Edge-Preserving Integration of a Normal Field: Weighted Least-squares, TV and L^1 Approaches

Yvain Quéau and Jean-Denis Durou

Université de Toulouse, IRIT, UMR CNRS 5505, Toulouse, France
yvain.queau@enseeiht.fr durou@irit.fr

Abstract. We introduce several new functionals, inspired from variational image denoising models, for recovering a piecewise-smooth surface from a dense estimation of its normal field[1]. In the weighted least-squares approach, the non-differentiable elements of the surface are *a priori* detected so as to weight the least-squares model. To avoid this detection step, we introduce reweighted least-squares for minimising an isotropic TV-like functional, and split-Bregman iterations for L^1 minimisation.

Keywords: Integration; Shape-from-gradient; Photometric stereo.

1 Introduction

Problem Statement. The normal field $\mathbf{n}$ of a surface can be estimated by 3D-reconstruction techniques such as photometric stereo [17]. To obtain a set of 3D points located on the surface, the estimated normal field must then be *integrated* into a depth map z, over a subset Ω of the image domain. This second step is crucial in the 3D-reconstruction process, since the accuracy of the recovered surface widely depends on the robustness of integration to noise and outliers.

Let us first recall the equations describing this integration problem, which are similar under both orthographic and perspective projections. In the orthographic case, z is related to the normal field $\mathbf{n}$, for every $(x,y) \in \Omega$, through [6]:

$$\mathbf{n}(x,y) = \frac{1}{\sqrt{\|\nabla z(x,y)\|_2^2 + 1}} \begin{bmatrix} -\nabla z(x,y) \\ 1 \end{bmatrix} \tag{1}$$

where $\nabla z = [\partial_x z, \partial_y z]^\top$ is the gradient of z. Denoting:

$$p_{\mathcal{O}}(x,y) = -\frac{n_1(x,y)}{n_3(x,y)},\ q_{\mathcal{O}}(x,y) = -\frac{n_2(x,y)}{n_3(x,y)},\ \mathbf{g}_{\mathcal{O}}(x,y) = [p_{\mathcal{O}}(x,y), q_{\mathcal{O}}(x,y)]^\top \tag{2}$$

where n_i, $i \in [1,3]$, is the i-th component of $\mathbf{n}$, we obtain from (1) and (2):

$$\nabla z(x,y) = \mathbf{g}_{\mathcal{O}}(x,y) \tag{3}$$

[1] Sample codes for testing the proposed methods can be found on http://ubee.enseeiht.fr/photometricstereo/

In the case of perspective projection, we need to know the focal length f of the camera, and to set the origin of image coordinates to the principal point. Introducing the change of variable $\tilde{z} = \log z$, we obtain [6]:

$$\mathbf{n}(x,y) = \frac{1}{\sqrt{\|\nabla\tilde{z}(x,y)\|_2^2 + \left(1+\nabla\tilde{z}(x,y)\cdot\frac{1}{f}\,[x,y]^\top\right)^2}} \begin{bmatrix} -\nabla\tilde{z}(x,y) \\ 1+\nabla\tilde{z}(x,y)\cdot\frac{1}{f}\,[x,y]^\top \end{bmatrix} \quad (4)$$

By setting $d(x,y) = xn_1(x,y) + yn_2(x,y) + fn_3(x,y)$, and:

$$p_{\mathcal{P}}(x,y) = -\frac{n_1(x,y)}{d(x,y)},\ q_{\mathcal{P}}(x,y) = -\frac{n_2(x,y)}{d(x,y)},\ \mathbf{g}_{\mathcal{P}}(x,y) = [p_{\mathcal{P}}(x,y), q_{\mathcal{P}}(x,y)]^\top \quad (5)$$

we get from (4) and (5), after some algebra:

$$\nabla\tilde{z}(x,y) = \mathbf{g}_{\mathcal{P}}(x,y) \quad (6)$$

Thus, for both these projection models, one has to solve, in every $(x,y) \in \Omega$, the same equation:

$$\nabla u(x,y) = \mathbf{g}(x,y) \quad (7)$$

where $(u,\mathbf{g}) = (z,\mathbf{g}_{\mathcal{O}})$ in the orthographic case, and $(u,\mathbf{g}) = (\tilde{z},\mathbf{g}_{\mathcal{P}})$ in the perspective one.

Integrating the normal field refers to the process of recovering the unknown u, which will be abusively referred to as "depth map" in the following, from an estimation $\mathbf{g} = [p,q]^\top$ of its gradient field over Ω. This problem, which has a long history since it dates back to the Dirichlet problem, has given rise to numerous studies in the area of mathematics for imaging, using many different approaches such as Fourier analysis [7,16], fast marching [8] or Sylvester equations [10,11]. In this paper, as in many recent works [1,2,4,5,14], we choose the energy minimisation way, which offers a natural framework for controlling the influence of noise and outliers.

Summary of our Contributions. We focus on the case where solving Eq. (7) makes sense only *almost everywhere*, which happens as soon as the surface to be reconstructed contains edges and depth discontinuities: the gradient ∇u of u cannot be defined on the neighborhood of such non-differentiable elements. In this case, classical least-squares solvers fail (Figure 1) and more robust estimation must be considered. Completing the study proposed in [5], we introduce three new functionals inspired by image denoising models, whose minimisation is shown to provide piecewise-smooth surfaces on an arbitrary connected domain Ω. They are based, respectively, on weighted least-squares (WLS), isotropic total variation (TV), and L^1 optimisation.

The rest of this paper is organized as follows. After reviewing in Section 2 the main energy minimisation methods for surface reconstruction from a gradient field, we detail in Section 3 the proposed edge-preserving approaches, which are eventually evaluated on both synthetic and real-world datasets in Section 4.

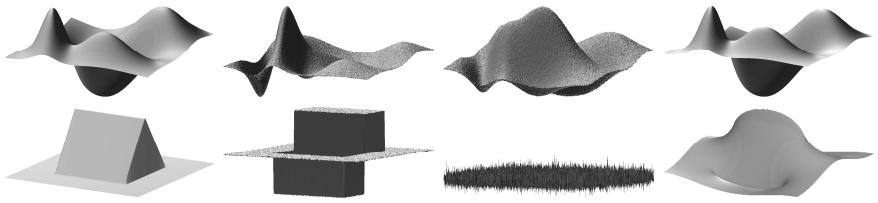

Fig. 1. Least-squares normal integration. First row, from left to right: ground truth $\mathcal{C}^\infty$ depth map u, analytical derivatives p and q corrupted by an additive zero-mean Gaussian noise with standard deviation $\sigma = 5\%$ of $\|\mathbf{g}\|_\infty$, and least-squares reconstruction [10]. Second row: same, for piecewise-$\mathcal{C}^\infty$ surface. Noise in the data is successfully handled by least-squares (first row), but discontinuities are smoothed (second row).

2 Related Work

2.1 Integrability of a Gradient Field

In the ideal case, $\mathbf{g} = [p, q]^\top$ is the true gradient of a $\mathcal{C}^2$ function u holding: $\partial_{yx} u = \partial_{xy} u$ (Schwarz' theorem). The distance from a gradient field $\mathbf{g}$ to an ideal (integrable) field holding $\partial_y p = \partial_x q$ can thus be measured by the *integrability* term [7]:

$$\mathcal{I}(x, y) = |\partial_y p(x, y) - \partial_x q(x, y)| \tag{8}$$

which is never null in real-world scenarios, because of noise and of depth discontinuities. In such cases, it makes sense to estimate an approximate solution u of Eq. (7) whose gradient ∇u is integrable, rather than to solve Eq. (7) exactly.

This can be performed efficiently through energy minimisation, by seeking u as the solution of an optimisation problem, lying in an appropriate function space. For instance, if u is sought in $L^2(\Omega)$, integrability of its gradient is implicitely granted (in the presence of discontinuities, the space of functions with bounded variations $BV(\Omega)$ should be preferred, so as to allow piecewise-smooth functions). We provide hereafter a brief overview of the main normal integration methods relying on energy minimisation.

2.2 Continuous Least-squares Formulation

The most natural energy minimisation approach to solve (7) consists in estimating u in a least-squares sense [7,16,6,10], by introducing the functional:

$$\mathcal{F}_{LS}(u) = \iint_\Omega \|\nabla u(x, y) - \mathbf{g}(x, y)\|_2^2 \, \mathrm{d}x \, \mathrm{d}y \tag{9}$$

According to the calculus of variations, minimising this functional is equivalent to solving the associated Euler-Lagrange equation on the interior part $\overset{\circ}{\Omega}$ of Ω:

$$\Delta u = \nabla \cdot \mathbf{g} \tag{10}$$

which is a Poisson equation ($\nabla\cdot$ is the divergence operator, which is the adjoint of the gradient, and $\Delta = \nabla\cdot\nabla$ is the Laplacian operator), along with the natural boundary condition (BC), which is of the Neumann type:

$$(\nabla u - \mathbf{g}) \cdot \mu = 0 \tag{11}$$

on the boundary $\partial\Omega$ of Ω, μ being normal to $\partial\Omega$.

Discretising Eqs (10) and (11) provides a linear system of equations which can be solved in linear time through Fast Fourier Transform (FFT), if Ω is rectangular. Indeed, replacing the natural BC (11) by a periodic one, Frankot and Chellappa's well-known algorithm [7] recovers the Fourier transform of the depth map analytically, and inverse FFT eventually provides a solution u of (10). This algorithm was extended by Simchony *et al.* in [16] to the natural BC, through the use of Discrete Cosine Transform (DCT).

2.3 Discretising the Functional, or the Optimality Conditions?

Instead of discretising the optimality conditions (10) and (11), the functional (9) itself can be discretised. This is the approach followed in [6], where it is shown that doing so, no explicit BC is needed (yet, the natural BC is *implicitely* satisfied). After proper discretisation, a new linear system is obtained, which is solved using Jacobi iterations. Alternatively, Harker and O'Leary show in [10] that the discrete least-squares functional can be minimised by solving a Sylvester equation, provided Ω is rectangular (this hypothesis is neither required in [6], nor in the present paper).

Examples of results obtained using the least-squares solver from [10] are shown in the last column of Figure 1. These experiments illustrate the robustness of least-squares against additive Gaussian noise, but also the edge smoothing which occurs using quadratic regularisation. As we shall see later, quadratic regularisation can be improved by introducing weights, or by replacing the squared L^2 norm by a non-differentiable regularisation.

Since the functional (9) is convex, discretising the functional or the associated optimality condition should be strictly equivalent, provided the natural BC is enforced. Yet, as noted by Harker and O'Leary in [11], Poisson-based integration relying on DCT suffers from a bias, due to inconsistent numerical approximations of the gradient ∇u in the discretisation of the natural BC (11). The choice of such inconsistent derivatives, as well as a rectangular domain Ω, are actually necessary for obtaining a matrix of the block-Toeplitz type, and thus allowing fast recovery by DCT. Choosing consistent numerical derivatives, or a non-rectangular domain Ω, the structure of this matrix is lost, and the system resulting from the discretisation of the continuous optimality condition must be solved using standard sparse solvers.

In this paper, as in [6,10], we choose to consider discrete functionals so as to avoid dealing with boundary conditions. Rather than relying on special matrix structures [10], we use standard solvers for the numerics, allowing us to deal with non-rectangular domains, as in [6].

2.4 Non-quadratic Regularisations

In [11], Harker and O'Leary extend the method from [10] to the case of spectral and Sobolev regularisations, improving the robustness of their method to Gaussian noise. Yet, such regularisations are not adapted to depth discontinuities, since they remain quadratic. In [1], Agrawal *et al.* study several functionals having the following general form:

$$\mathcal{F}_\Psi(u) = \iint_\Omega \Psi\left(\|\nabla u(x,y) - \mathbf{g}(x,y)\|_2\right) \,\mathrm{d}x\,\mathrm{d}y \tag{12}$$

where Ψ is chosen so as to reduce the influence of outliers. A numerical study of the discrete versions of several such functionals is presented in [5], where the Jacobi iterations used in [6] are extended to the minimisation of non-convex functionals through semi-implicit schemes. The use of sparse regularisations derived from the L^1 norm has also become an important research direction [4,14]: we will show how to accelerate such schemes using split-Bregman iterations. Extension to L^p minimisation, $p < 1$, is also presented in [2]. The results are indeed impressive in the presence of very noisy data, but involve setting numerous parameters, which is hardly tractable in real-world applications.

Furthermore, since photometric stereo is a technique which is mostly performed inside laboratories, the presence of a huge amount of Gaussian noise in the measurements is very unlikely, and thus greater care is given in this paper to *outliers* such as discontinuities, which cannot be avoided since they describe the surface itself and not the acquisition procedure. We introduce in the next section several new functionals related to this issue.

3 New functionals

3.1 Quadratic Prior

The functional (12) is not coercive, because of the ambiguity $u \mapsto u + k$, k constant, in the initial equation (7). In the literature, this ambiguity is usually solved *a posteriori*, for instance by manually setting the mean value of u. In this work, we proceed this way to first compute an approximate solution u_0 through DCT [16] (if Ω is not rectangular, $\mathbf{g}$ has to be completed with null values, which obviously creates a bias), before introducing u_0 as a quadratic prior to force coercivity.

This prior being biased in the presence of discontinuities and non-rectangular domains, it can be seen as an initial depth map that we want to denoise using an edge-preserving regularisation Φ which shall ensure diffusion along $\mathbf{g}$:

$$\mathcal{F}_\Phi(u) = \iint_\Omega \Phi\left(\nabla u(x,y) - \mathbf{g}(x,y)\right) + \frac{\lambda}{2}\left(u(x,y) - u_0(x,y)\right)^2 \,\mathrm{d}x\,\mathrm{d}y \tag{13}$$

with $\lambda > 0$ chosen according to the quality of the approximate solution u_0. In this paper, we consider three types of regularisation: $\Phi = w\|.\|_2^2$ (weighted least-squares), $\Phi = \|.\|_2$ (isotropic TV-like), and $\Phi = \|.\|_1$ (L^1).

3.2 Weighted Least-squares Functional

In a first approach, we assume that it is possible to *a priori* detect outliers through the evaluation of the integrability term (8). This *a priori* detection is used to weight the influence of discontinuities. Setting $\Phi(.) = w||.||_2^2$, where w is a weighting function depending only on the integrability term (8), and not on u, we obtain the weighted least-squares functional:

$$\mathcal{F}_{WLS}(u) = \iint_{\Omega} w(x,y) \|\nabla u(x,y) - \mathbf{g}(x,y)\|_2^2 + \frac{\lambda}{2} \left(u(x,y) - u_0(x,y)\right)^2 \, \mathrm{d}x \, \mathrm{d}y \quad (14)$$

Since we know that the integrability (8) is an indicator of the presence of discontinuities (though having null integrability does not imply being smooth: think of a piecewise flat shape), it seems natural to choose for w an integrability-based weighting function, which should be a decreasing function of $\mathcal{I}$. To choose effectively the weights, let us use the continuous optimality condition associated with $\mathcal{F}_{WLS}$. Assuming $w > 0$, and remarking that $\frac{\nabla w}{w} = \nabla(\log w)$, we obtain:

$$\Delta u + \nabla(\log w) \cdot (\nabla u - \mathbf{g}) - \lambda \left(u - u_0\right) = \nabla \cdot \mathbf{g} \quad (15)$$

Because of the presence of the logarithm, we consider:

$$w(x,y) = \exp(-\gamma \mathcal{I}(x,y)^2) \quad (16)$$

where $\gamma \geq 0$ is a parameter for controlling the weights ($\gamma = 0$ corresponds to the standard least-squares formulation).

We now discretise u uniformly over a grid (which does not need to be rectangular, unlike in [7,16,10,11]), with spacing 1, also denoted Ω for convenience. Extending the rationale in [6] for least-squares functionals, a consistent second-order accurate discretisation of (14) is obtained by first-order forward differences in u, and computation of the forward means of the components p and q of $\mathbf{g}$:

$$\mathrm{F}_{WLS}(u) = \sum_{(i,j) \in \Omega^{x+}} w^{i,j} \left(\partial_x^+ u^{i,j} - \bar{p}^{i,j}\right)^2 + \sum_{(i,j) \in \Omega^{y+}} w^{i,j} \left(\partial_y^+ u^{i,j} - \bar{q}^{i,j}\right)^2 + \frac{\lambda}{2} \sum_{(i,j) \in \Omega} \left(u^{i,j} - u_0^{i,j}\right)^2 \quad (17)$$

where we denote $u^{i,j}$ the value of u at discrete point (i,j), $\bar{p}^{i,j} = \frac{p^{i+1,j} + p^{i,j}}{2}$, $\bar{q}^{i,j} = \frac{q^{i,j+1} + q^{i,j}}{2}$, $\partial_x^+ u^{i,j} = u^{i+1,j} - u^{i,j}$, $\partial_y^+ u^{i,j} = u^{i,j+1} - u^{i,j}$, $\Omega^{x+} = \{(i,j) \in \Omega \text{ s.t. } (i+1,j) \in \Omega\}$ and $\Omega^{y+} = \{(i,j) \in \Omega \text{ s.t. } (i,j+1) \in \Omega\}$. The optimality condition in $u^{i,j} \in \Omega$ reads:

$$\begin{aligned} &\chi^{i+1,j} w^{i,j} \left(u^{i+1,j} - u^{i,j}\right) + \chi^{i,j+1} w^{i,j} \left(u^{i,j+1} - u^{i,j}\right) + \chi^{i-1,j} w^{i-1,j} \left(u^{i-1,j} - u^{i,j}\right) \\ &+ \chi^{i,j-1} w^{i,j-1} \left(u^{i,j-1} - u^{i,j}\right) - \frac{\lambda}{2} u^{i,j} = \chi^{i+1,j} w^{i,j} \bar{p}^{i,j} + \chi^{i,j+1} w^{i,j} \bar{q}^{i,j} \\ &\qquad - \chi^{i-1,j} w^{i-1,j} \bar{p}^{i-1,j} - \chi^{i,j-1} w^{i,j-1} \bar{q}^{i,j-1} - \frac{\lambda}{2} u_0^{i,j} \end{aligned} \quad (18)$$

where χ is the characteristic function of Ω. If w is constant and $\lambda = 0$, it is easily verified that (18) is a discrete approximation of both the Poisson equation (10) and the natural BC (11).

Stacking the $u^{i,j}$ column-wise in a vector $\mathbf{u}$ of size $n \times 1$, where n is the cardinal of Ω, the optimality condition (18) reads as a linear system $\mathbf{Au} = \mathbf{b}$, where $\mathbf{A}$ is a block-pentadiagonal $n \times n$ full-rank matrix with strictly dominant diagonal. We experimentally found that, for relatively small grids (up to 512×512), direct sparse solvers provide a fast solution to this system: since $\mathbf{A}$ has a small bandwidth (equal to the number of rows in Ω), computation of the sparse product $\mathbf{A}^\top\mathbf{A}$ is very fast, and the normal equation $\mathbf{A}^\top\mathbf{Au} = \mathbf{A}^\top\mathbf{b}$ can be solved through sparse Cholesky factorisation, though it artificially increases the order of points involved in the finite differences, leading to a small additional smoothing (see the numerical results on the peaks dataset in Section 4). Studying more efficient solvers for this problem, as for instance Krylov subspace methods applied to the initial $\mathbf{Au} = \mathbf{b}$ problem, will be the subject of a future research.

3.3 Isotropic TV Functional

The previous approach relies on *a priori* detection of the discontinuities, so that the corresponding points are "manually" discarded from the equality (7). Yet, *a priori* setting the weights might sometimes be tedious. The weights can also be automatically chosen as a function of $\|\nabla u - \mathbf{g}\|_2$ [1,5], but the problem cannot be solved directly anymore, and requires an iterative minimisation. We show how to use this idea to minimise a functional resembling the L^2-TV model [15].

It is well known in the image processing community that the isotropic total variation (TV) measure $TV(u) = \int_\Omega \|\nabla u(x,y)\|_2 \,\mathrm{d}x\mathrm{d}y$ has interesting edge-preserving properties, and tends to favor piecewise-smooth solutions. Considering the discontinuities as the equivalent of edges in image denoising, one would expect the residual $\nabla u - \mathbf{g}$ to be piecewise-smooth as well, with jumps located in discontinuities. This remark invites us to adapt the ROF model [15] to our problem: choosing $\Phi(.) = \|.\|_2$, we obtain from (13) the following functional:

$$\mathcal{F}_{TV}(u) = \iint_\Omega \|\nabla u(x,y) - \mathbf{g}(x,y)\|_2 + \frac{\lambda}{2}\left(u(x,y) - u_0(x,y)\right)^2 \,\mathrm{d}x\,\mathrm{d}y \qquad (19)$$

Remarking that $\|.\|_2 = \frac{\|.\|_2}{\|.\|_2^2}\|.\|_2^2$, this functional can be minimised through iteratively reweighted least-squares:

$$\begin{cases} w^k(x,y) = \dfrac{\|\nabla u^k(x,y) - \mathbf{g}(x,y)\|_2}{\|\nabla u^k(x,y) - \mathbf{g}(x,y) + \theta\|_2^2}, \ \forall (x,y) \in \Omega \\ u^{k+1} = \underset{u}{\operatorname{argmin}} \displaystyle\int_\Omega w^k(x,y)\|\nabla u(x,y) - \mathbf{g}(x,y)\|_2^2 + \frac{\lambda}{2}(u(x,y) - u_0(x,y))^2 \mathrm{d}x\mathrm{d}y \end{cases} \qquad (20)$$

with $u^0 = u_0$, $w^0 = 1$, and $\theta > 0$, small. The update in u, using Cholesky factorisation, has already been described in Section 3.2. Proceeding so, the normal equations are solved at each iteration, as in [1]. As a consequence, few iterations are needed, though this might become memory-hungry for large grids. Iterative Jacobi approximations, in the manner of what is proposed in [5], would probably offer a less memory-hungry solution. Alternatively, split-Bregman iterations can be considered: we show in the following paragraph how to use such iterations for minimising the anisotropic TV (L^1) model.

3.4 L^1 Functional

The discontinuities being sparsely distributed in essence, it seems natural to rely on the sparsity enhancing properties of the L^1 norm [4,14]. Considering the choice $\Phi(.) = \|.\|_1$, we get from (13):

$$\mathcal{F}_{L^1}(u) = \iint_\Omega \|\nabla u(x,y) - \mathbf{g}(x,y)\|_1 + \frac{\lambda}{2}\left(u(x,y) - u_0(x,y)\right)^2 \,\mathrm{d}x\,\mathrm{d}y \tag{21}$$

This new functional is still convex, but cannot be minimised through differentiable optimisation. Split-Bregman iterations [9] can be considered:

$$u^{k+1} = \underset{u}{\operatorname{argmin}}\ \frac{\alpha}{2}\|\mathbf{d}^k - (\nabla u - \mathbf{g}) - \mathbf{b}^k\|_2^2 + \frac{\lambda}{2}\|u - u_0\|_2^2 \tag{22}$$

$$\mathbf{d}^{k+1} = \underset{\mathbf{d}}{\operatorname{argmin}}\ \|\mathbf{d}\|_1 + \frac{\alpha}{2}\|\mathbf{d} - (\nabla u^{k+1} - \mathbf{g}) - \mathbf{b}^k\|_2^2 \tag{23}$$

$$\mathbf{b}^{k+1} = \mathbf{b}^k + (\nabla u^{k+1} - \mathbf{g}) - \mathbf{d}^{k+1} \tag{24}$$

where $(\mathbf{d}^k, \mathbf{b}^k) = \left([d_1^k, d_2^k]^\top, [b_1^k, b_2^k]^\top\right)$ are auxiliary variables related to the Bregman distance at iteration k. We solve the discrete version of (22) using the same kind of discretisation as in Section 3.2. Yet, unlike in Section 3.2 and 3.3, it is preferable not to solve the problem exactly [9], so as to improve convergence properties of the split-Bregman iterations: as advised in the literature, we perform only a few (typically 5) Gauss-Seidel updates at each iteration k. Regarding the basis pursuit problem (23), solution is obtained by shrinkage:

$$\begin{cases} d_1^{k+1} = \dfrac{\partial_x u^{k+1} - p + b_1^k}{|\partial_x u^{k+1} - p + b_1^k|} \max\left\{\left|\partial_x u^{k+1} - p + b_1^k\right| - \dfrac{1}{\alpha}, 0\right\} \\ d_2^{k+1} = \dfrac{\partial_y u^{k+1} - q + b_2^k}{|\partial_y u^{k+1} - q + b_2^k|} \max\left\{\left|\partial_y u^{k+1} - q + b_2^k\right| - \dfrac{1}{\alpha}, 0\right\} \end{cases} \tag{25}$$

where $|.|$ is the absolute value.

We now experimentally compare the proposed schemes on synthetic data, and show results of the L^1 approach on real-world datasets.

4 Results

4.1 Synthetic Data

We first evaluate the performances of the proposed algorithms on synthetic datasets (Figure 2 and Table 1). In each test, a small Gaussian noise with zero-mean and standard deviation $\sigma = 0.5\%$ of $\|\mathbf{g}\|_\infty$ was added to the gradient field, before it was integrated using, respectively, least-squares [16,10], spectral regularisation [11], weighted least-squares ($\lambda = 10^{-5}, \gamma = 10$), isotropic TV ($\lambda = 10^{-5}, \theta = 10^{-3}$) and L^1 ($\lambda = 10^{-4}, \alpha = 0.1$). The convergence criterion for the iterative methods was set to a 5.10^{-4} relative residual between u^k and u^{k+1}. For fair comparison, the integration constant was changed *a posteriori* so as to minimise the RMSE between the estimated depth map and the ground truth. The performances of each algorithm are evaluated by Matlab codes running on a I7 laptop at 2.9 Ghz.

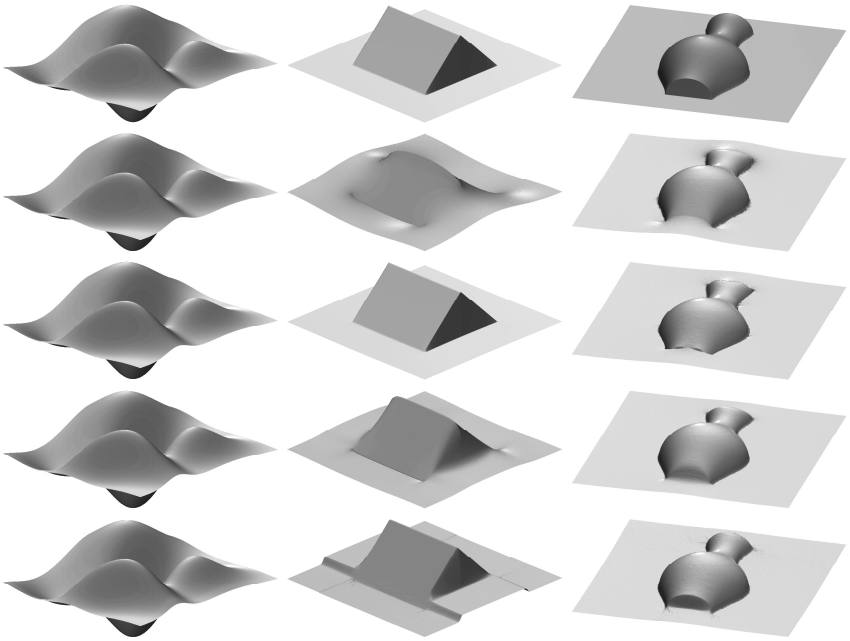

Fig. 2. Results on synthetic data. We show the ground truth surface (first row), the results using spectral regularisation [11] (second row), and those using the weighted least-squares (third row), TV (fourth row) and L^1 (fifth row) functionals. L^1 minimisation qualitatively offers the sharper edges, though a staircase effect appears. Weighted least squares provide accurate results for the "Canadian Tent", because the discontinuities correspond to very high integrability values, but it does not perform as well on the "Synthetic Vase", since a part of the discontinuity has null integrability.

	Peaks (512×512)	Canadian Tent (256×256)	Synthetic Vase (320×320)
Least-squares (DCT) [16]	0.30 (0.05 s)	10.76 (0.02 s)	4.55 (0.03 s)
Least-squares (Sylvester) [10]	0.14 (0.84 s)	10.76 (0.32 s)	4.56 (0.46 s)
Spectral regularisation [11]	**0.13** (0.22 s)	10.76 (0.12 s)	4.56 (0.15 s)
WLS (Cholesky)	0.16 (2.06 s)	**0.42** (0.55 s)	6.81 (0.93 s)
TV (reweighted least-squares)	0.15 (2.27 s)	4.91 (3.77 s)	3.15 (6.70 s)
L^1 (split-Bregman)	0.31 (1.82 s)	5.07 (12.85 s)	**2.89** (21.09 s)

Table 1. RMSE (in pixels) between the ground truth depth maps and those recovered using three state-of-the-art algorithms and our three new ones. The "Peaks" depth map being $\mathcal{C}^\infty$, all methods succeed at recovering accurate results for this dataset. Since we solve the normal equation (by means of Cholesky factorisation) in the WLS and TV approaches, additional smoothing is introduced, and thus these methods perform a little better than L^1 in this test.

4.2 Applications on Real Data

Photometric Stereo. The proposed split-Bregman scheme (Section 3.4) was applied to real-world gradient fields (the other proposed schemes provide comparable results), obtained by applying the photometric stereo technique [17] to the "Scholar"[2] and to the "Beethoven"[3] datasets (Figure 3). To emphasize the discontinuity-preserving properties of the scheme, as well as the staircasing effect appearing on large flat areas, we applied the method on the whole rectangular domain, rather than manually segmenting Ω.

It should be noted that a staircasing effect occurs on the background. This effect is well known and studied in the context of image denoising: adapting the total generalised variations schemes (TGV) [3], we could probably get rid of it.

Yet, staircasing seems to affect only the background, and should thus not be considered as a really damaging effect, since our method is able to deal with non-trivial integration domains (which is not the case in many algorithms [7,16,10,11]): staircase-free 3D-reconstructions can be obtained by manually segmenting the reconstruction domain (Figure 4).

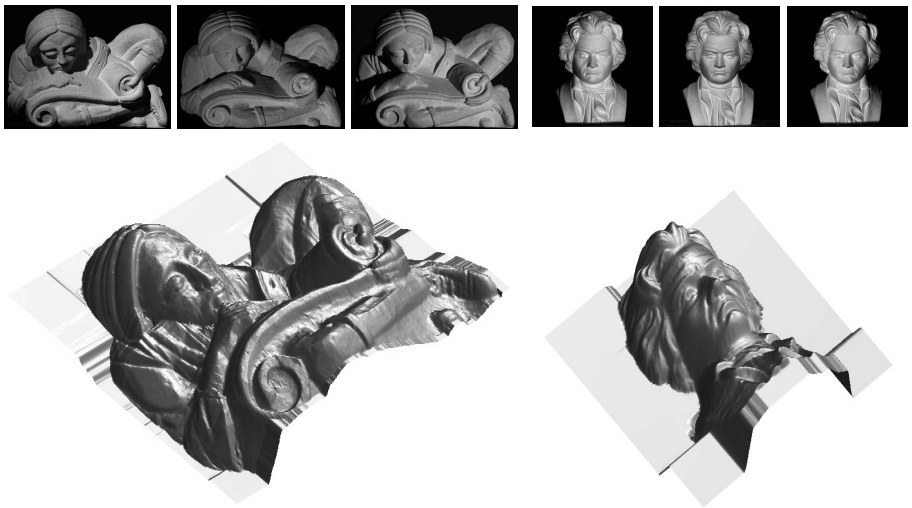

Fig. 3. Photometric stereo. A scene is captured from the same point of view, but under different lightings (top row), so that the normal field can be revealed using photometric stereo. By integrating this normal field (split-Bregman iterations), we obtain the surfaces on the bottom row. The staircase effect is clearly visible in these examples, though it only affects the background.

[2] `http://vision.seas.harvard.edu/qsfs/Data.html`
[3] `http://www.ece.ncsu.edu/imaging/Archives/ImageDataBase/Industrial/`

Surface Edition. In order to further illustrate the iterative normal integration through split-Bregman iterations, we consider a surface edition problem, consisting in inserting m small objects, whose gradient fields $\mathbf{g}^i$, $i = 1 \dots m$, are known over $\Omega^i \subset \Omega$, into a larger object represented by its gradient field $\mathbf{g}^0$ over Ω and the corresponding least-squares depth map u_0, while preserving thin details. Setting $\Omega^0 = \Omega \setminus \left(\cup_{i=1}^m \Omega^i\right)$, this reads as the minimisation of:

$$\mathcal{G}(u) = \sum_{i=0}^{m} \iint_{\Omega^i} \|\nabla u(x,y) - \mathbf{g}^i(x,y)\|_1 \,\mathrm{d}x\,\mathrm{d}y + \frac{\lambda}{2} \iint_{\Omega} (u(x,y) - u_0(x,y))^2 \,\mathrm{d}x\,\mathrm{d}y \tag{26}$$

which is an extension of the Poisson image editing problem [13]. Functionals (21) and (26) are the same, provided that $\mathbf{g} = \sum_{i=0}^{m} \chi_{\Omega^i} \mathbf{g}^i$, where χ_{Ω^i} is the characteristic function of Ω^i.

Now, we merge both the gradient fields $\mathbf{g}^0$ of the "Scholar" dataset (the reconstruction domain Ω is set to the non-rectangular domain of this dataset), and $\mathbf{g}^1$ of the "Beethoven" dataset, so as to replace a small area $\Omega^1 \subset \Omega$ of the reconstructed "Scholar" surface, by Beethoven's bust. In addition, we would like to remove some details inside another domain $\Omega^2 \subset \Omega$ (Figure 4). To this purpose, we choose $\mathbf{g}^2 = \mathbf{0}$: this will perform TV-"inpainting" inside Ω^2. Denoting $\Omega^0 = \Omega \setminus \left(\Omega^1 \cup \Omega^2\right)$, we form the gradient field $\mathbf{g} = \sum_{i=0}^{2} \chi_{\Omega^i} \mathbf{g}^i$, and apply the proposed split-Bregman scheme. As shown in Figure 4, a detail-preserving blending of the statues is obtained, while removing the details inside Ω^2.

Fig. 4. Surface edition. In the top-left figure, the colored part is Ω, the red part is Ω^0 ("Scholar" gradient field), the yellow one is Ω^1 ("Beethoven" gradient field), and the purple one is Ω^2 (inpainting area). We show the diffusion process at iterations 0, 10 (fine details begin to appear in the "hair"), 50 (the details inside the inpainted area disappear), 200 and 1000 (stable). Apart from the initialisation u_0 using the DCT solver [16], which required to add the background to Ω with null values of the gradient, the background was removed from the reconstruction domain: this considerably improves the boundaries of the surface, proving the importance of considering integration schemes able to deal with non trivial domains.

5 Conclusion and Perspectives

We studied weighted least-squares, TV and L^1 functionals in the context of normal field integration, and provided efficient numerics for minimising these functionals, through sparse Cholesky factorisation, reweighted least-squares and split-Bregman iterations, respectively. We experimentally showed that these functionals provide sharp depth maps, and demonstrated how to use them in the context of photometric stereo and surface edition. In future work, we plan to accelerate the iterative schemes by multigrid techniques [12], so as to allow real-time surface reconstruction and edition, and to more deeply study the staircase effect appearing with L^1 minimisation. We believe that introducing higher order regularisation terms, in the spirit of the total generalised variation regularisation (TGV) [3], will annihilate this effect, while providing a higher order of accuracy.

References

1. Agrawal, A., Raskar, R., Chellappa, R.: What is the range of surface reconstructions from a gradient field? In: ECCV (2006)
2. Badri, H., Yahia, H., Aboutajdine, D.: Robust surface reconstruction via triple sparsity. In: CVPR (2014)
3. Bredies, K., Kunisch, K., Pock, T.: Total generalized variation. SIIMS 3(3), 492–526 (2010)
4. Du, Z., Robles-Kelly, A., Lu, F.: Robust surface reconstruction from gradient field using the L1 norm. In: DICTA (2007)
5. Durou, J.D., Aujol, J.F., Courteille, F.: Integrating the normal field of a surface in the presence of discontinuities. In: EMMCVPR (2009)
6. Durou, J.D., Courteille, F.: Integration of a Normal Field without Boundary Condition. In: PACV (ICCV Workshops) (2007)
7. Frankot, R.T., Chellappa, R.: A Method for enforcing integrability in shape from shading algorithms. PAMI 10(4), 439–451 (1988)
8. Galliani, S., Breuß, M., Ju, Y.C.: Fast and robust surface normal integration by a discrete eikonal equation. In: BMVC (2012)
9. Goldstein, T., Osher, S.: The Split Bregman method for L1-regularized problems. SIIMS 2(2), 323–343 (2009)
10. Harker, M., O'Leary, P.: Least squares surface reconstruction from measured gradient fields. In: CVPR (2008)
11. Harker, M., O'Leary, P.: Regularized reconstruction of a surface from its measured gradient field. JMIV 51(1), 46–70 (2015)
12. Kimmel, R., Yavneh, I.: An algebraic multigrid approach for image analysis. SISC 24(4), 1218–1231 (2003)
13. Pérez, P., Gangnet, M., Blake, A.: Poisson image editing. In: SIGGRAPH (2003)
14. Reddy, D., Agrawal, A., Chellappa, R.: Enforcing integrability by error correction using L1-minimization. In: CVPR (2009)
15. Rudin, L.I., Osher, S., Fatemi, E.: Nonlinear total variation based noise removal algorithms. Physica D: Nonlinear Phenomena 60(1), 259–268 (1992)
16. Simchony, T., Chellappa, R., Shao, M.: Direct analytical methods for solving Poisson equations in computer vision problems. PAMI 12(5), 435–446 (1990)
17. Woodham, R.J.: Photometric method for determining surface orientation from multiple images. Optical Engineering 19(1), 139–144 (1980)