Shahidi and Moloney *EURASIP Journal on Advances in Signal Processing* (2016) 2016:116
DOI 10.1186/s13634-016-0415-2

EURASIP Journal on Advances in Signal Processing

RESEARCH Open Access

Introducing oriented Laplacian diffusion into a variational decomposition model

Reza Shahidi* and Cecilia Moloney

Abstract

The decomposition model proposed by Osher, Solé and Vese in 2003 (the OSV model) is known for its good denoising performance. This performance has been found to be due to its higher weighting of lower image frequencies in the H^{-1}-norm modeling the noise component in the model. However, the OSV model tends to also move high-frequency texture into this noise component. Diffusion with an oriented Laplacian for oriented texture is introduced in this paper, in lieu of the usual Laplacian operator used to solve the OSV model, thereby significantly reducing the presence of such texture in the noise component. Results obtained from the proposed oriented Laplacian model for test images with oriented texture are given, and compared to those from the OSV model as well as the Mean Curvature model (MCM). In general, the proposed oriented Laplacian model yields higher signal-to-noise ratios and visually superior denoising results than either the OSV or the MCM models. We also compare the proposed method to a non-local means model and find that although the proposed method generally yields slightly lower signal-to-noise ratios, it generally gives results of better perceptual visual quality.

Keywords: Image decomposition, Variational methods, Image processing, Osher-Solé-Vese model, Oriented Laplacian, Image denoising

1 Introduction

1.1 Image decomposition

According to the definition of Meyer [1], image decomposition is the process of splitting an image into components of different priorities, usually so that the original image is roughly equal to the sum of these components. Often the original image is denoted by *f*, which is considered to approximately equal the sum of two components, u and v, i.e. $f \approx u + v$. In Meyer's 2001 monograph [1], decomposition was deemed to be important for image coding and transmission. In addition to the problem of decomposition being interesting and important in its own right [2], decomposition has also proven to be useful for texture discrimination [3], image denoising [4], image inpainting [5] and image registration [6]. In this paper, the focus is on the application of decomposition to the problem of the denoising of images with oriented texture.

1.2 Meyer's decomposition framework

Meyer's decomposition framework models the various components of the decomposition as having small norms in different Banach spaces, i.e. complete normed vector spaces. In practical implementations of this framework [3], the energy method is used, in which an energy is defined over the image being decomposed and its components, and is subsequently minimized using the Calculus of Variations.

The first method that could be considered to fall under the image decomposition framework is the total variation model of Rudin, Osher and Fatemi [7]. The energy for the total variation model is defined as

$$\mathbf{E_{TV}}(u) = \int_{\Omega} |\nabla u| dxdy + \lambda \int_{\Omega} (f - u)^2 dxdy, \qquad (1)$$

where the original image is f and $f \approx u + v$. The first term $\left(\int_{\Omega} |\nabla u| dxdy\right)$ is included to produce a bounded variation (piecewise-smooth) image u upon energy minimization, while the second term is a fidelity term, which ensures that the final u component is close to the initial image f. Although the minimization of $\mathbf{E_{TV}}(u)$ preserves sharp edges, it destroys fine structure such as texture. However,

*Correspondence: rshahidi@gmail.com
Department of Electrical and Computer Engineering, Memorial University of Newfoundland, Prince Philip Avenue, St. John's, Canada

Eq. 1 has been used successfully for the denoising and deblurring of images of bounded variation.

In [1], Meyer proposed modifying the second term in the above energy of Eq. 1 by changing the L^2-norm of the residual v to the $*$-norm of this residual, where $*$ is a norm on a suitably defined Banach space G. The $*$-norm on G is defined by

$$\|v\|_* = \inf_{g_1,g_2} \left\| \sqrt{g_1^2(x,y) + g_2^2(x,y)} \right\|_{L^\infty}, \tag{2}$$

over all g_1 and g_2 such that $v = \operatorname{div}(\vec{g})$ where $\vec{g} = (g_1, g_2)$.

1.3 The Osher-Solé-Vese decomposition model

A second attempt at minimizing the $*$-norm of Eq. 2 was made by Osher, Solé and Vese in [4]. They used the Hodge decomposition of $\vec{g}$, which splits $\vec{g}$ into the sum of the divergence of a single valued function and a divergence-free vector field. The energy functional for their model is

$$\mathbf{E_{OSV}}(u) = \int_\Omega |\nabla u| dxdy + \lambda \int_\Omega \left|\nabla\left(\Delta^{-1}\right)(f-u)\right|^2 dxdy \tag{3}$$

The PDE that they obtained as a result of minimization of the above functional is

$$u_t = -\frac{1}{2\lambda}\Delta\left[\operatorname{div}\left(\frac{\nabla u}{|\nabla u|}\right)\right] - (u-f) \tag{4}$$

with adiabatic boundary conditions. This equation was found by approximating $\vec{g}$ as having no divergence-free part, and approximating the L^∞ part of the $*$-norm in Eq. 2 with the square of the L^2-norm of the same function. The resulting energy functional includes an inverse Laplacian of $f - u$; this was eliminated in [4] by showing that under some rather relaxed conditions, found in the second Remark of that paper, the gradient descent solution of an Euler-Lagrange equation $u_t = -\mathbf{E}'_{\mathbf{OSV}}(u)$, where $\mathbf{E}'_{\mathbf{OSV}}(u)$ is the first variation of $\mathbf{E_{OSV}}$ with respect to u, converges to the same solution as the equation $u_t = \Delta\mathbf{E}'_{\mathbf{OSV}}(u)$. Thus no inverse Laplacian appears in Eq. 4. Additionally, the auxiliary functions g_1 and g_2 of Eq. 2 are no longer involved in the PDE of Eq. 4, as they disappear in the derivation.

2 Incorporating oriented diffusion into the OSV model

2.1 Motivation and initial derivation

If we denote the curvature of the level lines of the cartoon component u by $K(u) = \operatorname{div}\left(\frac{\nabla u}{|\nabla u|}\right)$, Eq. 4 becomes

$$u_t = -\frac{1}{2\lambda}\Delta K(u) - (u-f). \tag{5}$$

Observe that Eq. 5 consists of two terms, one with the negative Laplacian of the curvature of the level lines of u; the other a fidelity term, which makes sure that the evolved image does not stray too far away from the original image f. Equation 5 has been shown to perform well for denoising [4]. Since the original image f is noisy, and the second term keeps the evolved image close to this noisy image, the denoising itself must be due to the first term; i.e. the negative Laplacian term.

Attention is now restricted to images with oriented texture. With this restricted focus, instead of including the Laplacian of the curvature, $\Delta K(u)$, as in Eq. 5, the oriented Laplacian [8] of the curvature can be taken instead. The new equation becomes

$$u_t = -\frac{1}{2\lambda}\left(c_\zeta K_{\zeta\zeta} + c_\eta K_{\eta\eta}\right) - (u-f) \tag{6}$$

Here ζ is the isophote direction of the curvature in the regions of the image where there is a dominant orientation, and η is the gradient direction of the curvature in such regions. An illustration of these directions is shown in Fig. 1.

There are also two conductivity coefficients c_ζ and c_η which determine how much diffusion there is in the isophote and gradient directions respectively.

The values of the second directional derivatives of the curvature in its isophote and gradient directions can be computed without actually calculating the isophote and gradient angles. This is done by using the general formula for the second-order directional derivative of the function K in the direction $\mathbf{w} = (w_x, w_y)$, i.e.

$$K_{\mathbf{ww}} = w_x^2 K_{xx} + 2w_x w_y K_{xy} + w_y^2 K_{yy},$$

and then substituting the unit gradient and isophote vector directions for $\mathbf{w}$. The unit gradient vector is simply

$$\mathbf{w_{grad}} = \left(\frac{K_x}{\sqrt{K_x^2 + K_y^2}}, \frac{K_y}{\sqrt{K_x^2 + K_y^2}}\right),$$

and the unit isophote vector is

$$\mathbf{w_{iso}} = \left(\frac{-K_y}{\sqrt{K_x^2 + K_y^2}}, \frac{K_x}{\sqrt{K_x^2 + K_y^2}}\right).$$

The formulae for the second directional derivative of the curvature in the isophote direction (ζ) and its gradient direction (η) are thus

$$K_{\zeta\zeta} = \frac{-2K_x K_{xy} K_y + K_{xx} K_y^2 + K_x^2 K_{yy}}{K_x^2 + K_y^2} \tag{7}$$

$$K_{\eta\eta} = \frac{K_x^2 K_{xx} + 2K_x K_{xy} K_y + K_y^2 K_{yy}}{K_x^2 + K_y^2}. \tag{8}$$

The expressions for the two directional derivatives in Eqs. 7 and 8 can be substituted into Eq. 6. Recall that

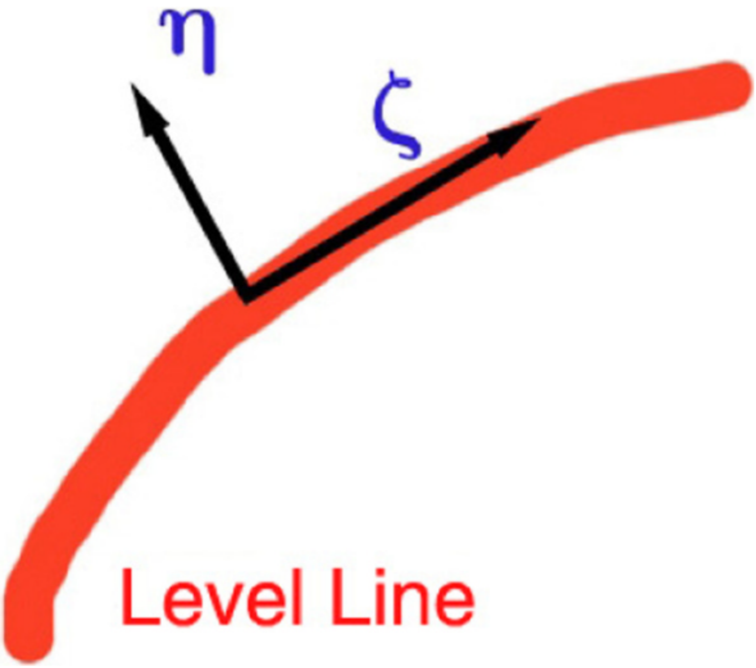


Fig. 1 Illustration of isophote and gradient directions

the function K represents the curvature of the cartoon component u.

In our experiments for this paper, better results were obtained in practice by using the normalized isophote orientation estimates of the original image f instead of those of the curvature image K. An initial explanation for the improvement in results is the sensitivity to noise of the formulae in Eqs. 7 and 8.

We now explain how estimates of the normalized isophote and gradient directions are extracted from the original noisy image using the linear structure tensor (Section 2.2.1), as well as how this image is separated into oriented and non-oriented regions (Section 2.2.2). This separation by region is important to decide whether anisotropic or isotropic diffusion should be performed at each image pixel, and the directional estimates are important to determine which directional diffusion parameters are used for anisotropic diffusion at each pixel in the oriented region. After this explanation in Section 2.2, we return to the derivation of the proposed model in Section 2.3 using these extracted gradient and isophote orientation estimates.

2.2 Orientation and coherence calculation

The proposed algorithm depends on the noise-resistant computation of orientation at each pixel in the image, in addition to the detection of the image regions that are coherent, or roughly consisting of one orientation. Two methods were considered in our experiments by which such calculations could be performed and be robust to noise. Both use the structure tensor, or the outer product of the image gradient with itself. The first is called the linear structure tensor, which corresponds to a linear diffusion or Gaussian blurring of the structure tensor elements, and the second is called the nonlinear structure tensor, an extension of the linear structure tensor which corresponds to nonlinear diffusion of the structure tensor entries. Only the linear structure tensor is described here, since it was found to be more efficient computationally than the nonlinear structure tensor, and gave results of similar quality. The linear structure tensor is now described, along with methods of determining the orientation coherence [9]. If not otherwise specified, it is assumed the orientation refers to the gradient orientation, or the orientation in which the image changes the most in intensity.

2.2.1 Linear structure tensor

The structure tensor is defined as the outer product of the image gradient vector with itself. Supposing that the image is f, then the structure tensor J at each pixel is defined as

$$J = \begin{pmatrix} f_x^2 & f_x f_y \\ f_x f_y & f_y^2 \end{pmatrix}.$$

It was found that better denoising results were obtained for the proposed method by using the following derivative filters (see e.g. [10]) to determine f_x and f_y

$$D_x = \frac{1}{32}\begin{pmatrix} -3 & 0 & 3 \\ -10 & 0 & 10 \\ -3 & 0 & 3 \end{pmatrix} \tag{9}$$

and

$$D_y = \frac{1}{32}\begin{pmatrix} 3 & 10 & 3 \\ 0 & 0 & 0 \\ -3 & -10 & -3 \end{pmatrix}. \tag{10}$$

To further increase robustness to noise, the resulting structure tensor J is blurred element-wise with a Gaussian filter G_σ of standard deviation σ, to obtain $J_\sigma = G_\sigma * J$. Then the gradient direction at each pixel, $\theta_{i,j}$, is calculated to be the angle between the eigenvector $\overrightarrow{w}_1$ of J_σ corresponding to its larger eigenvalue λ_1 and the horizontal axis, or more simply, if we consider $\overrightarrow{w}_1$ of J_σ to be a vector in the complex plane

$$\theta_{i,j} = \mathbf{arg}\,\overrightarrow{w}_1.$$

The isophote vector $\overrightarrow{f_{iso}} = (f_{iso,x}, f_{iso,y})$ is computed as the unit vector perpendicular to $\theta_{i,j}$, thus at an angle $\frac{\pi}{2}$ radians greater than $\theta_{i,j}$ from the horizontal axis; i.e. if we consider the isophote vector as a vector in the complex plane

$$\mathbf{arg}\overrightarrow{f_{iso}} = \mathbf{arg}\,\overrightarrow{w}_1 + \frac{\pi}{2}.$$

2.2.2 Orientation coherence and oriented region determination

For the proposed algorithm, it is necessary to separate the image into oriented and non-oriented regions, so that different variational models can be applied to each of the two region types. A region is defined to be non-oriented when its orientation coherence is less than a pre-determined threshold; this coherence function is a measure of how uniform gradient directions are around a pixel. In [11], the coherence of f is measured directly using a small window W around each pixel by the formula

$$\text{coher}(\theta_{i,j}) = |\nabla f|_{i,j} \frac{\sum_{(u,v)\in W} ||\nabla f|_{u,v} \cos(\theta_{i,j} - \theta_{u,v})|}{\sum_{(u,v)\in W} |\nabla f|_{u,v}},$$

where $\theta_{i,j}$ is the orientation calculated from the linear structure tensor J at pixel (i,j) (which can be found because the tensor determines the gradient vector at each pixel); generally W is chosen to be 7×7 pixels. In this paper, this formula is used to determine the orientation coherence at each pixel, thus allowing oriented and non-oriented regions to be separated.

The threshold used for the coherence of the orientations of f is dependent on the image. Call this parameter $\text{coher}_{\text{thresh}}$; we set $\text{coher}_{\text{thresh}} = 15$ for all experiments in this paper.

Now define the oriented region to be Ω_O, the union of pixels in the image where there is a dominant orientation in f. Similarly, we define the non-oriented region to be Ω_{NO}, the union of pixels in the image where there is no such dominant orientation.

2.3 The oriented Laplacian Osher-Solé-Vese model

After Eqs. 7 and 8 in Section 2.1, we stated that using the normalized isophote orientation estimates of the original image yielded better denoising results in our experiments. After this, in Section 2.2 we explained how these isophote orientation estimates were computed using the linear structure tensor. Now using the isophote orientation vectors $(f_{iso,x}, f_{iso,y})$ at each pixel, we obtain the following noise-resistant approximation of the second directional derivative of the curvature in the isophote direction

$$K_{\zeta\zeta} \approx f_{iso,x}^2 K_{xx} + 2 f_{iso,x} f_{iso,y} K_{xy} + f_{iso,y}^2 K_{yy} \tag{11}$$

which is subsequently substituted in the oriented Laplacian expression in Eq. 6. Note that the denominators in Eqs. 7 and 8 do not appear since we are dealing with the normalized isophote estimates, as opposed to Eqs. 7 and 8, where the partial derivatives of K are unnormalized.

For the coefficient c_η in Eq. 6, the usual Perona-Malik diffusivity function

$$c_\eta = g(|\nabla K(u)|) = \frac{1}{1 + \frac{|\nabla K(u)|^2}{K_d^2}}, \tag{12}$$

could be chosen (with K_d set to 10 after experimentation). However, we found that just setting c_η to zero gave similar results and was simpler; consequently the expression in Eq. 12 was not used. With the coefficient c_η set to zero, there is no diffusion or denoising in the image gradient direction. To promote smoothing in the image isophote direction, the coefficient c_ζ is set equal to 1. Therefore Eq. 6 becomes

$$u_t = -\frac{1}{2\lambda} K_{\zeta\zeta} - (u - f), \tag{13}$$

and substituting the approximation of Eq. 11 for $K_{\zeta\zeta}$ yields the final equation for the iterative solution of the proposed decomposition model

$$\begin{aligned} u_t &= -\frac{1}{2\lambda}\left(f_{iso,x}^2 K_{xx} + 2 f_{iso,x} f_{iso,y} K_{xy} + f_{iso,y}^2 K_{yy}\right) - (u - f) \\ &= -\frac{1}{2\lambda}\left(\zeta_x^2 K_{xx} + 2\zeta_x \zeta_y K_{xy} + \zeta_y^2 K_{yy}\right) - (u - f), \end{aligned} \tag{14}$$

where $(\zeta_x, \zeta_y) = (f_{iso,x}, f_{iso,y})$ is a unit vector defined for each pixel in the image pointing in the isophote direction.

Within the oriented region Ω_O, Eq. 14 is used, whereas within the non-oriented region Ω_{NO}, the ordinary evolution equation for Osher-Solé-Vese decomposition, as in Eq. 4 is utilized. Each equation and region involves a weighting parameter λ. This parameter becomes λ_O in Ω_O, and λ_{NO} in Ω_{NO}. In practice, each model is evolved over the entire image, but after convergence restricted to either Ω_O or Ω_{NO}, as initially defined by the coherence threshold. This avoids any noticeable artifacts at the boundaries of the regions by using the image itself to effectively pad each region for its model of diffusion along internal interfaces.

We call the decomposition model defined by the PDE of Eq. 14, the Oriented Laplacian Osher-Solé-Vese (OLOSV) decomposition model.

3 Denoising by mean curvature motion

A similar denoising model to the OLOSV model developed in Eq. 14 above is the mean curvature motion (MCM) denoising model. Both models are based on the idea of denoising edges, only along and not across them. However, it will be seen that MCM denoising does not lend itself readily to a dual norm formulation, and thus cannot avail of the better quality results obtained from such a formulation.

The equation defining Mean Curvature Motion denoising is

$$u_t = u_{\zeta\zeta} = |\nabla u| \mathrm{div}\left(\frac{\nabla u}{|\nabla u|}\right), \tag{15}$$

with $u_{\zeta\zeta}$ the second-order directional derivative of u in the isophote direction.

To make a fair comparison with the OSV and OLOSV models in Eqs. 4 and 14, respectively, a fidelity term is added to the basic MCM of Eq. 15, along with a parameter λ that controls the relative weighting of the diffusion and fidelity terms. Thus the version of MCM implemented in this paper for comparison with OSV and OLOSV is

$$u_t = \frac{1}{2\lambda} u_{\zeta\zeta} - (u - f). \tag{16}$$

The implementation of mean curvature motion that we used in our experiments was based on the following

$$u_{\zeta\zeta} = \zeta_x^2 u_{xx} + 2\zeta_x \zeta_y u_{xy} + \zeta_y^2 u_{yy}, \tag{17}$$

derived by setting $(\zeta_x, \zeta_y) = (f_{iso,x}, f_{iso,y})$ with this isophote direction measured in a noise-resistant manner, as in Section 2.2.1. We found from our experiments that using these image isophotes gave better denoising results.

3.1 Explicit timestepping

Then Eq. 14 (for OLOSV) and Eq. 16 (for MCM) are each separately solved via explicit timestepping (with the time step chosen by experimentation to be $\Delta t = 0.002$).

As with other decomposition/denoising schemes, with known texture/noise variance, the value of λ is dynamically updated with iteration number using a method based on gradient projection. In fact, in oriented regions (Ω_O) one dynamically updated coefficient λ_O is used, and in non-oriented regions, a separate dynamically updated coefficient λ_{NO} is calculated. The final formula that is obtained for λ_O is

$$\lambda_O = \frac{1}{2\sigma^2} \int_{\Omega_O} (f - u)\left(c_\xi K_{\xi\xi} + c_\eta K_{\eta\eta}\right) dxdy \tag{18}$$

In non-oriented regions, the expression for λ_{NO} at each iteration is

$$\lambda_{NO} = \frac{1}{2\sigma^2} \int_{\Omega_{NO}} \left(f - u\right) \Delta K(u) dxdy. \tag{19}$$

The above two equations are derived from Eqs. 14 and 4, respectively, by assuming we have reached equilibrium, so that $u_t = 0$, and then multiplying each side by $u - f$ and integrating over the image domain Ω.

For MCM denoising, a formula similar to Eq. 18 is used to determine λ_O at each iteration, also based on gradient projection

$$\lambda_O = \frac{1}{2\sigma^2} \int_{\Omega_O} (f - u) u_{\zeta\zeta} dxdy. \tag{20}$$

4 Numerical implementation

Two sets of experiments were conducted using the OSV and OLOSV methods. The first involved an explicit timestepping solution of the two models, along with the mean curvature denoising model. The second set of experiments was based on the dual formulation of total variation minimization of Chambolle [12], which was further generalized in [13] to deal with the OSV model, and to an even more general TV-Hilbert model in [14]

$$p_{i,j}^{n+1} = \frac{p_{i,j}^n + \Delta t \left(\nabla \left(K^{-1} \mathrm{div}\left(p^n\right) - \lambda f\right)\right)_{i,j}}{1 + \Delta t | \left(\nabla \left(K^{-1} \mathrm{div}\left(p^n\right) - \lambda f\right)\right)_{i,j}|}, \tag{21}$$

with initial condition

$$p^0 = 0.$$

Then, similar to [14], $f + \frac{1}{\lambda_{NO}} \Delta \mathrm{div} p^n$ converges to the u component of the decomposition in Ω_{NO}, the non-oriented section of the image, which implies that $-\frac{1}{\lambda_{NO}} \Delta \mathrm{div} p^n$ converges to the component $v = f - u$ in Ω_{NO}. Similarly, in the oriented section Ω_O of the image, $f + \frac{1}{\lambda_O} \Delta_{\zeta\zeta} \mathrm{div} p^n$ converges to the u component of the decomposition and thus $-\frac{1}{\lambda_O} \Delta_{\zeta\zeta} \mathrm{div} p^n$ converges to the component $v = f - u$ in Ω_O.

When used in conjunction with the OLOSV model, such an implicit numerical method gives rise to a new denoising method, which we call the Implicit Oriented Laplacian Osher-Solé-Vese, or IOLOSV method.

5 TV-Hilbert space formulation

The proposed IOLOSV model presented in this paper differs from the general TV-Hilbert model in [14] and the TV-Gabor model in [15], in that, in the proposed model, the operator K is spatially varying, whereas in the two other models, the operator K is constant throughout the entire image, and not data-dependent.

For two discrete zero-mean image functions f and g, Aujol and Gilboa [14] defined the inner product $< \cdot, \cdot >$ on a Hilbert space $\mathcal{H}$ as

$$< f, g >_{\mathcal{H}} = < f, Kg >_{L^2}, \tag{22}$$

where $< f, Kg >_{L^2} = \sum_{i,j} f_{i,j} (Kg)_{i,j}$. Then the Sobolev space H^{-1} used in the model of Osher, Solé and Vese fits into the TV-Hilbert space framework, with the operator $K = -\Delta^{-1}$ as the negative of the inverse Laplacian operator.

The proposed IOLOSV method can also be made to fit into the TV-Hilbert space decomposition framework, with the operator K spatially varying and equal to $K = -\Delta_{\zeta\zeta}^{-1}$, the negative of the spatially varying second-order directional derivative (in the direction of the image isophote). We denote the Hilbert space thus obtained as H_{ζ}^{-1} in order to emphasize its link with both the Sobolev space H^{-1} and its dependence on the spatially varying isophote direction $\zeta(x, y)$.

A flow chart is included below illustrating the overall procedure for denoising with the OLOSV and OSV models in the oriented and non-oriented parts of the image, respectively (Fig. 2).

5.1 Calculation of iteration-dependent fidelity parameter

It is shown in [14] that for an $M \times N$ pixel image n consisting only of white Gaussian noise of standard deviation σ, it can be assumed that

$$||n||_{\mathcal{H}}^2 = C_{\mathcal{H}} ||n||_{L^2}^2 = C_{\mathcal{H}} MN\sigma^2, \tag{23}$$

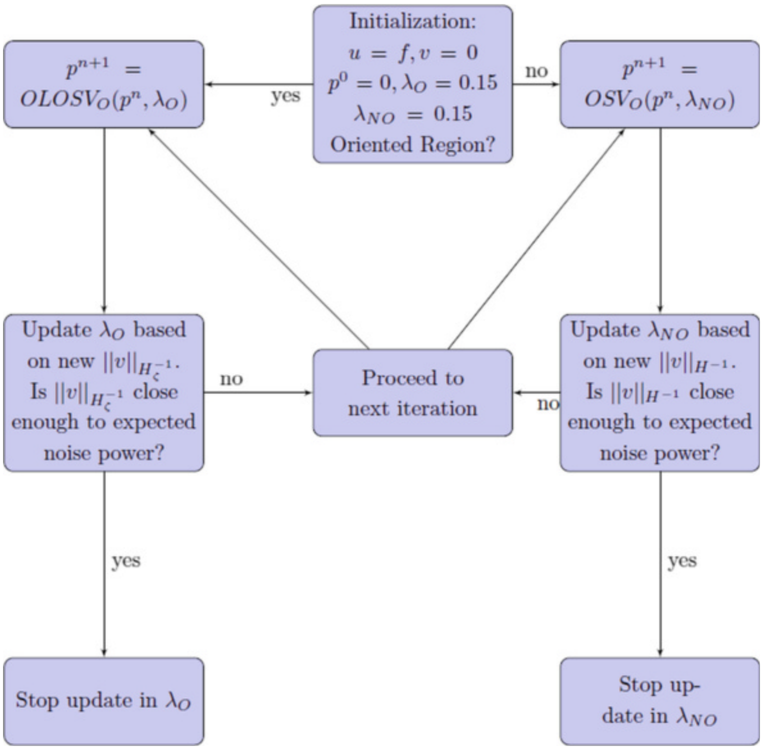
where $C_{\mathcal{H}}$ is a constant that only depends on the Hilbert space $\mathcal{H}$ over which we are taking the norm of the noise image n. Its calculation will be described in the next subsection.

The same procedure found in [14] is used in this paper to dynamically vary the fidelity parameters λ_O and λ_{NO}. This procedure is based on attempting to solve the problem

$$\min_u \left\{ \int_{\Omega} |\nabla u| dxdy \right\} \text{such that} ||u - f||_{\mathcal{H}}^2 = C_{\mathcal{H}} MN\sigma^2. \tag{24}$$

In practice, it was found in [14] that the problem posed by Eq. 24 leads to the oversmoothing of u; instead a positive factor $\alpha_{\mathcal{H}} < 1$ was added in [14] to the right-hand side of that equation to form the new problem

$$\min_u \left\{ \int_{\Omega} |\nabla u| dxdy \right\} \text{such that} ||u - f||_{\mathcal{H}}^2 = \alpha_{\mathcal{H}} C_{\mathcal{H}} MN\sigma^2. \tag{25}$$



Initialization:
$u = f, v = 0$
$p^0 = 0, \lambda_O = 0.15$
$\lambda_{NO} = 0.15$
Oriented Region?
yes
no
$p^{n+1} = OLOSV_O(p^n, \lambda_O)$
$p^{n+1} = OSV_O(p^n, \lambda_{NO})$
Update λ_O based on new $||v||_{H_\zeta^{-1}}$. Is $||v||_{H_\zeta^{-1}}$ close enough to expected noise power?
no
Proceed to next iteration
no
Update λ_{NO} based on new $||v||_{H^{-1}}$. Is $||v||_{H^{-1}}$ close enough to expected noise power?
yes
yes
Stop update in λ_O
Stop update in λ_{NO}

Fig. 2 A flowchart illustrating the overall procedure for denoising with OLOSV and OSV models in the oriented and non-oriented parts of the image, respectively

In both equations, the noise component ν is defined to be equal to $f - u$, where f is the original noise-contaminated image and u is the obtained cartoon component.

Based on our experimentation with the IOLOSV coefficients, both λ_O and λ_{NO} are initialized to 0.15, and then at each iteration, their values are updated using the following method. At every iteration, $||\nu_{O,n}||_{H_\zeta^{-1}}$ and $||\nu_{NO,n}||_{H^{-1}}$ are estimated by calculating $||\nu||_{L^2}$, easily done by taking the square-root of the sum of squares of the ν component of the decomposition, and then multiplying by the factor $\sqrt{C_{H^{-1}}}$ for the non-oriented section of the image, and $\sqrt{C_{H_\zeta^{-1}}}$ for the oriented part. As in [14], given λ_n and $||\nu_n||_{\mathcal{H}}$, λ_{n+1} is obtained from these quantities by the formula

$$\lambda_{n+1} = \frac{||\nu_n||_{\mathcal{H}}}{\sqrt{\alpha_{\mathcal{H}} C_{\mathcal{H}}} MN\sigma}\lambda_n. \tag{26}$$

In the above equation, $||\nu_n||_{\mathcal{H}}$ is the norm of ν_n in the Hilbert space $\mathcal{H}$; the calculation of this norm is detailed below in Section 5.3. Then ν is calculated using this new value of λ for the next iteration. The procedure is implemented separately in Ω_O, where $\mathcal{H} = H_\zeta^{-1}$, and in Ω_{NO}, where $\mathcal{H} = H^{-1}$. Moreover, the constant $\alpha_{\mathcal{H}}$ is chosen separately in each of Ω_O and Ω_{NO}, with the selected values being 0.6 and 0.98, respectively, based on experimental findings.

For Ω_O we slightly alter the calculation of the numerator of Eq. 26 to ensure enough noise is extracted from oriented parts of the image. We calculate the L^2-norm of the noise in the non-oriented region Ω_{NO} and then compute the expected L^2-norm of the noise in the oriented region Ω_O by balancing the value of this norm in Ω_O and the sizes of both regions, by assuming that the noise power is spatially invariant across both regions. Finally, in the denominator of Eq. 26, we use the maximum of the actual measured L^2-norm of the noise in Ω_O and 0.7 times the expected L^2-norm of this noise in this region, based on that in Ω_{NO}.

Execution of the proposed algorithm is stopped when the numerator and denominator of the multiplicative factor in Eq. 26 become close to one another. More specifically, for the non-oriented region Ω_{NO}, iterations are stopped once the numerator becomes greater than 0.95 times the denominator, while for the oriented region Ω_O we stop iterating once the numerator becomes greater than 0.99 of the denominator.

5.2 Calculation of constant $C_{\mathcal{H}}$

In [13] it is shown that when $\mathcal{H} = H^{-1}$, as is the case for the OSV model,

$$C_{H^{-1}} = \frac{1}{MN}\sum_{\substack{p=0,q=0\\(p,q)\neq(0,0)}}^{p=M,q=N} \frac{1}{2\left(2-\cos\left(\frac{2\pi}{M}p\right)-\cos\left(\frac{2\pi}{N}q\right)\right)}. \tag{27}$$

We first observe that the denominator of the expression for $C_{H^{-1}}$ corresponds to the discretization of the Laplacian operator in the frequency domain. Therefore, similar to the H^{-1} case and following the methodology in [13], a similar summation expression is obtained for $C_{H_\zeta^{-1}}$, with the denominator of the summation now corresponding to the discretization of the oriented Laplacian in the frequency domain instead of the discretization of the plain Laplacian in the frequency domain. Since this only corresponds to one directional derivative, instead of two orthogonal directional derivatives, as is the case with the usual Laplacian operator, the denominator will be half the value of that for the usual Laplacian. Therefore, we will have

$$C_{H_\zeta^{-1}} = 2C_{H^{-1}}. \tag{28}$$

5.3 Calculation of $\mathcal{H}$-norms

Recall from elementary Hilbert space theory that the norm of a function on a Hilbert space is defined as the square root of the inner product of the function with itself. From the inner product definition of Eq. 22, the square of the $\mathcal{H}$-norm of ν, $||\nu||_{\mathcal{H}}$ is thus defined by

$$||\nu||^2_{\mathcal{H}} =< \nu, \nu >_{\mathcal{H}}=< \nu, K\nu >_{L^2} \tag{29}$$

In the case of the OSV model, where $\mathcal{H} = H^{-1}$, and $K = -\Delta^{-1}$,

$$||\nu||^2_{H^{-1}} =< \nu, -\Delta^{-1}\nu >_{L^2} . \tag{30}$$

This equation was shown in [13] to simplify, based on Parseval's identity, to

$$||\nu||^2_{H^{-1}} = \frac{1}{MN}\sum_{\substack{p=0,q=0\\(p,q)\neq(0,0)}}^{p=M,q=N} \frac{1}{2\left(2-\cos\left(\frac{2\pi}{M}p\right)-\cos\left(\frac{2\pi}{N}q\right)\right)}\left(|\mathcal{F}(\nu)(p,q)|\right)^2, \tag{31}$$

with $\mathcal{F}(\nu)$ being the discrete Fourier transform of ν, and the image assumed to be MxN pixels large. From Eqs. 23 and 28 above, we find that

Table 1 Final SNRs for local OSV denoising vs. OLOSV, MCM, IOLOSV and NLM denoising on test images

	Denoising method				
Image	OSV	OLOSV	MCM	IOLOSV	NLM
`barbara`	14.219	15.101	15.716	15.845	*17.313*
`fingerprint`	10.990	10.994	*11.963*	11.939	11.930
`bridges`	15.991	16.143	16.214	16.423	*16.834*

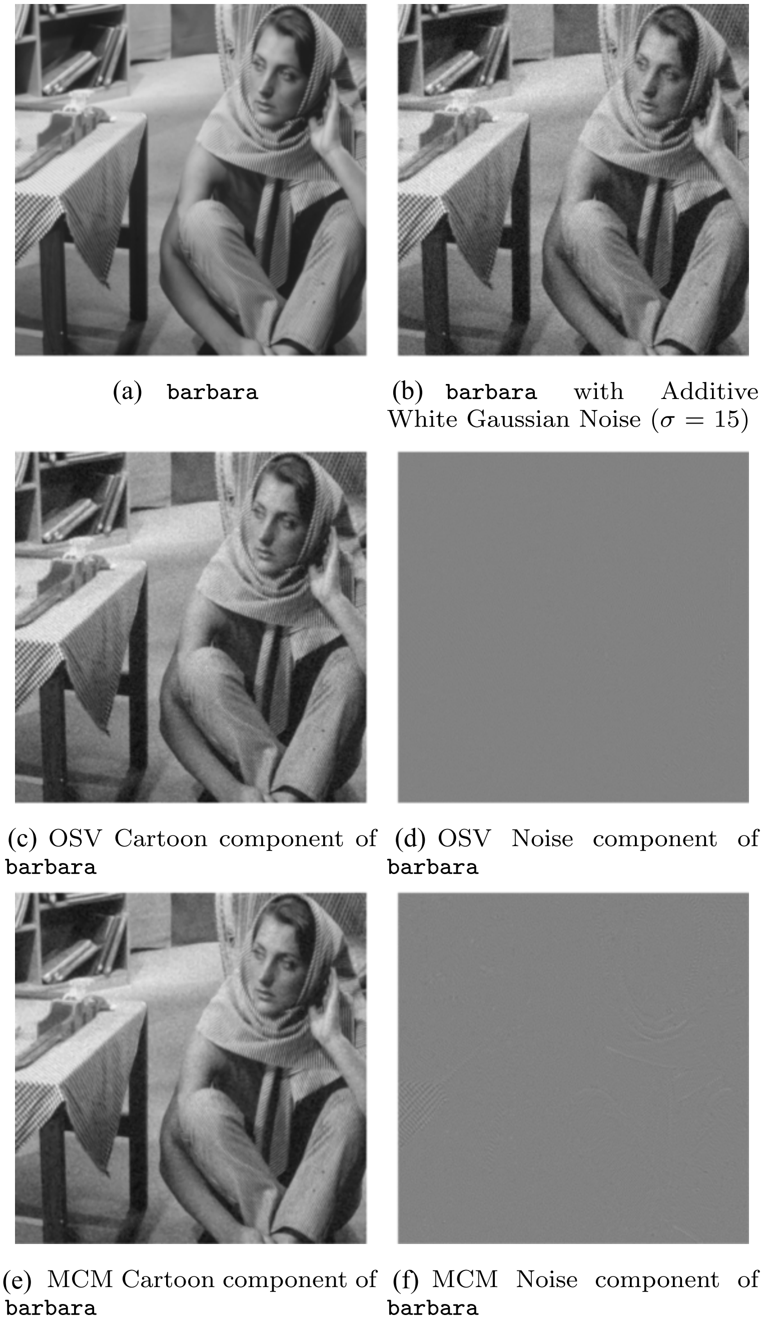
For each test image, the highest SNR value amongst the denoised results from each of the five algorithms tested is shown in italics

$$||\nu||^2_{H^{-1}_{\zeta}} = 2||\nu||^2_{H^{-1}}, \tag{32}$$

both of these squared-norm values being used in the dynamic update of the fidelity parameter λ, for their respective Hilbert spaces.

6 Non-local means

To set this paper within the forefront of research on image denoising, we also compare the proposed OLOSV algorithm with non-local means, a recent addition to the field. Non-local means denoising, introduced in [16] by Buades et al. is known to give good denoising results. It does so by



(a) barbara

(b) barbara with Additive White Gaussian Noise ($\sigma = 15$)

(c) OSV Cartoon component of barbara

(d) OSV Noise component of barbara

(e) MCM Cartoon component of barbara

(f) MCM Noise component of barbara

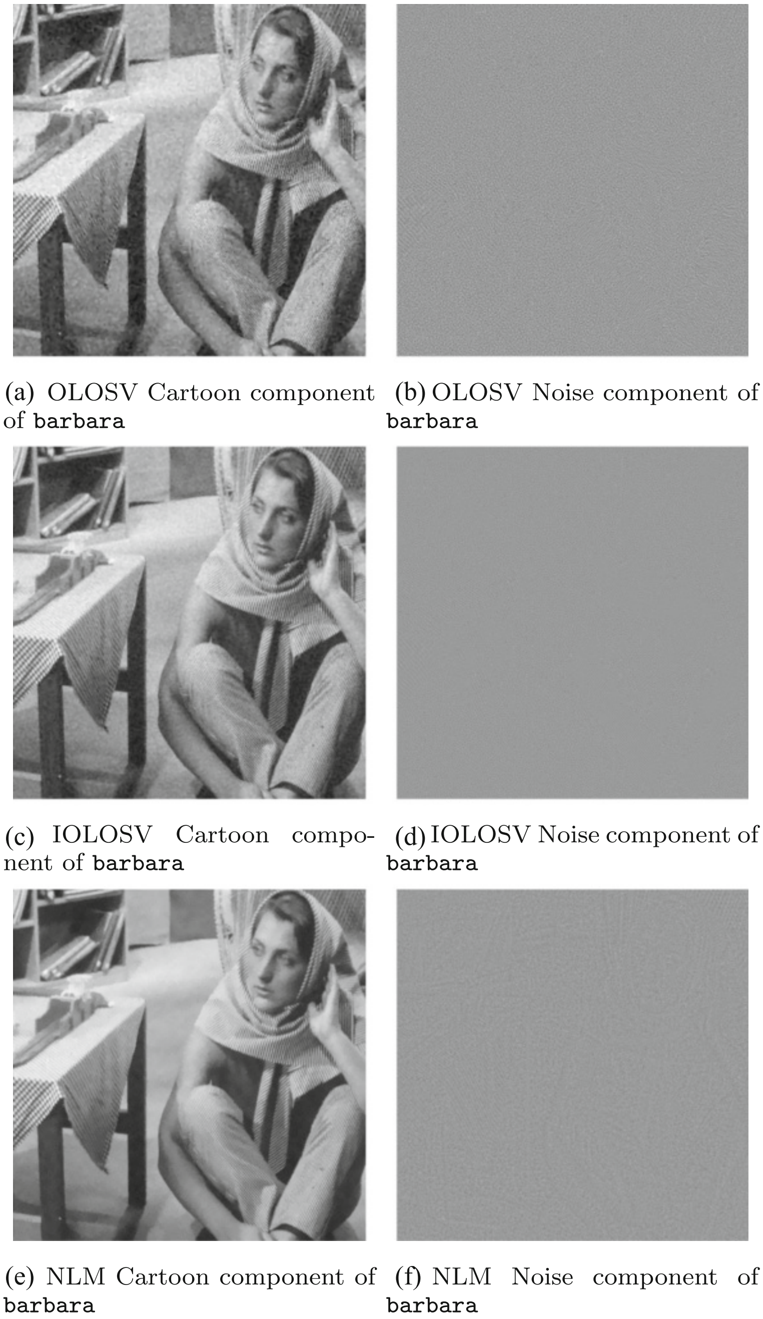
Fig. 3 Original, noisy, OSV and MCM denoising results for barbara image. **a** barbara. **b** barbara with additive Gaussian noise ($\sigma = 15$). **c** OSV cartoon component of barbara. **d** OSV noise component of barbara. **e** MCM cartoon component of barbara. **f** MCM noise component of barbara

comparing image patches from different parts of an image and using all or many of these patches to improve denoising performance by performing a weighted average on these patches based on their similarity in pixel intensities.

From [16], the non-local (NL) mean of a function $u : \Omega \rightarrow \mathbb{R}$ is defined as

$$\bar{u} := \frac{\int_{\Omega} u(y)\omega(x,y)dy}{\int_{\Omega} \omega(x,y)dy}, \tag{33}$$

where $\omega(x,y)$ is a weighting function measuring the similarity between the image patches centred at pixels x and y.



(a) OLOSV Cartoon component of barbara

(b) OLOSV Noise component of barbara

(c) IOLOSV Cartoon component of barbara

(d) IOLOSV Noise component of barbara

(e) NLM Cartoon component of barbara

(f) NLM Noise component of barbara

Fig. 4 OLOSV, IOLOSV, and non-local means denoising results for barbara image. **a** OLOSV cartoon component of barbara. **b** OLOSV noise component of barbara. **c** IOLOSV cartoon component of barbara. **d** IOLOSV noise component of barbara. **e** NLM cartoon component of barbara. **f** MCM noise component of barbara

7 Test images

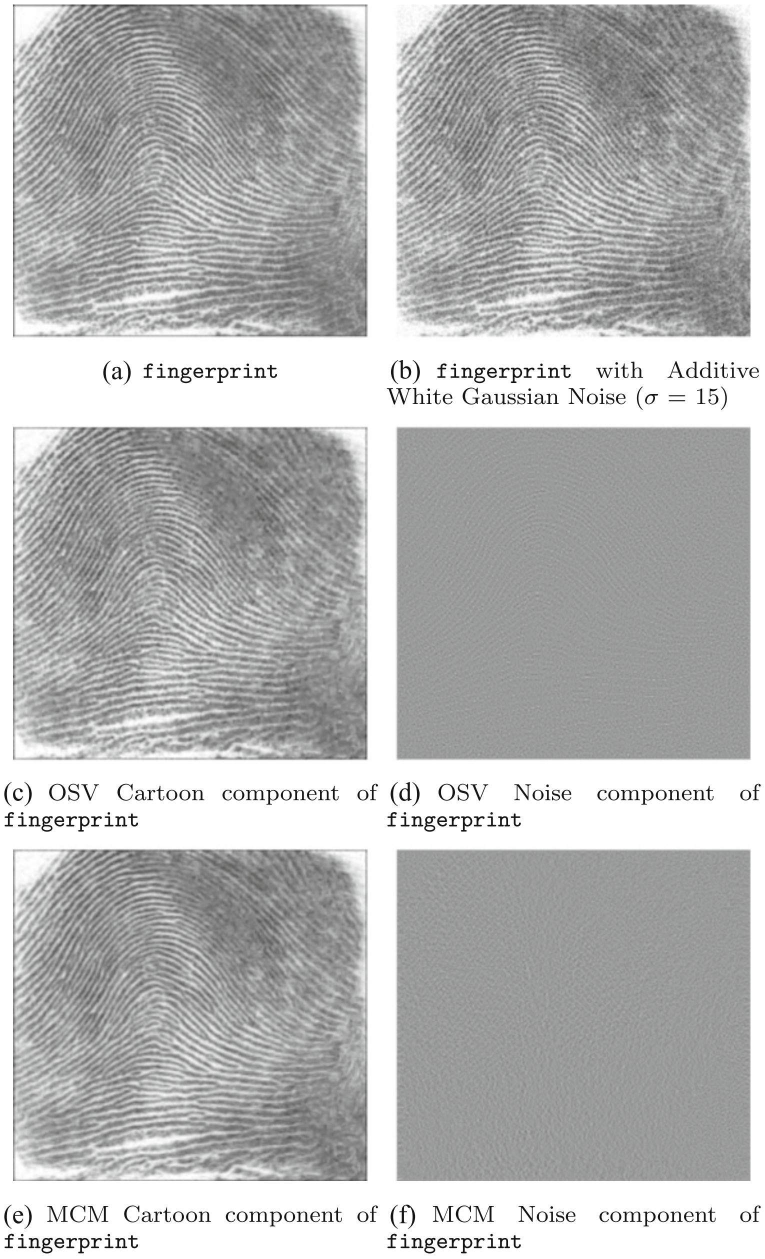
Among the test images used in this paper are several well-known images with significant oriented texture, i.e. `barbara` which is 512 × 512 pixels, `fingerprint` which is 256 × 240 pixels, and `bridges` which is 1024 × 683 pixels, each with bitdepth of 8 bits.

Results obtained for the three test images `barbara`, `bridges` and `fingerprint` are presented in the next section; these results demonstrate that in general, denoised images with higher signal-to-noise ratios are obtained with IOLOSV over both the MCM and OSV algorithms, and the results are visually superior.

8 Experimental results

8.1 Experimental method

To each of the three original test images, additive white Gaussian noise (AWGN) was added with a standard deviation $\sigma = 15$.



(a) `fingerprint`

(b) `fingerprint` with Additive White Gaussian Noise ($\sigma = 15$)

(c) OSV Cartoon component of `fingerprint`

(d) OSV Noise component of `fingerprint`

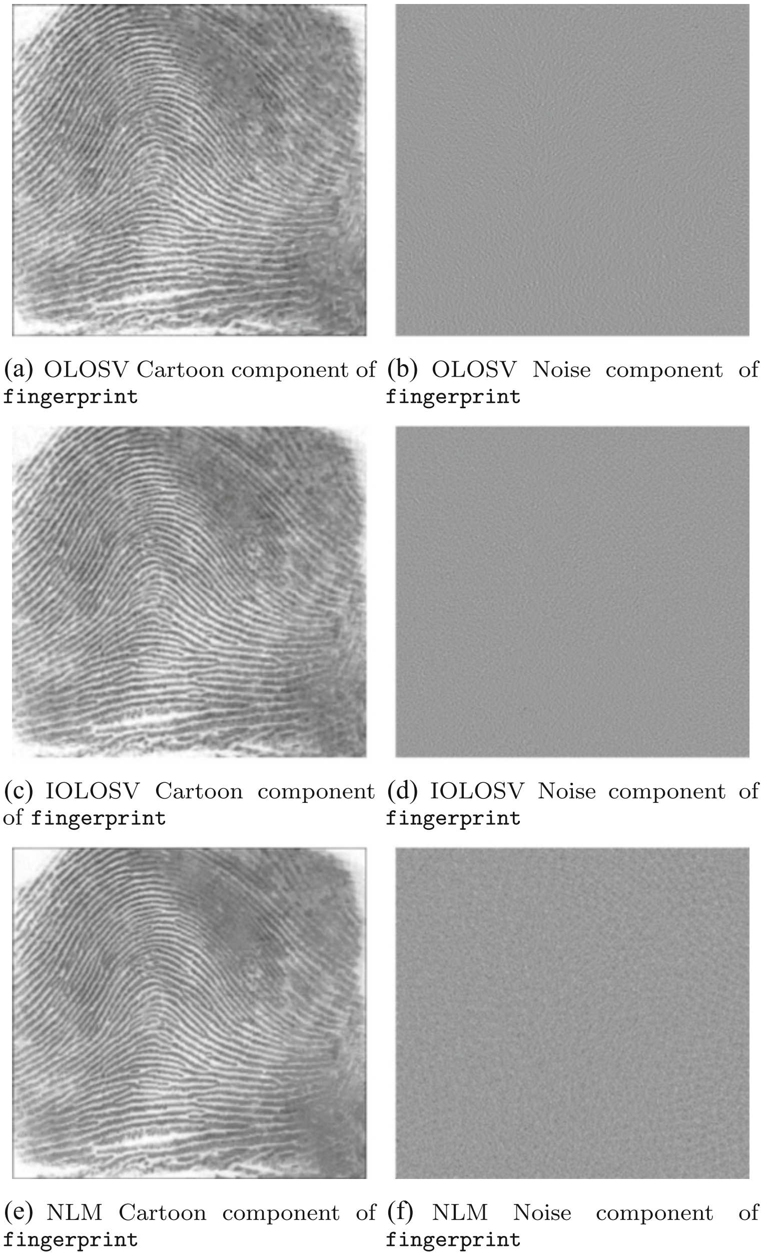
(e) MCM Cartoon component of `fingerprint`

(f) MCM Noise component of `fingerprint`

Fig. 5 Original, noisy, OSV and MCM denoising results for `fingerprint` image. **a** `fingerprint`. **b** `fingerprint` with additive Gaussian noise ($\sigma = 15$). **c** OSV cartoon component of `fingerprint`. **d** OSV noise component of `fingerprint`. **e** MCM cartoon component of `fingerprint`. **f** MCM noise component of `fingerprint`

Then each of the four local iterative denoising algorithms (OSV, OLOSV, IOLOSV and MCM) was run on the noisy test images to obtain a denoised image; the signal-to-noise ratio is then calculated against the noise-free original image. We also ran the non-local means (NLM) algorithm on all three images. For non-local means, 5×5 pixel image patches were used to calculate the weight function measuring similarity between patches centred at each pixel in 11×11 neighborhoods.

Of the four iterative algorithms, only the proposed OLOSV model with implicit timestepping (IOLOSV) had a well-defined stopping condition, whereas the other 3 algorithms were run until a peak in the calculated signal-to-noise ratio (SNR) was detected. This procedure could actually artificially inflate the SNRs of the other 3 algorithms with respect to IOLOSV, since in practice, the SNR is not easily calculated for an image being denoised without having the denoised image at our disposal.



(a) OLOSV Cartoon component of fingerprint

(b) OLOSV Noise component of fingerprint

(c) IOLOSV Cartoon component of fingerprint

(d) IOLOSV Noise component of fingerprint

(e) NLM Cartoon component of fingerprint

(f) NLM Noise component of fingerprint

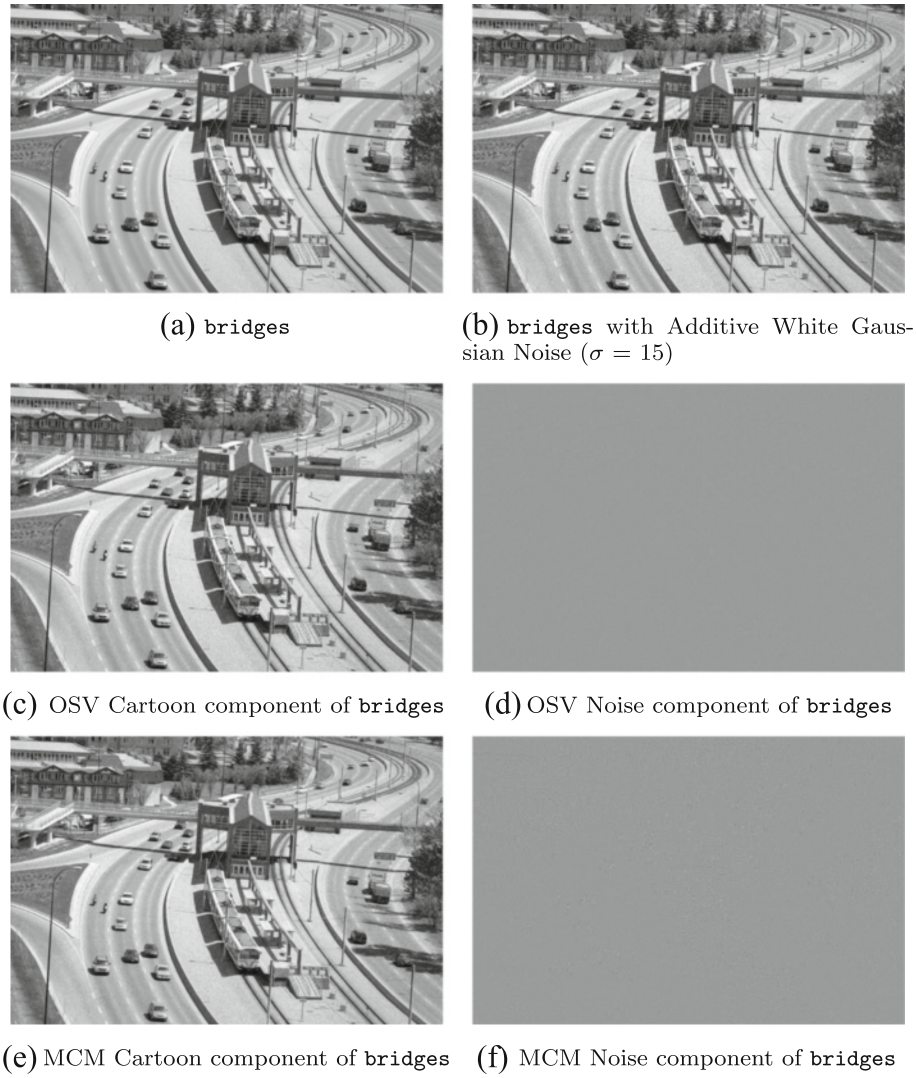
Fig. 6 OLOSV, IOLOSV, and non-local means denoising results for fingerprint image. **a** OLOSV cartoon component of fingerprint. **b** OLOSV noise component of fingerprint. **c** IOLOSV cartoon component of fingerprint. **d** IOLOSV noise component of fingerprint. **e** NLM cartoon component of fingerprint. **f** NLM noise component of fingerprint

As already mentioned in the description of the proposed IOLOSV method in Section 5.1, the values of the fidelity parameter λ for the oriented and non-oriented regions of the image are updated dynamically for each of these regions. We stop the denoising process for the IOLOSV method separately for each of the oriented and non-oriented image regions, based on when the dynamic value of λ for each of these regions stabilizes.

8.2 Results and discussion

Table 1 gives the SNRs of the denoised results from Osher-Solé-Vese decomposition (OSV), Oriented Laplacian OSV with explicit timestepping (OLOSV), Mean Curvature Motion (MCM), and Oriented Laplacian OSV with implicit timestepping (IOLOSV). As can be seen from that table, the NLM SNRs are generally the best, with the IOLOSV and MCM SNRs close behind.

Figures 3, 4, 5, 6, 7 and 8 show the cartoon and noise components obtained from the various denoising algorithms on the three test images, along with the original and noisy images used for testing all of these methods. The appearance of the ν component is much less ordered in the OLOSV and IOLOSV results than the results from the other algorithms. Despite the good SNR of NLM, it must be remembered that SNR is a global measure, which does not always accurately capture visual image quality. The noise components for all images processed with NLM exhibit very visible structure from the underlying noiseless image; this is not to be expected in typical white noise, indicating that the denoising results from NLM are not as


(a) bridges

(b) bridges with Additive White Gaussian Noise ($\sigma = 15$)

(c) OSV Cartoon component of bridges

(d) OSV Noise component of bridges

(e) MCM Cartoon component of bridges

(f) MCM Noise component of bridges

Fig. 7 Original, noisy, OSV and MCM denoising results for bridges image. **a** bridges. **b** bridges with additive Gaussian noise ($\sigma = 15$). **c** OSV cartoon component of bridges. **d** OSV noise component of bridges. **e** MCM cartoon component of bridges. **f** MCM noise component of bridges

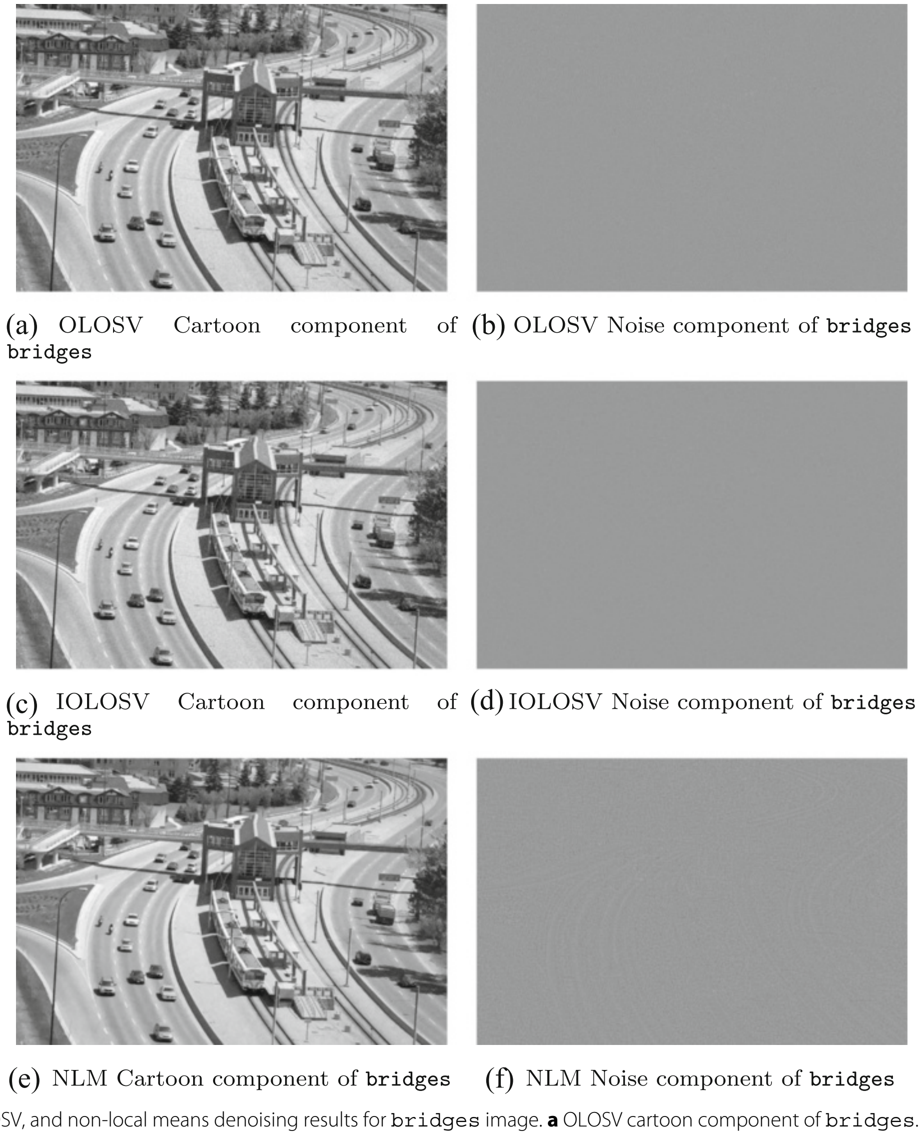
good as the signal-to-noise ratio would suggest, and not as good as the results from the iterative variational methods.

This is further illustrated in Fig. 9, which depicts a zoomed and contrast-enhanced portion of the noise component obtained from the proposed IOLOSV method and the NLM algorithm. The contrast enhancement is done using the `imadjust` command found in Matlab®'s Image Processing Toolbox. In the left panel, the NLM result contains visible structure from the original image which has been removed as part of the noise component, whereas in the right panel, corresponding to the same region from the IOLOSV result, such structures are not visible. Additionally, the NLM result takes longer to generate than the IOLOSV one, due to the high complexity of the non-local means algorithm.

9 Conclusions

The OLOSV decomposition scheme, in both its explicit and implicit forms of implementation, has been found to be extremely good at denoising oriented texture, and is a substantial improvement over the model of Osher, Solé and Vese on which it is based. It also has some other good properties.

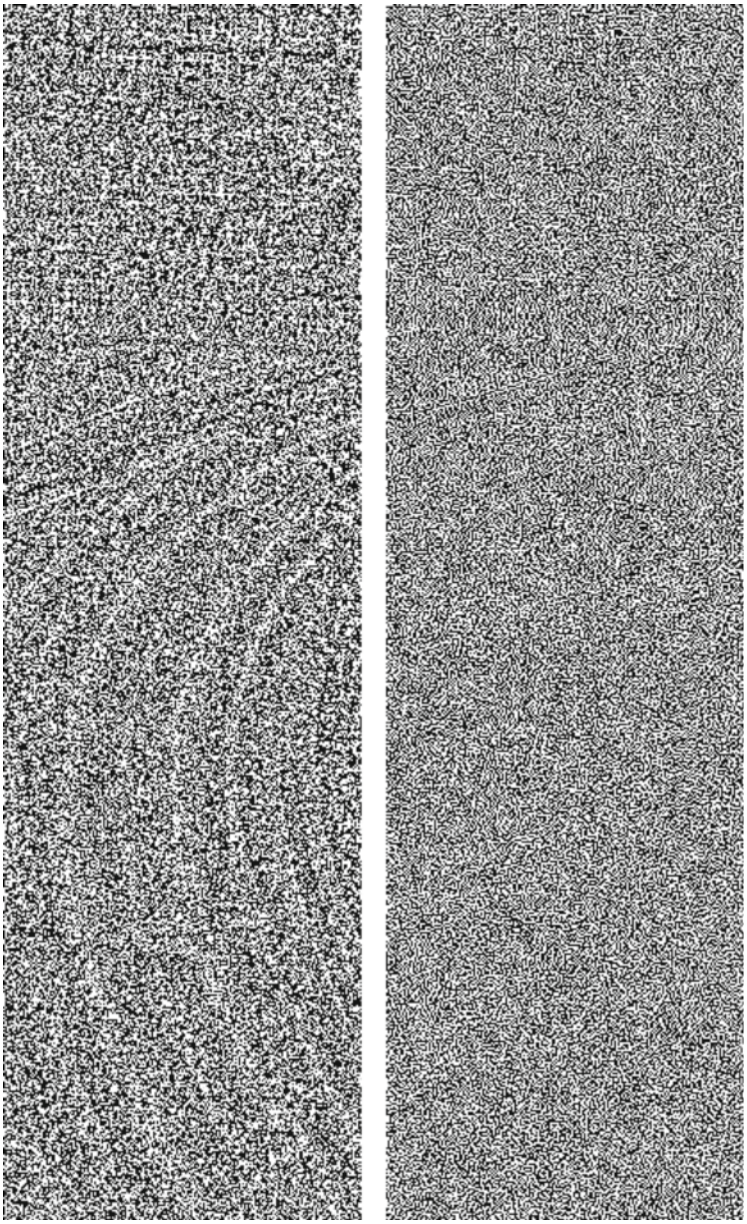
For example, one good property of OLOSV is that it can be generalized to images where there are up to two dominant isophote orientations at a point. These


(a) OLOSV Cartoon component of `bridges` (b) OLOSV Noise component of `bridges`

(c) IOLOSV Cartoon component of `bridges` (d) IOLOSV Noise component of `bridges`

(e) NLM Cartoon component of `bridges` (f) NLM Noise component of `bridges`

Fig. 8 OLOSV, IOLOSV, and non-local means denoising results for `bridges` image. **a** OLOSV cartoon component of `bridges`. **b** OLOSV noise component of `bridges`. **c** IOLOSV cartoon component of `bridges`. **d** IOLOSV noise component of `bridges`. **e** NLM cartoon component of `bridges`. **f** NLM noise component of `bridges`



(a) Contrast-enhanced zoomed portion of NLM Noise component of `bridges`

(b) Contrast-enhanced zoomed portion of IOLOSV Noise component of `bridges`

Fig. 9 Comparison of contrast-enhanced and zoomed portion of noise components from **a** NLM and **b** IOLOSV models

orientations can be calculated using the multiple orientation framework in [17]. In regions where there are two orientations present, the model's evolution equation consists of the sum of two oriented Laplacians with isophote directions ζ_1 and ζ_2 and gradient directions η_1 and η_2.

Additionally, it was found from the experiments in this paper that cartoon edges were much less visible in the noise components v of the OLOSV and IOLOSV results when compared with the OSV noise components (e.g. compare Fig. 5d to Fig. 6b, d). This is because a directional diffusion was performed close to these cartoon edges (if they were calculated as being in the coherent regions of the image), and along the direction of the edges, and not perpendicular to them. Unfortunately, there tends to be some slight visible smearing of the noise close to the edges, but this could be reduced by only including pixels *on* cartoon edges in the region in the image computed to have a coherent orientation, and not those around the edges. The detection of such cartoon edges could be implemented with a Total-Variation-like diffusion, similar to [18].

It may be argued that the Oriented Laplacian OSV model used for denoising is very similar to ordinary Oriented Laplacian diffusion, which when the diffusion across edges is set to zero, degenerates to the mean curvature motion denoising model [19] tested in this paper. However, the OLOSV model has two main differences with standard Oriented Laplacian diffusion:

1. Mathematically OLOSV is a novel nonlinear diffusion flow based on the Laplacian of the level-set curvature of the cartoon component, instead of the Laplacian of the image itself, and
2. OLOSV is based on a model for image decomposition, and thus variations of it could potentially be used for other applications of image decomposition, e.g. inpainting.

In addition to the above two distinguishing properties of the proposed OLOSV model with respect to mean curvature motion, OLOSV also has two distinct advantages:

1. The proposed OLOSV model fits into the Hilbert space formulation of [14], and therefore admits an implicit timestepping implementation, which we called IOLOSV in this paper, as opposed to mean curvature motion which does not. This means that the proposed method is far more efficient; this is confirmed by our experiments, that required only on the order of ten iterations for convergence, as opposed to several hundred for mean curvature motion.
2. The proposed IOLOSV method admits an easy-to-define stopping criterion which allows automatic cessation of execution of the algorithm when the SNR is at or close to its peak value. Such a stopping criterion is not as easily defined for mean curvature motion, necessitating human intervention to determine when the image has in fact been denoised.

Finally, the OLOSV decomposition model can be extended quite easily to colour images by using the colour total variation model of Chan and Blomgren [20]. This remains as future work.

Acknowledgements
The authors would like to acknowledge the partial support of this work by a Natural Sciences and Engineering Research Council (NSERC) of Canada Discovery Grant to Cecilia Moloney.

Competing interests
The authors declare that they have no competing interests.

Received: 2 June 2016 Accepted: 28 October 2016
Published online: 10 November 2016

References

1. Y Meyer, Oscillating Patterns in Image Processing and Nonlinear Evolution Equations: The Fifteenth Dean Jacqueline B. Lewis Memorial Lectures. Am. Math. Soc (2001)
2. R Shahidi, C Moloney, Decorrelating the structure and texture components of a variational decomposition model. IEEE Trans. Image Process. **18**(2), 299–309 (2009)
3. LA Vese, SJ Osher, Modeling textures with total variation minimization and oscillating patterns in image processing. J. Sci.Comput. **19**(1-3) (2003)
4. SJ Osher, A Solé, LA Vese, Image decomposition and restoration using total variation minimization and the H^{-1} norm. Multiscale Model. Simul. SIAM Interdiscip. J. **1**(3), 349–370 (2003)
5. M Bertalmio, G Sapiro, V Caselles, C Ballester, in *Proceedings of the 27th Annual Conference on Computer Graphics and Interactive Techniques*. Image inpainting (ACM Press/Addison-Wesley Publishing Co., New York, 2000), pp. 417–424. doi:10.1145/344779.344972. http://dx.doi.org/10.1145/344779.344972
6. D Paquin, D Levy, E Schreibmann, L Xing, Multiscale image registration. Math. Biosci. Eng. **3**, 389–418 (2006)
7. L Rudin, S Osher, E Fatemi, Nonlinear total variation based noise removal algorithms. Phys. D. **60**, 259–268 (1992)
8. D Tschumperlé, Pde's based regularization of multivalued images and applications. PhD thesis, Université de Nice - Sophia Antipolis (2002)
9. J Weickert, Coherence-enhancing diffusion filtering. Int. J. Comp. Vis. **31**(2-3), 111–127 (1999)
10. L Florack, R Duits, G Jongbloed, MC van Lieshout, L Davies, *Mathematical Methods for Signal and Image Analysis and Representation*. (Springer, 2012). https://books.google.ca/books?id=oby5nrg72iQC. Accessed 22 Aug 2016
11. R Terebes, O Lavialle, P Baylou, M Borda, Mixed anisotropic diffusion. ICPR. **3**, 760–763 (2002). doi: 10.1109/ICPR.2002.1048102
12. A Chambolle, An algorithm for total variation minimization and applications. J. Math. Imaging Vis. **20**, 89–97 (2004)
13. J-F Aujol, A Chambolle, Dual norms and image decomposition models. International Journal of Computer Vision. **63**(1), 85–104 (2005)
14. J-F Aujol, G Gilboa, Constrained and snr-based solutions for tv-hilbert space image denoising. J. Math. Imaging Vis. **26**(1-2), 217–237 (2006). doi: 10.1007/s10851-006-7801-6
15. J-F Aujol, G Gilboa, T Chan, S Osher, in *Proceedings of Variational, Geometric, and Level Set Methods in Computer Vision 2005, LNCS Vol. 3752*. Structure-Texture decomposition by a TV-Gabor model (Springer Berlin Heidelberg, Berlin, 2005), pp. 85–96
16. A Buades, B Coll, J-M Morel, in *Proceedings of the 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)*. A Non-Local Algorithm for Image Denoising, vol. 2 (IEEE Computer Society, Washington, 2005), pp. 60–65. doi:10.1109/CVPR.2005.38. http://dx.doi.org/10.1109/CVPR.2005.38
17. T Aach, C Mota, I Stuke, M Muhlich, E Barth, Analysis of superimposed oriented patterns. IEEE Trans. Image Process. **15**(12), 3690–3700 (2006)
18. R Shahidi, C Moloney, in *Proceedings of the International Conference on Image Processing*. Variational Textured Image Decomposition with Improved Edge Segregation (IEEE, Atlanta, 2006)
19. F Catte, PL Lions, JM Morel, T Coll, Image selective smoothing and edge detection by nonlinear diffusion. SIAM J. Numer. Anal. **29**(1), 182–193 (1992)
20. P Blomgren, TF Chan, Color TV: total variation methods for restoration of vector-valued images. IEEE Trans. Image Process. **7**(3), 304–309 (1998)