Int J Comput Vis (2009) 84: 130–145
DOI 10.1007/s11263-008-0132-x

Reversible Interpolation of Vectorial Images by an Anisotropic Diffusion-Projection PDE

Anastasios Roussos · Petros Maragos

Received: 21 September 2007 / Accepted: 14 March 2008 / Published online: 26 July 2008
© Springer Science+Business Media, LLC 2008

Abstract In this paper, a nonlinear model for the interpolation of vector-valued images is proposed. This model is based on an anisotropic diffusion PDE and performs an interpolation that is reversible. The interpolation solution is restricted to the subspace of functions that can recover the discrete input image, after an appropriate smoothing and sampling. The proposed nonlinear diffusion flow lies on this subspace while its strength and anisotropy adapt to the local variations and geometry of image structures. The derived method effectively reconstructs the real image structures and yields a satisfactory interpolation result. Compared to classic and other existing PDE-based interpolation methods, our proposed method seems to increase the accuracy of the result and to reduce the undesirable artifacts, such as blurring, ringing, block effects and edge distortion. We present extensive experimental results that demonstrate the potential of the method as applied to graylevel and color images.

Keywords Partial differential equations (PDEs) · Nonlinear anisotropic diffusion · Diffusion-projection PDE · Image interpolation · Vector-valued images · Reversibility · Sampling · Anti-aliasing filter

The authors acknowledge the financial support of the Future and Emerging Technologies (FET) programme 'ASPI' within the Sixth Framework Programme for Research of the European Commission, under FET-Open contract No. 021324.

A. Roussos (✉) · P. Maragos
School of ECE, National Technical University of Athens, 15773 Athens, Greece
e-mail: troussos@cs.ntua.gr

P. Maragos
e-mail: maragos@cs.ntua.gr

1 Introduction

Image interpolation is among the fundamental image processing problems and is often required for various image analysis operations. It is therefore of interest for many applications such as biomedical image processing, aerial and satellite imaging, text recognition, high definition video and high quality image display and printing. In this paper, the term *image interpolation* is used in the sense of the operation that takes as input a discrete image and recovers a continuous image or a discrete one with higher resolution. The case where the output image is discrete appears in the literature with several other names: *digital zooming*, *image magnification*, *upsampling*, *resolution enhancement*.

There exists a large variety of image interpolation methods, which can be classified in two main classes, linear and nonlinear methods (see Meijering 2002 for a detailed review). The *linear* methods (e.g. bicubic, quadratic and spline interpolations) perform convolution of the image samples with a single kernel, equivalent to a lowpass filtering. These methods yield fast algorithms, but they cannot effectively reconstruct the high-frequency part of images and inevitably introduce artifacts. *Nonlinear* methods perform a processing adapted to the local geometric structure of the image, with main goal to accurately reconstruct image edges. This class includes variational (e.g. Guichard and Malgouyres 1998; Malgouyres and Guichard 2001; Aly and Dubois 2005) and PDE-based (e.g. Belahmidi and Guichard 2004; Tschumperlé and Deriche 2005) methods, some of which will be presented in the following sections. Such methods have also been developed for two closely related problems, image inpainting (Bertalmio et al. 2000; Caselles et al. 1998; Chan and Shen 2002) and scattered data interpolation (Weickert and Welk 2006).

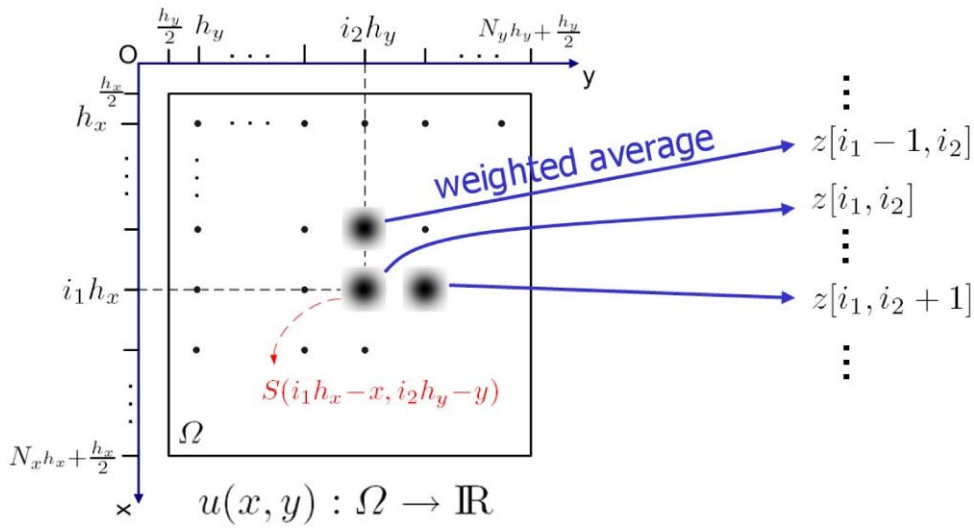


Fig. 1 Reversibility condition for the unknown interpolation solution $u(\boldsymbol{x})$

In this paper a novel nonlinear method for the interpolation of vector-valued images is proposed. We pose a constraint, which effectively exploits the available information of the input image. Then, we design a nonlinear anisotropic diffusion PDE, which performs adaptive smoothing but also complies with this constraint, thanks to an appropriate projection operation. The diffusion strength and anisotropy adapt to the local variations and geometry of image structures. This method yields a plausible result even when the resolution of the input image is relatively low and reduces the artifacts that usually appear in image interpolation. The paper is organized as follows: In Sect. 2, some interpolation models related to the proposed method are discussed. Section 3 presents our novel interpolation PDE model. In Sect. 4, we demonstrate and comment the results from various interpolation experiments, that show the efficacy of the new method.

2 Preliminaries and Background

2.1 Reversibility Condition for Interpolation

The problem of image interpolation is viewed here in a way similar to (Guichard and Malgouyres 1998; Malgouyres and Guichard 2001). The continuous solution of the interpolation $u(x, y)$ should yield the known, low resolution discrete image $z[i_1, i_2]$, after an anti-aliasing lowpass filtering followed by sampling. Namely, it is required that the operation of interpolation is reversible.

To pose this *reversibility condition* formally, let us consider that $z[i_1, i_2]$ takes real values and is defined on an orthogonal grid of $N_x \times N_y$ points with vertical and horizontal steps h_x and h_y respectively. Also let $u(\boldsymbol{x})$, where $\boldsymbol{x} = (x, y)$, be defined in the orthogonal domain $\Omega = h_x[\frac{1}{2}, N_x + \frac{1}{2}] \times h_y[\frac{1}{2}, N_y + \frac{1}{2}]$, which contains the grid points.

Then, the reversibility condition for the solution $u(\boldsymbol{x})$ can be written as (see also Fig. 1):

$$(S * u)(i_1 h_x, i_2 h_y) = z[i_1, i_2],$$
$$\forall (i_1, i_2) \in \{1, \ldots, N_x\} \times \{1, \ldots, N_y\}, \tag{1}$$

where "$*$" denotes continuous 2D convolution and hereafter we set $h_x = h_y = 1$. Also, $S(\boldsymbol{x}) : \mathbb{R}^2 \to \mathbb{R}$ is a smoothing kernel that performs the lowpass filtering. Let $\hat{S}(\omega_1, \omega_2) : \mathbb{R}^2 \to \mathbb{C}$ be the 2D Fourier transform of $S(\boldsymbol{x})$. We assume that $\hat{S}(\omega_1, \omega_2)$ takes nonzero values for all the baseband frequencies, i.e.:

$$\hat{S}(\omega_1, \omega_2) \neq 0, \quad \forall (\omega_1, \omega_2) \in [-\pi, \pi]^2. \tag{2}$$

For example, $S(\boldsymbol{x})$ could be the *mean kernel*, i.e. $S(\boldsymbol{x}) = \mathbb{1}_{[-\frac{1}{2}, \frac{1}{2}]^2}(\boldsymbol{x})$, where $\mathbb{1}_B(\boldsymbol{x})$ denotes the indicator function for any set $B \subset \mathbb{R}^2$, or an *isotropic Gaussian kernel* with appropriate variance.

Although the convolution values in (1) are mainly influenced from the values of $u(\boldsymbol{x})$ inside Ω, an extension of $u(\boldsymbol{x})$ to all $\mathbb{R}^2$ must be assumed, in order that $S * u$ is well-defined. In the subsequent presentation, we consider two alternatives for this extension:

Extension 1 (*Simple Periodization*) This is the straightforward choice that is made implicitly when Fourier analysis of such signals is applied (Oppenheim et al. 1984; Dudgeon and Mersereau 1984), but unfortunately it brings close image values from distant regions of Ω.

Extension 2 (*Reflection and then Periodization*)[1] This is a standard choice in image analysis that avoids the dis-

[1] First, $u(\boldsymbol{x})$ is extended to $\widetilde{\Omega} = [-N_x + \frac{1}{2}, N_x + \frac{1}{2}] \times [-N_y + \frac{1}{2}, N_y + \frac{1}{2}]$ by reflection across the lines of image borders $\{(x, y) : x = 0.5\}$ and $\{(x, y) : y = 0.5\}$:

$$u(0.5 \pm x, 0.5 \pm y) = u(0.5 + x, 0.5 + y), \quad \forall \boldsymbol{x} \in [0, N_x] \times [0, N_y].$$

Then, $u(\boldsymbol{x})$ is extended to $\mathbb{R}^2$ by periodization:

$$u(x + k_1 \cdot 2N_x, y + k_2 \cdot 2N_y) = u(\boldsymbol{x}), \quad \text{for all } (k_1, k_2) \in \mathbb{Z}^2.$$

advantage of simple periodization, as $u(\boldsymbol{x})$ is continuous in image borders. When used for the solution of a PDE, this is equivalent to imposing zero Neumann boundary conditions. This extension will be used in our proposed method.

Therefore, the extended $u(\boldsymbol{x})$ is $(\widetilde{N}_x, \widetilde{N}_y)$-periodic, where $(\widetilde{N}_x, \widetilde{N}_y) = (N_x, N_y)$ for Extension 1, whereas $(\widetilde{N}_x, \widetilde{N}_y) = (2N_x, 2N_y)$ for Extension 2. In addition, we consider that the discrete input $z[i_1, i_2]$ is also extended to all $\mathbb{Z}^2$, using the same type of extension as in $u(\boldsymbol{x})$. Thus, (1) becomes valid for all $(i_1, i_2) \in \mathbb{Z}^2$.

Note that (1) degenerates to the exact interpolation condition when $S(\boldsymbol{x}) = \delta(\boldsymbol{x})$ (2D unit impulse). However, condition (1) with an appropriate smoothing kernel, such as Gaussian, can be more realistic, as it can better model the digitization process, which is the final step of image acquisition systems (Aly and Dubois 2005). In addition, such a lowpass filtering is desirable, as it reduces the aliasing effects in the acquired image.

2.2 Reversibility Condition in Frequency Domain

We can derive a simple expression of the reversibility condition in the frequency domain, useful for the subsequent analysis. So, let: (i) $\hat{u}_{m_1,m_2} : \mathbb{Z}^2 \to \mathbb{C}$ be the 2D Fourier series coefficients of the $(\widetilde{N}_x, \widetilde{N}_y)$-periodic extension of $u(\boldsymbol{x})$ (using either Extension 1 or Extension 2) and (ii) $\hat{z}_{n_1,n_2} : \mathbb{Z}^2 \to \mathbb{C}$ be the discrete-space 2D Fourier series coefficients of the extended $z[i_1, i_2]$, using the same method of extension as in $u(\boldsymbol{x})$. Note that, since $z[i_1, i_2]$ is discrete, $\hat{z}_{n_1,n_2}$ is $(\widetilde{N}_x, \widetilde{N}_y)$-periodic.

Using the convolution property of the 2D Fourier transform (Oppenheim et al. 1984; Dudgeon and Mersereau 1984), it can be shown that $S * u(\boldsymbol{x})$ is also $(\widetilde{N}_x, \widetilde{N}_y)$-periodic and its Fourier series coefficients are $\hat{S}(\frac{2\pi m_1}{\widetilde{N}_x}, \frac{2\pi m_2}{\widetilde{N}_y}) \cdot \hat{u}_{m_1,m_2}$. The extended $z[i_1, i_2]$ is derived from sampling $S * u(\boldsymbol{x})$, therefore we can conclude (Oppenheim et al. 1984; Dudgeon and Mersereau 1984) that the reversibility condition (1) is equivalent to the following:

$$\sum_{(k_1,k_2)\in\mathbb{Z}^2} \left\{ \hat{S}\left(\frac{2\pi}{\widetilde{N}_x}(n_1 + k_1\widetilde{N}_x), \frac{2\pi}{\widetilde{N}_y}(n_2 + k_2\widetilde{N}_y)\right) \cdot \hat{u}_{n_1+k_1\widetilde{N}_x, n_2+k_2\widetilde{N}_y} \right\} = \hat{z}_{n_1,n_2}, \quad \forall (n_1, n_2) \in \mathbb{Z}^2. \tag{3}$$

Note that, because of the periodicity of both sides of condition (3), it is sufficient for $\hat{u}_{m_1,m_2}$ to satisfy this condition only for (n_1, n_2) that belong to any $\widetilde{N}_x \times \widetilde{N}_y$ rectangular grid. Additionally, if the unknown image $u(\boldsymbol{x})$ in condition (1) is considered to be real, which is a desirable constraint for the interpolation solution, then $\hat{u}_{m_1,m_2}$ must also satisfy:

$$\hat{u}_{-m_1,-m_2} = \overline{\hat{u}_{m_1,m_2}}, \quad \forall (m_1, m_2) \in \mathbb{Z}^2,$$

where $\overline{(\cdot)}$ denotes complex conjugation.

The problem of finding $u(\boldsymbol{x})$ in (1) or equivalently (3) is ill-posed, as the reversibility condition is satisfied by infinitely many functions. Let $\mathcal{U}_{z,S}$ be the set of these functions. It is clear that some extra criterion must be posed to choose among the functions of $\mathcal{U}_{z,S}$.

2.3 Frequency Zero-Padding Interpolation

A simple linear interpolation method arises by imposing the additional constraint that $u(\boldsymbol{x})$ is a bandpass 2D signal, similarly to Shannon's theory. Then, the solution of (1), which we refer to as *frequency zero-padding interpolation*, is unique and can be easily derived using the expression (3).

Let us consider only the case that $u(\boldsymbol{x})$ is extended to $\mathbb{R}^2$ using Extension 2. Then, the bandpass constraint can be written as:

$$\hat{u}_{m_1,m_2} = 0, \quad \forall (m_1, m_2) \notin \{-N_x, \ldots, N_x\} \times \{-N_y, \ldots, N_y\} \tag{4}$$

In addition, because of the reflection applied to $z[i_1, i_2]$, it follows that $\hat{z}_{\pm N_x,n_2} = 0$, $\forall n_2$ and $\hat{z}_{n_1,\pm N_y} = 0$, $\forall n_1$. Therefore, we can also state that $\hat{u}_{\pm N_x,m_2} = 0$, $\forall m_2$ and $\hat{u}_{m_1,\pm N_y} = 0$, $\forall m_1$. Consequently, $\hat{u}_{m_1,m_2}$ is finally constrained to take nonzero values only in $B_1 = \{-N_x + 1, \ldots, N_x - 1\} \times \{-N_y + 1, \ldots, N_y - 1\}$.

Thus, if we evaluate the condition (3) at the values $(n_1, n_2) \in B_1$, only the term of the sum that corresponds to $(k_1, k_2) = (0, 0)$ is nonzero. In conclusion, the Fourier series of frequency zero-padding are given by:

$$\hat{u}_{m_1,m_2} = \begin{cases} \dfrac{\hat{z}_{m_1,m_2}}{\hat{S}(\frac{2\pi m_1}{\widetilde{N}_x}, \frac{2\pi m_2}{\widetilde{N}_y})}, & \text{if } (m_1, m_2) \in B_1, \\ 0, & \text{if } (m_1, m_2) \in \mathbb{Z}^2 \setminus B_1, \end{cases} \tag{5}$$

since the values in the denominator $\hat{S}(\frac{2\pi m_1}{\widetilde{N}_x}, \frac{2\pi m_2}{\widetilde{N}_y}) \neq 0$, because of (2).

In the case of Extension 1, an expression very similar to (5) can be derived. The only difference in this case is that, if N_x or N_y is even, the definition of values $\hat{u}_{m_1,m_2}$ at the borders of bandpass zone is more complicated.

The frequency zero-padding interpolation reconstructs image edges without significantly blurring or distorting

them, but usually introduces strong oscillations around edges (Malgouyres and Guichard 2001). The cutoff of high frequencies that it performs is undesirable, as the bandlimited assumption is not true for most real-world images. Consequently, a more appropriate method of selection among the functions of $\mathcal{U}_{z,S}$ is needed. Such methods will be presented in the following sections.

2.4 Total Variation Based Interpolation

The authors of (Guichard and Malgouyres 1998; Malgouyres and Guichard 2001) proposed to choose as solution of the interpolation the image that minimizes the Total Variation (TV), $E[u] = \iint_\Omega \|\nabla u(\boldsymbol{x})\| \mathrm{d}\boldsymbol{x}$, under the constraint that $u \in \mathcal{U}_{z,S}$. This minimization problem is solved in (Guichard and Malgouyres 1998) by applying a constrained gradient descent flow, described by the following PDE:

$$\frac{\partial u(\boldsymbol{x},t)}{\partial t} = P_{\mathcal{U}_{0,S}}\left\{\operatorname{div}\left(\frac{\nabla u}{\|\nabla u\|}\right)\right\}, \tag{6}$$

supplemented with the initial condition that $u(\boldsymbol{x},0)$ is the frequency zero-padding interpolation of $z[i_1,i_2]$. We note that, in order to use the reversibility condition, this method makes the implicit assumption that $u(\boldsymbol{x})$ is extended to $\mathbb{R}^2$ using Extension 1. $P_{\mathcal{U}_{0,S}}\{\cdot\}$ denotes the operator of orthogonal projection on the subspace $\mathcal{U}_{0,S}$, which corresponds to the condition (1) with $z[i_1,i_2]=0$ for all (i_1,i_2). This projection ensures that $u(\boldsymbol{x},t) \in \mathcal{U}_{z,S}, \forall t > 0$, since $u(\boldsymbol{x},0) \in \mathcal{U}_{z,S}$. The authors propose two options for the smoothing kernel of condition (1): (i) the *mean kernel* $\mathbb{1}_{[-\frac{1}{2},\frac{1}{2}]^2}(\boldsymbol{x})$ or (ii) the *sinc kernel*, which provides an ideal lowpass filter as its Fourier transform is $\mathbb{1}_{[-\pi,\pi]^2}(\omega_1,\omega_2)$.

This method leads to reconstructed images without notable blurring effects, as it allows discontinuities and preserves 1D image structures. However, TV minimization is based on the assumption that the desirable image is almost piecewise constant, which yields a result with oversmoothed homogeneous regions. In addition, the diffusion in (6) is controlled by the simple coefficient $1/\|\nabla u\|$, therefore it cannot remove block effects, especially in the regions with high image variations. Further, the mean kernel vanishes too sharply, so the corresponding projection $P_{\mathcal{U}_{0,S}}\{\cdot\}$ reintroduces block effects and the sinc kernel is badly localized in space and oscillates, so $P_{\mathcal{U}_{0,S}}\{\cdot\}$ causes formation of oscillations in reconstructed edges.

2.5 Belahmidi-Guichard (BG) Method

Belahmidi and Guichard (2004) have tried to improve the TV-based interpolation by developing a nonlinear anisotropic PDE, hereafter referred as *BG interpolation method*. In order to enhance edge preservation, this PDE performs a diffusion with strength and orientation adapted to image structures. The PDE is initialized by the simple zero order hold (ZOH), which yields strong block effects. The reversibility condition (1) is taken into account (with the choice of mean kernel for $S(\boldsymbol{x})$) by adding to the PDE an appropriate fidelity term, so that the flow $u(\boldsymbol{x},t)$ stays close to the subspace $\mathcal{U}_{z,S}$ (see Belahmidi and Guichard 2004 for details). This method balances linear zooming on homogeneous regions and anisotropic diffusion near edges, trying to combine the advantages of these two processes. Nevertheless, the diffusion is not always desirably adapted to real image structures and the fact that the PDE flow is not constrained to lie inside $\mathcal{U}_{z,S}$ may decrease the accuracy of the result.

2.6 PDE Model of Tschumperlé and Deriche (TD)

In (Tschumperlé 2002; Tschumperlé and Deriche 2005), the authors propose an effective PDE method for vector-valued image regularization. This PDE scheme, which we refer to as *TD PDE*, is mainly designed for image restoration applications, but it is presented here because we utilize it to the design of the new interpolation PDE (Sect. 3). This model is an anisotropic diffusion flow, which uses tensors to adapt the diffusion to the image structure. Let $\boldsymbol{u}(\boldsymbol{x},t) = [u_1,\ldots,u_M]^{\mathrm{T}}$ be the output vector-valued image at (artificial) time t and M be the number of vector components. Then, the TD PDE model can be described by the following set of coupled PDEs:

$$\frac{\partial u_m(\boldsymbol{x},t)}{\partial t} = \operatorname{trace}\big(T(J_\rho(\nabla \boldsymbol{u}_\sigma))\, D^2 u_m\big), \quad m = 1,\ldots,M, \tag{7}$$

with initial condition that $\boldsymbol{u}(\boldsymbol{x},0)$ is the input vector-valued image. $D^2 u_m$ denotes the spatial Hessian matrix of the component $u_m(\boldsymbol{x},t)$ and T is the 2×2 *diffusion tensor*:

$$T(J_\rho(\nabla \boldsymbol{u}_\sigma)) = \big[1+(\mathcal{N}/K)^2\big]^{-\frac{1}{2}} \boldsymbol{w}_- \boldsymbol{w}_-^{\mathrm{T}} + \big[1+(\mathcal{N}/K)^2\big]^{-1} \boldsymbol{w}_+ \boldsymbol{w}_+^{\mathrm{T}}, \tag{8}$$

where $\mathcal{N} = \sqrt{\lambda_+ + \lambda_-}$ and K is a threshold constant similar to the constant in the diffusivity of (Perona and Malik 1990).[2] Also, $\lambda_- \le \lambda_+$ and $\boldsymbol{w}_-, \boldsymbol{w}_+$ are the eigenvalues and unit eigenvectors of the 2×2 *structure tensor*:

$$J_\rho(\nabla \boldsymbol{u}_\sigma) = G_\rho * \sum_{m=1}^{M} \nabla(G_\sigma * u_m)\,(\nabla(G_\sigma * u_m))^{\mathrm{T}}. \tag{9}$$

[2]This is a slightly more general version of the original model (Tschumperlé 2002; Tschumperlé and Deriche 2005), where $K=1$.

The 2D isotropic Gaussian kernels $G_\sigma(\boldsymbol{x})$ and $G_\rho(\boldsymbol{x})$ are of standard deviation σ and ρ respectively:[3]

$$G_\sigma(\boldsymbol{x}) \triangleq \frac{1}{2\pi\sigma^2}\exp\left(-\frac{x^2+y^2}{2\sigma^2}\right).$$

The structure tensor $J_\rho(\nabla \boldsymbol{u}_\sigma)$ measures the local geometry of image structures (Weickert 1998). Due to the convolutions with G_σ and G_ρ, $J_\rho(\nabla \boldsymbol{u}_\sigma)$ is insensitive to image details smaller than $O(\sigma)$ and is affected by the image variation within a neighborhood of size $O(\rho)$. The eigenvectors $\boldsymbol{w}_-$ and $\boldsymbol{w}_+$ describe the orientation of minimum and maximum vectorial variation of $\boldsymbol{u}$ and the eigenvalues λ_- and λ_+ describe measures of these variations. The term $\mathcal{N}$ derived from these eigenvalues is an edge-strength predictor that effectively generalizes the gradient magnitude $\|\nabla u\|$. Thus, the diffusion is strong and isotropic in homogeneous regions (small $\mathcal{N}$), but weak and mainly oriented by image structures near the edges (big $\mathcal{N}$). Consequently, this method offers a flexible and effective control on the diffusion process (see Tschumperlé and Deriche 2005 for more details).

Among various applications, the authors of (Tschumperlé 2002; Tschumperlé and Deriche 2005) also apply their generic PDE model to image interpolation (we refer to the derived method as *TD interpolation method*). This method casts image interpolation as a special case of the image inpainting problem (Bertalmio et al. 2000). It imposes the constraint that the solution must coincide with the input at the appropriate pixels in the new finer grid (exact interpolation condition). Thus, the inpainting domain (i.e. the domain where the image values are unknown) consists of the remaining pixels. The image values in this domain are processed according to PDE (7), with a modified diffusion tensor (Tschumperlé 2002):

$$T(J_\rho(\nabla \boldsymbol{u}_\sigma)) = \left[1+(\mathcal{N}/K)^2\right]^{-\frac{1}{2}} \boldsymbol{w}_- \boldsymbol{w}_-^{\mathrm{T}}. \tag{10}$$

The bilinear interpolation of the input image is chosen as initial condition $\boldsymbol{u}(\boldsymbol{x},0)$ and the interpolation solution is derived from the equilibrium state. Contrary to the effectiveness of the TD PDE model for image restoration, the derived interpolation method suffers from some inefficiencies. The initialization by the bilinear interpolation contains edges with significant blurring. Also, the information of each input value $z[i_1,i_2]$ is not spread to all the corresponding pixels of the finer grid: During the PDE evolution, the values of $\boldsymbol{u}$ are fixed at the pixels of the initial coarser grid but vary without any constraint at the rest pixels. Furthermore, the diffusion tensor (10) is fully anisotropic even in regions with small image variations, therefore it may distort image structures and create false edges.

[3]The original model corresponds to $\sigma = 0$, but we use the more general version of (Weickert 1998).

3 Proposed Anisotropic Diffusion-Projection PDE

The aforementioned PDE interpolation methods usually perform better than classic linear methods, as they reconstruct the edges without significantly blurring them. In some cases though, they yield artifacts such as over-smoothing of homogeneous regions, block effects or edge distortion. In order to improve the effectiveness of these methods, we propose a novel PDE model that performs a nonlinear interpolation based on a combination of the reversibility condition approach and nonlinear anisotropic diffusion. The model is designed to deal with vector-valued images in general and processes the different channels in a coupled manner.

More precisely, the design of our model has been based on the observation that the TV-based interpolation PDE (6) can be derived from a non-minimization point of view: it is in fact a modification of the zero-fidelity ($\lambda = 0$) TV PDE (Rudin et al. 1992):

$$\frac{\partial u(\boldsymbol{x},t)}{\partial t} = \operatorname{div}\left(\frac{\nabla u}{\|\nabla u\|}\right), \tag{11}$$

which can be viewed as a special case of the general nonlinear diffusion of (Perona and Malik 1990). This modification is done by replacing the right hand side (RHS) of the PDE with its projection to $\mathcal{U}_{0,S}$. Thanks to this projection, the whole flow remains into the subspace $\mathcal{U}_{z,S}$, provided that $u(\boldsymbol{x},0) \in \mathcal{U}_{z,S}$.

We propose to follow a similar approach, using the RHS of some other, more effective for this application, PDE instead of the TV PDE (11). In (Roussos and Maragos 2007), we have used the RHS of the TD PDE (7), which performs a more flexible regularization. In this paper though, we use the RHS of a modified version of the TD PDE, namely:

$$\frac{\partial u_m(\boldsymbol{x},t)}{\partial t} = \operatorname{div}\big(T(J_\rho(\nabla \boldsymbol{u}_\sigma))\,\nabla u_m\big), \quad m = 1,\ldots,M, \tag{12}$$

with the tensor $T(J_\rho(\nabla \boldsymbol{u}_\sigma))$ given again by (8). This modification is based on the general anisotropic diffusion model of (Weickert 1998). As our experiments revealed (c.f. Sect. 4), using this divergence-based expression of anisotropic diffusion instead of the trace-based version (7), the interpolation result is slightly improved, mainly in small regions where the tensor T has a strong spatial variation. Thus, in the divergence-based expression, the adaptation of the anisotropic diffusion to the image structures seems to be more desirable.

At this point, we must note that the PDE (12), whose RHS we use in the design of our proposed method, is only one possible choice. Any other regularization PDE could be used and the subsequent analysis of the method would be very similar.

3.1 Reversibility Condition for the Proposed Model

We use a straightforward generalization of condition (1) to vector-valued images: Every channel of the interpolation solution $\boldsymbol{u}(\boldsymbol{x})$ is constrained to satisfy (1) independently. This generalized reversibility condition can thus be written as:

$$(S * u_m)(i_1, i_2) = z_m[i_1, i_2], \tag{13}$$

for all points $(i_1, i_2) \in \{1, \ldots, N_x\} \times \{1, \ldots, N_y\}$ and channels $m \in \{1, \ldots, M\}$. $z_m[i_1, i_2]$ and $u_m(\boldsymbol{x})$ are the m-th of M components of the discrete input and interpolated image respectively. Let $\boldsymbol{\mathcal{U}}_{z,S}$ be the set of vector-valued images $\boldsymbol{u}(\boldsymbol{x})$ that satisfy the generalized reversibility condition (13).

Note that we consider that $\boldsymbol{u}(\boldsymbol{x})$ is extended to $\mathbb{R}^2$ using Extension 2 (see Sect. 2.1). Also, in the reversibility condition (13) we use a *Gaussian smoothing kernel* $S(\boldsymbol{x}) = G_r(\boldsymbol{x})$ of standard deviation r. With this choice, the digitization process during the image acquisition can be reliably modeled. The standard deviation r must be neither too small nor too big. If r is too small, $S(\boldsymbol{x})$ is too localized in space and the information of each input value $z[i_1, i_2]$ is not spread properly to the neighborhood of the point (i_1, i_2). On the other hand, if r is too big, $S(\boldsymbol{x})$ is too extended in space and performs a lowpass filtering with very small bandwidth, therefore the reversibility condition is not realistic.

Compared to the mean and sinc kernel of TV based and BG methods (see Sects. 2.4 and 2.5), the Gaussian kernel $G_r(\boldsymbol{x})$ leads to a more efficient version of the reversibility condition, when an appropriate r is chosen. More precisely, it avoids the sharp cutoff either in space domain (encountered by the mean kernel) or frequency domain (encountered by the sinc kernel), therefore it vanishes smoothly and is well localized in space.

The use of the Gaussian smoothing kernel is a slight difference from the previous version of our work: In (Roussos and Maragos 2007), we had used a *truncated* Gaussian kernel, which has support of size $h_x \times h_y = 1 \times 1$ in the continuous space (same as the size of discrete input's pixels). In fact, the use of Gaussian without truncation is an improvement, as the truncation may introduce a rather sudden cutoff in the kernel (although the mean kernel is much more sudden). This truncation was done in (Roussos and Maragos 2007) in order to derive a simple expression for the projection operator in the space domain. Here though, we derive a *generic* expression for the projection (without need to make any assumption about the kernel's support), using a more sophisticated analysis in frequency domain (see Sect. 3.3 and Appendix).

3.2 PDE Scheme of the Proposed Model

We derive the interpolated image from the equilibrium solution of the following system of coupled PDEs (which we name *Anisotropic Diffusion-Projection PDE*):

$$\frac{\partial u_m(\boldsymbol{x}, t)}{\partial t} = P_{\mathcal{U}_{0,S}}\big\{\operatorname{div}\big(T(J_\rho(\nabla \boldsymbol{u}_\sigma))\,\nabla u_m\big)\big\}, \quad m = 1, \ldots, M, \tag{14}$$

where $P_{\mathcal{U}_{0,S}}\{\cdot\}$ denotes the operator of orthogonal projection on the subspace $\mathcal{U}_{0,S}$ and the tensors $T(J_\rho(\nabla \boldsymbol{u}_\sigma))$ and $J_\rho(\nabla \boldsymbol{u}_\sigma)$ are again given by (8) and (9) respectively. We have chosen the following initial conditions for (14): every $u_m(\boldsymbol{x}, 0)$ is derived from the frequency zero-padding interpolation of $z_m[i_1, i_2]$ (5). Note that $\boldsymbol{u}(\boldsymbol{x}, 0) \in \boldsymbol{\mathcal{U}}_{z,S}$, so $P_{\mathcal{U}_{0,S}}\{\cdot\}$ ensures that $\boldsymbol{u}(\boldsymbol{x}, t) \in \boldsymbol{\mathcal{U}}_{z,S}$, $\forall t > 0$.

The above initialization, which is similar to the one of PDE (6) proposed in (Guichard and Malgouyres 1998), can be easily computed numerically and contains quite efficient reconstructions of image edges (see also the discussion in Sect. 3.4). As mentioned above, we assume that $u(\boldsymbol{x})$ is extended to $\mathbb{R}^2$ using the Extension 2 instead of the Extension 1. The reflection that is added to the Extension 2 offers a slight improvement to the frequency zero-padding, as it eliminates the ringing effects near the image borders.

3.3 Expression for the Projection Operator

It is clear that an expression for the projection operator in (14) is needed. As we show in the Appendix, the projection $P_{\mathcal{U}_{0,S}}\{v\}$ of an $(\widetilde{N}_x, \widetilde{N}_y)$-periodic function $v(\boldsymbol{x})$ on the subspace $\mathcal{U}_{0,S}$ can be derived from the following equations:

$$P_{\mathcal{U}_{0,S}}\{v\} = v(\boldsymbol{x}) - w(\boldsymbol{x}), \tag{15}$$

where $w(\boldsymbol{x})$ is also an $(\widetilde{N}_x, \widetilde{N}_y)$-periodic function with 2D Fourier series coefficients:

$$\hat{w}_{m_1,m_2} = \Bigg\{ \sum_{(k_1,k_2)\in\mathbb{Z}^2} \hat{\phi}\left(\frac{2\pi m_1}{\widetilde{N}_x} + k_1 2\pi, \frac{2\pi m_2}{\widetilde{N}_y} + k_2 \pi\right) \cdot \hat{v}_{m_1+k_1\widetilde{N}_x, m_2+k_2\widetilde{N}_y} \Bigg\} \cdot \hat{\phi}\left(\frac{2\pi m_1}{\widetilde{N}_x}, \frac{2\pi m_2}{\widetilde{N}_y}\right), \tag{16}$$

and $\phi(\boldsymbol{x})$ is a function with Fourier transform:

$$\hat{\phi}(\omega_1, \omega_2) = \Bigg\{ \sum_{(k_1,k_2)\in\mathbb{Z}^2} \left|\hat{S}(\omega_1 + k_1 2\pi, \omega_2 + k_2 2\pi)\right|^2 \Bigg\}^{-\frac{1}{2}} \cdot \overline{\hat{S}(\omega_1, \omega_2)}. \tag{17}$$

Note that most of the spectral energy of $\phi(\boldsymbol{x})$ is concentrated in low frequencies. In practice, the definition (16) of

$w(\boldsymbol{x})$ in the frequency domain is more suitable for the implementation of the projection. But we can also define it in the space domain, in order to obtain more intuition. Thus, using the convolution property of the 2D Fourier transform and the relation between the 2D Fourier transforms of a signal and its sampled version (Oppenheim et al. 1984; Dudgeon and Mersereau 1984), it can be shown that (16) can be equivalently expressed as:

$$w(\boldsymbol{x}) = \big(\mathrm{III} \cdot (\phi * v)\big) * \phi(\boldsymbol{x}), \tag{18}$$

where $\mathrm{III}(x, y) = \sum_{(i,j)} \delta(x-i, y-j)$ is a 2D impulse train for sampling on a rectangular grid with unit horizontal and vertical sampling periods. As a simple example, if $S(\boldsymbol{x})$ is the sinc kernel, then $\phi(\boldsymbol{x}) = S(\boldsymbol{x})$ and the above equation reduces to $w = \phi * v$; i.e. w becomes a bandlimited version of v and $P_{\mathcal{U}_{0,S}}\{v\}$ contains only the highpass part of v.

It follows from (17) that ϕ is real. Therefore, if v is real, then $w(\boldsymbol{x})$ and $P_{\mathcal{U}_{0,S}}\{v\}$ are also real, because of (18) and (15). This is the desirable property for the application studied in this paper, as we deal with images that take real values and we want the interpolation result to take also real values.

In the special case where the smoothing kernel $S(\boldsymbol{x})$ is the Gaussian kernel $G_r(\boldsymbol{x})$, as we have chosen in our method, then $\hat{S}(\omega_1, \omega_2) = \frac{2\pi}{r^2} G_{1/r}(\omega_1, \omega_2)$. From this, one can compute an analytic expression for $\hat{\phi}(\omega_1, \omega_2)$ and then use it in (16).

3.4 Properties of the Model

As already mentioned, the frequency zero-padding interpolation, which we use as initial condition of (14), reconstructs image edges without significantly blurring or distorting them, but also introduces strong oscillations around edges (c.f. Malgouyres and Guichard 2001). It can thus be viewed as a desirable interpolation result degraded by a significant amount of noise. The scope of the proposed PDE (14) is to effectively regularize the image $\boldsymbol{u}(\boldsymbol{x}, 0)$ by removing these oscillations.

Also note that (15) shows that the projection $P_{\mathcal{U}_{0,S}}\{v\}$ subtracts the component of v that does not comply with the reversibility condition. This subtraction does not affect the basic characteristics of the regularization that the velocity $v_m = \mathrm{div}(T \nabla u_m)$ tends to apply to the image. Therefore, similarly to the TD PDE (7), the anisotropic tensor T forces the diffusion to be strong and isotropic in homogeneous regions, but relatively weak and oriented by image structures, near the edges. In this way, the proposed PDE removes the undesirable oscillations and simultaneously preserves the important image structures. Namely, the proposed PDE can be considered as a diffusion flow towards elements of $\boldsymbol{\mathcal{U}}_{z,S}$ with "better" visual quality.

Additionally, we have experimentally observed that the best regularized image is usually reached at a finite time, where practically the flow reaches an equilibrium (see Fig. 8(a) for an example), thanks to the term that $P_{\mathcal{U}_{0,S}}\{v\}$ subtracts from the velocity v. Therefore, there is no need to specify the stopping time as an additional parameter.

3.5 Numerical Implementation

In the numerical implementation, the continuous result $\boldsymbol{u}(\boldsymbol{x})$ of the proposed model is approximated by a discrete image $\boldsymbol{u}[i_1', i_2']$, defined to a finer grid than the input image $z[i_1, i_2]$. Thus, $\boldsymbol{u}[i_1', i_2']$ provides a discrete interpolation (upsampling) of the input $z[i_1, i_2]$. We consider only the case where the vertical and horizontal grid steps of $z[i_1, i_2]$ are multiples of the grid steps of $\boldsymbol{u}[i_1', i_2']$ by an integer factor d, which we call *zoom factor*. Namely, the input image is magnified $d \times d$ times. For the sake of simplicity, we hereafter assume that the finer grid of $\boldsymbol{u}[i_1', i_2']$ has unit steps, hence the grid of input $z[i_1, i_2]$ has steps d.

To discretize the PDE (14), we use an explicit numerical scheme with finite differences, similar to (Tschumperlé 2002). The discrete time step δt is chosen sufficiently small for stability purposes (the typical value of $\delta t = 0.2$ is used). Due to the fact that the output image is given at the equilibrium, we stop the iterative process when $\boldsymbol{u}^{n+1}$ differs from $\boldsymbol{u}^n$ by a small constant, with respect to an appropriate norm, such as the root mean square (RMS) value over all the grid points and image channels.

Note that, since during the numerical implementation of the proposed PDE the derived interpolation solution is discrete instead of continuous, we can develop a totally discrete framework for the reversibility condition, the frequency zero-padding and the projection operator, very similar to the continuous framework presented in Sects. 2.1, 2.3 and 3.3. The discrete expressions derived from this framework have an exact correspondence to their continuous counterparts that we have presented. Therefore, in our experiments we implement the frequency zero-padding interpolation and the projection operator based on this discrete framework and using the 2D Fast Fourier Transform.

4 Experimental Results and Comparisons

4.1 Framework for the Experiments

In order to compare the interpolation methods and extract performance measures, we use the following framework: We choose a reference image with a relatively good resolution and negligible noise. We reduce the dimensions of this image by an integer factor d (i.e. the image is reduced to $\frac{1}{d} \times \frac{1}{d}$ of its size), using a decimation process, i.e. (anti-aliasing) lowpass filtering followed by discrete sampling. We implement the lowpass filtering by a convolution with a bicubic

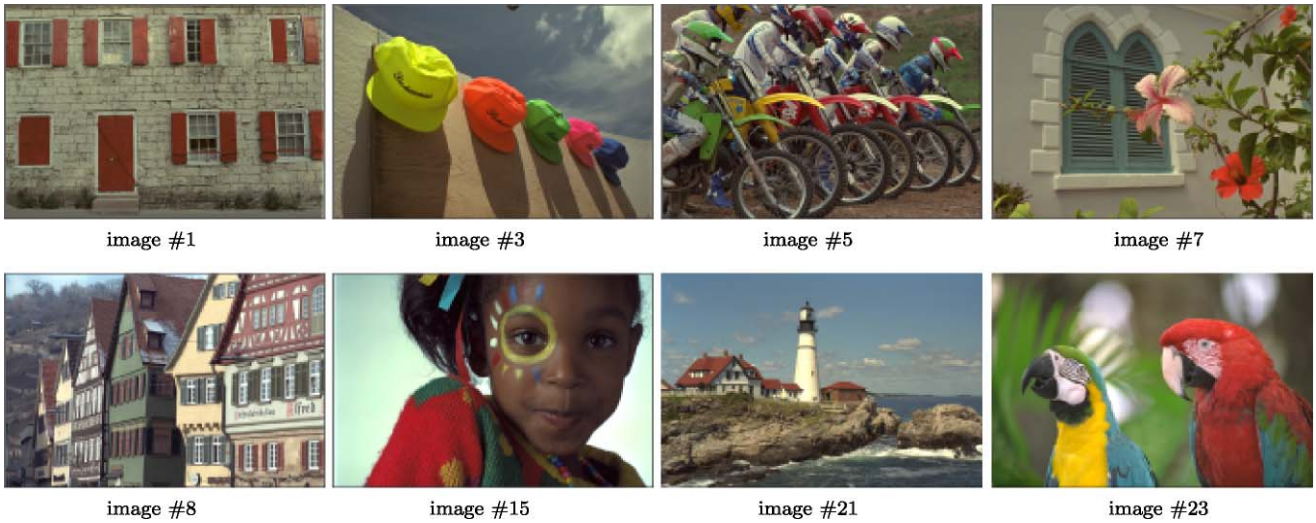


Fig. 2 Color versions of 8 out of 23 reference images from the dataset used (768 × 512 pixels per image)

Fig. 3 Details of reference images used in the demonstrations of next figures. (**a**) Detail of image #7 (graylevel version). (**b**) Detail of image #5 (color version)

spline, which results to a reliable decimation process. Finally, we apply the interpolation methods to enlarge the decimated image by the zoom factor d, so that the output images have the same size as the reference image. Note that we used the range [0, 255] for image values and in the case of color, we applied the methods representing the images in the RGB color space. The reference image can be considered as the ideal output of the interpolation, as it is noiseless. Therefore the difference between the reference image $\boldsymbol{r}[i_1, i_2]$ and the output of a method $\boldsymbol{u}[i_1, i_2]$ can be viewed as reconstruction error and is quite representable of the method's performance.

We use two measures for this error, the classic peak signal-to-noise-ratio (PSNR)[4] and the mean structural similarity (MSSIM) index (Wang et al. 2004), which seems to better approximate the perceived visual quality of an image than PSNR or various other measures. MSSIM index takes values in [0, 1] and increases as the quality increases.[5] In the case of color images, we extend MSSIM with the simplest way: we calculate the MSSIM index of each RGB channel and then take the average.

We repeat the above procedure for different reference images from a dataset and for zoom factors $d = 2, 3$ and 4. For every zoom factor and interpolation method, we compute the averages of PSNR and MSSIM for all the images in the set, which we consider as final measures of performance.

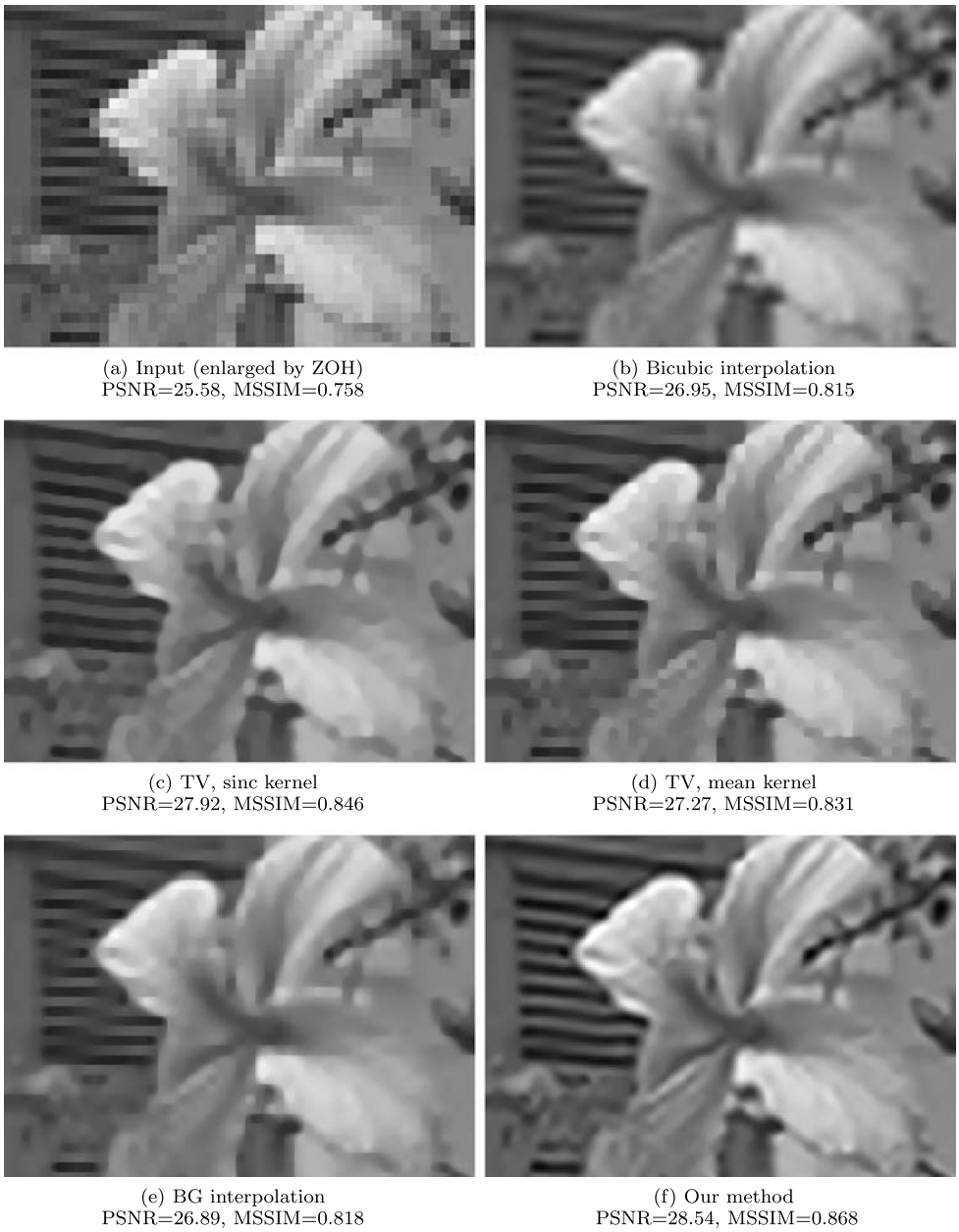
We followed the above experimental framework using a dataset of 23 natural images of size 768 × 512 pixels[6] (see Figs. 2 and 3). We run two series of experiments: (i) for the graylevel versions of images, where we applied the bicubic, the TV-based, the BG method, the previous (Roussos and Maragos 2007) and the current version of our method[7] and

[4]We use the definition $\mathrm{PSNR} = 10\log_{10}(255^2 M/\mathrm{var}\{\|\boldsymbol{u}[i_1, i_2] - \boldsymbol{r}[i_1, i_2]\|\})$, where $\|\cdot\|$ denotes here the Euclidean norm of vectors with M components.

[5]We calculate the MSSIM index based on the code available at http://www.cns.nyu.edu/~lcv/ssim/, using the default parameters.

[6]We took this dataset from http://www.cipr.rpi.edu/resource/stills/kodak.html.

[7]Recall that the current version has two slight differences from the version of (Roussos and Maragos 2007) (see Sect. 3): (i) it uses a Gaussian



(a) Input (enlarged by ZOH)
PSNR=25.58, MSSIM=0.758

(b) Bicubic interpolation
PSNR=26.95, MSSIM=0.815

(c) TV, sinc kernel
PSNR=27.92, MSSIM=0.846

(d) TV, mean kernel
PSNR=27.27, MSSIM=0.831

(e) BG interpolation
PSNR=26.89, MSSIM=0.818

(f) Our method
PSNR=28.54, MSSIM=0.868

Fig. 4 Visual details and error measures of 4×4 graylevel interpolation results using image #7 as reference (see Fig. 3(a))

(ii) for their color counterparts, where we applied the bicubic, the TD method and again the previous and current version of our method. In the second series, we did not apply the TV-based and BG methods because they are designed for the interpolation of graylevel images only.

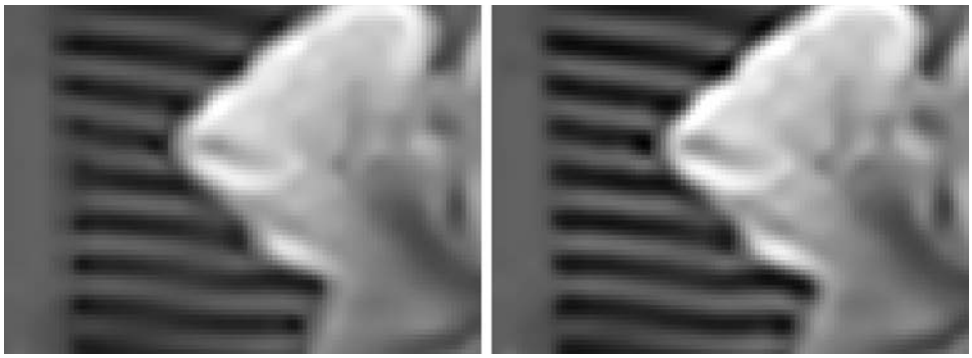
Note that we implemented the other PDE-based interpolation methods with a way similar to the implementation of the proposed method, as briefly described in Sect. 3. For the methods that needed specification of parameter(s), we utilized fixed values in all the dataset, which we empirically derived based on the quality of the results, as perceived by a human observer. We have hence chosen the parameters $\sigma = 0.3d$, $\rho = 0.4d$, $r = 0.35d$ and $K = 1$ for both versions of our method. Also, in the TD method we used the same values $\rho = 0.4d$, $K = 1$ and in the BG method we used $K = 3$ for the corresponding threshold constant.

A complete demonstration of the experimental results using the above dataset can be found at: http://cvsp.cs.ntua.gr/~tassos/PDEinterp/ssvm07res.

4.2 Examples from the Experimental Results

Figure 4 is an example of the results for graylevel image interpolation. The input image, which is a 4×4 decima-

smoothing kernel instead of *truncated* Gaussian and (ii) the PDE of the model contains a divergence-based diffusion term instead of a trace-based one.



(a) Our method, previous version [14] PSNR=28.08, MSSIM=0.852

(b) Our method, current version PSNR=28.54, MSSIM=0.868

Fig. 5 Comparison of the two versions of our method, for the interpolation example of Fig. 4 (small details of 95 × 70 pixels)

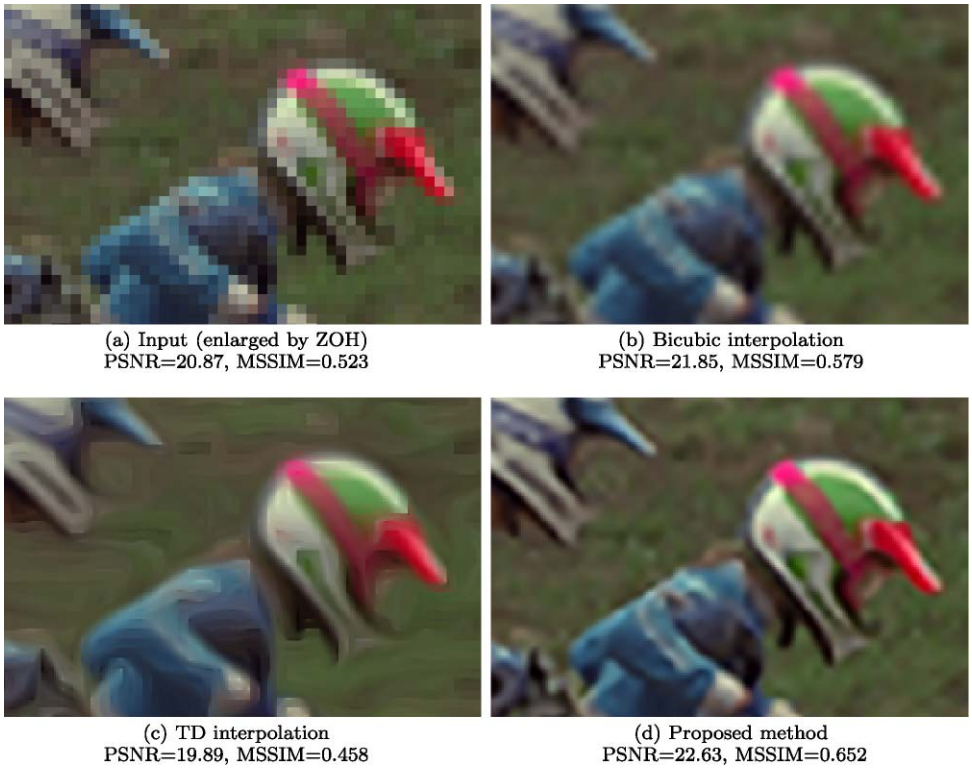
tion of the reference of Fig. 3(a), is shown in Fig. 4(a). For the sake of demonstration, it has been enlarged by the simple zero order hold (ZOH), whose error measures are displayed below this image. It can be observed that the bicubic interpolation significantly blurs the edges (e.g. note the flower boundary in Fig. 4(b)). The TV-based interpolation over-smooths some homogeneous areas (e.g. the interior of the flower in Fig. 4(c)), creates block effects (e.g. the thin black branch at the upper right of Fig. 4(d)) and oscillations in reconstructed edges (e.g. the shutter behind the flower in Fig. 4(c)). In the result of BG interpolation, the block effects are reduced, but they are still present in some regions (e.g. the branch at the upper right of Fig. 4(e)). In addition, this result is more blurred than the results of TV-based method. Finally, Fig. 4(f) shows that our proposed method yields the most effective reconstruction of image structures, with the less intense block and blurring effects. For example, observe in Fig. 4 how the shutter is most desirably reconstructed by our method. Also, quantitatively, our method performs the best PSNR and MSSSIM measures.

Note that, if we had also demonstrated the previous version (Roussos and Maragos 2007) of our method in the example of Fig. 4, then the result would have looked almost identical with the result of the current version (Fig. 4(f)), as these two results exhibit slight differences only. But in Fig. 5, which shows an even smaller detail of these results, the differences become clearer. More precisely, we see that in the current version of our method, the reconstructed edges are even sharper and better localized (compare e.g. the reconstructions of the shutter's parallel line segments). This improvement is due to the different smoothing kernel used, as in the current version this kernel vanishes more smoothly, but also due to the fact that the divergence-based operator makes the diffusion term to adapt more effectively to image structures than the trace-based operator.

Figure 6 demonstrates an example of the results for color image interpolation. We observe that the bicubic interpolation gives again a result with blurring but also significant staircase effects. Figure 6(c) shows that TD interpolation yields an excessively synthetic aspect to the result, as it has distorted image edges and created false thin edges around the real ones. Again, the result of the proposed method (Fig. 6(d)) seems the most aesthetically satisfying and has the highest performance measures. This result contains sharper and better localized edges than the bicubic interpolation (e.g. note the more effective reconstruction of casques' edges) and looks much more natural than the result of TD interpolation.

Let us now provide more details about the application of the proposed method to the above experiment, which will better explain how this method outputs an effective result. Figure 7 shows the evolution of the proposed PDE flow (14). In the frequency zero-padding interpolation (Fig. 7(b)), which is used as initialization, the reconstructed edges are fairly sharp and have plausible shapes. But there are also strong oscillations, because all the high frequencies have been set to zero. In the intermediate result of Fig. 7(c), the oscillations have been removed from some parts of the image, but they are still present near strong edges. In the final result of Fig. 7(d), it seems that they have been totally removed from the image. This is due to the anisotropic diffusion term of our PDE. We also observe that the reconstructed edges of frequency zero-padding are preserved during the whole PDE evolution, without any undesirable distortion or blurring (e.g. note the similarity between the casque boundaries of Figs. 7(b), (c) and (d)). This is achieved thanks to the effective adaptation of the anisotropic diffusion tensor to the image structures, but also thanks to the projection operator of the proposed PDE (14), which increases the accuracy of the result.

In Fig. 8, we can see the variations of 3 measurements during the evolution of the proposed PDE, when applied to the aforementioned example. In Fig. 8(a), the RMS of PDE solution's rate of change $\frac{\partial \boldsymbol{u}}{\partial t}(x, y, n\delta t)$ (computed from the discretization of PDE's RHS) is plotted. We observe that this quantity is descending during the evolution and finally tends to zero, which indicates that the flow equilibrates when $t \to \infty$. This is a desirable property, as it provides an evidence about the stability of the proposed PDE scheme



(a) Input (enlarged by ZOH)
PSNR=20.87, MSSIM=0.523

(b) Bicubic interpolation
PSNR=21.85, MSSIM=0.579

(c) TD interpolation
PSNR=19.89, MSSIM=0.458

(d) Proposed method
PSNR=22.63, MSSIM=0.652

Fig. 6 Visual details and error measures of 4×4 color interpolation results using image #5 as reference (see Fig. 3(b))

(a) Input (enlarged by ZOH)

(b) initialization, $t = 0$ (freq. zero-padding)

(c) 20 iterations, $t = 4$

(d) 280 iterations, $t = 56$ (final result)

Fig. 7 Details from the evolution of the proposed PDE flow (14) when applied to the 4×4 color interpolation using image #5 as reference (the time step $\delta t = 0.2$)

and its numerical implementation. Figures 8(b) and 8(c) show that both PSNR and MSSIM measures are increasing during the PDE evolution. This property is desirable as well, since it means that the image with the best quality is derived from the equilibrium, which is the final result of our method. The improvement on the image quality may be attributed to the fact that the PDE flow gradually removes the oscillations without destroying the image structures, thus brings the image closer and closer to the reference.

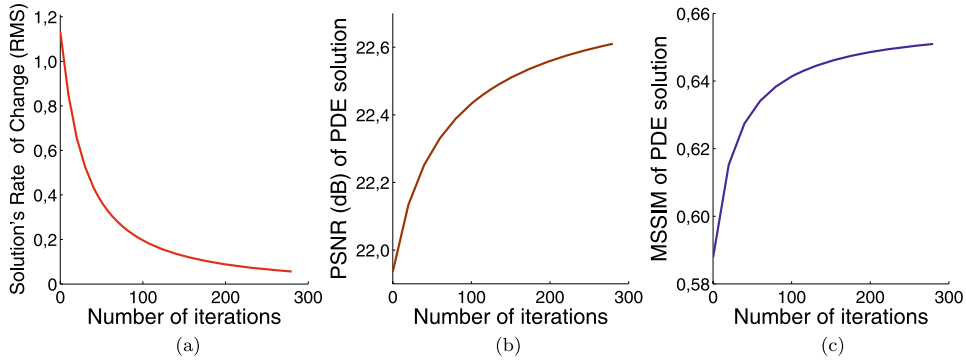
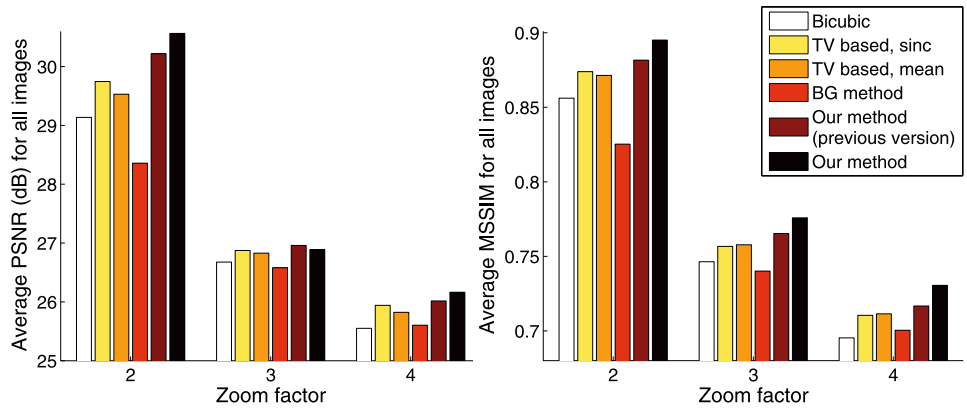
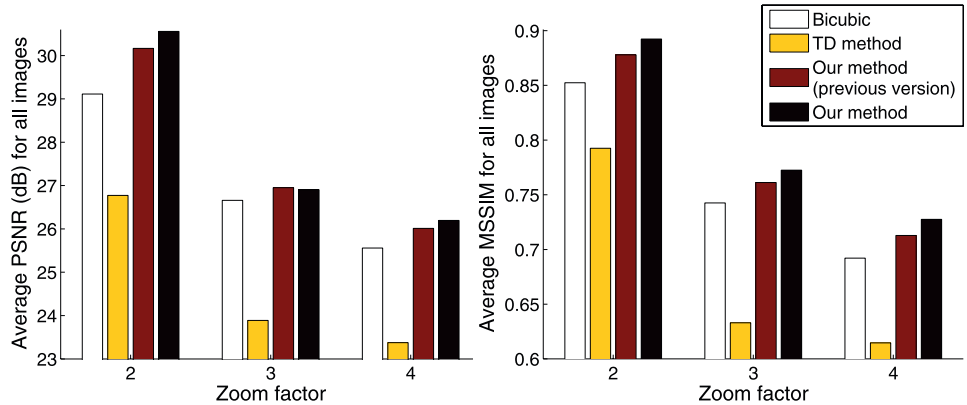


Fig. 8 Measurements during the evolution of the proposed PDE flow (14), for the example of Figs. 6 and 7. (**a**) RMS of PDE solution's rate of change $\partial \boldsymbol{u}(x, y, n\delta t)/\partial t$. (**b**) PSNR and (**c**) MSSIM error measures



(a) Experiments with graylevel images



(b) Experiments with color images

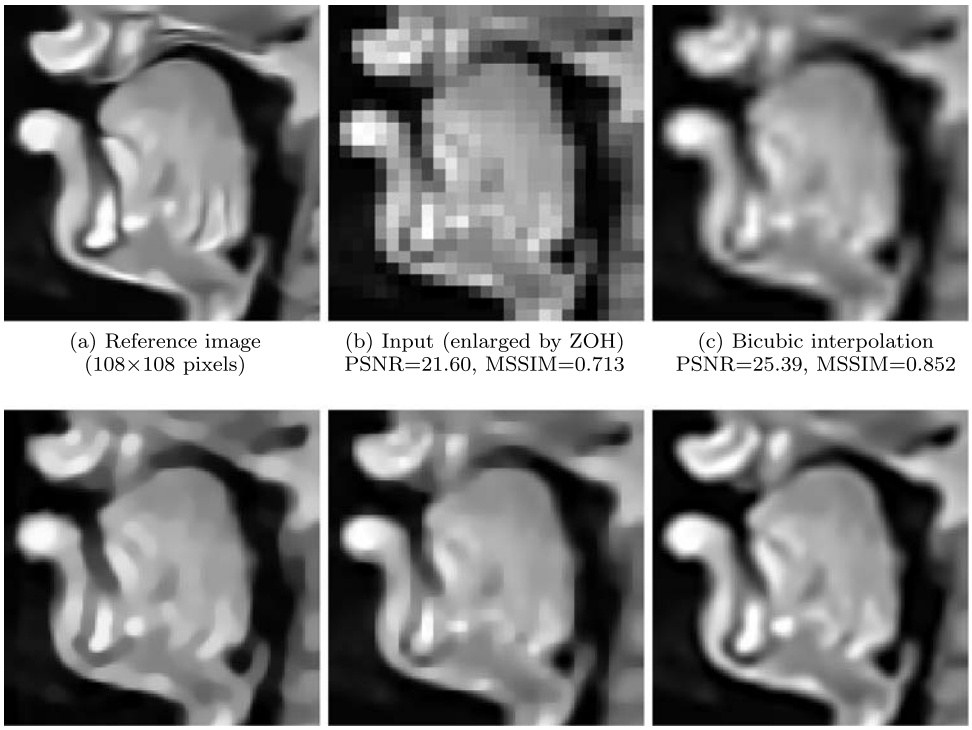
Fig. 9 Average error measures in all results using the 23 images, for different zoom factors

4.3 Overall Performance Measures

Figure 9 contains the overall performance measures of the interpolation methods, for the two series of experiments in the dataset with the 23 images.

We observe in Fig. 9(a) that the worst error measures in the experiments with graylevel images are provided by the bicubic and the BG method. Bicubic interpolation yields better error measures than the BG method when the zoom factor $d = 2$, but this relation is inverted when $d = 4$. This is due to the fact the bicubic method introduces blurring effects which become more intense as d increases. On the other hand, the BG method oversmooths the homogeneous areas of the image even when d is small, but the corresponding error measures decrease slower as d increases, because it performs an adaptive processing. Also, the BG method yields worse error measures than the other PDE-based methods because the PDE flow of this method is not constrained to satisfy exactly the reversibility condition, thus it introduces oversmoothing effects and reduces the result's accuracy. In addition, the TV-based interpolation accomplishes in all the cases better error measures than the bicubic and BG methods. These measures are fairly similar for the two choices of the smoothing kernel (sinc and mean kernel). Finally, we see

Fig. 10 Interpolation (4×4) of a biomedical vocal tract image



(a) Reference image (108×108 pixels)
(b) Input (enlarged by ZOH) PSNR=21.60, MSSIM=0.713
(c) Bicubic interpolation PSNR=25.39, MSSIM=0.852
(d) TV, sinc kernel PSNR=26.14, MSSIM=0.870
(e) BG interpolation PSNR=25.88, MSSIM=0.870
(f) Proposed method PSNR=27.69, MSSIM=0.904

that both versions of the proposed method yield better average PSNR and MSSIM measures than all the other methods. This improved performance may be attributed to the fact that our method performs a more flexible adaptive smoothing and reliably exploits the input image data to increase the accuracy of the result. Also note that the current version of our method has further improved the average PSNR and MSSIM measures in almost all the cases. This further improvement is due to the two modifications made from the current method.

Figure 9(b) shows that, in the experiments with color images, the proposed method scores again the best average PSNR and MSSIM measures. In addition, we see that the measures of the TD method are much worse even from bicubic interpolation. This is due to the excessively synthetic aspect of this method's results.

4.4 Interpolation of Biomedical Images

We have also applied our interpolation method to a special class of biomedical data that are MRI images of a speaker's vocal tract. Image data of this type are important for the analysis and modeling of the human speech production system. Often, these images have low spatial resolution, because of limitations in the image acquisition system. Thus, the application of an effective interpolation as a preprocessing step could be crucial for the success of the subsequent image analysis steps.

Figure 10 shows an example using an MRI midsagittal vocal tract image.[8] Figure 10(a) is a close-up of a denoised (using anisotropic diffusion) version of this image. Similarly to the above experiments, we used this image as reference and we reduced it to $\frac{1}{4} \times \frac{1}{4}$ of its size (see Fig. 10(b)). Finally, we applied the bicubic, TV-based, BG and our method to 4×4 interpolate the decimated image (Figs. 10(c)–(f)). The observation of the results of the Fig. 10 leads to conclusions about the methods similar to the conclusions from Fig. 4. The proposed method seems to yield again the most satisfying result, with the best reconstruction of the vocal tract shape. Note as well that the best PSNR and MSSIM index is achieved by the result of our method. This simple example reveals that the proposed model can be also used to effectively enhance the resolution of medical image data of the vocal tract.

5 Conclusions

In this paper, we have proposed a model for the interpolation of vector-valued images, based on an anisotropic diffusion PDE. Our main contribution is an efficient combination of the reversibility condition approach (Guichard and Malgouyres 1998) with a nonlinear anisotropic diffusion

[8] We have taken this image from: http://www.speech.kth.se/~olov/.

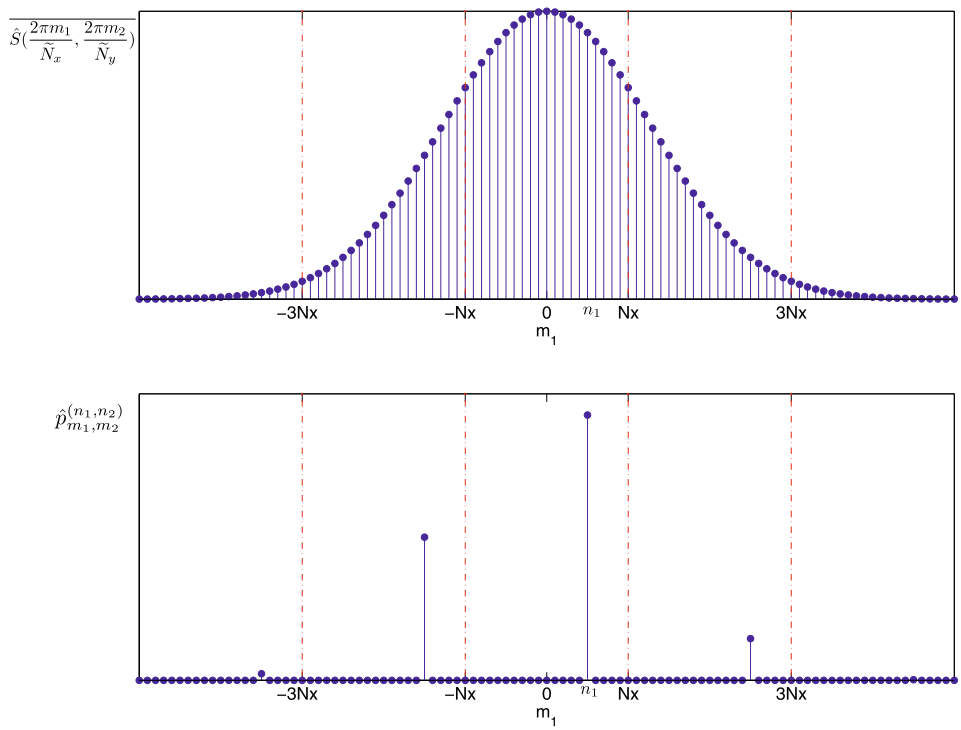


Fig. 11 Derivation of $\hat{p}^{(n_1,n_2)}_{m_1,m_2}$ (for the sake of demonstration, m_2 is constant to $n_2 + k_2\widetilde{N}_y$, for some $k_2 \in \mathbb{Z}$)

(Tschumperlé and Deriche 2005; Weickert 1998). For this purpose, we have introduced an appropriate projection operator and we have derived an expression for it using the Fourier transform. The proposed model reduces the undesirable effects of classic linear and similar PDE-based interpolation methods. Thorough experimental results have demonstrated the potential of the method as applied to graylevel and color images.

Note that important theoretical issues, such as existence, uniqueness and stability of the solution of the proposed PDE, remain still open. On the other hand, the extensive experiments that we performed indicate that the discretization of the proposed PDE evolution is stable and practically reaches an equilibrium after a finite time. Finally, it must be noted that our proposed model assumes that the input image is noise free. Thus, it could be modified to handle also images with non-negligible noise.

Appendix

We show here the validity of the expressions (15), (16) and (17), which describe the projection $P_{\mathcal{U}_{0,S}}\{\cdot\}$ on the subspace $\mathcal{U}_{0,S}$. We assume that all the input functions of the projection $P_{\mathcal{U}_{0,S}}\{\cdot\}$ in our proposed PDE (14) lie in a Hilbert space H.

The subspace $\mathcal{U}_{0,S}$ can be defined as the set of functions $v(\boldsymbol{x})$ that satisfy (1) with $z[i_1, i_2] = 0$ everywhere, or equivalently (3) with $\hat{z}_{n_1,n_2} = 0$ everywhere, namely:

$$\sum_{(k_1,k_2)\in\mathbb{Z}^2} \hat{S}\left(\frac{2\pi}{\widetilde{N}_x}(n_1 + k_1\widetilde{N}_x), \frac{2\pi}{\widetilde{N}_y}(n_2 + k_2\widetilde{N}_y)\right) \cdot \hat{v}_{n_1+k_1\widetilde{N}_x, n_2+k_2\widetilde{N}_y} = 0, \tag{19}$$

for all $(n_1, n_2) \in \mathbb{Z}^2$. As mentioned before, it is sufficient for $\hat{v}_{m_1,m_2}$ to satisfy (19) only for (n_1, n_2) that belong to any $\widetilde{N}_x \times \widetilde{N}_y$ rectangular grid. We choose the grid $B_2 = \{-N_x, \ldots, N_x - 1\} \times \{-N_y, \ldots, N_y - 1\}$ that surrounds the origin.

Thus, for any $(n_1, n_2) \in B_2$, let $p^{(n_1,n_2)}(\boldsymbol{x})$ be the $(\widetilde{N}_x, \widetilde{N}_y)$-periodic complex function, whose Fourier series coefficients are (see Fig. 11):

$$\hat{p}^{(n_1,n_2)}_{m_1,m_2} = \begin{cases} \overline{\hat{S}\left(\frac{2\pi m_1}{\widetilde{N}_x}, \frac{2\pi m_2}{\widetilde{N}_y}\right)}, & \text{if } \exists (k_1, k_2) \in \mathbb{Z}^2 \text{ s.t.:} \\ & (m_1, m_2) = (n_1 + k_1\widetilde{N}_x, n_2 + k_2\widetilde{N}_y), \\ 0, & \text{else} \end{cases} \tag{20}$$

for all $(m_1, m_2) \in \mathbb{Z}^2$. Then, condition (19) is equivalent to:

$$\forall (n_1, n_2) \in B_2: \quad \langle \hat{v}, \hat{p}^{(n_1,n_2)} \rangle_{\ell^2} = 0 \Leftrightarrow \langle v, p^{(n_1,n_2)} \rangle_{L^2} = 0, \tag{21}$$

where $\langle\cdot,\cdot\rangle_{\ell^2}$ and $\langle\cdot,\cdot\rangle_{L^2}$ denote the inner products of $\ell^2(\mathbb{Z}^2)$ and $L^2(\widetilde{\Omega})$ respectively.[9] The equivalence in (21) is due to the Parseval equality (Naylor and Sell 1982):

$$\langle \hat{v}, \hat{p}^{(n_1,n_2)}\rangle_{\ell^2} = (\widetilde{N}_x\widetilde{N}_y)^{-1}\langle v, p^{(n_1,n_2)}\rangle_{L^2}. \tag{22}$$

Since (19), which defines the subspace $\mathcal{U}_{0,S}$, is equivalent to (21), $\mathcal{U}_{0,S}$ is in fact the set of functions that are orthogonal to all $p^{(n_1,n_2)}$. Thus, it can be shown that the set $\{p^{(n_1,n_2)} : (n_1,n_2)\in B_2\}$ is a basis of $\mathcal{U}_{0,S}^{\perp}$, which is the orthogonal complement of $\mathcal{U}_{0,S}$. This basis is orthogonal, because:

$$\begin{aligned}
&\langle p^{(n_1,n_2)}, p^{(n_1',n_2')}\rangle_{L^2}\\
&\quad= (\widetilde{N}_x\widetilde{N}_y)\langle \hat{p}^{(n_1,n_2)}, \hat{p}^{(n_1',n_2')}\rangle_{\ell^2}\\
&\quad= \|p^{(n_1,n_2)}\|_{L^2}^2\delta_{n_1-n_1',n_2-n_2'},\\
&\quad\forall (n_1,n_2)\in B_2, \forall (n_1',n_2')\in B_2,
\end{aligned} \tag{23}$$

since, if $(n_1',n_2')\neq(n_1,n_2)$, $\hat{p}^{(n_1',n_2')}$ takes nonzero values for different (m_1,m_2) than $p^{(n_1,n_2)}$ (see (20) and Fig. 11). The norm $\|p^{(n_1,n_2)}\|_{L^2}$ in (23) is given by:

$$\begin{aligned}
\|p^{(n_1,n_2)}\|_{L^2} &= (\widetilde{N}_x\widetilde{N}_y)^{\frac{1}{2}}\cdot\|\hat{p}^{(n_1,n_2)}\|_{\ell^2} = (\widetilde{N}_x\widetilde{N}_y)^{-\frac{1}{2}}\\
&\cdot\left\{\sum_{(k_1,k_2)\in\mathbb{Z}^2}\left|\hat{S}\left(\frac{2\pi}{\widetilde{N}_x}(n_1+k_1\widetilde{N}_x)^{-1},\frac{2\pi}{\widetilde{N}_y}(n_2+k_2\widetilde{N}_y)\right)\right|^2\right\}^{-\frac{1}{2}}.
\end{aligned} \tag{24}$$

We form the set $\{q^{(n_1,n_2)} : (n_1,n_2)\in B_2\}$, which is an orthonormal basis of $\mathcal{U}_{0,S}^{\perp}$ by normalizing each $p^{(n_1,n_2)}$: For every $(n_1,n_2)\in B_2$, $q^{(n_1,n_2)}$ is defined as:

$$q^{(n_1,n_2)}(\boldsymbol{x}) = \frac{1}{\|p^{(n_1,n_2)}\|_{L^2}}\cdot p^{(n_1,n_2)}(\boldsymbol{x}),\quad \forall \boldsymbol{x}\in\mathbb{R}^2. \tag{25}$$

Now, we can straightforwardly compute the projection in $\mathcal{U}_{0,S}^{\perp}$ (Naylor and Sell 1982):

$$\begin{aligned}
w(\boldsymbol{x}) &\triangleq P_{\mathcal{U}_{0,S}^{\perp}}\{v\}\\
&= \sum_{(n_1,n_2)\in B_2}\langle v, q^{(n_1,n_2)}\rangle_{L^2}\cdot q^{(n_1,n_2)}(\boldsymbol{x}),\quad \forall \boldsymbol{x}\in\mathbb{R}^2,
\end{aligned}$$

which is also $(\widetilde{N}_x,\widetilde{N}_y)$-periodic, like all $q^{(n_1,n_2)}(\boldsymbol{x})$ functions. Thus, the Fourier series coefficients of w are given by:

$$\hat{w}_{m_1,m_2} = (\widetilde{N}_x\widetilde{N}_y)\sum_{(n_1,n_2)\in B_2}\langle \hat{v}, \hat{q}^{(n_1,n_2)}\rangle_{\ell^2}\cdot\hat{q}^{(n_1,n_2)}_{m_1,m_2}, \tag{26}$$

for all $(m_1,m_2)\in\mathbb{Z}^2$.

[9] $\langle f,g\rangle_{\ell^2} = \sum_{(m_1,m_2)\in\mathbb{Z}^2} f_{m_1,m_2}\overline{g_{m_1,m_2}}$, $\langle f,g\rangle_{L^2} = \iint_{\widetilde{\Omega}} f(\boldsymbol{x})\overline{g(\boldsymbol{x})}\mathrm{d}\boldsymbol{x}$.

As we said, the Fourier series of each element of $\{p^{(n_1,n_2)} : (n_1,n_2)\in B_2\}$ is nonzero at different points (m_1,m_2) from all the other elements. The same property holds also for the basis $\{q^{(n_1,n_2)} : (n_1,n_2)\in B_2\}$, due to (25). This means that, for each $(m_1,m_2)\in\mathbb{Z}^2$, only one term is nonzero in the above sum of (26): the one that corresponds to the pair $(n_1,n_2)\in B_2$ that can be written as $(n_1,n_2) = (m_1-k_1\widetilde{N}_x, m_2-k_2\widetilde{N}_y)$, for some $(k_1,k_2)\in\mathbb{Z}^2$. Using this statement and (20), (24), (25), we can write (26) as:

$$\begin{aligned}
\hat{w}_{m_1,m_2} &= \left\{\sum_{(k_1,k_2)\in\mathbb{Z}^2}\hat{\phi}\left(\frac{2\pi m_1}{\widetilde{N}_x}+k_1 2\pi, \frac{2\pi m_2}{\widetilde{N}_y}+k_2 2\pi\right)\right.\\
&\qquad\left.\cdot\,\hat{v}_{m_1+k_1\widetilde{N}_x, m_2+k_2\widetilde{N}_y}\right\}\cdot\hat{\phi}\left(\frac{2\pi m_1}{\widetilde{N}_x},\frac{2\pi m_2}{\widetilde{N}_y}\right),\\
&\forall (m_1,m_2)\in\mathbb{Z}^2,
\end{aligned}$$

where it has been introduced the function $\phi(\boldsymbol{x})$ whose Fourier transform is:

$$\begin{aligned}
\hat{\phi}(\omega_1,\omega_2) &= \left\{\sum_{(k_1,k_2)\in\mathbb{Z}^2}|\hat{S}(\omega_1+k_1 2\pi, \omega_2+k_2 2\pi)|^2\right\}^{-\frac{1}{2}}\\
&\quad\cdot\,\overline{\hat{S}(\omega_1,\omega_2)}.
\end{aligned}$$

Finally, the projection $P_{\mathcal{U}_{0,S}}\{v\}$ can be computed using the expression of $P_{\mathcal{U}_{0,S}^{\perp}}\{v\}$: From the projection theorem (Naylor and Sell 1982), it follows that:

$$P_{\mathcal{U}_{0,S}}\{v\} = v - P_{\mathcal{U}_{0,S}^{\perp}}\{v\} = v - w.$$

References

Aly, H. A., & Dubois, E. (2005). Image up-sampling using total-variation regularization with a new observation model. *IEEE Transactions on Image Processing*, *14*(10), 1647–1659.

Belahmidi, A., & Guichard, F. (2004). A partial differential equation approach to image zoom. In *Proceedings of the international conference on image processing* (Vol. 1, pp. 649–652).

Bertalmio, M., Sapiro, G., Caselles, V., & Ballester, C. (2000). Image inpainting. In *Proceedings of the SIGGRAPH 2000* (pp. 417–424).

Caselles, V., Morel, J. M., & Sbert, C. (1998). An axiomatic approach to image interpolation. *IEEE Transactions on Image Processing*, *7*(3), 376–386.

Chan, T. F., & Shen, J. (2002). Mathematical models for local nontexture inpaintings. *SIAM Journal on Applied Mathematics*, *62*(3), 1019–1043.

Dudgeon, D. E., & Mersereau, R. M. (1984). *Multidimensional digital signal processing*. New York: Prentice-Hall.

Guichard, F., & Malgouyres, F. (1998). Total variation based interpolation. In *Proceedings of the EUSIPCO* (Vol. 3, pp. 1741–1744).

Malgouyres, F., & Guichard, F. (2001). Edge direction preserving image zooming: a mathematical and numerical analysis. *SIAM Journal on Numerical Analysis*, *39*(1), 1–37.

Meijering, E. (2002). A chronology of interpolation: from ancient astronomy to modern signal and image processing. *Proceedings of the IEEE*, *90*(3), 319–342.

Naylor, A. W., & Sell, G. R. (1982). *Linear operator theory in engineering and science*. Berlin: Springer.

Oppenheim, A. V., Willsky, A. S., & Young, I. T. (1984). *Signals and systems*. Berlin: Prentice Hall.

Perona, P., & Malik, J. (1990). Scale space and edge detection using anisotropic diffusion. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *12*(7), 629–639.

Roussos, A., & Maragos, P. (2007) Vector-valued image interpolation by an anisotropic diffusion-projection PDE. In *Lecture notes in computer science: Vol. 4485. Scale space and variational methods in computer vision, first international conference, SSVM 2007 proceedings* (pp. 104–115). Berlin: Springer.

Rudin, L., Osher, S., & Fatemi, E. (1992). Nonlinear total variation based noise removal algorithms. *Physica D*, *60*, 259–268.

Tschumperlé, D. (2002). *PDE's based regularization of multivalued images and applications*. PhD thesis, Univ. of Nice-Sophia Antipolis.

Tschumperlé, D., & Deriche, R. (2005). Vector-valued image regularization with PDE's: a common framework for different applications. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *27*(4), 506–517.

Wang, Z., Bovik, A., Sheikh, H., & Simoncelli, E. (2004). Image quality assessment: from error visibility to structural similarity. *IEEE Transactions on Image Processing*, *13*(4), 600–612.

Weickert, J. (1998). *Anisotropic diffusion in image processing*. Stuttgart: Teubner.

Weickert, J., & Welk, M. (2006) Tensor field interpolation with PDEs. In *Visualization and processing of tensor fields* (pp. 315–325). Berlin: Springer.