International Journal of Information, Control and Computer Sciences
ISSN: 2517-9942
Vol:2, No:11, 2008

Adaptive Bidirectional Flow for Image Interpolation and Enhancement

Shujun Fu, Qiuqi Ruan and Wenqia Wang

Abstract**—**Image interpolation is a common problem in imaging applications. However, most interpolation algorithms in existence suffer visually the effects of blurred edges and jagged artifacts in the image to some extent. This paper presents an adaptive feature preserving bidirectional flow process, where an inverse diffusion is performed to sharpen edges along the normal directions to the isophote lines (edges), while a normal diffusion is done to remove artifacts ("jaggies") along the tangent directions. In order to preserve image features such as edges, corners and textures, the nonlinear diffusion coefficients are locally adjusted according to the directional derivatives of the image. Experimental results on synthetic images and nature images demonstrate that our interpolation algorithm substantially improves the subjective quality of the interpolated images over conventional interpolations.

Keywords**—**anisotropic diffusion, bidirectional flow, directional derivatives, edge enhancement, image interpolation, inverse flow, shock filter.

I. INTRODUCTION

IMAGE interpolation (magnification) is an image processing to gain its high-resolution image from a low-resolution version. Conventional interpolations treat the problem primarily as either fitting a space-invariant function (e.g., bilinear and bicubic) or extrapolating in frequency domain [1]. The former employs similarly a low-pass filtering process and blurs the edges of the interpolated image; the latter introduces false high-frequency components and produces annoying artifacts ("jaggies" and "mosaics"). Adaptive interpolation techniques [2, 3] spatially adapt the interpolation coefficients to better match the local structures around the edges. However, it may bring errors in selecting and estimating the edges of interest. Edge-directed interpolations [4, 5, 6, 7] fit the sub-pixel edges of the image utilizing the limited quantification of the directions and the widths of the edges, while preventing interpolations from across the edges. Although they can produce sharp edges, they fit the edges too simply, and they may also lose some features of the image. Among others are projection onto convex sets (POCS) scheme, approaches based on wavelet analysis and morphological filtering, etc [8, 9, 10].

In section II, we presents an adaptive feature preserving bidirectional flow (BDF) process, that is, an anisotropic diffusion equation, where an inverse diffusion is performed to enhance edges along the normal directions to the isophote lines (edges), while a normal diffusion is done to remove artifacts ("jaggies") along the tangent directions. In section III, having investigated the character of the typical edge, we discuss the backward flow and the forward flow along different directions, and, to preserve image features, detail this section by properly designing the diffusion coefficients using the first and second order directional derivatives of the image. In section IV, we implement the scheme and test it on synthetic images and nature images. Conclusions are presented in section V.

II. A UNIFIED BIDIRECTIONAL FLOW

The use of partial differential equations (PDEs) in image processing has grown significantly over the past years [11]. Initially proposed by P. Perona and J. Malik [12], the nonlinear anisotropic diffusion filters have been widely used in image denoising, enhancement, and sharpening. The grey levels of an image $u(x, y, t):\Omega\times[0,+\infty)\rightarrow R$, are diffused according to:

$$\frac{\partial u(x, y, t)}{\partial t} = div(g(|\nabla u(x, y, t)|)\nabla u(x, y, t)) \ , \qquad (1)$$

The scalar diffusivity $g(|\nabla u|)$, chosen as a non-increasing function, governs the behaviour of the diffusion process. A typical choice for the diffusivity function is:

$$g(|\nabla u|) = 1/(1+(|\nabla u|/K)^2) \ , \qquad (2)$$

with K some gradient threshold. Practical implementations of the P-M filter are giving impressive results, noise is eliminated and edges are kept or even enhanced provided that their gradient value is greater than K.

By formally developing the divergence term, (1) can be put in iterms of second order derivatives taken in the directions of the gradient vectors ($\vec{n}$) and in the orthogonal tangent ones ($\vec{t}$):

$$\frac{\partial u}{\partial t} = g(|\nabla u|)u_{tt} + (g'(|\nabla u|)|\nabla u| + g(|\nabla u|))u_{nn} \ , \qquad (3)$$

Manuscript received Novemble 4, 2004. This work was supported by the Key Laboratory Project of Information Science & Engineering of Railway of National Ministry of Railways, China (No.TDXX0510), the Technological Innovation Fund of Excellent Doctorial Candidate of Beijing Jiaotong University, China (No. 48007).

Shujun Fu and Qiuqi Ruan are with the Institute of Information Science, Beijing Jiaotong University, Beijing, 100044, China (corresponding author, E-mail: shujunfu@163. com).

Shujun Fu and Wenqia Wang are with the School of Mathematics and System Science, Shandong University, Jinan, 250100, China.

$$u_{tt}=\frac{1}{|\nabla u|^2}(u_x^2u_{yy}+u_y^2u_{xx}-2u_xu_yu_{xy})=k|\nabla u|, \quad (4)$$

$$u_{nn}=\frac{1}{|\nabla u|^2}(u_x^2u_{xx}+u_y^2u_{yy}+2u_xu_yu_{xy}). \quad (5)$$

with k the isophote curvature [11]. This expression allows an easier interpretation of the original equation: (1) acts like a low pass filter diffusing along the edge directions and, selectively, for diffusion functions as (2), can preserve edges without diffusing across edges. Results obtained with the P-M process paved the way for a variety of PDE-based methods that were applied to various problems in low-level vision.

G. Gilboa et al. [13] present a forward-and-backward (FAB) adaptive diffusion process, and apply it in signal and image enhancement and sharpening. They switch the diffusion process from a forward to a backward (inverse) mode according to local gradients of the image, which can enhance features while locally denoising smoother segments of the signal or image. Different from (2), they choose the following diffusion coefficient:

$$g(s)=1/(1+(s/k_f)^n)-\alpha/(1+((s-k_b)/w)^{2m}), \quad (6)$$

with parameters (k_f, k_b, w) and (n, m), which are chosen such that a forward diffusion ($g(s)>0$) is used to smooth low gradients, while a backward one ($g(s)<0$) to enhance high gradients.

Another PDE-based enhancement process is proposed by L. Alvarez and L. Mazorra [14], which couples an anisotropic diffusion with the shock filter (we call it ADSF) of S. J. Osher and L. I. Rudin [15], yielding an equation of the form:

$$\frac{\partial u}{\partial t}=-sign(G_\sigma * u_{nn})|\nabla u|+cu_{tt}, \quad (7)$$

with c a positive constant, G_σ a Gaussian of standard deviation σ. The first term on the right side creates solutions approaching piecewise constant regions separated by shocks at the zero-crossings of the smoothed second derivative of the image along $\vec{n}$. The second term is an anisotropic diffusion along the level-set lines $\vec{t}$.

In fact, both (6) and (7) are based on the same idea, which can be shown more clearly below. Noticing the equation:

$$sign(s)=s/|s|, \quad s\neq 0, \quad (8)$$

we can now define a unified bidirectional flow (BDF) equation covering both (6) and (7):

$$\frac{\partial u}{\partial t}=\alpha(-c_n(u_n,u_{nn},u_{tt})u_{nn})+\beta(c_t(u_n,u_{nn},u_{tt})u_{tt}), \quad (9)$$

where α, β are the backward and forward flow control coefficients, $c_n(s)$, $c_t(s)$ are diffusion coefficients of their arguments, which should be properly designed to preserve features of the image such as edges, corners and texture. We will discuss this issue immediately.

III. FEATURE PRESERVING BIDIRECTIONAL FLOW FOR IMAGE INTERPOLATION AND ENHANCEMENT

Image interpolation means "reading between the original pixels", which also can be considered as a diffusion process: "to diffuse gray levels from pixels of the original image to the blank interpolated pixels between them". Therefore, this paper further extends the nonlinear PDE-based flow methods, and applies them to image interpolation and enhancement.

We divide our BDF process into two steps. First, the image is interpolated to the new desired size. In our implementation, we use bilinear interpolation. The first step provides good results over smooth areas, but edges are smeared, and artifacts ("jaggies") are also introduced. Then, we perform the BDF process to enhance the edges and smooth the interpolation byproducts. Most importantly, we detail this section by designing the diffusion coefficients c_n and c_t properly to preserve image features.

A. Backward flow

1) 1D backward flow. For interpreting the backward flow clearly, we begin with one dimensional (1D) signal case, then extend to 2D image. Because conventional interpolations (e.g., bilinear and bicubic) result in inevitably blurred edges of the image, we first analyze the differential properties of 1D typical slope edge. In Fig.1, **a** is a slope edge, whose center is **o**, and **b**, **c** are its first and second order differential curves. It is evident that **b** increases from 0 gradually, reaches its maximum at **o**, then decreases to 0; while **c** changes its symbol at **o**, from positive to negative. Here we want to control the variety of gray levels beside the edge center **o**. More precisely, we want to minish gray levels of pixels on the left of **o**, while to add that on the right of **o**, by which we can enhance the edge reducing its width (see Fig.2).

Thus, being contrary to the classic non-linear anisotropic diffusion, here we perform a backward flow (inverse diffusion):

$$\frac{\partial u}{\partial t}=-c_x sign(u_{xx}), \quad (10)$$

with c_x the positive diffusion coefficient.

We can further understand (10) clearly by discretizing u_{xx} at a point (i) in the central difference scheme:

$$(u_x)_i=(u_{i+1}-u_{i-1})/2h, \quad (11)$$

$$(u_{xx})_i=(u_{i+1}+u_{i-1}-2u_i)/h^2=3(\bar{u}_i-u_i)/h^2,$$
$$\bar{u}_i=(u_{i-1}+u_i+u_{i+1})/3, \quad (12)$$

with h the spacial step. When $u_i>\bar{u}_i$ (the local mean of u_i), u_i will increase by evolving (10); contrariwise, u_i will decrease. For **o**, it does not vary because $(u_{xx})_o\approx 0$. This makes edges and corners (singularities) emphasized ("stretch- ed") while preserving the locations of edges. At the same time, we can see that the forward flow means "flowing to the local mean", while the backward flow means "flowing from the local mean".

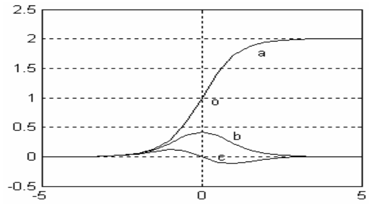


Fig.1 The differentials of 1D typical slope edge **a**, with center **o**, and the first and second order differential curves **b**, **c**.

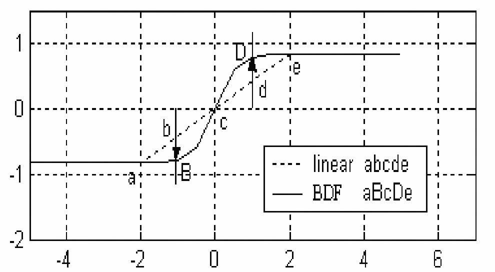


Fig.2 Edge enhanced BDF process (the solid line **aBcDe**), compared with linear interpolation (the broken line **abcde**).

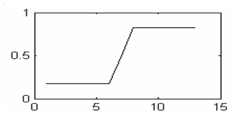
2) Comparison with FAB. Here we will clarify the differences between our scheme and the forward-and-backward (FAB) diffusion process [13]. In [13], FAB performs selectively forward or backward diffusion according to local gradients of the image. However, the bidirectional flow (BDF) performs differently forward or backward diffusion according to tangent or normal directions to the local isophote lines of the image.

In 1D signal case, when we adopt FAB to a step edge, we find that overshoot or ringing artifacts appear on the edge (see Fig.3). Thus, to decide diffusion speed c_x by the gradient information does not work. In section 1), we see that the backward flow means "flowing from the local mean", which manifests itself as increasing of u_{xx} at overshoot pixels more and more largely with iteration times. For suppressing this plague, we add the second order derivative information to c_x:

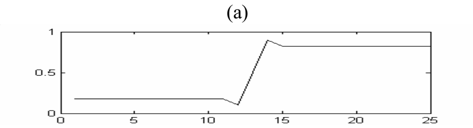
$$c_x = |u_x| / (1 + l_1 u_{xx}^2), \tag{13}$$

with l_1 a constant.

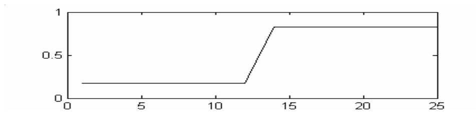
In fig.3, a step edge (a) is interpolated by FAB (b) and BDF (c) respectively. Having suppressed overshoots successfully by our scheme is shown.



(a)



(b)



(c)

Fig.3 BDF processing of a 1D step edge: (a) original 1D step edge; (b) result by FAB; (c) result by BDF, l_1 = 0.008. Both are obtained after 25 time steps.

3) Comparison with ADSF. Now we proceed our discussion with 2D image interpolation and enhancement. In [14], indicating edges by the zero-crossing is a binary decision process, by which, unfortunately, the obtained result is a false piecewise constant image whose texture and fine details are lost (see Fig.5). For this reason, we substitute $sign(s)$ by a hyperbolic tangent function $th(s)$, controlling softly the variety of gray levels of the image beside the edge center, and propose a backward flow of the form:

$$\frac{\partial u}{\partial t} = -c_n th(l u_{nn}), \tag{14}$$

with diffusion coefficient:

$$c_n = |u_n| / (1 + l_1 u_{nn}^2), \tag{15}$$

where $th(x) = (e^x - e^{-x}) / (e^x + e^{-x})$ is a hyperbolic tangent function, l is a constant to control its gradient.

B. *Forward flow*

1) Forward flow. Conventional interpolations also result in artifacts ("jaggies") in the image [16]. Now we perform a forward flow (normal diffusion) along the tangent directions to the isophote lines (edges):

$$\frac{\partial u}{\partial t} = c_t u_{tt}, \tag{16}$$

with c_t the diffusion coefficient. Because the forward flow means "flowing to the local mean", (16) can produce smooth level curves of the interpolated image [see Fig.4].

2) Comparison with level-set reconstruction. An image magnification method using level-set reconstruction is presented in [16], where instead of assuming a smoothness prior for the underlying intensity function, it assumes smooth- ness of the level curves, and produces appealing visually images. However, with $c_t = 1$, it may smooth away corners and small details at the same time. From (12), we can see that u_{tt} at the angle is much bigger than one at the edge line in value along the tangent direction. Therefore, for preventing over smoothness to corners, we add the second order derivative information to c_t:

$$c_t = 1 / (1 + l_2 u_{tt}^2), \tag{17}$$

with l_2 a constant.

In Fig.4, it shows that the weighted forward flow (17) not only has better smoothed image contours, but also has preserved three corners of the triangle in a synthetic image (b), and yet which has been over smoothed by the level-set reconstruction method (a).

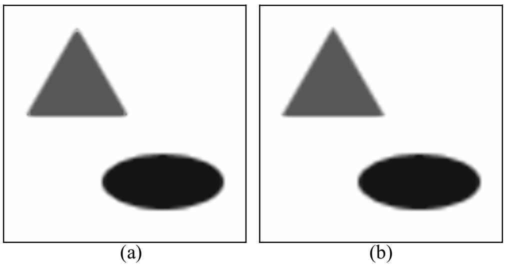


Fig.4 Interpolating a synthetic image: (a) the level-set reconstruction, (b) the weighted forward flow (17).

C. *Bidirectional flow*

Based on preceding discussion, we write the adaptive feature preserving bidirectional flow as:

$$\frac{\partial u}{\partial t} = \alpha(-c_n th(lu_{nn})) + \beta(c_t u_{tt}), \qquad (18)$$

with Neumann boundary condition, backward and forward flow control coefficients α, β, where we adopt the following diffusion coefficients:

$$c_n = |u_n| / (1 + l_1 u_{nn}^2), \quad c_t = 1 / (1 + l_2 u_{tt}^2). \qquad (19)$$

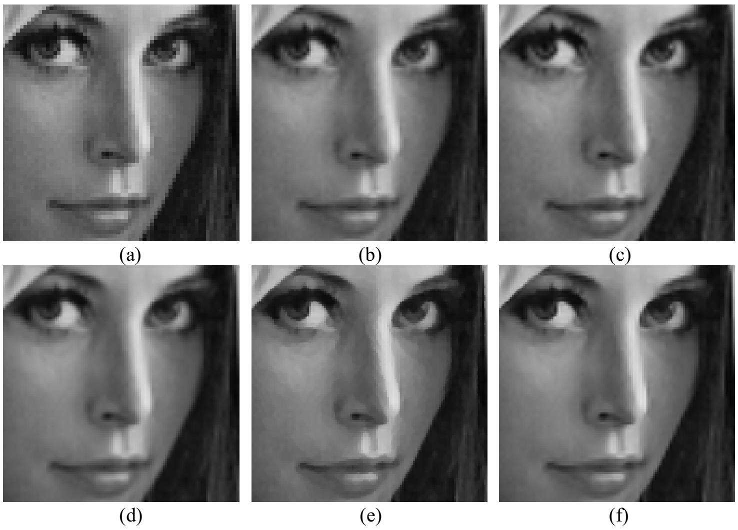


Fig.5 BDF processing of Lena image, compared with others: (a) nearest, (b) bilinear, (c) bicubic, (d) the level-set reconstruction, (e) ADSF, and (f) BDF.

Ⅳ. EXPERIMENTAL RESULTS

We used the explicit Euler method with the central difference scheme. A number of images have been used to test our scheme (18). Examples shown in Fig.5 are Lena image, where we interpolate it by a factor 2 with parameters: $[l, l_1, l_2]$= [200, 0.008, 0.003], $[\alpha, \beta]$ = [3, 2].

It is generally agreed that peak signal-to-noise ratio (PSNR) does not always provide an accurate measure of the visual quality for natural images [6, 7]. Therefore, we shall only rely on subjective evaluation to assess the visual quality of the interpolated images in this paper. Comparing with the level-set reconstruction, ADSF, and conventional interpolation methods: nearest, bilinear and bicubic, it can be seen that the best visual quality is obtained by interpolating the image using the proposed method, which preserves most features of the image, and produces sharp edges and smooth contours (see Lena's brim, cheek and eyeballs).

Ⅴ. CONCLUSIONS

This paper presents an adaptive feature preserving bidirectional flow process, by which we not only can effectively sharpen edges, but also can smooth contours of the image. Preserving image features such as edges, corners and textures, this method produces better visual results of the interpolated images than conventional interpolations.

REFERENCES

[1] K. R. Castleman, Digital Image Processing. Prentice Hall, 1995.
[2] S.W. Lee and J.K. Paik, "Image interpolation using adaptive fast B-spline filtering," In Proc. IEEE Int. Conf. Acoustics, Speech, Signal Processing, vol. 5, pp. 177–180, 1993.
[3] S. Carrato, G. Ramponi, and S. Marsi, "A simple edge-sensitive image Interpolation filter," In Proc. IEEE Int. Conf. Image Processing, vol. 3, pp. 711–714, 1996.
[4] K. Jensen and D. Anastassiou, "Subpixel edge localization and the Interpolation of still images," IEEE Trans. on Image Processing, vol. 4, pp. 285–295, Mar. 1995.

[5] J. Allebach and P.W. Wong, "Edge-directed interpolation," In Proc. IEEE Int. Conf. Image Processing, vol. 3, pp. 707–710, 1996.

[6] S. Battiato, G. Gallo, F. Stanco, "A locally adaptive zooming algorithm for digital images," Image Vision and Computing, Elsevier Science. Inc., Vol. 20, pp. 805-812, 2002.

[7] Xin Li, M.T. Orchard, "New edge-directed interpolation," IEEE transactions on image processing, 10(10):1521-1527, 2001.

[8] K. Ratakonda, N. Ahuja, "POCS based adaptive image magnification," In Proc. IEEE Int. Conf. Image Processing, vol. 3:203–207, 1998.

[9] Zhu Chang-Qing, Wang Qian, etc, "Image magnification based on multi-band wavelet transformation," China Journal of Image and Graphics, 7(A)(3): 653-656, 2003.

[10] D.A. Florencio and R.W. Schafer, "Post-sampling aliasing control for natural images," In Proc. IEEE Int. Conf. Acoustics, Speech, Signal Processing, vol. 2, pp. 893–896, 1995.

[11] G. Aubert , P. Kornprobst. Mathematical Problems in Image Processing: Partial Differential Equations and the Calculus of Variations. Applied Mathematical Sciences, volume 147, Springer-Verlag, 2001.

[12] P. Perona, J. Malik, "Scale-space and edge detection using anisotropic diffusion," IEEE Trans. Pattern Anal. Machine Intell, 12(7): 629–639, 1990.

[13] G. Gilboa, N. Sochen, and Y.Y. Zeevi, "A forward-and-backward diffusion process for adaptive image enhancement and denoising," IEEE Trans. Image Processing, vol. 11, no. 7, pp. 689-703, 2002.

[14] L. Alvarez and L. Mazorra, "Signal and image restoration using shock filters and anisotropic diffusion," SIAM J. Numer. Anal., 31(2): 590–605, 1994.

[15] S.J. Osher and L.I. Rudin, "Feature-oriented image enhancement using shock filters," SIAM J. Numer. Anal., vol. 27, pp. 919–940, 1990.

[16] B.S. Morse and D. Schwartzwald, "Image magnification using level-set reconstruction," Proceedings of the 2001 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, vol.1:333-340, 2001.

Shujun Fu received the B.S. degree in power engineering and the M.S. degree in computational mathematics from Shandong University, Jinan, China, in 1990 and 1999 respectively. Since 1999 he is a lecturer of the School of Mathematics and System Science, Shandong University. He is currently pursuing the Ph.D. degree in signal and information processing at the Beijing Jiaotong University, Beijing, China. His research interests include image processing, partial differential equations, and wavelet analysis.

Qiuqi Ruan received the B.S. and M.S. degrees from Northern Jiaotong University in 1969 and 1981 respectively. From January 1987 to May 1990, he was a visiting scholar in the University of Pittsburgh, and the University of Cincinnati. Subsequently, he has been a visiting professor in USA for several times. Now he is a professor, doctorate supervisor and the head of the institute of information science in the school of computer and information technology, Beijing Jiaotong University, China. He is a senior member of IEEE. His main research interests include digital signal processing, computer vision, pattern recognition, and virtual reality etc.

Wenqia Wang is a professor and doctorate supervisor in the School of Mathematics and System Science, Shandong University, Jinan, China. His research interests include partial differential equations, numerical analysis.