An anisotropic nonlinear diffusion approach to image interpolation

S.Morigi‡, F.Sgallari‡
†Dept. Mathematics, University of Bologna, Italy
{morigi,sgallari}@dm.unibo.it

C. Zannoni ♮
♮CINECA, Casalecchio di Reno, Bologna, Italy
c.zannoni@cineca.it

A. Cappello†
†D.E.I.S, University of Bologna, Italy
acappello@deis.unibo.it

Keywords: Image interpolation, PDE, Image Processing.

Abstract

In medical imaging, a three dimensional (3D) object geometry together with its density features must often be reconstructed from serial cross sections. Usually, these cross sections are not evenly spaced and are affected by noise intrinsically linked to the acquisition method. In this paper we present a new approach to the image interpolation problem based on anisotropic nonlinear diffusion. The method filters the anisotropic noisy data and processes them to get a 3D isotropic data reconstruction of geometry and density features. Computational results with syntetic and computed tomographic (CT) real images are presented and discussed, focusing on X-Ray examinations of long bones.

1 Introduction

Image interpolation is an important technique in the field of the 3D reconstruction from cross sectional images. Medical techniques such as CT, magnetic resonance imaging (MRI) or direct sectioning, for example, generate a series of cross-sectional images from which 3D object geometry and density features have to be extracted and reconstructed. Usually, the cross sections are not evenly spaced in order to capture the most significant parts of the object, the number of scans is limited to reduce the radiation dose delivered to the patient, moreover, these images are often affected by the noise intrinsically linked to the acquisition method. For improving visualization and 3D-feature enhancement and reconstruction, the obtained anisotropic acquired data set needs to be filtered and interpolated to a isotropic image data set.

The aim of this work is to present a new approach to the image interpolation problem based on anisotropic nonlinear diffusion. The proposed nonlinear Partial Differential Equation (PDE) models, representing a multiscale analysis, filter and interpolate the sequence of slices preserving and enhancing structures and converting the acquired data set into data of uniform discretization. PDE models and methods have recently emerged in image processing as an alternative to traditional transform and statistically based methods [1, 5, 21, 22]. The application of traditional pre-processing algorithms (moving average, median and gaussian filtering) reduces the noise superimposed to the image but does not preserve a good definition of the details. The PDEs are designed to possess certain desirable geometrical properties and allow a more systematic approach for restoring images with sharp edges and segmentation. The preliminary results we have obtained investigating the use of a PDE approach to the image interpolation and reconstruction problem look promising even if some more work still needs to be done to get a suitable PDE model for the problem.

The paper is organized as follows. Section 2 introduces some basic notations together with a problem formulation. Section 3 presents classical interpolation techniques, applied either on slices either on objects. A brief introduction on the application of PDE models to image processing is presented in Section 4, while Section 5 provides a description of the PDE methods proposed for the interpolation problem. Section 6 presents some experimental results of using the proposed methods to process syntetic and CT cross sectional images.

2 Notation and problem statement

The input image sequence we need to elaborate can be considered both as a volume (or 3D image) and as a single two dimensional (2D) image evolving in time (image sequence).

A volume can be represented by a real function $u_0(x)$, $u_0 : \Omega \to \mathbb{R}$, where $\Omega \subset \mathbb{R}^3$ represents a spatial domain. A 2D image sequence can be modeled by a real function $u_0(x, \theta)$, $u_0 : \Omega \times [0, T] \to \mathbb{R}$, where $\Omega \subset \mathbb{R}^2$ represents a spatial domain, and $[0, T]$ the time interval.

A slice is equivalently represented by $u_0(x_1, x_2, x_3)$, with constant $x_3$ in the 3D image notation, and by $u_0(x_1, x_2, \theta)$ for a constant $\theta$ value, in the 2D image sequence.
The input data set we wish to interpolate consists of a sequence of $n$ slices. Interpolation transforms a volume (or 2D image sequence) $u_0$, into another volume (or 2D image sequence) $u$, composed by $m > n$ slices. After interpolation, in fact, $m - n$ new slices will appear in between the adjacent input ones.

3 Related work

Traditional interpolation techniques can be divided into two groups: scene-based and object-based. Scene-based methods determine pixel values for interpolated cross-sections directly from the density values of given slices. In object-based methods, some object information extracted from the slices, such as shape, are used in guiding the interpolation process. We should note that none of the methods belonging to these two classes deals with the denoising problem.

The most commonly used scene-based methods are the straightforward nearest-neighbor and linear interpolation [14, 6]. Higher order techniques such as spline-based image interpolation have been used to fill the interslice spaces [12], obtaining slightly superior results. However, if the cross-section morphology of individual structures changes or is displaced significantly from slice to slice, image intensities from one tissue type may be interpolated with those of another, producing artifacts in the images and in the resulting 3D reconstruction (see Figure 2 in Section 6). Hence, image interpolation methods of this class often do not provide an appropriate correspondence between structures of the same tissue type in adjoining slices, especially near the object boundaries. This problem is accentuated as the distance between adjacent slices increases. Summarizing, this class presents three major disadvantages: low frequency information is introduced, artifacts appear, structures and anatomy of the object tend to be lost. Thus, when objects are extracted and displayed from the interpolated image, these objects often exhibit a staircase and generally unsatisfactory appearance.

Object-based interpolation methods offer an improvement over scene-based interpolation [3, 13, 15, 7, 8]. Raya and Udupa in [18] have introduced the shape-based interpolation technique that has become the most relevant of the object-based interpolation class. These methods apply the distance transform to the grey-level images, then interpolate the distance maps between the slices and finally convert back into grey-level images to get the interpolated slices. The image space is interpolated by incorporating the correspondence between objects on adjacent slices that express changes in the shape and position. Recently, shape-based interpolation methods have been improved using distance transform and morphing [4]. Methods belonging to the object-based interpolation class preserve the anatomy and structures of the objects and solve the artifact problems. However, a fundamental problem with each of these techniques is that each computation is very time consuming. Moreover, they don’t use the original gray-scale information and have difficulties with images containing three-like structures (see Figure 3 in Section 6).

Our methods can be considered an example of object-based method that works directly on the density values of the objects. The method will be applied to a sequence of CT images of a femur bone. These tomographic slices present the following peculiarities: adjacent slices differ only for slight changes due to translations, rotations and scaling. This allows us to search for correspondence between points only over a small neighborhood of the reference point. Moreover, regions in consecutive slices have quite similar shape and area, so big variations can be only due to errors in the choice of the scanning sequence.

4 PDE for image processing

The main reason for using nonlinear diffusion filtering in image processing is provided by the very good results known in the literature in many fields of application. Unlike linear diffusion filtering (which is equivalent to convolving the original image with a Gaussian function) edges remain well localized and can even be enhanced. For mathematical fundamentals and an extensive literature on this class of filters we refer the reader to [21, 23].

PDEs turn to be the appropriate mathematical tool that leads to well-defined and powerful algorithms for preprocessing of raw data. In particular, PDEs encode adaptive behaviour in a purely data-driven way that is flexible enough to cope with the rich image structure commonly found in medical images.

Nonlinear PDE-based image processing has been introduced to the field of computer vision by Perona and Malik [17] and after them applied to the field of medical imaging (see [22, 19] and the references therein) and studied in theoretical and computational details (see [2, 1, 5]).

The image multiscale analysis, as it has been introduced in [1], associates with $u_0$ a family $u(t, x, \theta)$ of smoothed - simplified images (or a family of smoothed sequences) depending on an abstract parameter $t$, the scale. As it has been proved in [1], if such family fulfills basic assumptions - pyramidal structure, regularity and local comparison principle - then, it can be represented as a solution of a second order parabolic PDE with the initial condition given by the original noisy images $u_0(x, \theta)$.

In this paper we propose to process a single image (2D/3D) using the model due to Alvarez et al. in [2] represented by the nonlinear PDE:

$$\frac{\partial u}{\partial t} = g(|\nabla G_\sigma * u|)|\nabla u| \cdot \frac{\nabla u}{|\nabla u|},$$

(1)

while the model we consider for 2D image sequences (representing the 2D slices $\theta_i, i = 1, \ldots, m$ of the volume), has been introduced in [19], and is given by

$$\frac{\partial u}{\partial t} = c t u \nabla \cdot (g(|\nabla G_\sigma * u|) \nabla u).$$

(2)
These equations are accompanied with zero Neumann boundary conditions on the spatial boundary and suitable boundary conditions in time.

The parameters in (1–2) fulfill the following assumptions

- \( g \) is a continuous function, \( g(0) = 1 \) and \( 0 < g(s) \to 0 \) for \( s \to \infty \)
- \( G_\sigma \subseteq C^\infty(\mathbb{R}^N) \) is a smoothing kernel, \( N = 2, 3 \), \( \int_{\mathbb{R}^N} G_\sigma(x) dx = 1 \), \( G_\sigma(x) \to \delta_x \) for \( \sigma \to 0 \), \( \delta_x \) - the Dirac measure at point \( x \).

Common choices for the two previous parameters are e.g.

\[
g(s) = \frac{1}{1 + K * s^2}
\]

with some constant \( K \), and \( N \)-dimensional Gauss function

\[
G_\sigma(x) = \frac{1}{(2\sqrt{\pi} \sigma)^N} e^{-\frac{|x|^2}{\sigma^2}}.
\]

The first equation (1) is related to the so called morphological multiscale analysis models and it has the peculiarity of preserving image geometrical features like silhouettes in the smoothing process given by diffusion [2].

Notice that the case \( g(s) \equiv 1 \) reduces equation (1) to the well-known level set equation proposed by Osher & Sethian [16, 20]. The level set equation moves each level set of \( u \) (namely, level line in 2D and level surface in 3D) in the normal direction with velocity proportional to its mean curvature field. Moreover, it yields the so called morphological principle: if \( u \) is a solution then, for any nondecreasing function \( \varphi \), \( \varphi(u) \) is a solution as well.

The term \( |\nabla u| \nabla \cdot (\frac{\nabla u}{|\nabla u|}) \) in (1) represents a degenerate diffusion term, which diffuses \( u \) in the direction orthogonal to its gradient \( \nabla u \), and does not diffuse at all in the direction of \( \nabla u \). The function \( g(s) \) depending on \( |\nabla G_\sigma * u| \) is an edge indicator and is used for the "enhancement" of the edges.

The second equation (2) was proposed in [19] and applied to 3D echocardiographic image sequences in order to consider a time coherence of successive frames. The proposed model combines ideas of the regularized Perona–Malik anisotropic diffusion model [17] and Galilean invariant movie multiscale analysis equation of Alvarez, Guichard, Lions and Morel [1]. It filters the space-time image sequence while respecting the spatial as well as temporal coherent structures. The assumptions considered in [19] are that the structures in the image are formed by points that preserve their intensity along the motion trajectory. Such objects are called Lambertian structures. Moreover it is assumed that motion is Galilean locally in time, i.e. the trajectories are smooth.

The term \( clt(u) \) has been presented by Guichard in order to respect the Lambertian structures; the name \( clt \), in fact, indicates the relation to the curvature of Lambertian trajectory. Guichard in [1, 10] proposed the following quantity:

\[
clt(u) = \min_{w_1, w_2} \frac{1}{\Delta \theta^2} \left| \nabla u, w_1 - w_2 > \right| + \quad (5)
\]

where \( w_1, w_2 \) are arbitrary vectors in \( \mathbb{R}^N \), \( \nabla u \) denotes partial derivatives with respect to all space variables and \( \Delta \theta \) is the time increment.

The scalar function \( clt(u) \) introduces a measure of coherence in time for the moving structures, in such a way that low values of \( clt(u) \) correspond high coherence in time. The value of \( clt(u) \) is small for the so-called Galilean trajectories, i.e. for Lambertian points moving by a constant velocity field. On the other hand, for noisy points, there is no motion coherence and thus the value of \( clt(u) \) will be large there. It consists of the sum of three positive parts and we want to find the minimum in all possible directions \( w_1, w_2 \). The last two terms in the sum on the right hand side of (5) are related to the differences in the intensities of end-points of candidate Lambertian velocity vectors \( w_1, w_2 \). To find the directions of such vectors we look at the points which have the closest intensity value to the intensity \( u(x, \theta) \) in the previous frame (term \( |u(x - w_1, \theta - \Delta \theta) - u(x, \theta)| \) in the next frame (term \( |u(x + w_2, \theta + \Delta \theta) - u(x, \theta)| \)). Note that, if we find corresponding Lambertian points both terms vanish. The first term in the sum, namely \(|\nabla u, w_1 - w_2|/|\Delta \theta|^2\), corresponds to the so called apparent acceleration, i.e. to the difference between candidate Lambertian velocity vectors \( w_1 \) and \( w_2 \) in the direction of \( \nabla u \). For details we refer to [1].

The spatial smoothing in equation (2) is obtained by a slight modification of the well-known Perona–Malik equation called also anisotropic diffusion in computer vision community. It selectively diffuses an image in the regions where the signal is of constant mean in contrast with those regions where the signal changes its tendency. Such diffusion process is governed by the shape of function \( g \) and by its dependence on \( \nabla u \) which is in a sense an edge indicator [17].

Catté et al. introduced in [5] the convolution with Gaussian kernel \( G_\sigma \) and they have established the well-posedness of the mathematical formulation.

5 Outline of the proposed method

Applying PDE models to solve the image interpolation problem, we require to preserve points moving by a constant velocity field through the slices and keeping their intensity along the motion trajectory. In the spatial variables, that is in each slice, the smoothing process is desirable to conserve edges and filter out the noise and the artifacts which appear in the initial reconstructed images due to the linear interpolation process.

To respect these requirements we have chosen the nonlinear degenerate diffusion equation of mean curvature flow type (1) for the processing of the entire volume, and PDE equation model (2) for the processing of the 2D image sequence.

In input our methods consider a sequence of images \( u_0(\cdot, \theta), \quad \theta = 1, \ldots, m \), with \( m \) the total number of slices;
Figure 1: The original slices (1-9) and the interpolated slices (1-9) between the slices (1,9) of the truncated cone volume, using PDE model (2).

This sequence is considered as a volume for model (1) and as a sequence of 2D images for model (2). Some of these slices are the input data set, and we will refer to these as reference slices, while in between we consider the slices coming from a trivial interpolation method, such as, for example, nearest-neighbor or linear interpolation. We suppose that, under a certain resolution, the shape of objects should change smoothly. The proposed algorithm processes the 2D image sequence (or volume) and is able to distinguish the noise from the contours of the different structures by using in a distinct way spatial and sequence coherence. In this approach, correspondence is found between points in consecutive reference slices, and then this correspondence is used to estimate data between the slices. The method takes advantage of the fact that consecutive slices have small geometric differences and carries out the search in predicted small neighborhoods.

We consider the model equation (2) where the \( \text{clt}(u) \) has been modified as follows:

\[
\text{clt}(u) = \min_{w_1,w_2} \frac{1}{(\alpha \Delta \theta)^2} \left( |\nabla u, w_1 - w_2| + (6) \right)
\]

\[
|u(x - w_1, \theta - \alpha \Delta \theta) - u(x, \theta)| + |u(x + w_2, \theta + \alpha \Delta \theta) - u(x, \theta) |
\]

where \( \alpha \) is used to choose as previous and next slices only the reference slices. Moreover, we added to the right-hand side of equation (2) the term

\[
f(u - u_r)
\]

where \( u_r \) represents the nearest reference slice to \( u \) and \( f \) is a Lipschitz continuous, non-decreasing function, such that \( f(0)=0 \). By means of \( f \) on the right-hand side of (2), the solution \( u \) is forced to be close to \( u_r \).

The discretization of the presented methods is based on semi-implicit approximation in scale and finite volume in space [19, 11]. Such discretization leads to the solution of linear systems of equations for each slice of the sequence at each discrete scale step. This can be done efficiently by using preconditioned iterative methods [19].

6 Examples

Experiments were performed on synthetic and real sequences of images. The synthetic sequence was selected to present the features of the algorithm. First the PDE model (2) was applied on slices of size 64 \( \times \) 64 that contain circular cuts of an object that can be described as a truncated cone (see Figure 1). We used a scale step of size 0.01, and we looked at a 5 \( \times \) 5 neighborhood to compute \( \text{clt}(u) \). There are 7 interpolated slices generated. The 9 first images starting from the top left corner are the original noisy images, while the other 9 images are the result after 6 scale steps of the proposed PDE method (2), with initial conditions given by the result of a simple linear interpolation between slices 1 and 9.

Figure 2: 3D display of interpolated slices: (a) original data set (b) grey-level linear interpolated data set.

In the following experiments we have applied the PDE approach to an \textit{in vitro} acquired CT sequence of a human femur.
selecting a subset of \( n \) scan locations upon the set of \( m \) available slices. This would simulate the in vivo investigations where the number of CT scans has to be limited because it determines the quantity of radiation absorbed by the patient [24]. The 3D data set was obtained by scanning the cadaver femur in vitro. The femur was positioned with the longitudinal axis parallel to the CT scanning direction and was scanned with a 1mm step on a GE Sytec 3000 scanner. Thus, the 3D data set consists of 420 CT images corresponding to the femur longitudinal dimension of 420mm. Each slice is a \( 512 \times 512 \) gray level image.

The first experiment on the CT images shows isosurface visualization of the femur data set. The results displayed in Figure 2, show the original data set of the femur (Figure 2(a)) and the reconstructed data set using linear interpolation using step 10 (see Figure 2(b)). Both data sets consist of 200 slices of size \( 256 \times 256 \) with pixel grey levels in the range \([0, 4095]\).

The second CT sequence of slices (1-4) is presented in Figure 3. We show in Figure 4 the result of applying shape-based interpolation algorithm using slices (1,4) as reference slices. Notice the lost of gray-level details in the reconstructed slices in between. The result of processing PDE model (1) on the same data set is shown in Figure 5 after 4 scale steps.

The third experiment on real CT sequences is illustrated in Figure 6 and Figure 7. Reconstructed slices using PDE model (2) are illustrated in Figure 6, while the reconstruction obtained by shape-based interpolation is shown in Figure 7.

In Figure 6 results are shown from the reconstruction of a part of the femur bone using 8 reference slices equally spaced for reconstructing a sequence of 40 slices. The 3D display of isosurfaces has been obtained by shape-based interpolation (Figure 6(a)), by linear interpolation (Figure 6(b)), and by the PDE model (2) (Figure 6(c)). Significantly smoother and less ”staircasing” representations of the femur bone are produced using PDE models than those created using linear interpolation. Finally we should notice that these PDE methods turn to be more efficient than the computationally expensive shape-based interpolation techniques. The efficiency of the latter technique is function of the number of different gray levels in each slice, while the PDE approach is influenced only by the dimensions of the slice. In these preliminary experimentations we have noticed that PDE methods perform better 20 times than the shape-based ones.

7 Conclusion

In this paper, we have dealt with the important problem of image sequence interpolation by applying the PDE based methodology, which has been already proved to be very successful in anisotropically restoring images and segmentation. Our approach allows us to deal simultaneously with the problem of denoising and interpolation. Results demonstrate that the algorithm provides good estimation of interpolated data from a visual qualitative point of view. Future work will
Figure 6: Interpolated slices: PDE model (2) interpolation.

Figure 7: Interpolated slices: shape-based interpolation.
investigate a serious comparison of the performance of the methods on the basis of quantitative analysis, according to the common figures of performance merit [9].

References


Figure 8: 3D display of interpolated slices: (a) shape based interpolation (b) grey-level linear interpolation (c) PDE model (2) interpolation.


