Discrete Total Variation: New Definition and Minimization

Laurent Condat

Discrete Total Variation: New Definition and Minimization

Laurent Condat*

Abstract. We propose a new definition for the gradient field of a discrete image, defined on a twice finer grid. The differentiation process from the image to its gradient field is viewed as the inverse operation of linear integration, and the proposed mapping is nonlinear. Then, we define the total variation of an image as the ℓ_1 norm of its gradient field amplitude. This new definition of the total variation yields sharp edges and has better isotropy than the classical definition.

Key words. total variation, variational image processing, coarea formula, finite-difference schemes

AMS subject classifications. 68Q25, 68R10, 68U05

1. Introduction.

In their seminal paper, Rudin, Osher, and Fatemi [1] introduced the total variation (TV) regularization functional for imaging problems. Since then, a variety of papers has demonstrated the effectiveness of TV minimization to recover sharp images, by preserving strong discontinuities, while removing noise and other artifacts [2–4]. TV minimization also appears in clustering and segmentation problems, by virtue of the coarea formula [5, 6]. The TV can be defined in other settings than image processing, for instance on graphs [7]. Numerical minimization of the TV has long been challenging, but recent advances in large-scale convex nonsmooth optimization, with efficient primal–dual splitting schemes and alternating directions methods, have made the implementation of TV minimization relatively easy and efficient [3, 8–19]. Yet, the rigorous definition of the TV for discrete images has received little attention. For continuously defined two-dimensional functions, the TV is simply the L_1 norm of the gradient amplitude. But for discrete images, it is a nontrivial task to properly define the gradient using finite differences, as is well known in the community of computer graphics and visualization [20, 21]. The classical, so called "isotropic" definition of the discrete TV, is actually far from being isotropic, but it performs reasonably well in practice. In this paper, we propose a new definition of the discrete TV, which corrects some drawbacks of the classical definition and yields sharper edges and structures. The key idea is to associate, in a nonlinear way, an image with a gradient field on a twice finer grid. The TV of the image is then simply the ℓ_1 norm of this gradient field amplitude.

In section 2, we review the classical definitions of the discrete TV and their properties. In section 3, we introduce our new definition of the TV in the dual domain and in section 4, we study the equivalent formulation in the primal domain. An algorithm to solve problems regularized with the proposed TV is presented in section 5. The good performances of the proposed TV on some test imaging problems are demonstrated in section 6.

2. Classical definitions of the discrete TV and their properties.

A function $s(t_1, t_2)$ defined in the plane $\mathbb{R}^2$, under some regularity assumptions, has a gradient field $\nabla s(t_1, t_2) = \big(\frac{\partial s}{\partial t_1}(t_1, t_2), \frac{\partial s}{\partial t_2}(t_1, t_2)\big)$, defined in $\mathbb{R}^2$ as well. We can then define the TV of s as the $L_{1,2}$ norm of the gradient: $\mathrm{TV}(s) = \int_{\mathbb{R}^2} |\nabla s(t_1, t_2)|\, \mathrm{d}t_1 \mathrm{d}t_2$, where $|(a, b)|$ is a shorthand notation for the

*Univ. Grenoble Alpes, GIPSA-Lab, F-38000 Grenoble, France (http://www.gipsa-lab.fr/~laurent.condat/).

2-norm $\sqrt{a^2+b^2}$. The TV has the desirable property of being isotropic, or rotation-invariant: a rotation of s in the plane does not change the value of its TV.

A (grayscale) discrete image x of size $N_1 \times N_2$ has its pixel values $x[n_1,n_2]$ defined at the locations (n_1,n_2) in the domain $\Omega = \{1,\dots,N_1\} \times \{1,\dots,N_2\}$, where n_1 and n_2 are the row and column indices, respectively, and the pixel with index $(1,1)$ is at the top left image corner. The pixel values are supposed to lie between 0 (black) and 1 (white). The challenge is then to define the discrete TV of x, using only its pixel values, while retaining the mathematical properties of the continuous TV. The so-called *anisotropic* TV is defined as

$$\mathrm{TV}_{\mathrm{a}}(x) = \sum_{n_1=1}^{N_1}\sum_{n_2=1}^{N_2} \big|x[n_1+1,n_2]-x[n_1,n_2]\big| + \big|x[n_1,n_2+1]-x[n_1,n_2]\big|, \tag{1}$$

assuming Neumann (symmetric) boundary conditions: a finite difference across a boundary, like $x[N_1+1,n_2]-x[N_1,n_2]$, is assumed to be zero. The anisotropic TV is well known to be a poor definition of the discrete TV, as it yields metrication artifacts: its minimization favors horizontal and vertical structures, because oblique structures make the TV value larger as it should be. Therefore, one usually uses the so-called *isotropic* TV defined as

$$\mathrm{TV}_{\mathrm{i}}(x) = \sum_{n_1=1}^{N_1}\sum_{n_2=1}^{N_2} \sqrt{\big(x[n_1+1,n_2]-x[n_1,n_2]\big)^2 + \big(x[n_1,n_2+1]-x[n_1,n_2]\big)^2}, \tag{2}$$

using Neumann boundary conditions as well.

It is hard to quantify the isotropy of a functional like the TV, since the grid $\mathbb{Z}^2$ is not isotropic and there is no unique way of defining the rotation of a discrete image. However, it is natural to require, at least, that after a rotation of $\pm 90^{\mathrm{o}}$, or a horizontal or vertical flip, the TV of the image remains unchanged. It turns out that this is not the case with the isotropic TV, with a change factor as large as $\sqrt{2}$ after a horizontal flip, see in Table 1 the TV of an edge at $+45^{\mathrm{o}}$ and at -45^{o}. In spite of this significant drawback, the isotropic TV is widely used, for its simplicity. We can note that a straightforward way to restore the four-fold symmetry is to define the TV as the average of TV_{i} applied to the image rotated by 0^{o}, 90^{o}, -90^{o}, 180^{o}. But the drawbacks of TV_{i}, stressed below, would be maintained, like the tendency to blur oblique edges and the too low value for an isolated pixel or a checkerboard.

An attempt to define a more isotropic TV has been made with the *upwind* TV [22], defined as

$$\mathrm{TV}_{\mathrm{u}}(x) = \sum_{n_1=1}^{N_1}\sum_{n_2=1}^{N_2} \sqrt{\begin{array}{l}\big(x[n_1,n_2]-x[n_1+1,n_2]\big)_+^2 + \big(x[n_1,n_2]-x[n_1-1,n_2]\big)_+^2 + \\ \big(x[n_1,n_2]-x[n_1,n_2+1]\big)_+^2 + \big(x[n_1,n_2]-x[n_1,n_2-1]\big)_+^2\end{array}}, \tag{3}$$

where $(a)_+$ means $\max(x,0)$. The upwind TV is indeed more isotropic and produces sharp oblique edges, but as shown below, it is not invariant by taking the image negative, i.e. replacing the image x by $1-x$. Since $\mathrm{TV}_{\mathrm{u}}(x) \neq \mathrm{TV}_{\mathrm{u}}(1-x) = \mathrm{TV}_{\mathrm{u}}(-x)$, the upwind TV is not a seminorm, contrary to the other forms considered in this paper. In practice, it penalizes correctly small dark structures over a light background, but not the opposite, see the striking example in Figure 10 (e).

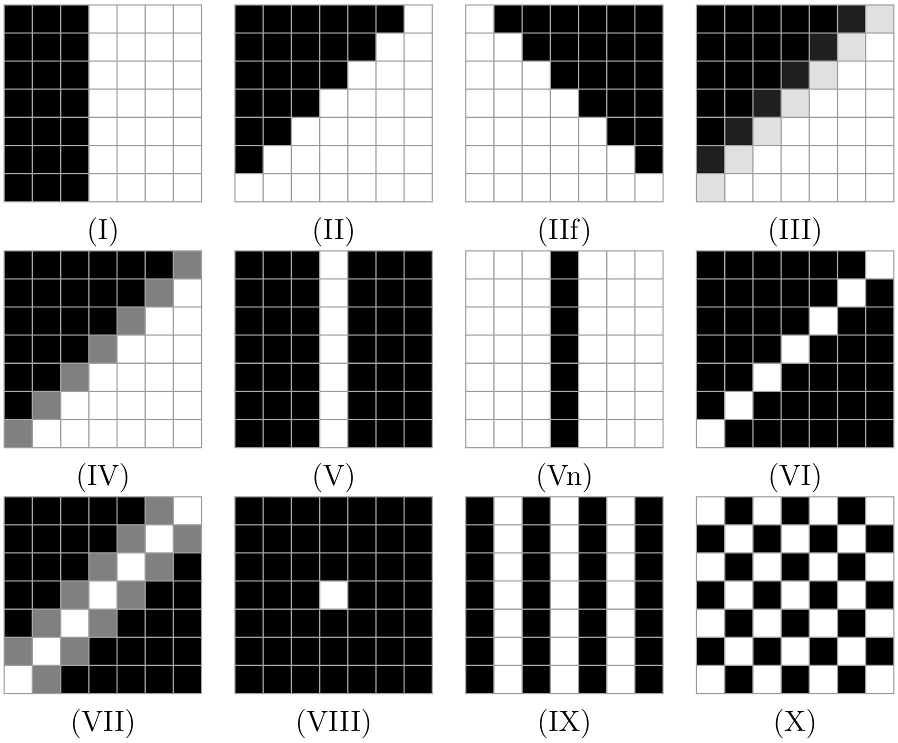


Figure 1. *Some patterns, for which we report the value of the TV in Table 1. Black and white correspond to 0 and 1, respectively. In (III), the transition goes through the levels 0, 1/8, 7/8, 1. In (IV), the transition goes through the levels 0, 1/2, 1. In (VII), the transition goes through the levels 0, 1/2, 1, 1/2, 0.*

Another formulation of the discrete TV, called "Shannon Total Variation" was proposed recently [23], at the time the present paper was finalized; so, this formulation, which has good isotropy properties, is not included in our comparisons. It aims at estimating the continuous TV of the Shannon interpolate of the image, by using a Riemann sum approximation of the corresponding integral. This way, aliasing is removed from the images, at the price of slightly more blurred edges.

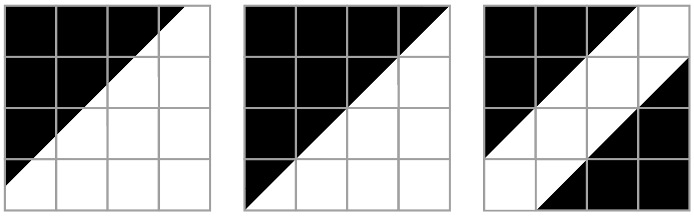
To evaluate the different definitions of the discrete TV, we consider typical patterns of size $N \times N$, depicted in Figure 1, and we report the corresponding value of the TV in Table 1, when N is large, i.e. ignoring the influence of the image boundaries. For some patterns, we consider its horizontally flipped version, denoted by a 'f', see patterns (II) and (IIf) in Figure 1, and its negative version, denoted by a 'n', see patterns (V) and (Vn). In Table 1, the value is in green if it is an appropriate value for this case, and in red if not. In this respect, some considerations must be reported. An isolated pixel, like in patterns (VIII) or (VIIIn), can be viewed as the discretization by cell-averaging, i.e. $x[n_1, n_2] = \int_{n_1-1/2}^{n_1+1/2} \int_{n_2-1/2}^{n_2+1/2} s(t_1, t_2) \mathrm{d}t_1 \mathrm{d}t_2$, of the indicator function (1 inside, 0 outside) $s(t)$ of a square of size one pixel. According to the coarea formula, the continuous TV of the indicator function of a set is equal to the perimeter of that set. So, it is natural to ask the TV value in pattern (VIII) to be equal to 4. The isotropic TV

Table 1

Asymptotic value, when the image is of size $N \times N$ and $N \to +\infty$, of the TV, for the examples depicted in Figure 1. A 'f' means a horizontal flip and a 'n' means taking the image negative. $\mathrm{TV_a}$, $\mathrm{TV_i}$, $\mathrm{TV_u}$, $\mathrm{TV_p}$ *are the anisotropic, isotropic, upwind, proposed TV, defined in (1), (2), (3), (8), respectively.*

	$\mathrm{TV_a}$	$\mathrm{TV_i}$	$\mathrm{TV_u}$	$\mathrm{TV_p}$
(I)	N	N	N	N
(II)	$2N$	$\sqrt{2}N$	$\sqrt{2}N$	$2N$
(IIf)	$2N$	$2N$	$\sqrt{2}N$	$2N$
(III)	$2N$	$\sqrt{2}N$	$\sqrt{2}N$	$\sqrt{2}N$
(IIIf)	$2N$	$(\sqrt{37}+1)N/4$	$\sqrt{2}N$	$\sqrt{2}N$
(IV)	$2N$	$\sqrt{2}N$	$\sqrt{2}N$	$\sqrt{2}N$
(IVf)	$2N$	$(1+1/\sqrt{2})N$	$\sqrt{2}N$	$\sqrt{2}N$
(V)	$2N$	$2N$	$\sqrt{2}N$	$2N$
(Vn)	$2N$	$2N$	$2N$	$2N$
(VI)	$4N$	$2\sqrt{2}N$	$2N$	$4N$
(VIf)	$4N$	$(2+\sqrt{2})N$	$2N$	$4N$
(VIn)	$4N$	$2\sqrt{2}N$	$2\sqrt{2}N$	$4N$
(VII)	$4N$	$2\sqrt{2}N$	$(\sqrt{2}+1)N$	$2\sqrt{2}N$
(VIIf)	$4N$	$(3/\sqrt{2}+1)N$	$(\sqrt{2}+1)N$	$2\sqrt{2}N$
(VIIn)	$4N$	$2\sqrt{2}N$	$2\sqrt{2}N$	$2\sqrt{2}N$
(VIII)	4	$2+\sqrt{2}$	2	4
(VIIIn)	4	$2+\sqrt{2}$	4	4
(IX)	N^2	N^2	$N^2/\sqrt{2}$	N^2
(X)	$2N^2$	$\sqrt{2}N^2$	N^2	$2N^2$

and upwind TV take too small values. This is a serious drawback, since they do not penalize noise as much as they should, and penalizing noise is the most important property of a functional used to regularize ill-posed problems. For the checkerboard (X), it is natural to expect a value of $2N^2$. It is important that this value is not lower, because an inverse problem like demosaicking consists in demultiplexing luminance information and chrominance information modulated at this highest frequency [24, 25]. Interpolation on a quincunx grid also requires penalizing the checkerboard sufficiently. The isotropic TV gives a value of $\sqrt{2}N^2$, which is too small, and the upwind TV gives an even smaller value of N^2. Then, an important property of the TV is to be convex and one-homogeneous, so that the TV of a sum of images is less or equal than the sum of their TV. Consequently, viewing the checkerboard as a sum of diagonal lines, like the one in (VI), disposed at every two pixels, the TV of the diagonal line (VI) cannot be lower than $4N$. That is, the lower value of $2\sqrt{2}N$, achieved by the isotropic TV, is not compatible with the value of $2N^2$ for the checkerboard and with convexity of the TV. We can notice that the line in (VI) cannot be explained as the discretization by cell-averaging of a continuously defined diagonal ridge. So, it is coherent that its jaggy nature is penalized. By contrast, the pattern in (VII) can be viewed as the discretization by cell-averaging of a diagonal ridge, depicted in Figure 2 (c). So, a TV value of $2\sqrt{2}N$ is appropriate for this case. Further on, the line in (VI) can be viewed as the difference of two edges like in (II), one of



(a) (b) (c)

Figure 2. *In (a), (b), (c), continuously-defined images whose cell-average discretization yields Figure 1 (III), (IV), (VII), respectively.*

which shifted by one pixel. So, by convexity, the value of the TV for the edge in (II) cannot be lower than $2N$. The value of $\sqrt{2}N$, we could hope for by viewing (II) as a diagonal edge discretized by point sampling, is not accessible. Again, after a small blur, the discrete edges in (III) and (IV) become compatible with a diagonal edge discretized by cell-averaging, see the edges in Figure 2 (a) and (b), respectively. So, the expected value of the TV is $\sqrt{2}N$ in these cases. It is true that a TV value of $\sqrt{2}N$ would be nice for the binary edge (II), especially for partitioning applications [6], and that the isotropic TV achieves this value, but the price to pay with the isotropic TV is a higher value of $2N$ for the flipped case (IIf), which does not decrease much by blurring the edge to (IIIf) or (IVf). Therefore, minimizing the isotropic TV yields nice binary edges at the diagonal orientation like in (II), but significantly blurred edges for the opposite orientation, as can be observed in Figure 8 (c), Figure 5 (d), Figure 12 (b).

We can mention, mainly in the literature of computational fluid or solid mechanics, the use of staggered grid discretizations of partial differential equations, or marker and cell method [26], wherein different variables, like the pressure and velocity, are located at different positions on the grid, i.e. at cell centers or at cell edges. This idea is also applied in so-called mimetic finite difference methods [27, 28]. Transposed to the present context, pixel values are located at the pixel centers, whereas a finite difference like $x[n_1+1,n_2]-x[n_1,n_2]$ is viewed as the vertical component of the gradient at the spatial position $(n_1+\frac{1}{2},n_2)$, i.e. at an edge between two pixels [29]. This interpretation is insightful, but it does not specify how to define the norm of the gradient. The proposed approach is different from this framework in two respects. First, we define the image gradient field not only at the pixel edges, but also at the pixel centers. Second, a finite difference like $x[n_1+1,n_2]-x[n_1,n_2]$ is not viewed as an estimate of a partial derivative, but as its local integral; we develop this interpretation in section 4.

3. Proposed discrete TV: dual formulation.

It is well known that in the continuous domain, the TV of a function s can be defined by duality as

$$\mathrm{TV}(s)=\sup\Big\{\langle s,-\mathrm{div}(u)\rangle\ :\ u\in\mathcal{C}_c^1(\mathbb{R}^2,\mathbb{R}^2),\ |u(t)|\leq 1\ (\forall t\in\mathbb{R}^2)\Big\}, \tag{4}$$

where $\mathcal{C}_c^1(\mathbb{R}^2,\mathbb{R}^2)$ is the set of continuously differentiable functions from $\mathbb{R}^2$ to $\mathbb{R}^2$ with compact support and div is the divergence operator. So, the dual variable u has its amplitude bounded by one everywhere.

In the discrete domain, the TV can be defined by duality as well. First, let us define the discrete operator D, which maps an image $x \in \mathbb{R}^{N_1 \times N_2}$ to the vector field $Dx \in (\mathbb{R}^2)^{N_1 \times N_2}$ made of forward finite differences of x; that is,

$$(Dx)_1[n_1,n_2] = x[n_1+1,n_2] - x[n_1,n_2], \tag{5}$$

$$(Dx)_2[n_1,n_2] = x[n_1,n_2+1] - x[n_1,n_2], \tag{6}$$

for every $(n_1,n_2) \in \Omega$, with Neumann boundary conditions. Note that for ease of implementation, it is convenient to have all images and vector fields of same size $N_1 \times N_2$, indexed by $(n_1,n_2) \in \Omega$, keeping in mind that for some of them, the last row or column is made of dummy values equal to zero, which are constant and should not be viewed as variables; for instance, $(Dx)_1[N_1,n_2] = (Dx)_2[n_1,N_2] = 0$, for every $(n_1,n_2) \in \Omega$. So, $\mathrm{TV_i}(x) = \|Dx\|_{1,2}$, where the $\ell_{1,2}$ norm is the sum over the indices n_1, n_2 of the 2-norm $|(Dx)[n_1,n_2]|$.

Then, the isotropic TV of an image x can be defined by duality as

$$\mathrm{TV_i}(x) = \max_{u \in (\mathbb{R}^2)^{N_1 \times N_2}} \Big\{ \langle Dx, u \rangle \ : \ |u[n_1,n_2]| \leq 1,\ \forall (n_1,n_2) \in \Omega \Big\}, \tag{7}$$

with the usual Euclidean inner product.

The scalar dual variables $u_1[n_1,n_2]$ and $u_2[n_1,n_2]$, like the finite differences $(Dx)_1[n_1,n_2]$ and $(Dx)_2[n_1,n_2]$, can be viewed as located at the points $(n_1+\frac{1}{2},n_2)$ and $(n_1,n_2+\frac{1}{2})$, respectively. So, the anisotropy of the isotropic TV can be explained by the fact that these variables, which are combined in the constraint $|u[n_1,n_2]| \leq 1$, are located at different positions. We propose to correct this half-pixel shift by interpolation: we look for the dual images u_1 and u_2, whose values $u_1[n_1,n_2]$ and $u_2[n_1,n_2]$ are located at the pixel edges $(n_1+\frac{1}{2},n_2)$ and $(n_1,n_2+\frac{1}{2})$, respectively, such that, when interpolated, the constraint $|u[n_1,n_2]| \leq 1$ is satisfied both at pixel centers and at pixel edges. So, the proposed TV, denoted $\mathrm{TV_p}$, is defined in the dual domain as

$$\begin{aligned}\mathrm{TV_p}(x) = \max_{u \in (\mathbb{R}^2)^{N_1 \times N_2}} \Big\{ &\langle Dx, u \rangle \ : \\ &|(L_{\updownarrow} u)[n_1,n_2]| \leq 1,\ |(L_{\leftrightarrow} u)[n_1,n_2]| \leq 1,\ |(L_{\bullet} u)[n_1,n_2]| \leq 1,\ \forall (n_1,n_2) \in \Omega \Big\},\end{aligned} \tag{8}$$

where the three operators $L_{\updownarrow}$, $L_{\leftrightarrow}$, $L_{\bullet}$ interpolate bilinearly the image pair $u = (u_1,u_2)$ on the grids $(n_1+\frac{1}{2},n_2)$, $(n_1,n_2+\frac{1}{2})$, (n_1,n_2), for $(n_1,n_2) \in \Omega$, respectively. That is,

$$(L_{\updownarrow} u)_1[n_1,n_2] = u_1[n_1,n_2], \tag{9}$$

$$(L_{\updownarrow} u)_2[n_1,n_2] = (u_2[n_1,n_2] + u_2[n_1,n_2-1] + u_2[n_1+1,n_2] + u_2[n_1+1,n_2-1])/4, \tag{10}$$

$$(L_{\leftrightarrow} u)_1[n_1,n_2] = (u_1[n_1,n_2] + u_1[n_1-1,n_2] + u_1[n_1,n_2+1] + u_1[n_1-1,n_2+1])/4, \tag{11}$$

$$(L_{\leftrightarrow} u)_2[n_1,n_2] = u_2[n_1,n_2], \tag{12}$$

$$(L_{\bullet} u)_1[n_1,n_2] = (u_1[n_1,n_2] + u_1[n_1-1,n_2])/2, \tag{13}$$

$$(L_{\bullet} u)_2[n_1,n_2] = (u_2[n_1,n_2] + u_2[n_1,n_2-1])/2, \tag{14}$$

for every $(n_1,n_2) \in \Omega$, replacing the dummy values $u_1[0,n_2]$, $u_2[n_1,0]$, $u_1[N_1,n_2]$, $u_2[n_1,N_1]$, $(L_{\updownarrow} u)_1[N_1,n_2]$, $(L_{\updownarrow} u)_2[N_1,n_2]$, $(L_{\leftrightarrow} u)_1[n_1,N_2]$, $(L_{\leftrightarrow} u)_2[n_1,N_2]$ by zero.

Thus, we mimic the continuous definition (4), where the dual variable is bounded everywhere, by imposing that it is bounded on a grid three times more dense than the pixel grid. Actually, for the dual variable to be bounded everywhere after bilinear interpolation, the fourth lattice of pixel corners $(n_1 + \frac{1}{2}, n_2 + \frac{1}{2})$ must be added; that is, we can define a variant of the proposed approach, in which we add to the constraint set in (8), the additional constraints $|(L_+ u)[n_1, n_2]| \leq 1$, where the operator L_+ interpolates bilinearly the image pair u on the grid $(n_1 + \frac{1}{2}, n_2 + \frac{1}{2})$:

$$(L_+ u)_1[n_1, n_2] = (u_1[n_1, n_2] + u_1[n_1, n_2 + 1])/2, \tag{15}$$

$$(L_+ u)_2[n_1, n_2] = (u_2[n_1, n_2] + u_2[n_1 + 1, n_2])/2. \tag{16}$$

In the Matlab code accompanying the paper, available on the author's webpage, this variant is implemented, as well. The author observed empirically that this variant, in general, brings very minor changes in the images, which are not worth the extra computational burden. That is why, in the rest of the paper, we focus on the version with the dual variables and the gradient defined on three lattices, and not on this variant with four lattices.

Our definition of the discrete TV, using interpolation in the dual domain, is not new: it was proposed in [30] and called *staggered grid discretization* of the TV. With the isotropic TV, the projection of the image pair u onto the $l_{\infty,2}$ norm ball, which amounts to simple pixelwise shrinkage, can be used. But using the same algorithms with the proposed TV requires projecting u onto the set $\{u \ : \ \|L_{\updownarrow} u\|_{\infty,2} \leq 1, \|L_{\leftrightarrow} u\|_{\infty,2} \leq 1, \|L_{\bullet} u\|_{\infty,2} \leq 1\}$. There is no closed form for this projection. We emphasize that in [30], and certainly in other papers using this dual staggered grid discretization, this projection is not implemented, and is replaced by an approximate shrinkage, see [30, Eq. (64)]. This operation is not a projection onto the set above, since it is not guaranteed to yield an image pair satisfying the bound constraints, and it is not a firmly nonexpansive operator [31]; this means that the convergence guarantees of usual iterative fixed-point algorithms are lost, and that if convergence occurs, there is no way to characterize the obtained solution, which depends on the algorithm, the initial conditions, and the parameters. By contrast, we will propose a generic splitting algorithm, with proved convergence to exact solutions of problems involving the proposed TV, in section 5.

4. Proposed discrete TV: primal formulation.

We have defined the proposed TV implicitly in (8), as the optimal value of an optimization problem, expressed in terms of the dual image pair u. In the frame of the Fenchel–Rockafellar duality [31], we can define the proposed TV as the optimal value of an equivalent optimization problem, expressed in terms of what we will consider as the gradient field of the image.

Proposition 1. *Given an image x, the proposed TV has the following primal formulation, equivalent to the dual formulation (8):*

$$\mathrm{TV_p}(x) = \min_{v_{\updownarrow}, v_{\leftrightarrow}, v_{\bullet} \in (\mathbb{R}^2)^{N_1 \times N_2}} \Big\{ \|v_{\updownarrow}\|_{1,2} + \|v_{\leftrightarrow}\|_{1,2} + \|v_{\bullet}\|_{1,2} \ : \ L_{\updownarrow}^* v_{\updownarrow} + L_{\leftrightarrow}^* v_{\leftrightarrow} + L_{\bullet}^* v_{\bullet} = Dx \Big\}, \tag{17}$$

where $\cdot^*$ denotes the adjoint operator.

Before proving Proposition 1, we give more compact forms of the primal and dual definitions of the proposed TV. For this, let us define the linear operator L as the concatenation of $L_\updownarrow$, $L_\leftrightarrow$, $L_\bullet$, and the $\ell_{\infty,\infty,2}$ norm $\|\cdot\|_{\infty,\infty,2}$ of a field as the maximum over the three components and the pixels of the 2-norm of its vectors. Then we can rewrite (8) as

$$\mathrm{TV}_\mathrm{p}(x) = \max_{u\in(\mathbb{R}^2)^{N_1\times N_2}} \Big\{\langle Dx, u\rangle \ : \ \|Lu\|_{\infty,\infty,2} \leq 1\Big\}. \tag{18}$$

Let the vector field v be the concatenation of the three vector fields $v_\updownarrow$, $v_\leftrightarrow$, and $v_\bullet$, which appear in (17). Let the $\ell_{1,1,2}$ norm of v be the sum of the $\ell_{1,2}$ norm of its three components $v_\updownarrow$, $v_\leftrightarrow$, $v_\bullet$. We have $L^*v = L_\updownarrow^* v_\updownarrow + L_\leftrightarrow^* v_\leftrightarrow + L_\bullet^* v_\bullet$. Then we can rewrite (17) as

$$\mathrm{TV}_\mathrm{p}(x) = \min_{v\in\left((\mathbb{R}^2)^{N_1\times N_2}\right)^3} \Big\{ \|v\|_{1,1,2} \ : \ L^*v = Dx\Big\}. \tag{19}$$

Proof of Proposition 1. A (primal) convex optimization problem of the form: $\mathrm{minimize}_v\{F(L^*v) + G(v)\}$, for two convex, lower semicontinuous functions F and G and a linear operator L^*, has a Fenchel–Rockafellar dual problem of the form: $\mathrm{maximize}_u\{-F^*(u) - G^*(-Lu)\}$, where F^* and G^* are the Legendre–Fenchel conjugates of F and G, respectively [31]. Moreover, strong duality holds, and the primal and dual problems have the same optimal value; that is, if a minimizer $\hat{v}$ of the primal problem and a maximizer $\hat{u}$ of the dual problem exist, we have $-F^*(\hat{u}) - G^*(-L\hat{u}) = F(L^*\hat{v}) + G(\hat{v})$. In our case, F is the convex indicator function of the set $\{v \ : \ L^*v = Dx\}$; that is, the function which maps its variable to 0 if it belongs to this set, to $+\infty$ else. G is the $\ell_{1,1,2}$ norm. Then it is well known that F^* maps u to $\langle u, Dx\rangle$ and that the Legendre–Fenchel conjugate of the $\ell_{1,2}$ norm is the convex indicator function of the $\ell_{\infty,2}$ norm ball [2,3]. So, we see that (8), up to an unimportant change of sign of u, is indeed the dual problem associated to the primal problem (17); they share the same optimal value, which is $\mathrm{TV}_\mathrm{p}(x)$. ■

In the following, given an image x, we denote by $v_\updownarrow$, $v_\leftrightarrow$, and $v_\bullet$, the vector fields solution to (17) (or any solution if it is not unique). We denote by v the vector field, which is the concatenation of $v_\updownarrow$, $v_\leftrightarrow$, and $v_\bullet$. So, for every $(n_1,n_2)\in\Omega$, its elements $v_\updownarrow[n_1,n_2]$, $v_\leftrightarrow[n_1,n_2]$, $v_\bullet[n_1,n_2]$ are vectors of $\mathbb{R}^2$, located at the positions $(n_1+\frac{1}{2}, n_2)$, $(n_1, n_2+\frac{1}{2})$, (n_1,n_2), respectively. Then we call v the *gradient field* of x. Thus, the proposed TV is the $\ell_{1,2}$ norm of the gradient field v associated to the image x, solution to (17) and defined on a grid three times more dense than the one of x. The mapping from x to its gradient field v is nonlinear and implicit: given x, one has to solve the optimization problem (17) to obtain its gradient field and the value $\mathrm{TV}_\mathrm{p}(x)$. We can notice that the feasible set in (17) is nonempty, since the constraint is satisfied by the vector field defined by

$$v_{\updownarrow,1} = (Dx)_1, \quad v_{\updownarrow,2} = 0 \tag{20}$$

$$v_{\leftrightarrow,1} = 0, \quad v_{\leftrightarrow,2} = (Dx)_2 \tag{21}$$

$$v_{\bullet,1} = 0, \quad v_{\bullet,2} = 0. \tag{22}$$

This vector field has a $\ell_{1,2}$ norm equal to $\|(Dx)_1\|_1 + \|(Dx)_2\|_1$, which is exactly $\mathrm{TV}_\mathrm{a}(x)$, the value of the anisotropic TV of x. Therefore, we have the property: for every image x,

$$\mathrm{TV}_\mathrm{p}(x) \leq \mathrm{TV}_\mathrm{a}(x). \tag{23}$$

Further on, we have

$$\begin{aligned}(L_{\updownarrow}^* v_{\updownarrow} + L_{\leftrightarrow}^* v_{\leftrightarrow} + L_{\bullet}^* v_{\bullet})_1[n_1,n_2] = {}& v_{\updownarrow,1}[n_1,n_2] + (v_{\leftrightarrow,1}[n_1,n_2] + v_{\leftrightarrow,1}[n_1,n_2-1] + \\ & v_{\leftrightarrow,1}[n_1+1,n_2] + v_{\leftrightarrow,1}[n_1+1,n_2-1])/4 + \\ & (v_{\bullet,1}[n_1,n_2] + v_{\bullet,1}[n_1+1,n_2])/2,\end{aligned} \tag{24}$$

$$\begin{aligned}(L_{\updownarrow}^* v_{\updownarrow} + L_{\leftrightarrow}^* v_{\leftrightarrow} + L_{\bullet}^* v_{\bullet})_2[n_1,n_2] = {}& v_{\leftrightarrow,2}[n_1,n_2] + (v_{\updownarrow,2}[n_1,n_2] + v_{\updownarrow,2}[n_1,n_2+1] + \\ & v_{\updownarrow,2}[n_1-1,n_2] + v_{\updownarrow,2}[n_1-1,n_2+1])/4 + \\ & (v_{\bullet,2}[n_1,n_2] + v_{\bullet,2}[n_1,n_2+1])/2,\end{aligned} \tag{25}$$

using, again, zero boundary conditions. So, the quantity $(L_{\updownarrow}^* v_{\updownarrow} + L_{\leftrightarrow}^* v_{\leftrightarrow} + L_{\bullet}^* v_{\bullet})_1[n_1,n_2]$ is the sum of the vertical part of the elements of the vector field v falling into the square $[n_1,n_1+1] \times [n_2-\frac{1}{2}, n_2+\frac{1}{2}]$, weighted by $1/2$ if they are on an edge of the square, and by $1/4$ if they are at one of its corners. Similarly, $(L_{\updownarrow}^* v_{\updownarrow} + L_{\leftrightarrow}^* v_{\leftrightarrow} + L_{\bullet}^* v_{\bullet})_2[n_1,n_2]$ is the sum of the horizontal part of the elements of v falling into the square $[n_1-\frac{1}{2}, n_1+\frac{1}{2}] \times [n_2,n_2+1]$. Equating these two values to $(Dx)_1[n_1,n_2]$ and $(Dx)_2[n_1,n_2]$, respectively, is nothing but a discrete and 2-D version of the *fundamental theorem of calculus*, according to which the integral of a function on an interval is equal to the difference of its antiderivative at the interval bounds. So, we have defined the differentiation process from an image x to its gradient field v as the linear inverse problem of integration: integrating the gradient field v allows to recover the image x. Among all vector fields consistent with x in this sense, the gradient field v is selected as the simplest one, i.e. the one of minimal $\ell_{1,2}$ norm.

Let us be more precise about this integration property connecting v to x. We first note that it is incorrect to interpret the pixel value $x[n_1,n_2]$ as a point sample of an unknown function $s(t_1,t_2)$, i.e. $x[n_1,n_2] = s(n_1,n_2)$, and the values $v_{\updownarrow,1}[n_1,n_2]$, $v_{\leftrightarrow,1}[n_1,n_2]$, $v_{\bullet,1}[n_1,n_2]$ as point samples of $\partial s/\partial t_1$ at $(n_1+\frac{1}{2}, n_2)$, $(n_1, n_2+\frac{1}{2})$, (n_1,n_2), respectively. Indeed, if it were the case, and viewing (24) as a kind of extended trapezoidal rule for numerical integration, the right-hand side of (24) would be divided by three. Instead, one can view x as the cell-average discretization of an unknown function $s(t_1,t_2)$, i.e. $x[n_1,n_2] = \int_{n_1-1/2}^{n_1+1/2}\int_{n_2-1/2}^{n_2+1/2} s(t_1,t_2)\mathrm{d}t_1\mathrm{d}t_2$, and v as the gradient field of s, in a distributional sense. For this, let us define the 1-D box and hat functions

$$\Pi(t) = \begin{cases} 1 & \text{if } t \in (-\frac{1}{2}, \frac{1}{2}), \\ \frac{1}{2} & \text{if } t = \pm\frac{1}{2}, \\ 0 & \text{else} \end{cases}, \quad \Lambda(t) = \Pi(t) * \Pi(t) = \max(1-|t|, 0), \tag{26}$$

where $*$ denotes the convolution. We also define the 2-D box function $\Pi(t_1,t_2) = \Pi(t_1)\Pi(t_2)$ and the function $\psi(t_1,t_2) = \Lambda(t_1)\Pi(t_2)$. The function or distribution $\partial s/\partial t_1$ is such that

$$(Dx)_1[n_1,n_2] = x[n_1+1,n_2] - x[n_1,n_2] = (s*\Pi)(n_1+1,n_2) - (s*\Pi)(n_1,n_2) \tag{27}$$

$$= \int_{n_1}^{n_1+1} \Big(\frac{\partial s}{\partial t_1} * \Pi\Big)(t_1,n_2)\mathrm{d}t_1 \tag{28}$$

$$= \Big(\frac{\partial s}{\partial t_1} * \psi\Big)(n_1+\tfrac{1}{2}, n_2). \tag{29}$$

Then, the same equality holds, when replacing $\partial s/\partial t_1$ by the distribution

$$\widetilde{v}_1(t_1,t_2) = \sum_{(n_1,n_2)\in\Omega} v_{\updownarrow,1}[n_1,n_2]\delta(t_1-n_1-\tfrac{1}{2},t_2-n_2) + v_{\leftrightarrow,1}\delta(t_1-n_1,t_2-n_2-\tfrac{1}{2}) + v_{\bullet,1}[n_1,n_2]\delta(t_1-n_1,t_2-n_2), \tag{30}$$

where $\delta(t_1,t_2)$ is the 2-D Dirac distribution. Indeed,

$$\begin{aligned}(\widetilde{v}_1 * \psi)(n_1+\tfrac{1}{2},n_2) = v_{\updownarrow,1}[n_1,n_2] + (&v_{\leftrightarrow,1}[n_1,n_2] + v_{\leftrightarrow,1}[n_1,n_2-1] + \\ &v_{\leftrightarrow,1}[n_1+1,n_2] + v_{\leftrightarrow,1}[n_1+1,n_2-1])/4 + \\ &(v_{\bullet,1}[n_1,n_2] + v_{\bullet,1}[n_1+1,n_2])/2,\end{aligned} \tag{31}$$

which, according to (24), is equal to $(L_{\updownarrow}^* v_{\updownarrow} + L_{\leftrightarrow}^* v_{\leftrightarrow} + L_{\bullet}^* v_{\bullet})_1[n_1,n_2]$, which in turn is equal to $(Dx)_1[n_1,n_2]$, by definition of v in (17). Altogether, the scalar field v_1, the vertical component of the gradient field v, identified to the distribution $\widetilde{v}_1$, plays the same role as the partial derivative $\partial s/\partial t_1$ of s, in the sense that they both yield the pixel values of x by integration. The same relationship holds between v_2 and $\partial s/\partial t_2$. To summarize, v is the discrete counterpart of the gradient of the unknown continuously-defined scene s, whose cell-average discretization yields the image x. So, it is legitimate to call v the gradient field of x. Note that there exists no function s such that ∇s is the Dirac brush $(\widetilde{v}_1,\widetilde{v}_2)$, so v is no more than a discrete equivalent of ∇s.

We can notice that, given the image x, the gradient field v solution to (17) is not always unique. For instance, for the 1-D signal $x = (0,0,1/2,1,1)$, viewed as an image with only one row, one can set $v_{\updownarrow} = 0$, $v_{\leftrightarrow} = 0$, $v_{\bullet} = (0,0,1,0,0)$. Another possibility is to take $v_{\updownarrow} = 0$, $v_{\bullet} = 0$, $v_{\leftrightarrow} = (0,1/2,1/2,0,0)$. This possible nonuniqueness of v, which is very rare in practice, does not have any impact on the images obtained by TV minimization. We leave the study of theoretical aspects of the proposed TV and gradient field for future work, like showing Gamma-convergence of the proposed TV.

We end this section with a remark about the fact that the grid for the gradient field is twice finer than the one of the image. This factor of two appears naturally, according to the following sampling-theoretic argument. Let us consider a 2-D sine function $s(t_1,t_2) = \sin(at_1+bt_2+c)$, for some a, b, c in $(-\pi,\pi)$, which is sampled to give the image x, with $x[n_1,n_2] = s(n_1,n_2)$. We have $|\nabla s(t_1,t_2)|^2 = (a^2+b^2)\cos^2(at_1+bt_2+c) = (a^2+b^2)\cos(2at_1+2bt_2+2c)/2+(a^2+b^2)/2$. So, by taking the squared amplitude of the gradient, the frequency of the sine is doubled. According to Shannon's theorem, the function $|\nabla s|^2$ must be sampled on a grid twice finer than the one of x, for its information content to be kept. Since, by virtue of the Fourier transform, every sufficiently regular function can be decomposed in terms of sines, this argument applies to an arbitrary 2-D function s, not only to a sine. The picture does not change by applying the square root, passing from $|\nabla s|^2$ to $|\nabla s|$, the integral of which is the TV of s. Thus, as long as the amplitude of the gradient is the information of interest, it must be represented on a twice finer grid; else aliasing occurs and the value of the TV becomes unreliable.

5. Algorithms for TV minimization.

In this section, we focus on the generic convex optimization problem:

$$\text{Find } \hat{x} \in \operatorname*{arg\,min}_{x\in\mathbb{R}^{N_1\times N_2}} \left\{F(x) + \lambda\,\mathrm{TV}(x)\right\}, \tag{32}$$

where the sought-after image $\hat{x}$ has size $N_1 \times N_2$, $\lambda > 0$ is the regularization parameter, F is a convex, proper, lower semicontinuous function [31]. A particular instance of this problem is image denoising or smoothing: given the image y, one solves:

$$\text{Find } \hat{x} \in \operatorname*{arg\,min}_{x \in \mathbb{R}^{N_1 \times N_2}} \left\{ \tfrac{1}{2}\|x - y\|^2 + \lambda\, \mathrm{TV}(x) \right\}, \tag{33}$$

where the norm is the Euclidean norm. This problem is a particular case of (32) with $F(x) = \frac{1}{2}\|x - y\|^2$. More generally, many inverse problems in imaging can be written as: given the data y and the linear operator A,

$$\text{Find } \hat{x} \in \operatorname*{arg\,min}_{x \in \mathbb{R}^{N_1 \times N_2}} \left\{ \tfrac{1}{2}\|Ax - y\|^2 + \lambda\, \mathrm{TV}(x) \right\}. \tag{34}$$

Again, this problem is a particular case of (32) with $F(x) = \frac{1}{2}\|Ax - y\|^2$. Another instance is TV minimization subject to a linear constraint, for instance to regularize an ill-posed inverse problem in absence of noise: given the data y and the linear operator A, one solves:

$$\text{Find } \hat{x} \in \operatorname*{arg\,min}_{x \in \mathbb{R}^{N_1 \times N_2}} \{ \mathrm{TV} \;:\; Ax = y \}. \tag{35}$$

This problem is a particular case of (32) with $\lambda = 1$ and $F(x) = \imath_{\{x \,:\, Ax = y\}}(x)$, where the convex indicator function $\imath_\Gamma$ of a set Γ maps its variable x to 0 if $x \in \Gamma$, to $+\infty$ else.

When the TV is the anisotropic, isotropic, or upwind TV, which is a simple function composed with the finite differentiation operator D, there are efficient primal–dual algorithms to solve a large class of problems of the form (32), see e.g. [3, 17, 18] and references therein. In section 6, we use the overrelaxed version [32] of the Chambolle–Pock algorithm [3]. With the proposed TV, it is not straightforward to apply these algorithms. In fact, (32) can be rewritten as:

$$\text{Find } (\hat{x}, \hat{v}) \in \operatorname*{arg\,min}_{x \in \mathbb{R}^{N_1 \times N_2},\, v \in \left((\mathbb{R}^2)^{N_1 \times N_2}\right)^3} \{ F(x) + \lambda \|v\|_{1,1,2} \;:\; L^* v = Dx \}. \tag{36}$$

So, one has to find not only the image $\hat{x}$, but also its gradient field $\hat{v}$, minimizing a separable function, under a linear coupling constraint. Let us introduce the function $G(v) = \lambda \|v\|_{1,1,2}$ and the linear operator $C = -L^*$, so that we can put (36) under the standard form:

$$\text{Find } (\hat{x}, \hat{v}) \in \operatorname*{arg\,min}_{x \in \mathbb{R}^{N_1 \times N_2},\, v \in \left((\mathbb{R}^2)^{N_1 \times N_2}\right)^3} \{ F(x) + G(v) \;:\; Cv + Dx = 0 \}. \tag{37}$$

The dual problem is

$$\text{Find } \hat{u} \in \operatorname*{arg\,min}_{x \in u \in (\mathbb{R}^2)^{N_1 \times N_2}} \{ F^*(-D^* u) + G^*(-C^* u) \}, \tag{38}$$

which, in our case, is

$$\text{Find } \hat{u} \in \operatorname*{arg\,min}_{x \in u \in (\mathbb{R}^2)^{N_1 \times N_2}} \{ F^*(-D^* u) \;:\; \|Lu\|_{\infty,\infty,2} \le \lambda \}. \tag{39}$$

We now assume that the function F is simple, in the sense that it is easy to apply the proximity operator [31,33] $\mathrm{prox}_{\alpha F}$ of αF, for any parameter $\alpha > 0$. For the denoising problem (33), $\mathrm{prox}_{\alpha F}(x) = (x+\alpha y)/(1+\alpha)$. For the regularized least-squares problem (34), $\mathrm{prox}_{\alpha F}(x) = (\mathrm{Id} + \alpha A^* A)^{-1}(x + \alpha A^* y)$. For the constrained problem (35), $\mathrm{prox}_{\alpha F}(x) = x + A^\dagger (y - Ax)$, where $A^\dagger$ is the Moore-Penrose pseudo-inverse of A. We also need the proximity operator of $\alpha G = \alpha\lambda \|\cdot\|_{1,1,2}$, which is

(40)
$$\big(\mathrm{prox}_{\alpha G}(v)\big)_c[n_1,n_2] = v_c[n_1,n_2] - \frac{v_c[n_1,n_2]}{\max(|v_c[n_1,n_2]|/(\alpha\lambda),1)}, \quad \forall (n_1,n_2)\in\Omega,\ \forall c \in \{\updownarrow, \leftrightarrow, \bullet\}.$$

We can notice that $\|D\|^2 \leq 8$ [2] and $\|C\|^2 = \|L\|^2 \leq 3$. So, we have all the ingredients to use the Alternating Proximal Gradient Method [34], a particular case of the Generalized Alternating Direction Method of Multipliers [35]:

Algorithm 1 to solve (36)

Choose the parameters $0 < \tau < 1/\|D\|^2$, $0 < \gamma < 1/\|C\|^2$, $\mu > 0$, and the initial estimates $x^{(0)}$, $v^{(0)}$, $u^{(0)}$.
Then iterate, for $i = 0, 1, \ldots$

$$\left\lfloor \begin{array}{l} x^{(i+1)} := \mathrm{prox}_{\tau\mu F}\Big(x^{(i)} - \tau D^*(Dx^{(i)} + Cv^{(i)} + \mu u^{(i)})\Big), \\ v^{(i+1)} := \mathrm{prox}_{\gamma\mu G}\Big(v^{(i)} - \gamma C^*(Dx^{(i+1)} + Cv^{(i)} + \mu u^{(i)})\Big), \\ u^{(i+1)} := u^{(i)} + (Dx^{(i+1)} + Cv^{(i+1)})/\mu. \end{array}\right.$$

Assuming that there exists a solution to (36), for which a sufficient condition is that there exists a minimizer of F, Algorithm 1 is proved to converge [34,35]: the variables $x^{(i)}$, $v^{(i)}$, $u^{(i)}$ converge respectively to some $\hat{x}$, $\hat{v}$, $\hat{u}$, solution to (36) and (39).

It is easy to show that the same algorithm can be used to compute the gradient field v of an image x, solution to (17); we simply replace $x^{(i)}$ by x. This yields

Algorithm 2 to find v solution to (17), given x

Choose the parameters $0 < \gamma < 1/\|C\|^2$, $\mu > 0$, and the initial estimates $v^{(0)}$, $u^{(0)}$.
Then iterate, for $i = 0, 1, \ldots$

$$\left\lfloor \begin{array}{l} v^{(i+1)} := \mathrm{prox}_{\gamma\mu G}\Big(v^{(i)} - \gamma C^*(Dx + Cv^{(i)} + \mu u^{(i)})\Big), \\ u^{(i+1)} := u^{(i)} + (Dx + Cv^{(i+1)})/\mu. \end{array}\right.$$

In practice, we recommend setting $\tau = 0.99/8$ and $\gamma = 0.99/3$ in Algorithm 1 and Algorithm 2, so that there only remains to tune the parameter μ.

Further on, let us consider the regularized least-squares problem (34), in the case where the proximity operator of the quadratic term cannot be computed. It is possible to modify Algorithm 1, by changing the metric in the Generalized Alternating Direction Method of Multipliers [35], to obtain a fully split algorithm, which only applies A and A^* at every iteration, without having to solve any linear system. So, we consider the more general problem

$$\text{Find } \hat{x} \in \underset{x\in\mathbb{R}^{N_1\times N_2}}{\arg\min} \left\{F(x) + \tfrac{1}{2}\|Ax-y\|^2 + \lambda\,\mathrm{TV}_\mathrm{p}(x)\right\}, \tag{41}$$

or, equivalently,

$$\text{Find } (\hat{x},\hat{v}) \in \underset{x\in\mathbb{R}^{N_1\times N_2},\, v\in\left((\mathbb{R}^2)^{N_1\times N_2}\right)^3}{\arg\min} \left\{F(x) + \tfrac{1}{2}\|Ax-y\|^2 + G(v) \;:\; Cv + Dx = 0\right\}, \tag{42}$$

where, again, $G(v) = \lambda\,\|v\|_{1,1,2}$ and $C = -L^*$. The algorithm, with proved convergence to exact solutions of (41) and its dual, is:

Algorithm 3 to solve (42)

Choose the parameters $\tau > 0$, $\mu > 0$, such that $\tau < 1/(\|D\|^2 + \mu\|A\|^2)$, $0 < \gamma < 1/\|C\|^2$, and the initial estimates $x^{(0)}$, $v^{(0)}$, $u^{(0)}$.
Then iterate, for $i = 0, 1, \ldots$

$x^{(i+1)} := \mathrm{prox}_{\tau\mu F}\Big(x^{(i)} - \tau D^*(Dx^{(i)} + Cv^{(i)} + \mu u^{(i)}) - \tau\mu A^*(Ax^{(i)} - y)\Big)$,
$v^{(i+1)} := \mathrm{prox}_{\gamma\mu G}\Big(v^{(i)} - \gamma C^*(Dx^{(i+1)} + Cv^{(i)} + \mu u^{(i)})\Big)$,
$u^{(i+1)} := u^{(i)} + (Dx^{(i+1)} + Cv^{(i+1)})/\mu$.

The proposed TV, like the other forms, could be used as a constraint, instead of being used as a functional to minimize [36, 37].

Many other algorithms could be applied to solve problems involving the proposed TV. The most appropriate algorithm for a particular problem must be designed on a case-by-case basis. So, it is beyond the scope of this paper to do any comparison of algorithms in terms of convergence speed.

6. Experiments. In this section, we evaluate the proposed TV on several test problems. First, we report in Figure 1 the value of the proposed TV for the patterns shown in Figure 1. For each image, the value was determined by computing the associated gradient field, using Algorithm 2; these gradient fields are depicted in Figure 3. According to the discussion in section 2 and section 4, the proposed TV, which is a seminorm, takes appropriate values in all cases. We observe that, for binary patterns, the gradient field, and thus the value of the TV, is the same as with the anisotropic TV; that is, it is given by Eqs. (20)–(22). Thus, the staircased nature of oblique binary patterns is penalized.

In the remainder of this section, we study the behavior of the proposed TV in several applications, based on TV minimization. Matlab code implementing the corresponding optimization algorithms and generating the images in Figure 3 to Figure 12 is available on the author's webpage.

6.1. Smoothing of a binary edge. We consider the smoothing problem (33) with the proposed TV, where the initial image y ($N_1 = N_2 = 256$), is an oblique binary edge, obtained by point sampling a continuously defined straight edge with slope 5/16. The central part of y

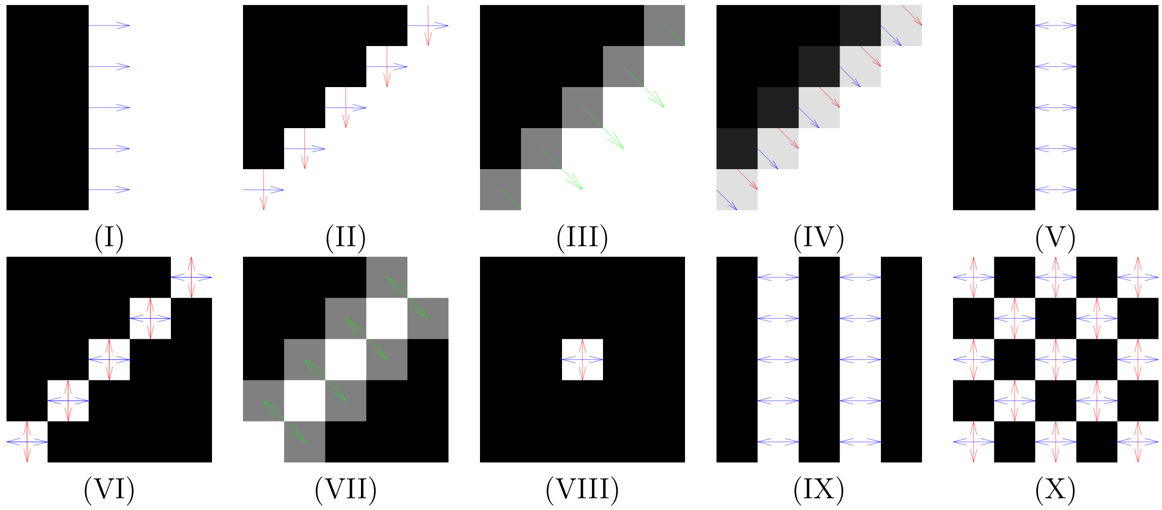


Figure 3. *Same patterns as in Figure 1, with the associated gradient fields, solutions to (17). The vectors $v_{\updownarrow}[n_1,n_2]$, $v_{\leftrightarrow}[n_1,n_2]$, $v_{\bullet}[n_1,n_2]$, are represented by red, blue, green arrows, starting at $(n_1+\frac{1}{2},n_2)$, $(n_1,n_2+\frac{1}{2})$, (n_1,n_2), respectively.*

is depicted in Figure 7 (a). So, we solve

$$\text{(43)} \qquad \text{Find } (\hat{x},\hat{v}) \in \mathop{\arg\min}_{x\in\mathbb{R}^{N_1\times N_2},v\in\left((\mathbb{R}^2)^{N_1\times N_2}\right)^3} \left\{\tfrac{1}{2}\|x-y\|^2+\lambda\,\|v\|_{1,1,2}\ :\ L^*v=Dx\right\},$$

using Algorithm 1 ($\mu = 0.05$, 2000 iterations). The central part of the smoothed image $\hat{x}$, as well as the corresponding gradient field $\hat{v}$, are depicted in Figure 7 (b), for $\lambda = 2$; see the caption of Figure 3 for the representation of the gradient field by colored arrows. The result for stronger smoothing with $\lambda = 20$ is depicted in Figure 7 (c).

We observe that the edge undergoes a slight blur, which remains concentrated over one or two pixels vertically, even for a strong smoothing parameter λ. This is expected, since such a slightly blurred edge has a lower TV value than the binary edge in y. Importantly, the minimization of the proposed TV tends to make all the gradient vectors of the field $\hat{v}$ aligned with the same orientation, which is exactly perpendicular to the underlying edge with slope 5/16. This shows that not only the amplitude, but also the orientation of the gradient vectors obtained with the proposed approach, is meaningful.

6.2. Smoothing of a disk. We consider the smoothing problem (33), with $\lambda = 6$, where y is the image of a white disk, of radius 32, over a black background ($N_1 = N_2 = 99$), depicted in Figure 8 (a). To simulate cell-average discretization, a larger $(16N_1)\times(16N_2)$ binary image was constructed by point sampling a 16 times larger disk, and then y was obtained by averaging over the 16×16 blocks of this image. In the continuous domain, it is known [38] that TV smoothing of a disk of radius R and amplitude one over a zero background, with zero/Dirichlet boundary conditions, gives the same disk, with lower amplitude $1-2\lambda/R$, assuming $\lambda < R/2$. Here, we consider a square domain of size $N_1\times N_2$ with symmetric/Neumann boundary conditions, so the background is expected to become lighter after smoothing, with amplitude

$2\pi\lambda R/(N_1 N_2 - \pi R^2)$. We can notice that the total intensity remains unchanged and equal to πR^2 after smoothing. Moreover, according to the coarea formula, the TV of the image of a disk is $2\pi R$, the perimeter of the disk, multiplied by the difference of amplitude between the disk and the background. Thus, in the discrete domain, we expect the smoothed image $\hat{x}$ to be similar to y, after an affine transform on the pixel values, so that the pixel values in the interior of the disk and in the background are $1 - 2\lambda/R = 0.625$ and $2\pi\lambda R/(N_1 N_2 - \pi R^2) \approx 0.183$, respectively; this reference image is depicted in Figure 8 (b).

The images $\hat{x}$ obtained by solving (33) with the anisotropic, isotropic, upwind, and proposed TV (using 2000 iterations of Algorithm 1 with $\mu = 0.1$), are shown in Figure 8.

– With the anisotropic TV the perimeter of the disk is evaluated in the sense of the Manhattan distance, and not the Euclidean distance. So, the TV of the disk is over-estimated. Since blurring an edge does not decrease the TV, TV minimization lets the TV value decrease by shrinking the shape of the disk and attenuating the amplitude of the edge more than it should.

– With the isotropic TV, the bottom, right, and top-left parts of the edge are sharp, but the other parts are significantly blurred. Contrary to the three other forms, the isotropic TV does not yield a symmetric image; the image is only symmetric with respect to the diagonal at -45°.

– The upwind TV performs relatively well.

– The proposed TV outperforms the three other forms. Except at the top, bottom, left, right ends, the edge is sharper than with the upwind TV. The edge has the same spread everywhere, independently of the local orientation, which is a clear sign of the superior isotropy of the proposed approach. Since the proposed TV does not blur a horizontal or vertical edge after smoothing, the fact that the top, bottom, left, right ends of the disk edge are blurred here shows the truly nonlocal nature of the proposed TV; this is due to the higher number of degrees of freedom optimized during TV minimization, with not only the image but also its three gradient subfields. The other forms of the TV have less flexibility, with the gradient fully determined by local finite differences on the image.

The gradient field $\hat{v}$, solution to (43), is depicted in Figure 4. We can observe its quality, with all the arrows pointing towards the disk center, showing that the gradient orientation is perpendicular to the underlying circular edge everywhere.

6.3. Smoothing of a square. We consider the smoothing problem (33), with $\lambda = 6$, where y is the image of a white square, of size 64×64, over a black background ($N_1 = N_2 = 100$), depicted in Figure 9 (a). In the continuous domain, the solution of the smoothing problem, when the function y is equal to 1 inside the square $[-1, 1]^2$ and 0 outside, $\lambda < 1/(1 + \sqrt{\pi}/2)$, and with zero boundary conditions, contains a square of same size, but with rounded and blurred corners, and lower amplitude [39, 40]. The following closed-form expression can be derived:

$$x(t_1, t_2) = \begin{cases} 0 & \text{if } |t_1| > 1 \text{ or } |t_2| > 1, \\ 0 & \text{else, if } r \leq \lambda, \\ 1 - \lambda(1 + \sqrt{\pi}/2) & \text{else, if } r \geq 1/(1 + \sqrt{\pi}/2), \\ 1 - \lambda/r & \text{else,} \end{cases} \tag{44}$$

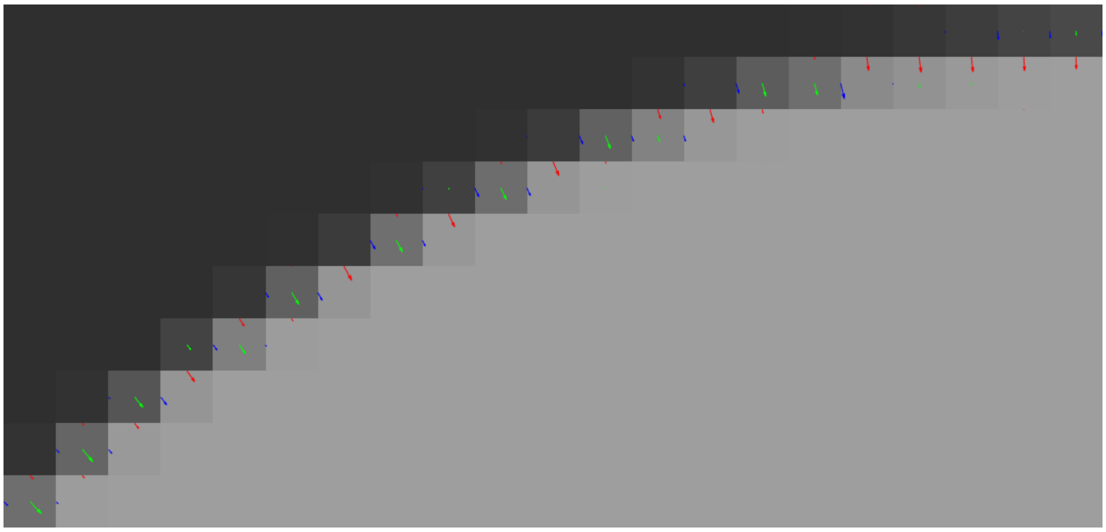
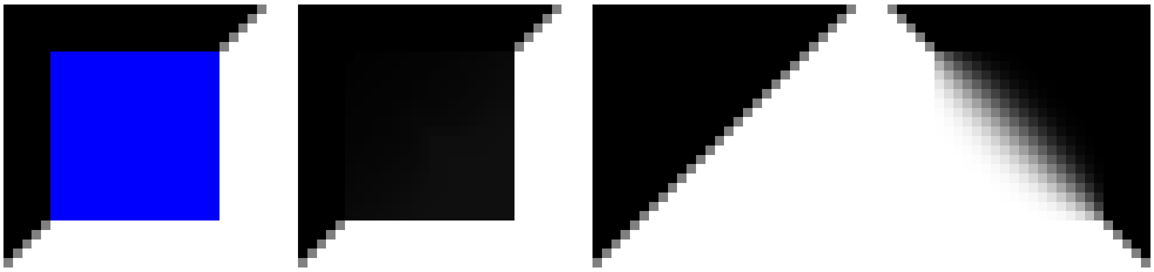


Figure 4. *Zoom on the top-left part of the disk edge in Figure 8 (f), with the associated gradient field.*

where $r = 2 - |t_1| - |t_2| + \sqrt{2(1-|t_1|)(1-|t_2|)}$. Since symmetric, instead of zero, boundary conditions are considered here, $x(t_1, t_2)$ is actually the maximum of this expression and a constant, which can be calculated. So, the reference result in the discrete case was simulated by point sampling this function $x(t_1, t_2)$ on a fine grid, with $\lambda = 6/32$, in a large 1600×1600 image, which was then reduced by averaging over its 16×16 blocks. This reference image is depicted in Figure 9 (b).

The image $\hat{x}$, solution to (33) with the anisotropic, isotropic, upwind, and proposed TV (using 2000 iterations of Algorithm 1 with $\mu = 0.3$), is shown in Figure 9. The anisotropic TV yields a square, without any rounding of the corners. This shows again that the metric underlying anisotropic TV minimization is not the Euclidean one. With the isotropic TV, the asymmetric blur of the corners contaminates the top and left sides of the square. Only the top-left corner has the correct aspect. With the upwind TV, the level lines at the corners are more straight than circular. The proposed TV yields the closest image to the reference image.

6.4. Denoising of the Bike. We consider the application of the smoothing/denoising problem (33), or (43) with the proposed TV, to remove noise in a natural image. The initial image y, depicted in Figure 10 (a), is a part of the classical *Bike* image, depicted in Figure 10 (b), corrupted by additive white Gaussian noise of standard deviation 0.18. λ is set to 0.16. With the anisotropic TV, the noise is removed, but the contrast of the spokes is more attenuated than with the other forms of the TV. With the isotropic TV, the noise is less attenuated and some small clusters of noise remain. This is also the case, to a much larger extent, with the upwind TV: the dark part of the noise is removed, but not the light part, and a lot of small light clusters of noise remain. This drawback of the isotropic and upwind TV can be explained by the too low penalization of a single isolated pixel, as reported in Table 1 and in section 2. The proposed TV (using 1000 iterations of Algorithm 1 with $\mu = 1$) yields the best result: the noise is removed, the spokes have an elongated shape with less artifacts and a good contrast.



(a) (b) (c) (d)

Figure 5. *Inpainting experiment, see subsection 6.5. The region to reconstruct is in blue in (a). In (b), one solution of anisotropic TV minimization. In (c), solution of isotropic, upwind, and proposed TV minimization. In (d), solution of isotropic TV minimization, for the flipped case.*

6.5. Inpainting of an edge. We consider an inpainting problem, which consists in reconstructing missing pixels by TV minimization. The image is shown in Figure 5 (a), with the missing pixels in blue. We solve the constrained TV minimization problem (35), where A is a masking operator, which sets to zero the pixel values in the missing region and keeps the other pixels values unchanged. We have $A^\dagger = A^* = A$. The image y, shown in Figure 5 (b), has its pixel values in the missing region equal to zero.

With the anisotropic TV, the solution is not unique, and every image with nondecreasing pixels values horizontally and vertically is a solution of the TV minimization problem. One solution, equal to y, is shown in Figure 5 (b). The result with the isotropic, upwind, and proposed TV (using 1000 iterations of Algorithm 1, with $\mu = 1$) is the same, and corresponds to what is expected; it is shown in Figure 5 (c). The gradient field $\hat{v}$ associated to the solution with the proposed TV is not shown, but it is the same as in Figure 3 (III).

We also consider the flipped case, where y is flipped horizontally. The solution with the isotropic TV is shown in Figure 5 (d). It suffers from a strong blur. Indeed, as reported in Table 1, the value of the isotropic TV for slightly blurred edges at this orientation, like in the cases (IIIf) and (IVf), is too high. So, when minimizing the TV, the TV value is decreased by the introduction of an important blur. By contrast, the anisotropic, upwind, and proposed TV are symmetric, so they yield flipped versions of the images shown in Figure 5 (b) and (c).

6.6. Upscaling of a disk. We consider the upscaling problem, which consists in increasing the resolution of the image y of a disk, shown in Figure 11 (a), by a factor of 4 in both directions. Upscaling is viewed as the inverse problem of downscaling: the downscaling operator A maps an image to the image of its averages over 4×4 blocks, and we suppose that $y = Ax^\sharp$, for some reference image $x^\sharp$, that we want to estimate. Here, y is of size 23×23 and the reference image $x^\sharp$, shown in Figure 11 (b), of size 92×92, was constructed like in subsection 6.2: to approximate cell-average discretization, a larger 1472×1472 image x^0 was constructed by point sampling a 16 times larger disk, and $x^\sharp$ was obtained by averaging over the 16×16 blocks of this image; that is, $x^\sharp = AAx^0$. Then y was obtained as $y = Ax^\sharp$. Hence, the upscaled image is defined as the solution to the constrained TV minimization problem (35). We have $A^\dagger = 16A^*$.

The results with the anisotropic, isotropic, upwind, and proposed TV (using 2000 iterations

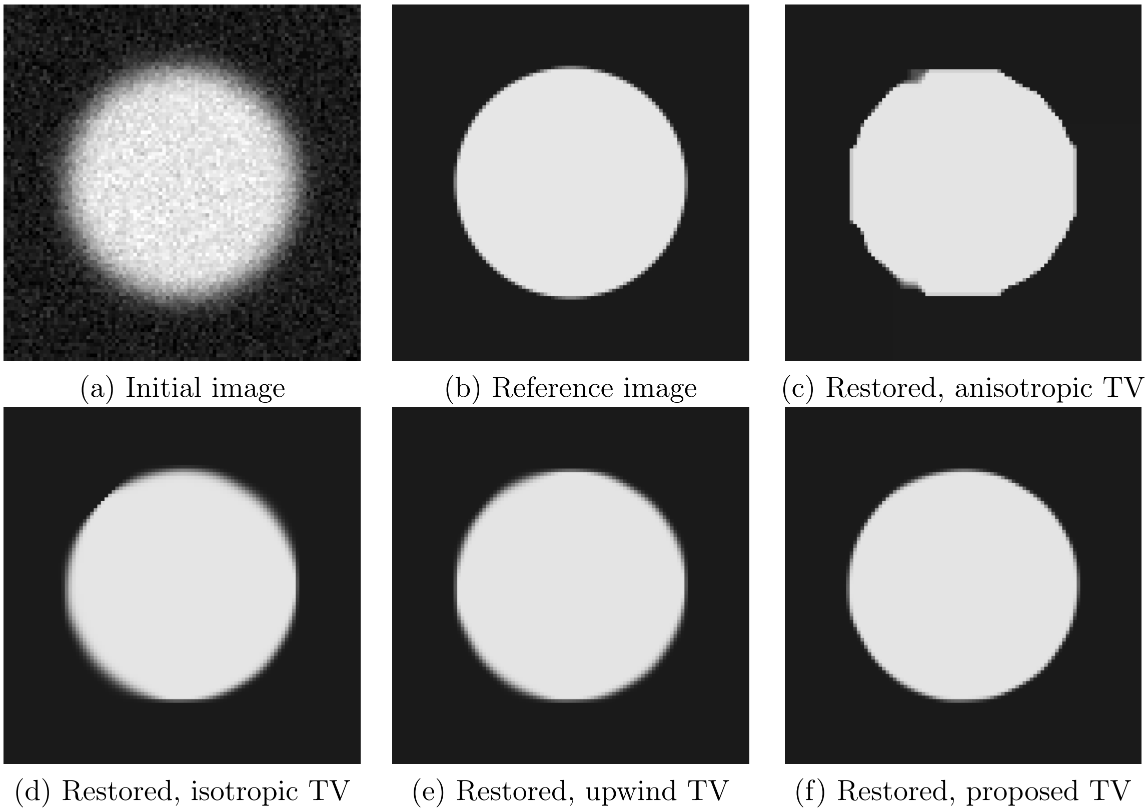


Figure 6. *Deconvolution experiment, see subsection 6.7. The initial image in (a) is obtained from the ground-truth image in (b), by convolution with a Gaussian filter and addition of white Gaussian noise.*

of Algorithm 1, with $\mu = 1$) are shown in Figure 11 (c)–(f). With the anisotropic TV, the result is very blocky. With the isotropic TV, the disk edge is jagged, except at the top-left and bottom-right ends. The result is much better with the upwind TV, and even better with the proposed TV, which has the most regular disk edge. The distance $\|\hat{x} - x^\sharp\|$ between the upscaled image and the reference image is 2.91, 1.59, 1.23, with the isotropic, upwind, proposed TV, respectively. So, this error is 23% lower with the proposed TV than with the upwind TV.

6.7. Deconvolution of a disk. We consider the deconvolution, a.k.a. deblurring, problem, which consists in estimating an image $x^\sharp$, given its blurred and noisy version y. $x^\sharp$ is the image of a disk, constructed like in subsection 6.2 and shown in Figure 6 (b). The initial image y, depicted in Figure 6 (a), was obtained by applying a Gaussian filter, of standard deviation (spread) 3.54 pixels, to $x^\sharp$ and adding white Gaussian noise of standard deviation 0.05. The image was restored by solving, for the proposed TV, (42) with $F = 0$ and, for the other TV forms, (34). In all cases, $A = A^*$ is convolution with the Gaussian filter, with symmetric boundary conditions, and λ is set to 0.1. Algorithm 3 was used in all cases, for simplicity, with $C = -\mathrm{Id}$ for all but the proposed TV. The distance $\|\hat{x} - x^\sharp\|$ between the restored image and the reference image is 4.10, 3.56, 2.78, 1.46, with the anisotropic, isotropic, upwind, proposed TV, respectively. We observe in Figure 6 (c)–(f) that the noise is well removed in all cases.

Again, the proposed TV provides the roundest and least blurred edge of the disk.

6.8. Segmentation of the Parrot. We consider a convex approach to color image segmentation. Given the set $\Sigma = \{c_k \in [0,1]^3 \ : \ k = 1,\dots,K\}$ of $K \geq 2$ colors c_k, expressed as triplets of R,G,B values, and the color image $y \in (\mathbb{R}^3)^{N_1 \times N_2}$, we would like to find the segmented image

$$\hat{x} = \underset{x \in \Sigma^{N_1 \times N_2}}{\arg\min} \left\{ \tfrac{1}{2}\|x - y\|^2 + \tfrac{\lambda}{2} \sum_{k=1}^{K} \mathrm{per}(\Omega_k) \right\}, \tag{45}$$

for some $\lambda > 0$, where $\Omega_k = \{(n_1, n_2) \in \Omega \ : \ x[n_1, n_2] = c_k\}$ and per denotes the perimeter. That is, we want a color image, whose color at every pixel is one of the c_k, close to y, but at the same time having homogeneous regions. However, this nonconvex "Potts" problem is very difficult, and even NP-hard [6]. And a rigorous definition of the perimeter of a discrete region is a difficulty in itself. So, we consider a convex relaxation of this problem [6]: we look for the object $\hat{z} \in \Delta^{N_1 \times N_2}$, such that, at every pixel, $\hat{z}[n_1, n_2] = (\hat{z}_k[n_1, n_2])_{k=1}^K$ is an assignment vector in the simplex $\Delta = \{(a_k)_{k=1}^K \ : \ \sum_{k=1}^K a_k = 1 \text{ and } a_k \geq 0, \forall k\}$. The elements $\hat{z}_k[n_1, n_2] \in [0,1]$ are the proportions of the colors c_k at pixel (n_1, n_2); that is, the segmented image $\hat{x}$ is obtained from $\hat{z}$ as

$$\hat{x}[n_1, n_2] = \sum_{k=1}^{K} \hat{z}_k[n_1, n_2] c_k, \quad \forall (n_1, n_2) \in \Omega. \tag{46}$$

Now, by virtue of the coarea formula, the segmentation problem can be reformulated as [6]

$$\text{Find } \hat{z} = \underset{z \in \Delta^{N_1 \times N_2}}{\arg\min} \left\{ \langle z, p \rangle + \lambda \sum_{k=1}^{K} \mathrm{TV}(z_k) \right\}, \tag{47}$$

where the Euclidean inner product is

$$\langle z, p \rangle = \sum_{(n_1, n_2) \in \Omega} \sum_{k=1}^{K} z_k[n_1, n_2] p_k[n_1, n_2], \tag{48}$$

$$\text{with} \ \ p_k[n_1, n_2] = \|y[n_1, n_2] - c_k\|^2. \tag{49}$$

The problem (47) can be put under a form similar to (32):

$$\text{Find } \hat{z} \in \underset{z \in (\mathbb{R}^K)^{N_1 \times N_2}}{\arg\min} \big\{ \mathbf{F}(z) + \lambda \, \mathbf{TV}(z) \big\}, \tag{50}$$

with the TV of z having a separable form with respect to k, i.e. $\mathbf{TV}(z) = \sum_{k=1}^K \mathrm{TV}(z_k)$, and $\mathbf{F}(z)$ having a separable form with respect to the pixels, i.e. $\mathbf{F}(z) = \sum_{(n_1, n_2) \in \Omega} F_{n_1, n_2}(z[n_1, n_2])$, where

$$F_{n_1, n_2}(a) = \imath_\Delta(a) + \langle a, p[n_1, n_2] \rangle. \tag{51}$$

For any $\alpha > 0$, we have $\mathrm{prox}_{\alpha F_{n_1,n_2}}(a) = P_\Delta(a - \alpha p[n_1, n_2])$, where P_Δ is the projection onto the simplex, which can be computed efficiently [41]. So, the primal–dual algorithms described in section 5 can be used for the segmentation problem, as well. With the proposed TV, we must introduce K gradient fields v_k, associated to the images z_k. We used 1000 iterations of Algorithm 1, with $\mu = 50$.

We compare the performances of the anisotropic, isotropic, upwind, proposed TV on this problem, with y a part, of size 399×400, of the classical *Parrot* image, shown in Figure 12 (a). We set $\lambda = 0.09$ and we set the $K = 6$ colors as some kind of black, white, yellow, blue, green, brown, visible in Figure 12 (b)–(e). In this respect, we would like the edges, which are the interfaces between the regions Ω_k, to be sharp, and their perimeter to be correctly measured by the TV of the assignment images $\hat{z}_k$. But these two goals are antagonist: the coarea formula is not well satisfied for discrete binary shapes, as we have seen in section 2: the length of oblique binary edges is overestimated by the anisotropic, isotropic, and proposed TV, and the length of small structures, like in the extreme case of a single isolated pixel, is underestimated by the upwind TV. This seems like an intrinsic limitation and the price to pay for convexity, in a spatially discrete setting. As visible in Figure 12 (b), the anisotropic TV yields sharp edges, but their length is measured with the Manhattan distance, not the Euclidean one. So, the edges tend to be vertical and horizontal. With the isotropic TV, for half of the orientations, the edges are significantly blurred, as is visible on the dark region over a green background, in the bottom-left part of the image in Figure 12 (c). The upwind TV tends to introduce more regions made of a few pixels, because their perimeter is underestimated, see the eye of the parrot in Figure 12 (d). The best tradeoff is obtained with the proposed TV: there is a slight, one or two pixel wide blur at the edges, but this blur cannot be avoided, for the perimeter of the regions to be correctly evaluated.

7. Conclusion.

We proposed a new formulation for the discrete total variation (TV) semi-norm of an image. Indeed, the classical, so-called isotropic, TV suffers from a poor behavior on oblique structures, for half of the possible orientations. It is important to have a sound definition of the TV, not least to be able to compare different convex regularizers for imaging problems, based on their intrinsic variational and geometrical properties, and not on the quality of their implementation.

Our new definition of the gradient field of an image has potential applications going far beyond TV minimization; for instance, one can consider edge detection based on the gradient amplitude, nonlinear diffusion and PDE flows based on the gradient orientation, one can define higher order differential quantities… We will explore some of these problematics in future work. The extension of the proposed TV to color or multichannel images will be investigated, as well.

REFERENCES

[1] L. Rudin, S. Osher, and E. Fatemi, *Nonlinear total variation based noise removal algorithms*, Phys. D, 60 (1992), pp. 259–268.

[2] A. Chambolle, V. Caselles, D. Cremers, M. Novaga, and T. Pock, *An introduction to total variation for image analysis*, in Theoretical Foundations and Numerical Methods for Sparse Recovery, vol. 9, De Gruyter, Radon Series Comp. Appl. Math., 2010, pp. 263–340.

[3] A. Chambolle and T. Pock, *A first-order primal-dual algorithm for convex problems with applications to imaging*, J. Math. Imaging Vision, 40 (2011), pp. 120–145.

[4] M. Burger and S. Osher, *A guide to the TV zoo*, in Level Set and PDE Based Reconstruction Methods in Imaging, Springer, 2013, pp. 1–70.

[5] T. Goldstein, X. Bresson, and S. Osher, *Geometric applications of the split Bregman method: Segmentation and surface reconstruction*, J. Sci. Comput., 45 (2010), pp. 272–293.

[6] A. Chambolle, D. Cremers, and T. Pock, *A convex approach to minimal partitions*, SIAM J. Imaging Sci., 5 (2012), pp. 1113–1158.

[7] C. Couprie, L. Grady, L. Najman, J.-C. Pesquet, and H. Talbot, *Dual constrained TV-based regularization on graphs*, SIAM J. Imaging Sci., 6 (2013), pp. 1246–1273.

[8] T. Goldstein and S. Osher, *The split Bregman method for L1-regularized problems*, SIAM J. Imaging Sci., 2 (2009), pp. 323–343.

[9] M. K. Ng, P. Weiss, and X. Yuan, *Solving constrained total-variation image restoration and reconstruction problems via alternating direction methods*, SIAM J. Sci. Comput., 32 (2010), pp. 2710–2736.

[10] M. Afonso, J. Bioucas-Dias, and M. Figueiredo, *Fast image recovery using variable splitting and constrained optimization*, IEEE Trans. Signal Processing, 19 (2010), pp. 2345–2356.

[11] P. L. Combettes, D. Dũng, and B. C. Vũ, *Dualization of signal recovery problems*, Set-Valued Var. Anal., 18 (2010), pp. 373–404.

[12] X. Zhang, M. Burger, and S. Osher, *A unified primal–dual algorithm framework based on Bregman iteration*, J. Sci. Comput., 46 (2011), pp. 20–46.

[13] L. M. Briceño-Arias and P. L. Combettes, *A monotone+skew splitting model for composite monotone inclusions in duality*, SIAM J. Optim., 21 (2011), pp. 1230–1250.

[14] L. M. Briceño-Arias, P. L. Combettes, J.-C. Pesquet, and N. Pustelnik, *Proximal algorithms for multicomponent image recovery problems*, J. Math. Imaging Vision, 41 (2011), pp. 3–22.

[15] P. L. Combettes, D. Dũng, and B. C. Vũ, *Proximity for sums of composite functions*, J. Math. Anal. Appl., 380 (2011), pp. 680–688.

[16] B. C. Vũ, *A splitting algorithm for dual monotone inclusions involving cocoercive operators*, Adv. Comput. Math., 38 (2013), pp. 667–681.

[17] L. Condat, *A generic proximal algorithm for convex optimization—Application to total variation minimization*, IEEE Signal Processing Lett., 21 (2014), pp. 1054–1057.

[18] P. L. Combettes, L. Condat, J.-C. Pesquet, and B. C. Vũ, *A forward–backward view of some primal–dual optimization methods in image recovery*, in Proc. of IEEE ICIP, Paris, France, Oct. 2014.

[19] N. Komodakis and J.-C. Pesquet, *Playing with duality: An overview of recent primal–dual approaches for solving large-scale optimization problems*, IEEE Signal Processing Mag., 32 (2015), pp. 31–54.

[20] U. R. Alim, T. Möller, and L. Condat, *Gradient estimation revitalized*, IEEE Trans. Visual. Comput. Graphics, 16 (2010), pp. 1494–1503.

[21] Z. Hossain, U. R. Alim, and T. Möller, *Towards high quality gradient estimation on regular lattices*, IEEE Trans. Visual. Comput. Graphics, 17 (2011), pp. 426–439.

[22] A. Chambolle, S. E. Levine, and B. J. Lucier, *An upwind finite-difference method for total variation-based image smoothing*, SIAM J. Imaging Sci., 4 (2011), pp. 277–299.

[23] R. Abergel and L. Moisan, *The Shannon total variation.* Research report MAP5 2016-19 <hal-01349516>, 2016.

[24] D. Alleysson, S. Süsstrunk, and J. Hérault, *Linear demosaicing inspired by the human visual system*, IEEE Trans. Image Processing, 14 (2005), pp. 439–449.

[25] L. Condat and S. Mosaddegh, *Joint demosaicking and denoising by total variation minimization*, in Proc. of IEEE ICIP, Orlando, USA, Sept. 2012.

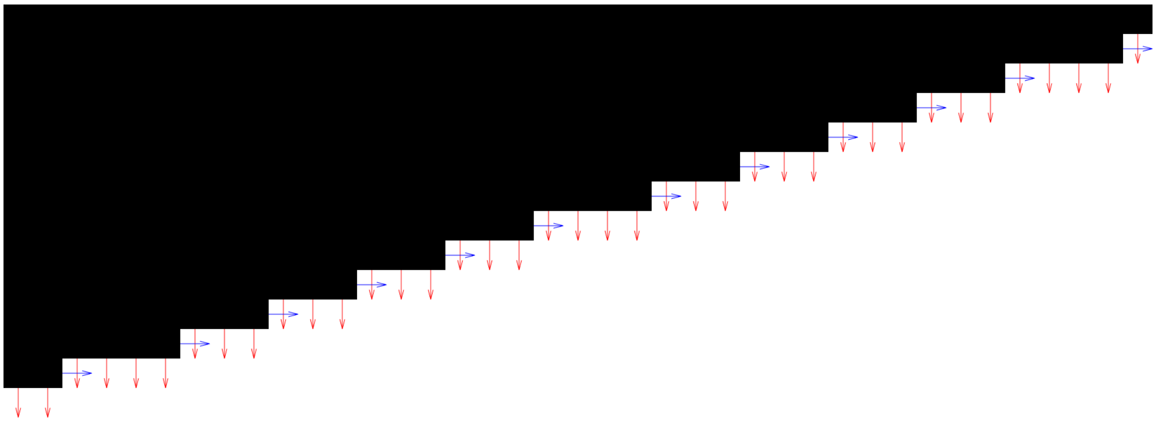
[26] F. H. Harlow and J. E. Welch, *Numerical calculation of time-dependent viscous incompressible flow of fluid with free surface*, Phys. Fluids, 8 (1965), pp. 2182–2189.

[27] P. B. Bochev and J. M. Hyman, *Principles of mimetic discretizations of differential operators*, in Compatible Spatial Discretizations, Springer New York, 2006, pp. 89–119.

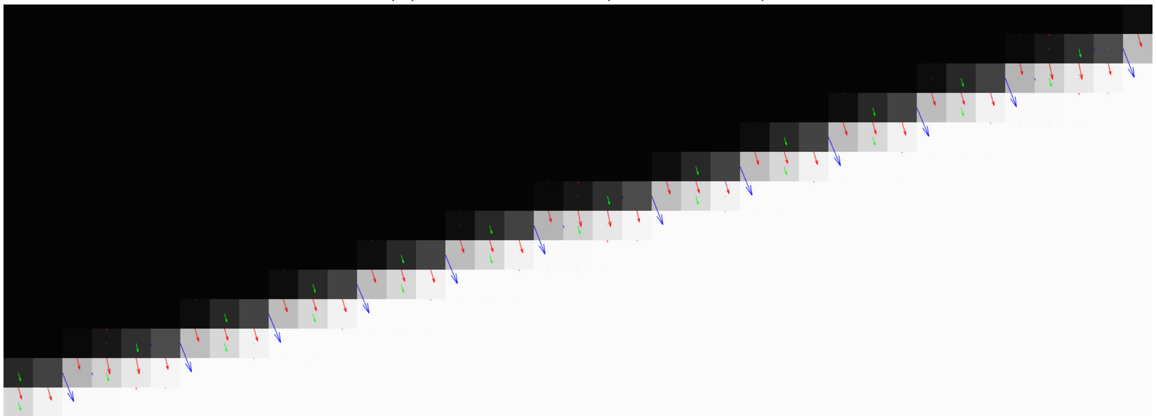
[28] C. Bazan, M. Abouali, J. Castillo, and P. Blomgren, *Mimetic finite difference methods in image processing*, Comput. Appl. Math., 30 (2011), pp. 701–720.

[29] J. F. Garamendi, F. J. Gaspar, N. Malpica, and E. Schiavi, *Box relaxation schemes in staggered discretizations for the dual formulation of total variation minimization*, IEEE Trans. Image

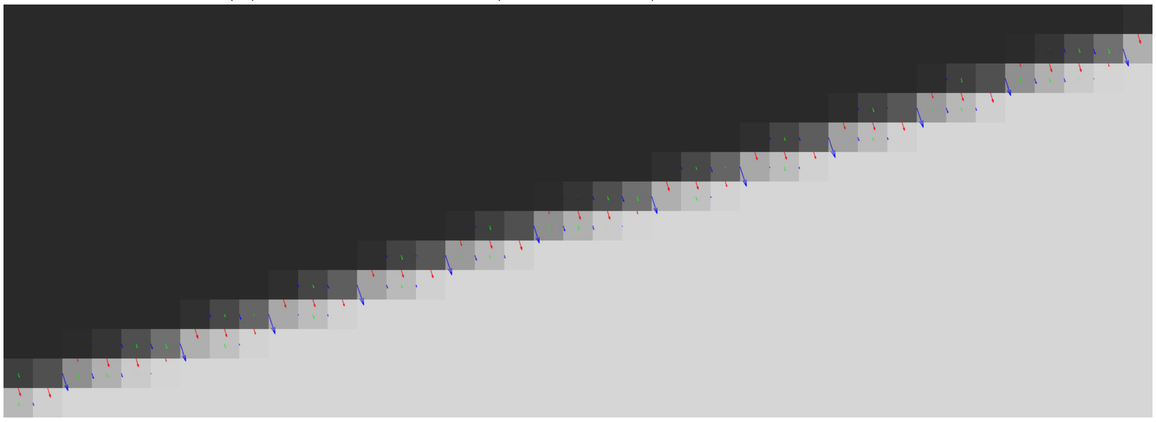
Processing, 22 (2013), pp. 2030–2043.
[30] M. Hintermüller, C. N. Rautenberg, and J. Hahn, *Functional-analytic and numerical issues in splitting methods for total variation-based image reconstruction*, Inverse Problems, 30 (2014).
[31] H. H. Bauschke and P. L. Combettes, *Convex Analysis and Monotone Operator Theory in Hilbert Spaces*, Springer, New York, 2011.
[32] L. Condat, *A primal-dual splitting method for convex optimization involving Lipschitzian, proximable and linear composite terms*, J. Optim. Theory Appl., 158 (2013), pp. 460–479.
[33] P. L. Combettes and J.-C. Pesquet, *Proximal splitting methods in signal processing*, in Fixed-Point Algorithms for Inverse Problems in Science and Engineering, H. H. Bauschke, R. Burachik, P. L. Combettes, V. Elser, D. R. Luke, and H. Wolkowicz, eds., Springer-Verlag, New York, 2010.
[34] S. Ma, *Alternating proximal gradient method for convex minimization*, J. Sci. Comput., (2015). to be published.
[35] W. Deng and W. Yin, *On the global and linear convergence of the generalized alternating direction method of multipliers*, J. Sci. Comput., 66 (2015), pp. 889–916.
[36] P. Combettes and J.-C. Pesquet, *Image restoration subject to a total variation constraint*, IEEE Trans. Image Processing, 13 (2004), pp. 1213–1222.
[37] J. M. Fadili and G. Peyré, *Total variation projection with first order schemes*, IEEE Trans. Image Processing, 20 (2011), pp. 657–669.
[38] W. Ring, *Structural properties of solutions to total variation regularization problems*, ESAIM Math. Model. Numer. Anal., 34 (2000), pp. 799–810.
[39] F. Alter, V. Caselles, and A. Chambolle, *Evolution of characteristic functions of convex sets in the plane by the minimizing total variation flow*, Interfaces Free Bound., 7 (2005), pp. 29–53.
[40] W. K. Allard, *Total variation regularization for image denoising. I. Geometric theory*, SIAM J. Math. Anal., 39 (2008), pp. 1150–1190.
[41] L. Condat, *Fast projection onto the simplex and the l1 ball*, Math. Program. Series A, 158 (2016), pp. 575–585.



(a) Initial image (central part)

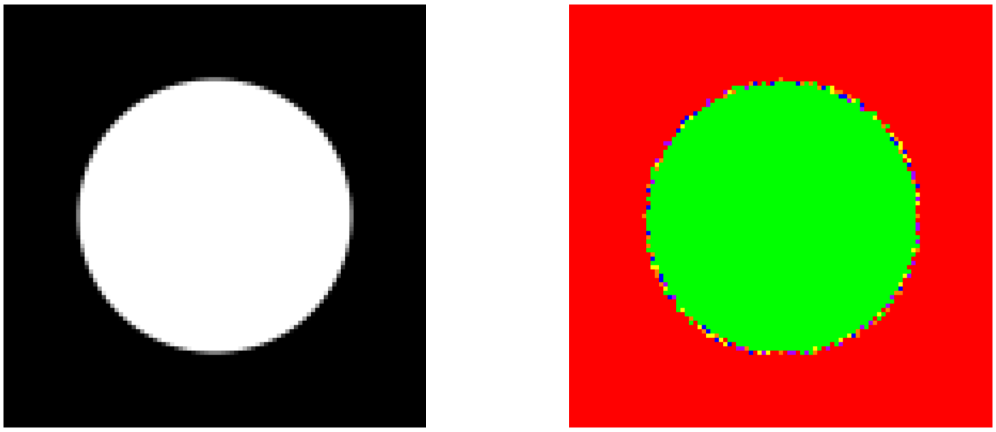


(b) Smoothed image (central part), proposed TV, $\lambda = 2$

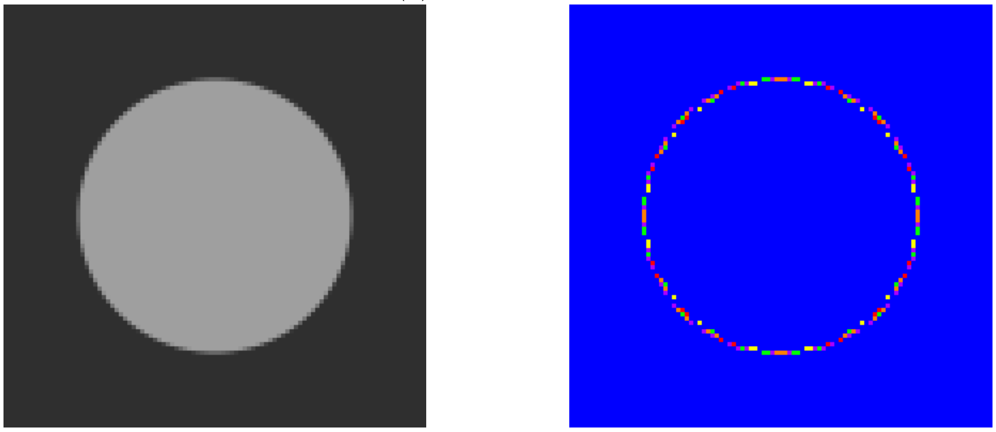


(c) Smoothed image (central part), proposed TV, $\lambda = 20$

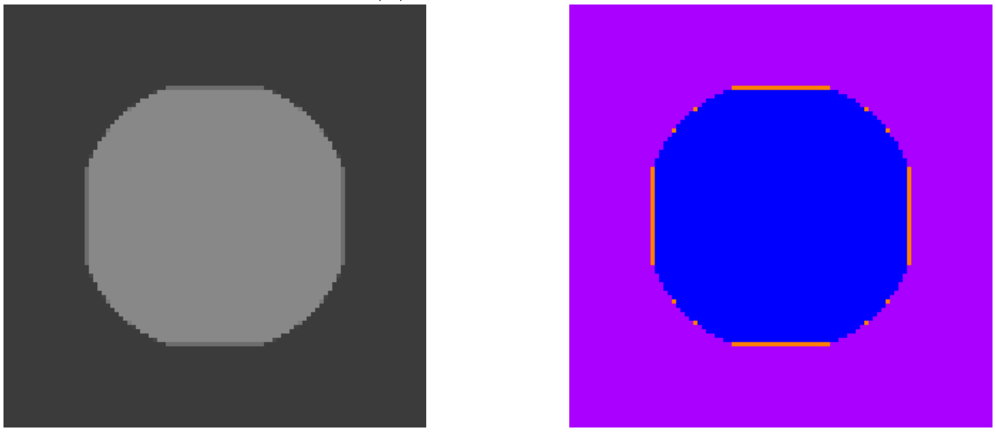
Figure 7. *Smoothing experiment, see subsection 6.1. In (b) and (c), central part of the images and their gradient fields obtained by smoothing the binary edge in (a), with $\lambda = 2$ and $\lambda = 20$, respectively.*



(a) Initial image

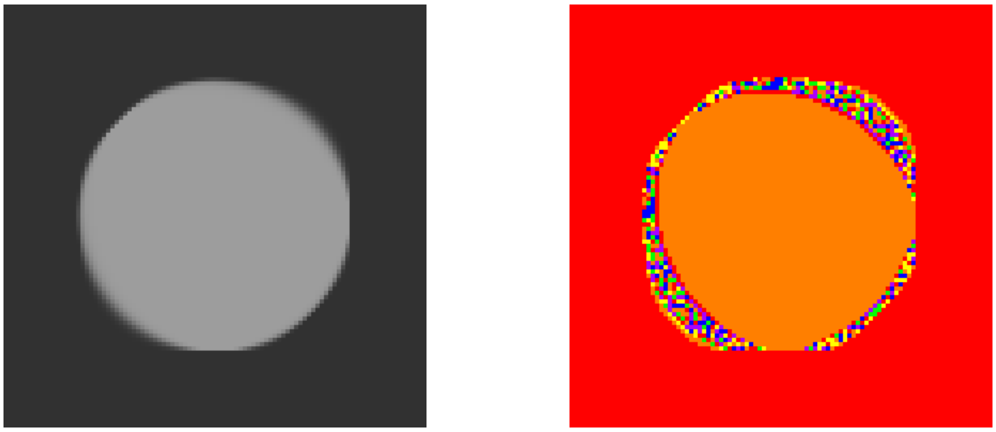


(b) Reference image

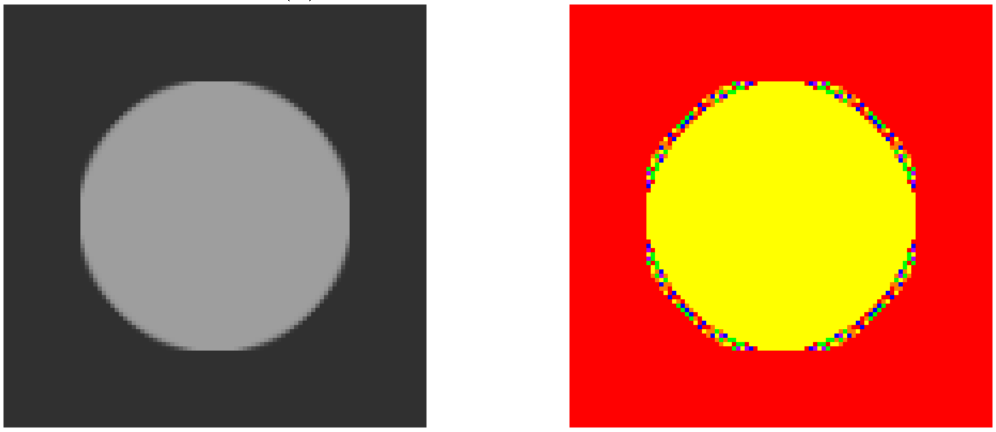


(c) Smoothed image, anisotropic TV

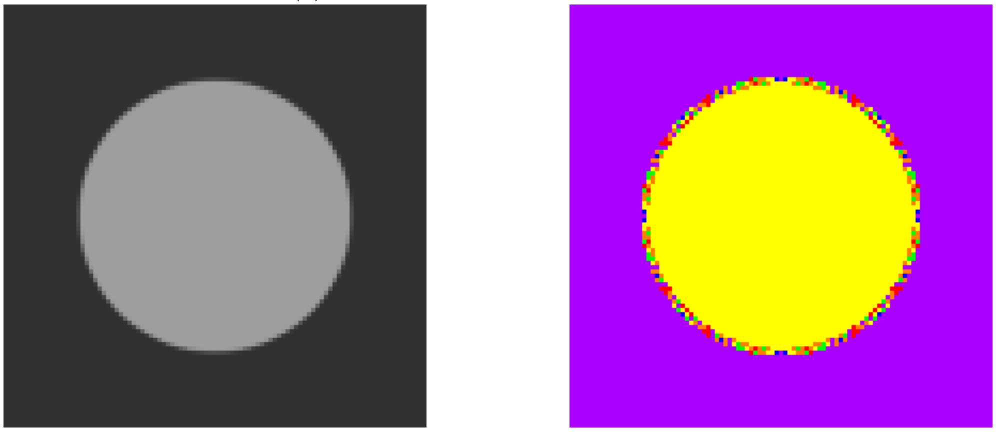
Figure 8. *Smoothing experiment, see subsection 6.2. In (c), the image obtained by smoothing the image in (a), using the anisotropic TV. In (b), the ideal result one would like to obtain. Every image is represented in grayscale on the left and in false colors on the right, to better show the spread of the edges. The fact that in (b) the disk interior and the background are rendered with the same blue false color is a coincidence, due to the limited number of colors in the colormap 'prism' of Matlab.*



(d) Smoothed image, isotropic TV

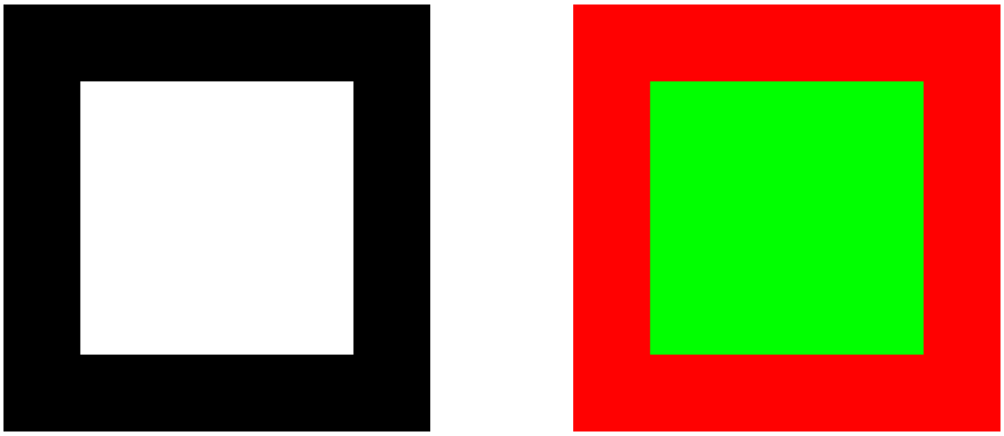


(e) Smoothed image, upwind TV

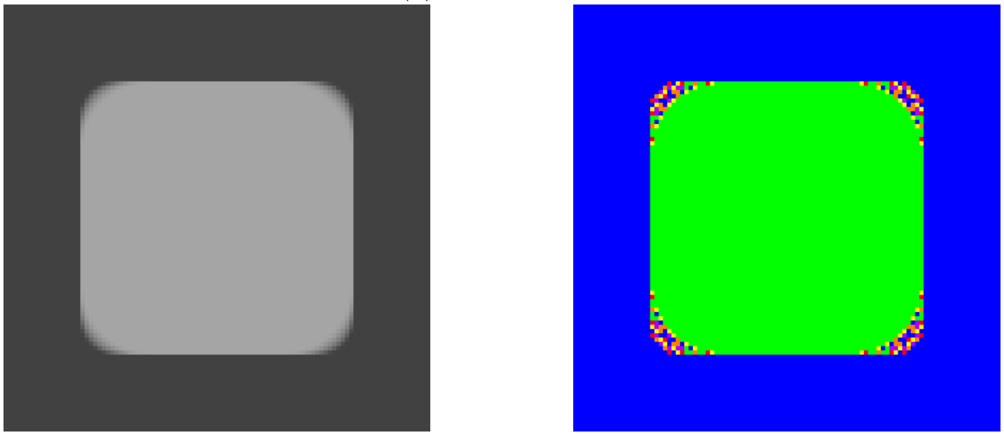


(f) Smoothed image, proposed TV

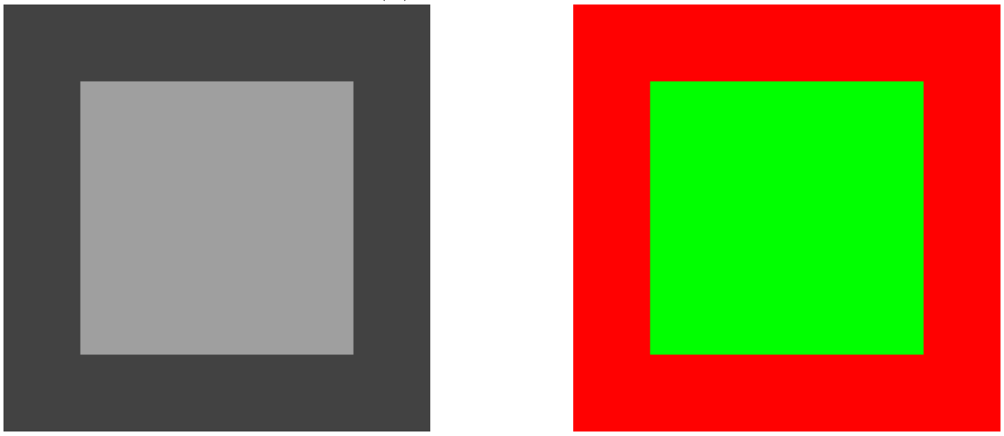
Figure 8, continued. *In (d), (e), (f), the images obtained by smoothing the image in (a), using the isotropic TV, upwind TV, proposed TV, respectively.*



(a) Initial image

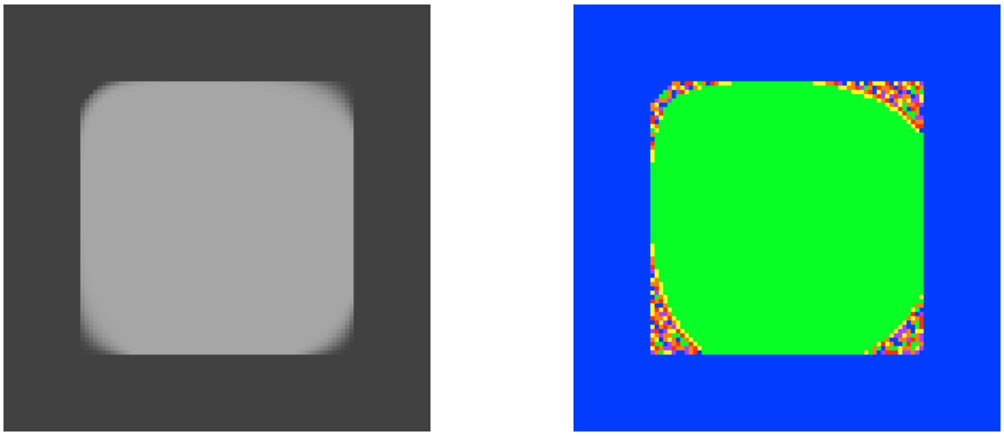


(b) Reference image

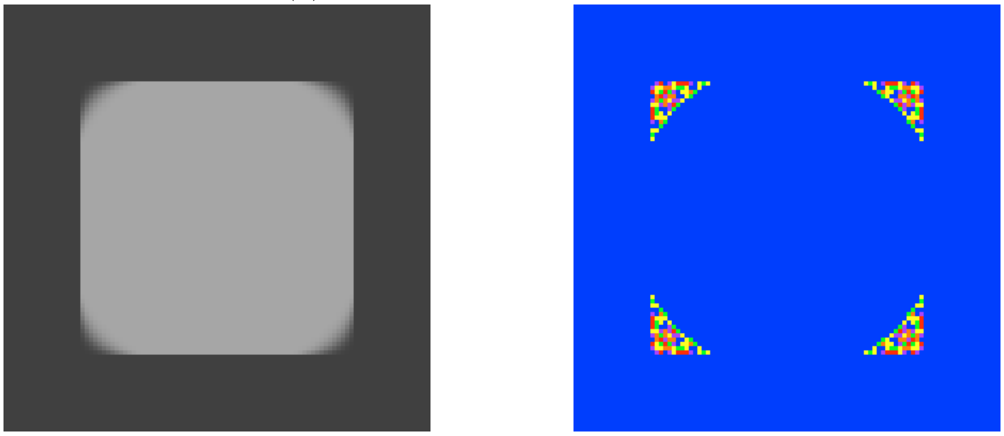


(c) Smoothed image, anisotropic TV

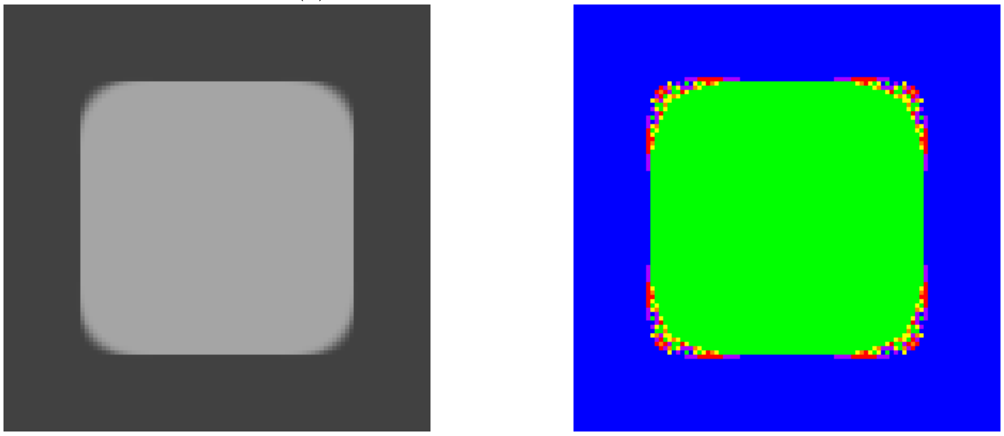
Figure 9. *Smoothing experiment, see* subsection 6.3*. In (c), the image obtained by smoothing the image in (a), using the anisotropic TV. In (b), the ideal result one would like to obtain. Every image is represented in grayscale on the left and in false colors on the right, to better show the spread of the corners.*



(d) Smoothed image, isotropic TV

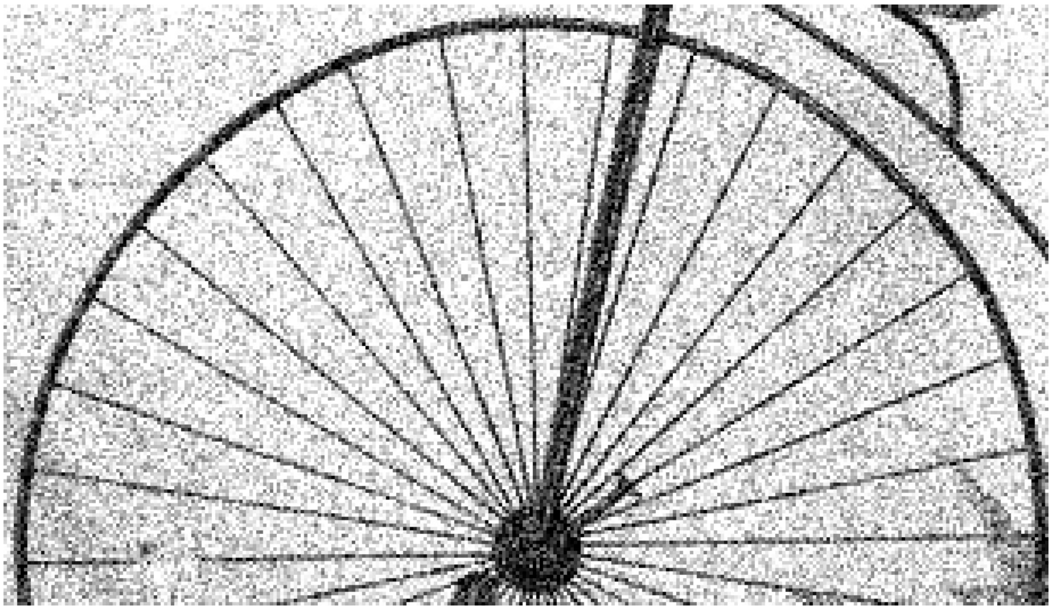


(e) Smoothed image, upwind TV



(f) Smoothed image, proposed TV

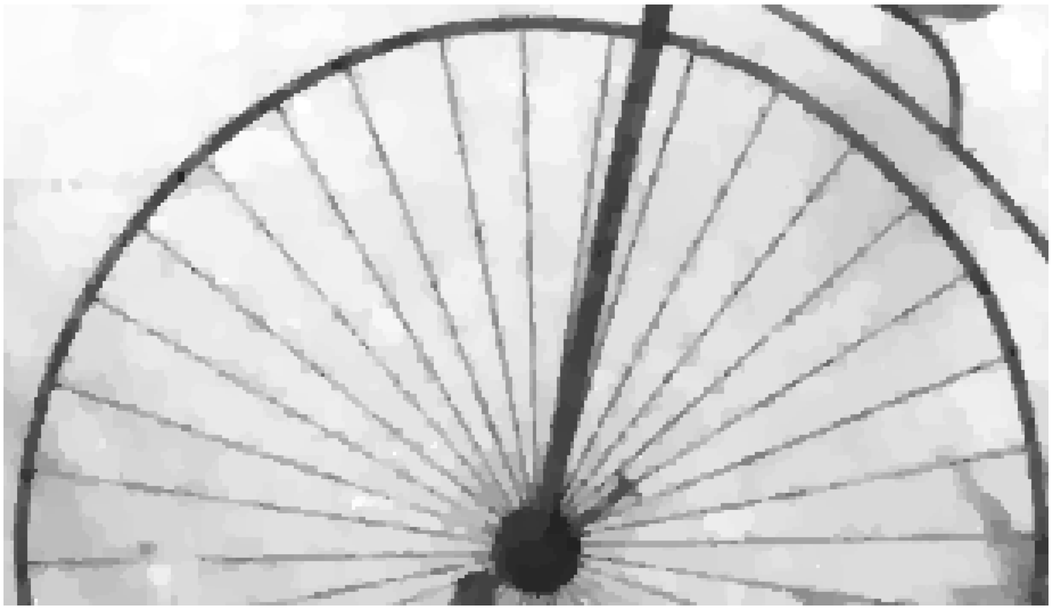
Figure 9, continued. *In (d), (e), (f), the images obtained by smoothing the image in (a), using the isotropic TV, upwind TV, proposed TV, respectively. The fact that in (e) the square interior and the background are rendered with the same blue false color is a coincidence, due to the limited number of colors in the colormap 'prism' of Matlab.*



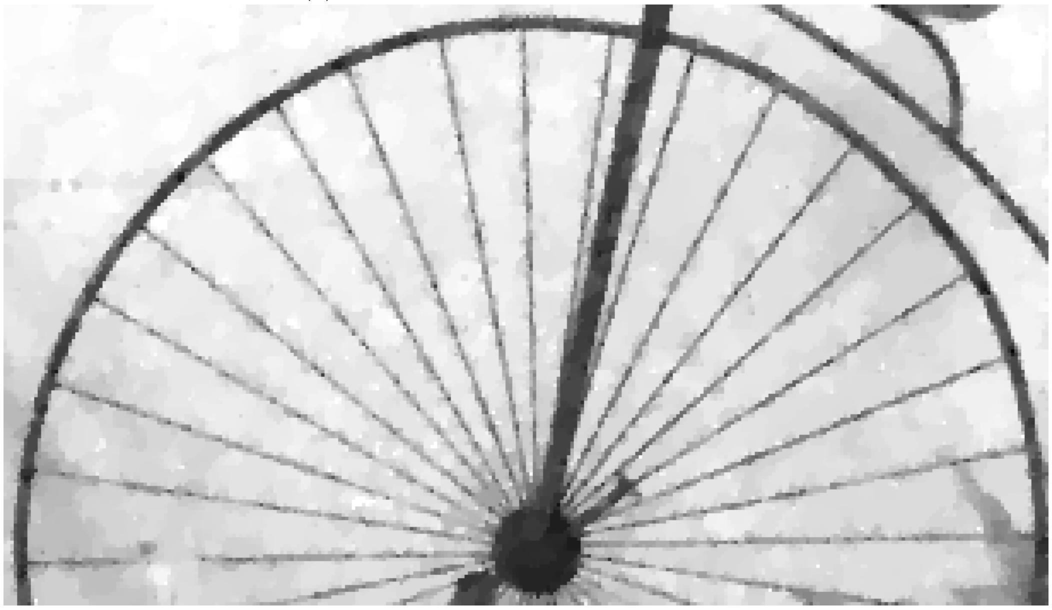
(a) Initial image

(b) Reference image

Figure 10. *Denoising experiment, see subsection 6.4. The initial noisy image in (a) is the ground-truth image in (b), after corruption by additive white Gaussian noise.*

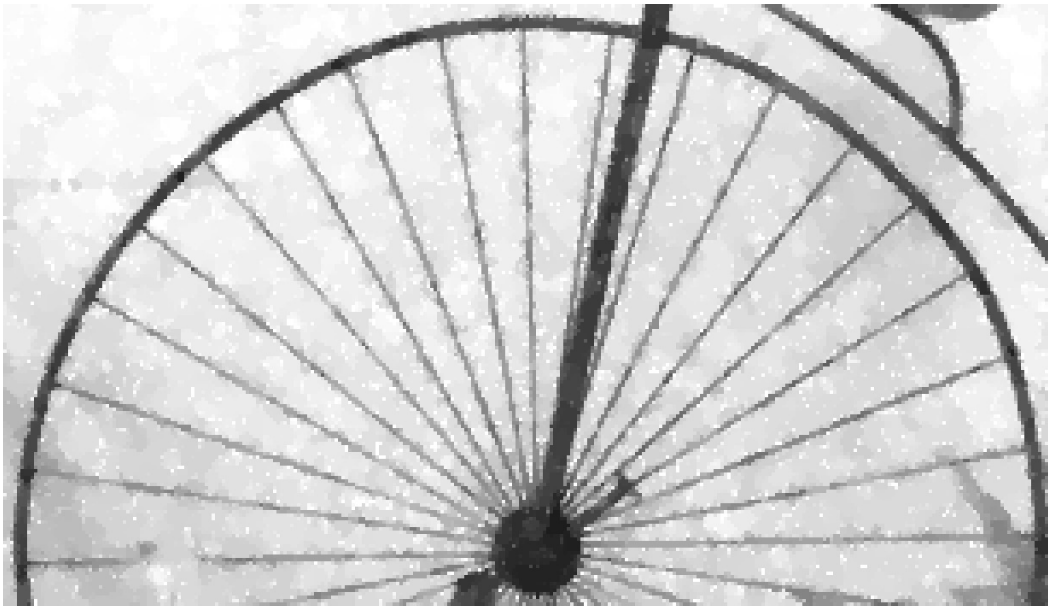


(c) Denoised image, anisotropic TV

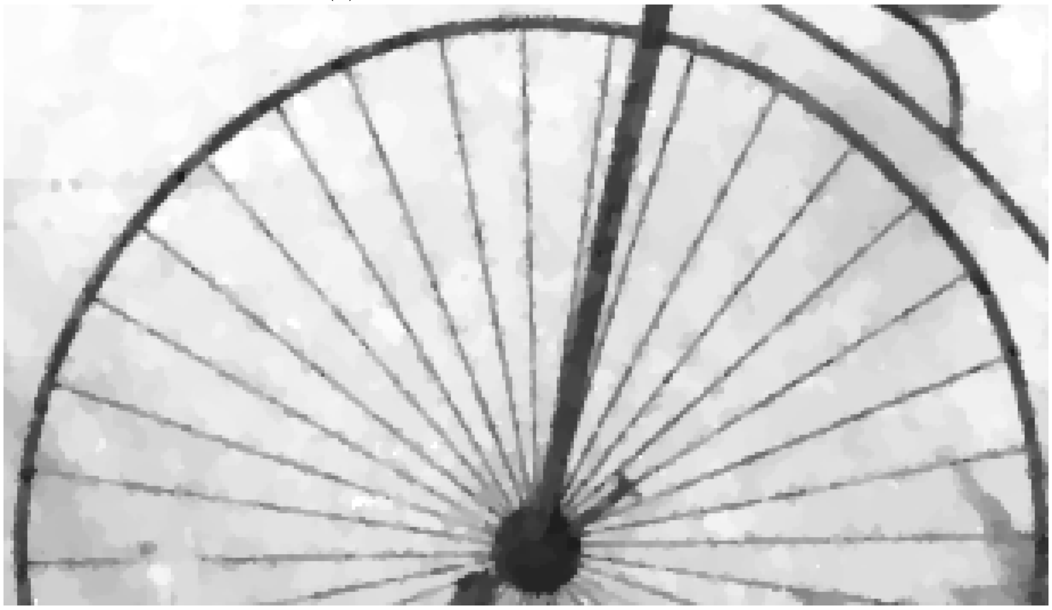


(d) Denoised image, isotropic TV

Figure 10, continued. *In (c), (d), the images obtained by denoising the image in (a), using the anisotropic TV and isotropic TV, respectively.*

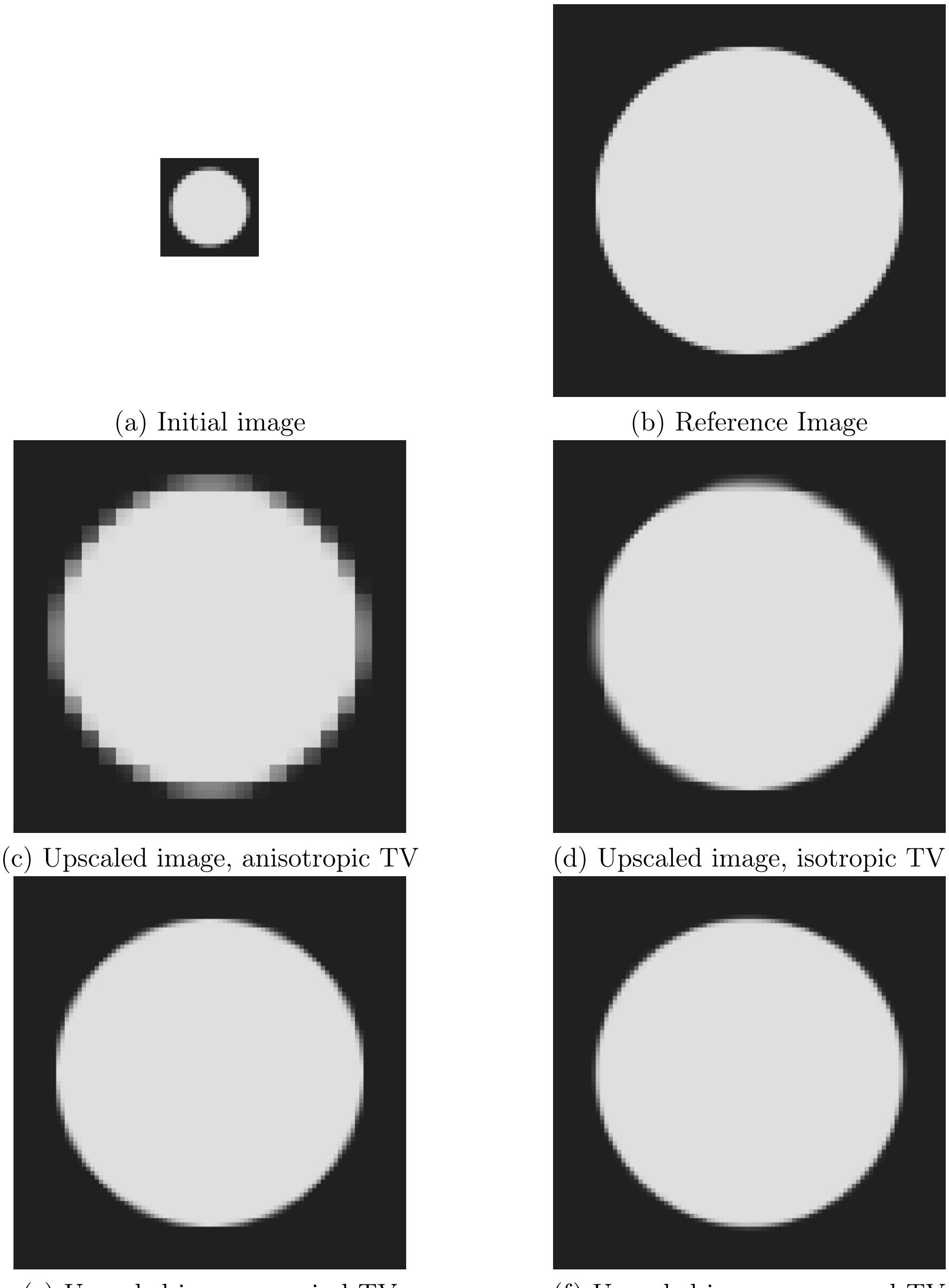


(e) Denoised image, upwind TV



(f) Denoised image, proposed TV

Figure 10, continued. *In (e), (f), the images obtained by denoising the image in (a), using the upwind TV and proposed TV, respectively.*



(a) Initial image

(b) Reference Image

(c) Upscaled image, anisotropic TV

(d) Upscaled image, isotropic TV

(e) Upscaled image, upwind TV

(f) Upscaled image, proposed TV

Figure 11. *Upscaling experiment, see subsection 6.6. The images in (b)–(f), when reduced by averaging over 4×4 blocks, yield the image in (a), exactly.*

(a) Initial image

(b) Segmented image, anisotropic TV

(c) Segmented image, isotropic TV

(d) Segmented image, upwind TV

(e) Segmented image, proposed TV

Figure 12. *Segmentation experiment, see subsection* 6.8.