 1, 2, \dots, n-1, B_{i,1}(x, y)$ and $B_{i,n-1}(x, y), i = 2, 3, \dots, n-2$. As symmetry, we only discuss the construction of $B_{1,j}(x, y), j = 1, 2, \dots, n-1$, the rest of the cases can be handled similarly. $B_{1,1}(x, y)$ is defined by $f_{2,2}(x, y)$, for $j = 2, 3, \dots, n-2, B_{1,j}(x, y)$ is defined by

$$B_{1,n-1}(x, y) = f_{2,n-1}(x, y)$$

$$B_{1,j}(x, y) = f_{2,j}(x, y)(1-w) + f_{2,j+1}(x, y)w \quad (13)$$

Based on the Theorem 2 and the definition of $B_{i,j}(x, y)$ (10) and (11), it is easy to know that $f(x, y)$ has a quadratic polynomial approximation precision.

4. EXPERIMENTS

In this section, we will compare the efficiency of the new method(CCEM) with CCM^[4], NEI method^[6] and IIAS method^[8], where, λ in (11) is set to 5. The comparison is carried out by enlarging 6 standard images (as shown in Figure 3) with size 256×256 .

We first compare the vision quality of the enlarged images produced by the four methods. The comparison results are shown in Figure 4. Images (A), (B), (C) are parts of the 512×512 images which are produced using (from top to bottom) CCEM, CCM, NEI and IIAS, respectively, by enlarging the 256×256 images. Images (D), (E), (F) are parts of the 1024×1024 images using the above four methods by enlarging the 256×256 images. Figure 4 shows that the images produced by CCEM have better visual quality than the images produced by CCM, NEI and IIAS. The Medical, Peppers and Lenna images by CCM have staircases along image edges, while the Rail, Medical, Peppers and Couple images by NEI and IIAS, respectively have texture distortion.

Then, the images enlarged by the four methods are compared in term of PSNR defined by

$$PSNR = \frac{N \times N \times 255 \times 255}{\sum_{j=1}^n \sum_{i=1}^n (R_{i,j} - P_{i,j})^2} \quad (14)$$

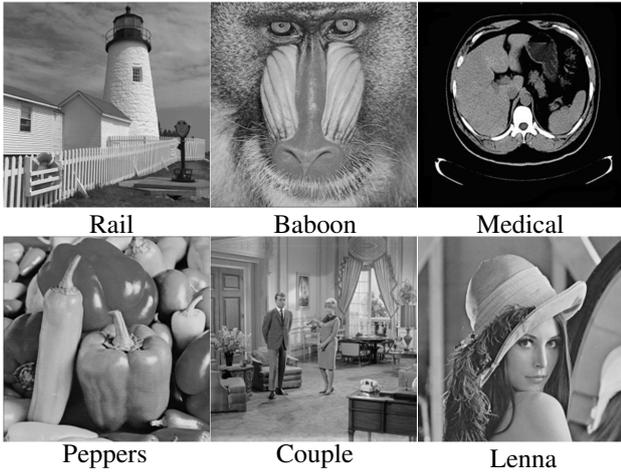


Fig. 3. Four standard images

Table 1. PSRN of the four methods (size: 256×256)

| Image | CCEM | CCM | NEI | IIAS |
|---------|-------|-------|-------|-------|
| Rail | 26.67 | 26.49 | 24.51 | 24.19 |
| Baboon | 23.95 | 23.95 | 22.70 | 22.55 |
| Medical | 29.88 | 29.55 | 26.24 | 26.09 |
| Peppers | 34.05 | 33.72 | 29.83 | 29.18 |
| Couple | 29.79 | 29.63 | 27.62 | 27.29 |
| Lenna | 32.50 | 32.24 | 29.16 | 28.98 |

which is a normalized measure for testing the image quality, where $P_{i,j}$ is the accurate image element, and $R_{i,j}$ is the enlarged image element produced by one of the four methods. The PSRN of the four methods applied to the 6 images (size 256×256) are given in Table 1.

The experiments have been done on other images. The results are similar to the ones as shown in Figure 4. Moreover, we also compared CCEM with the methods in papers[5][7][9] and PDE-based interpolation methods, CCEM got better vision quality too.

5. CONCLUSIONS

Conventional polynomial interpolation methods generally make the enlarged images have blurred edges and annoying artifacts. Edge-directed methods make enlarged image with good quality around edges while with the detail distortion of the portion surrounded by edges. In order to get rid of these two shortcomings, a new method is presented for image resizing. The new method constructs the fitting surface locally by the combination of the quadratic polynomial patches. The quadratic patches are constructed by reversing the process of sampling and with edges as constraints, which makes the surface have the shape suggested by the image and a better approximation precision. The new method has the advantage in that it can easily zoom the image into multiples. The com-

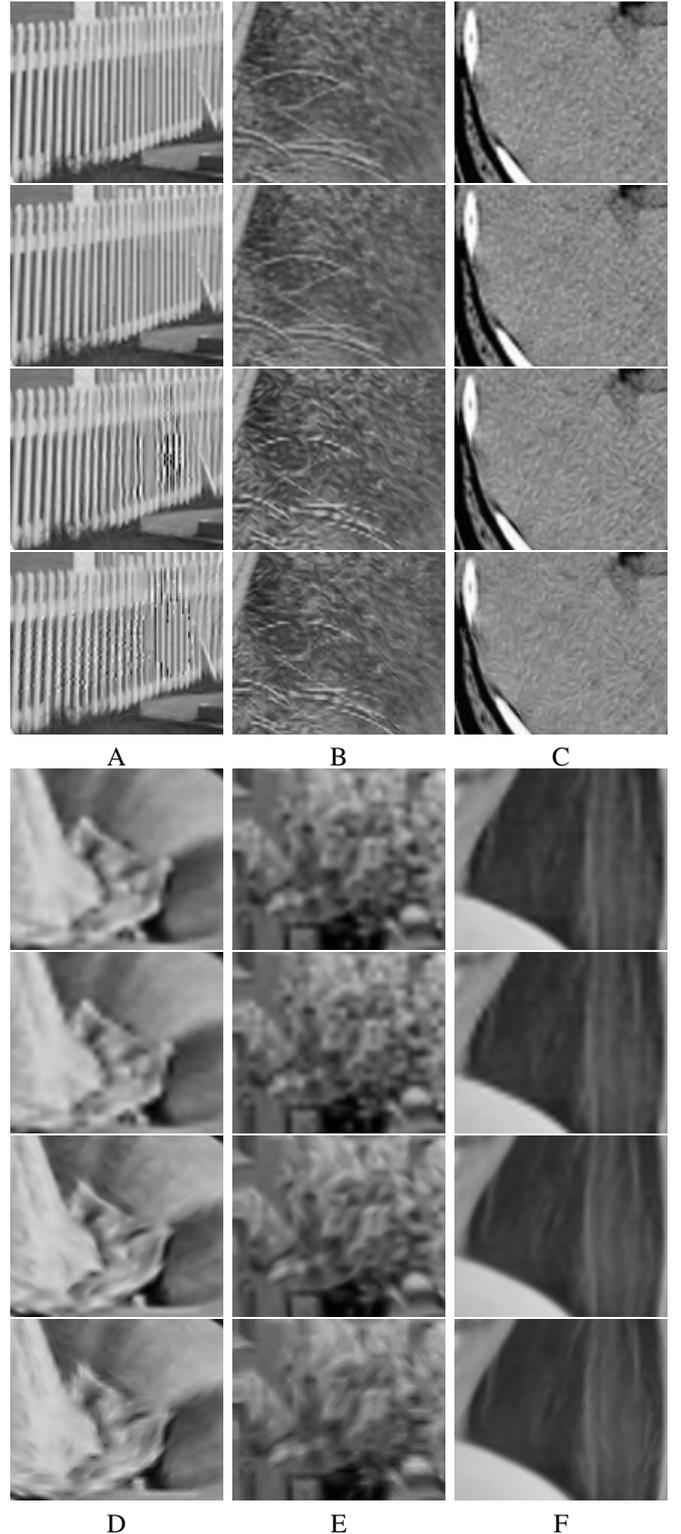


Fig. 4. Parts of the enlarged images with size 256×256 .

parison results also indicate that the new method produces resized images with better quality.

6. REFERENCES

- [1] Gonzalez R C. and Woods R E, *Digital Image Processing*, 1992, Addison-Wesley, Reading MA.
- [2] Park S K and Schowengerdt R A, "Image reconstruction by parametric cubic convolution," *Computer Vision, Graphics, and Image Proc*, vol. 23, pp. 258–272, September 1983.
- [3] Meijering E H W, Niessen W J, and Viergever M A, "Piecewise polynomial kernels for image interpolation: A generalization of cubic convolution," in *Proc IEEE Int Conf Image Processing*. IEEE, 1999, pp. 647–651.
- [4] Shi J and Reichenbach S E, "Image interpolation by two-dimensional parametric cubic convolution," *IEEE Trans Image Proc*, vol. 15, pp. 1857–1870, July 2006.
- [5] Munoz A, Blu T, and Unser M, "Least-squares image resizing using finite differences," *IEEE Trans Image Proc*, vol. 10, pp. 1365–1378, September 2001.
- [6] Li X and Orchard M T, "New edge-directed interpolation," *IEEE Trans Image Proc*, vol. 10, pp. 1521–1527, October 2001.
- [7] Li M. and Truong Q. Nguyen, "Markov random field model-based edge-directed image interpolation," *IEEE Trans Image Proc*, vol. 17, pp. 1121–1128, July 2008.
- [8] Zhang X. and Wu X., "Image interpolation by adaptive 2-d autoregressive modeling and soft-decision estimation," *IEEE Trans Image Proc*, vol. 17, pp. 887–896, June 2008.
- [9] Zhang C., Li X. Liu H., and Zhang C., "Cubic surface fitting to image by combination," *Science in China F*, vol. 41, pp. 1101, September 2011.