# Robust Image Filtering Using Joint Static and Dynamic Guidance

Bumsub Ham[1,*] Minsu Cho[1,*] Jean Ponce[2,*]

[1]Inria [2]École Normale Supérieure / PSL Research University

## Abstract

*Regularizing images under a guidance signal has been used in various tasks in computer vision and computational photography, particularly for noise reduction and joint upsampling. The aim is to transfer fine structures of guidance signals to input images, restoring noisy or altered structures. One of main drawbacks in such a data-dependent framework is that it does not handle differences in structure between guidance and input images. We address this problem by jointly leveraging structural information of guidance and input images. Image filtering is formulated as a nonconvex optimization problem, which is solved by the majorization-minimization algorithm. The proposed algorithm converges quickly while guaranteeing a local minimum. It effectively controls image structures at different scales and can handle a variety of types of data from different sensors. We demonstrate the flexibility and effectiveness of our model in several applications including depth super-resolution, scale-space filtering, texture removal, flash/non-flash denoising, and RGB/NIR denoising.*

## 1. Introduction and Background

Many tasks in computer vision and computational photography can be formulated as ill-posed inverse problems, and thus theoretically and practically, require regularization. In the classical setting, this is used to obtain a smoothly varying solution and/or ensure stability [4]. Recent work on joint regularization (or joint filtering) [10, 15, 33] provides a new perspective on the regularization process, with a great variety of applications including stereo correspondence [28, 34], optical flow [28], joint upsampling [8, 15, 20, 25], dehazing [10], noise reduction [27, 33], and texture removal [35]. The basic idea of joint regularization is that the structure of a guidance image is transferred to an input image, e.g., for preserving sharp structure transitions while smoothing the input image. It assumes that the guidance image has enough structural information to restore noisy or altered structures in the input image.

Joint regularization has been used with either static or dynamic guidance images. Static guidance regularization (e.g., [10]) provides an output image by modulating the input image with an affinity function that depends on the similarity of features in the guidance signal. This *static guidance* is fixed during the optimization. It can reflect internal properties of the input image itself, e.g., its gradient [15, 26], or be another signal aligned with the input image, e.g., a near infrared (NIR) image [33]. This framework determines the structure of the output image by referring to that of the guidance image only, and does not consider structural (or statistical) dependencies and inconsistencies between input and guidance images. This is problematic, especially in the case of data from different sensors, e.g., depth and color images. Dynamic guidance regularization (e.g., [35]) uses an affinity function obtained from the regularized input image. It is assumed that the affinity between neighboring pixels can be determined more accurately from already regularized images, than from the input image itself [2, 35]. This method is inherently iterative, and *dynamic guidance* (the regularized input image, i.e., a potential output image) is updated at every step. In contrast to static guidance regularization, dynamic guidance regularization does not use static guidance and takes into account of the properties of the input image. Data-dependent static guidance is needed to impose structures on the input image, especially when the input image is not enough in itself to pull out reliable information, e.g., joint upsampling [8, 15, 20, 25].

We present a unified framework for image filtering taking advantage of both static and dynamic guidances. We address the aforementioned problems by *fusing* appropriate structures of static and dynamic guidance images, rather than unilaterally *transferring* structures of guidance images to the input image. To encourage comparison and future work, our source code is available at our project webpage[1].

## 2. Related Work

Static or dynamic guidance can be implicit or explicit. Implicit regularization stems from a filtering framework. The input image is filtered using a weight function that depends on the similarity of features in the guidance image

[*] WILLOW project-team, Département d'Informatique de l'Ecole Normale Supérieure, ENS/Inria/CNRS UMR 8548.

[1] http://www.di.ens.fr/willow/research/sdfilter

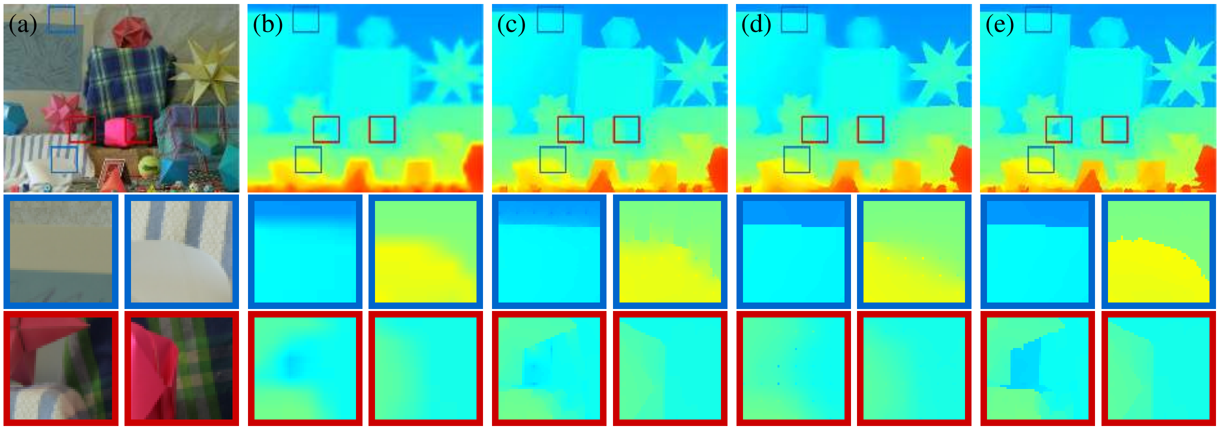


Figure 1. Comparison of static and dynamic guidance regularization methods. Given (a) a HR color image, (b) a LR depth map is upsampled (×8) by our model using (c) static guidance only, (d) dynamic guidance only, and (e) joint static and dynamic guidance. See Sec. 4.1 for details. (Best viewed in color.)

[15]. In this way, the structure of the guidance image is transferred to the input image. The bilateral filter (BF) [30], guided filter (GF) [10], and weighted median filter (WMF) [20] have been successfully adapted to static guidance regularization. Two representative filtering methods using dynamic guidance are iterative nonlocal means (INM) [2] and the rolling-guidance filter (RGF) [35]. They share the same filtering framework, but differ in that INM is for preserving textures during noise reduction, while the RGF aims at removing textures through scale-space filtering. This implicit regularization is simple and easy to implement, but the filtering formalization prevents its wide applicability. For example, it is hard to handle input images where information is sparse, e.g., in image colorization [18]. The local nature of this approach might introduce artifacts, e.g., halos and gradient reversal [10]. Accordingly, implicit regularization has been applied in a highly controlled condition, and usually employed as a pre- and/or post-processing for further applications [17, 20]. An alternative approach is to explicitly encode the regularization process into an objective functional, while taking advantage of a guidance signal. The objective functional typically consists of two parts: A fidelity term describes the consistency between input and output images, and a regularization term encourages the output image to have a similar structure to the guidance image. The weighted least-squares (WLS) framework [7] is the most popular explicit regularization method that has been used in static guidance regularization [25]. The regularization term is modeled as a weighted $l_2$ norm. Anisotropic diffusion (AD) [26] is an explicit regularization framework using dynamic guidance. In contrast to INM [2] and the RGF [35], AD updates both input and guidance images at every step; The regularization is performed iteratively with regularized input and updated guidance images. This explicit regularization enables formulating a task-specific model, with more flexibility than using implicit regularization. Furthermore, this type of regularization overcomes several limitations of implicit regularization, such as halos and gradient reversal, at the cost of global intensity shifting [7, 10].

Existing regularization methods typically apply to a limited range of applications and suffer from various artifacts: For example, the RGF is applicable to scale-space filtering only, and suffers from poor edge localization [35]. In contrast, our approach provides a unified model for many applications, gracefully handles most of these artifacts, and outperforms the state of the art in all the cases considered in the paper. Although the proposed model may look similar to WLS [7] and the RGF [35], our nonconvex objective function needs a different solver. Contrary to iteratively reweighted least-squares (IRLS) [5], we do not split a nonconvex regularizer but approximate the objective function by a surrogate (upper-bound) function.

# 3. Proposed Approach

## 3.1. Motivation and Problem Statement

There are pros and cons in regularizing images under static or dynamic guidance. Let us suppose the example of depth super-resolution, where a high-resolution (HR) color image (the guidance image) of Fig. 1 (a) is used to upsample (×8) a low-resolution (LR) depth map (the input image) of Fig. 1 (b). Regularization with static guidance reconstructs the destroyed depth edges by using the color image with high signal-to-noise ratio (SNR) [8, 15], as in the red boxes of Fig. 1 (c). However, this method has difficulties with handling differences in structure between depth and color images, transferring all the structural information of the color image to the depth map, as in the blue boxes of Fig. 1 (c). For regions of high contrast in the color image, e.g., textures, the depth is altered according to the texture pattern [3]. Similarly, the gradient of the depth map becomes similar to that of the color image for regions of low contrast in the color image, e.g., weak edges. This smoothes depth edges, and causes jagged artifacts [21]. Regularization with dynamic guidance utilizes the contents of the depth maps[2], avoiding the problems of static guidance regularization, as in the blue boxes of Fig. 1 (d). The depth edges are preserved, and unwanted structures are not transferred. A limitation is that dynamic guidance only does not utilize the abundant structural information that exists in the color image. Thus, depth edges are smoothed, and even eliminated for regions of low contrast in the depth map, as in the red boxes of Fig. 1 (d). This example illustrates the fact that static and dynamic guidance complement each other, and exploiting only one of them is not sufficient to infer high quality structural information from the input image. This problem becomes even worse when input and guidance images come from various types of data and have different statistical characteristics. Our model jointly leverages the structures of static (color image) and dynamic (depth map) guidance, taking advantage of both of them, as shown in Fig. 1 (e).

### 3.2. Model

Given the input image $f$, static guidance $g$, and the output image $u$ itself (dynamic guidance), we denote by $f_i$, $g_i$, and $u_i$; the corresponding image values at pixel $i$, with $i$ ranging over the image domain $\mathcal{I} \subset \mathbb{N}^2$. Our objective is to infer the structure of the input image by jointly using static and dynamic guidance. The influence of the guidance images on the input image varies spatially, and is controlled by affinity functions that measure similarities between adjacent vertices. Various features (e.g., spatial location, intensity, and textures [13, 25]) and metrics (e.g., Euclidian and geodesic distances [7, 19]) can be utilized to represent distinctive characteristics of vertices on images, and measure their similarities.

We minimize an objective function of the form:

$$\mathcal{E}(u) = \sum_i c_i (u_i - f_i)^2 + \lambda \Omega(u). \tag{1}$$

It consists of fidelity and regularization terms, balanced by the regularization parameter $\lambda$. The fidelity term helps the solution $u$ to harmonize well with the input image $f$ with confidence $c_i \geq 0$. The regularization term smoothes the solution $u$, and makes it have structures similar to static and/or dynamic guidance, $g$ and $u$. In static guidance regularization, $\Omega(u) = \sum_{i,j} \phi_\mu(g_i - g_j)(u_i - u_j)^2$ where $\phi_\mu(x) = \exp(-\mu x^2)$, and $\mu$ controls the smoothness bandwidth. In a purely equivalent dynamic guidance setting, one would take $\Omega(u) = \sum_{i,j} \phi_\nu(u_i - u_j)(u_i - u_j)^2$ for some $\nu$, and in a mixed setting: $\Omega(u) = \sum_{i,j} \phi_\mu(g_i - g_j)\phi_\nu(u_i - u_j)(u_i - u_j)^2$. These regularizers may be hard to optimize and be unstable. Instead, we choose

$$\Omega(u) = \sum_{i,j \in \mathcal{N}} \phi_\mu(g_i - g_j)\psi_\nu(u_i - u_j), \tag{2}$$

where $\psi_\nu(x) = (1 - \phi_\nu(x))/\nu$, i.e., Welsch's function [11], and $\mathcal{N}$ is in our implementation, the set of image adjacencies, defined in a local 8-neighborhood system.

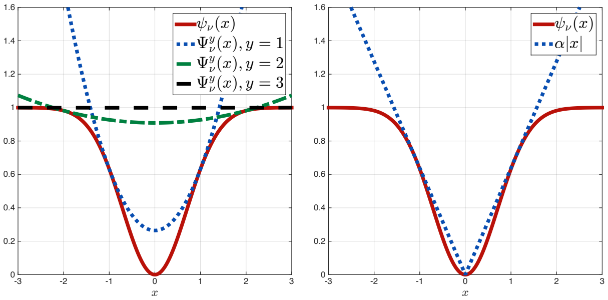


Figure 2. A nonconvex regularizer $\psi_\nu(x)$, its surrogate functions $\Psi_\nu^y(x)$ [left], and a $l_1$ regularizer $\alpha|x|$ (a tight upper bound function of $\psi_\nu(x)$) [right] when $\nu = 1$ and $\alpha = 0.6382$. (Best viewed in color.)

**Nonconvex Regularizer.** As shown in Fig. 2, $\psi_\nu(x)$ is a nonconvex regularizer: Welsch's function acts as a robust regularizer [11], and thus our objective function makes joint filtering robust to outliers. Inverse diffusion occurs when $\psi'_\nu(x)$ decreases, which enhances features having high-frequency structures (e.g., edges and corners) during regularization [6, 9, 26]. It may enhance noise as well, but the static guidance image with high SNR in our model avoids this problem. The combination of a fixed weight function and a nonconvex regularizer is common in variational approaches [37], and they are typically referred to as image- and flow-driven regularizers, respectively. Our model differs from image- and flow-driven regularization in its new regularizer and solver. It is also more flexible (e.g., we can handle sparse data of different resolutions). This leads to new applications such as joint upsampling. Note that most image- and flow-driven regularization techniques have been applied to optical flow only. Finally, we are not aware of existing *joint image filters* using a nonconvex regularizer.

### 3.3. Solver

**Optimization.** Let $\mathbf{f} = [f_i]_{N\times1}$, $\mathbf{g} = [g_i]_{N\times1}$, and $\mathbf{u} = [u_i]_{N\times1}$ denote vectors representing the input image, static guidance and the output image (or dynamic guidance), respectively, where $N = |\mathcal{I}|$ is the size of images. Let $\mathcal{W}_g = [\phi_\mu(g_i - g_j)]_{N\times N}$, $\mathcal{W}_u = [\phi_\nu(u_i - u_j)]_{N\times N}$, and $\mathcal{C} = diag\left([c_1, \ldots, c_N]\right)$. We can rewrite our objective function in matrix/vector form as:

$$\mathcal{E}(\mathbf{u}) = (\mathbf{u} - \mathbf{f})^T \mathcal{C}\,(\mathbf{u} - \mathbf{f}) + \frac{\lambda}{\nu} \mathbb{1}^T \left(\mathcal{W}_g - \mathcal{W}\right) \mathbb{1}, \tag{3}$$

where $\mathcal{W} = \mathcal{W}_g \circ \mathcal{W}_u$, and $\circ$ denotes the Hadamard product of the matrices. $\mathbb{1}$ is a $N \times 1$ vector, where all the

[2] Dynamic guidance is initially set to the upsampled depth map obtained from static guidance regularization as shown in Fig. 1 (c), not the LR depth map itself.

entries are 1. The diagonal entries $c_i$ of $\mathcal{C}$ are confidence values for the pixels $i$ of the input image. Minimizing $\mathcal{E}$ is a nonconvex optimization problem, which can be solved by the majorization-minimization algorithm (Fig. 3) as follows [16, 22, 36]:

1. **Majorization Step:** Construct a surrogate function $\mathcal{Q}^k(\mathbf{u})$ of $\mathcal{E}(\mathbf{u})$ such that

$$\begin{cases} \mathcal{E}(\mathbf{u}) \leq \mathcal{Q}^k(\mathbf{u}), \text{for all } \mathbf{u}, \mathbf{u}^k \in \Theta \\ \mathcal{E}(\mathbf{u}^k) = \mathcal{Q}^k(\mathbf{u}^k), \text{for all } \mathbf{u}^k \in \Theta \end{cases}, \tag{4}$$

where $\Theta \subset [0,1]^N$[3]. The nonconvexity in our objective function comes from the regularizer $\psi_\nu(x)$ in (2), which has a convex surrogate function $\Psi_\nu^y(x)$ defined by (see the supplementary material):

$$\Psi_\nu^y(x) = \psi_\nu(y) + (x^2 - y^2)(1 - \nu\psi_\nu(y)), \tag{5}$$

that is, the curve $x \mapsto \Psi_\nu^y(x)$ lies above the curve $\psi_\nu(x)$ and is tangent to it at the point $x = y$ [12], as shown in Fig. 2. The surrogate objective function $\mathcal{Q}^k(\mathbf{u})$ can then be found using (5) as follows:

$$\begin{aligned} \mathcal{Q}^k(\mathbf{u}) = \mathbf{u}^T \left[\mathcal{C} + \lambda\mathcal{L}^k\right] \mathbf{u} - 2\mathbf{f}^T\mathcal{C}\mathbf{u} + \mathbf{f}^T\mathcal{C}\mathbf{f} \\ - \lambda\mathbf{u}^{kT}\mathcal{L}^k\mathbf{u}^k + \frac{\lambda}{\nu}\mathbb{1}^T\left(\mathcal{W}_g - \mathcal{W}^k\right)\mathbb{1}. \end{aligned} \tag{6}$$

$\mathcal{L}^k = \mathcal{D}^k - \mathcal{W}^k$ is a *dynamic* Laplacian matrix at step $k$, where $\mathcal{W}^k = \mathcal{W}_g \circ \mathcal{W}_{u^k}$ and $\mathcal{D}^k = diag\left(\left[d_1^k, \ldots, d_N^k\right]\right)$ where $d_i^k = \sum_{j=1}^N \phi_\mu(g_i - g_j)\phi_\nu(u_i^k - u_j^k)$. Note that the affinity matrix of static guidance is fixed regardless of steps, and that of dynamic guidance is iteratively updated.

2. **Minimization Step:** Obtain the next estimate $\mathbf{u}^{k+1}$ by minimizing the surrogate function $\mathcal{Q}^k(\mathbf{u})$ w.r.t. $\mathbf{u}$ as follows[4]:

$$\mathbf{u}^{k+1} = \underset{\mathbf{u}\in\Theta}{\arg\min}\, \mathcal{Q}^k(\mathbf{u}) = (\mathcal{C} + \lambda\mathcal{L}^k)^{-1}\mathcal{C}\mathbf{f}. \tag{7}$$

The above iterative scheme decreases the value of $\mathcal{E}(\mathbf{u})$ monotonically in each step, i.e.,

$$\mathcal{E}(\mathbf{u}^{k+1}) \leq \mathcal{Q}^k(\mathbf{u}^{k+1}) \leq \mathcal{Q}^k(\mathbf{u}^k) = \mathcal{E}(\mathbf{u}^k), \tag{8}$$

where the first and the second inequalities follow from (4) and (7), respectively, and it can be shown to converge to a local minimum of $\mathcal{E}$ [31].

**Initialization.** Our solver finds a local minimum, and thus different initializations for $\mathbf{u}^0$ (dynamic guidance at $k = 0$) may give different solutions. In our work, two initializations are used: The initial solution $\mathbf{u}^0$ can be set to a constant vector, e.g., $\mathbf{u}^0 = \mathbb{1}$. Note that the constant initialization, regardless of its value, makes $\mathcal{W}^0 = \mathcal{W}_g$. This initialization is simple, but shows a slow convergence rate as shown in Fig. 4 (a). A good initial solution accelerates the convergence of our solver. We propose to use the following regularizer to compute the initial solution $\mathbf{u}^0$:

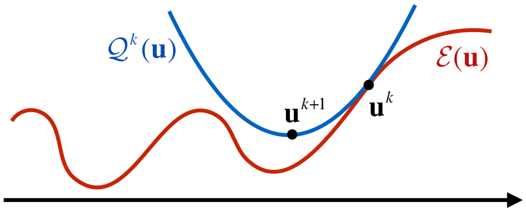


Figure 3. Sketch of the majorization-minimization algorithm. Given some estimate $\mathbf{u}^k$ of the minimum of $\mathcal{E}$, a surrogate function $\mathcal{Q}^k(\mathbf{u})$ is constructed. The next estimate $\mathbf{u}^{k+1}$ is then computed by minimizing $\mathcal{Q}^k$

$$\Omega_{l_1}(u) = \sum_{i,j\in\mathcal{N}} \phi_\mu(g_i - g_j)\alpha|u_i - u_j|, \tag{9}$$

where $\alpha$ is set to a positive constant, chosen so $\alpha|x|$ is a tight upper bound of $\psi_\nu(x)$ (Fig. 2). This regularizer is convex, and the global minimum of (9) is guaranteed.

### 3.4. Properties

**Convergence.** We show the convergence rate of (7) as $k$ increases, and observe its behavior with different initializations ($\mathbf{u}^0 = \mathbb{1}$ and $\mathbf{u}^0 = \mathbf{u}_{l_1}$, a global minimum of (1) using $\Omega = \Omega_{l_1}$). Figure 4 shows how (a) the energy and (b) the intensity differences (i.e., $\|\mathbf{u}^k - \mathbf{u}^{k+1}\|_1$) evolve at each step given the input image in (c). Our solver converges in fewer steps with the $l_1$ initialization ($\mathbf{u}^0 = \mathbf{u}_{l_1}$) than with the constant one ($\mathbf{u}^0 = \mathbb{1}$), with faster overall speed, despite the overhead of the $l_1$ minimization. On this example, our solver with the constant and $l_1$ initializations converges in 30 and 7 steps (Fig. 4 (d) and (e)), each of which takes 45 and 20 seconds, respectively. Although our solver with $\mathbf{u}^0 = \mathbb{1}$ converges more slowly, the per-pixel intensity difference decreases monotonically, and 5 steps are typically enough to get satisfactory results in both cases[5]. It should be noted that most filtering methods, except the recently proposed RGF [35], if they are applied repeatedly, eventually converges to a trivial solution, i.e., a constant signal [35], regardless of whether they have implicit or explicit forms (e.g., BF [30] and AD [26]). In contrast, repeatedly solving the linear system of (7) still gives a meaningful solution in the steady-state.

**Scale Adjustment.** There are two approaches to incorporating scale information in image regularization [7, 26, 35].

[3] A range of intensity values is normalized such that they exist between 0 and 1.

[4] In case of a color image, the linear system is solved in each channel.

[5] After 5 steps, an average (maximum) value of the per-pixel intensity difference is $9.4\times10^{-5}$ ($1.7\times10^{-3}$) with $\mathbf{u}^0 = \mathbb{1}$ and $4.3\times10^{-5}$ ($8.7\times10^{-4}$) with $\mathbf{u}^0 = \mathbf{u}_{l_1}$. Current un-optimized MATLAB implementation on 2.5 GHz CPU takes about 9 seconds ($\mathbf{u}^0 = \mathbb{1}$) and 16 seconds ($\mathbf{u}^0 = \mathbf{u}_{l_1}$) to filter an image of size 500 × 400 with a 8-neighborhood system and $k = 5$.

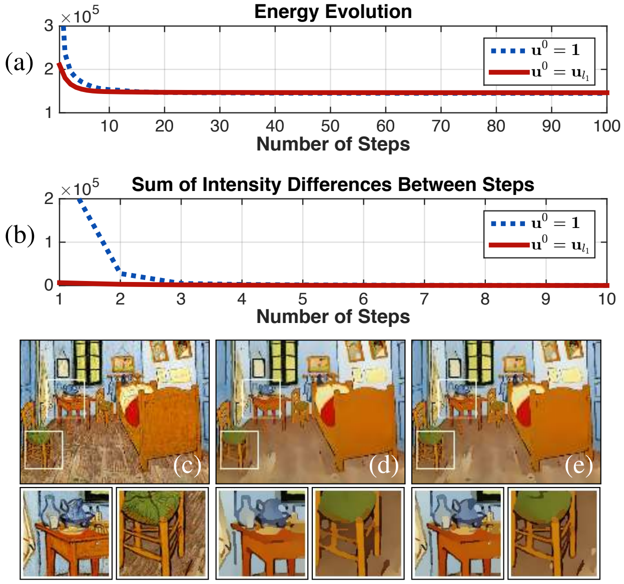


Figure 4. An example of (a) energy evolution and (b) a sum of intensity difference between successive steps, i.e., $\|\mathbf{u}^k - \mathbf{u}^{k+1}\|_1$, given (c) the input image. Our model monotonically converges, and guarantees a meaningful solution in the steady-state: (d) $\mathbf{u}^0 = \mathbb{1}$, $k = 30$, and (e) $\mathbf{u}^0 = \mathbf{u}_{l_1}$, $k = 7$. In this example, for removing textures, $\mathbf{g}$ is set to the Gaussian filtered version (standard deviation, 1) of the input image [$\lambda = 50$, $\mu = 5$, $\nu = 40$]. See Sec. 4.2 for details.

In filtering methods, an intuitive way to adjust a degree of smoothing is to *explicitly* use an isotropic Gaussian kernel. Due to the space-invariant property of the kernel, it regularizes both noise and features evenly regardless of the local structure [26]. The RGF addresses this problem in two phases: Small structures are removed by the Gaussian kernel, and large structures are then recovered [35]. Since RGF is based on the Gaussian kernel, it inherits its limitations; This leads to poor localization at coarse scales, which causes corners to be rounded and boundaries to be shifted. The regularization parameter is empirically used to adjust scales in explicit regularization methods [7]. It balances the degree of influence of fidelity and regularization terms in such a way that a large value leads to more regularized results than a small one. Now, we will show how the regularization parameter controls scales in our case, and how it relates to the standard deviation of the Gaussian kernel. Let $\mathbf{u}^{k+1} \to \mathbf{u}^\star$, $\mathcal{D}^k \to \mathcal{D}^\star$ and $\mathcal{W}^k \to \mathcal{W}^\star$ as $k \to \infty$. Then, it follow from (7) that

$$(\mathcal{C} + \lambda\mathcal{D}^\star)\mathbf{u}^\star - \lambda\mathcal{W}^\star\mathbf{u}^\star = \mathcal{C}\mathbf{f}. \tag{10}$$

Let us define diagonal matrices $\mathcal{A}$ and $\mathcal{A}'$ as

$$\mathcal{A} = \lambda(\mathcal{C} + \lambda\mathcal{D}^\star)^{-1}\mathcal{D}^\star, \tag{11}$$

and

$$\mathcal{A}' = (\mathcal{C} + \lambda\mathcal{D}^\star)^{-1}\mathcal{C}, \tag{12}$$

such that $\mathcal{A} + \mathcal{A}' = \mathbf{I}$. By multiplying the left- and right-hand sides of (10) by $(\mathcal{C} + \lambda\mathcal{D}^\star)^{-1}$, we obtain

$$\begin{aligned}\mathbf{u}^\star &= (\mathbf{I} - \underbrace{\lambda(\mathcal{C} + \lambda\mathcal{D}^\star)^{-1}\mathcal{D}^\star}_{\mathcal{A}}\,\mathcal{P})^{-1}\,\underbrace{(\mathcal{C} + \lambda\mathcal{D}^\star)^{-1}\mathcal{C}}_{\mathcal{A}'=\mathbf{I}-\mathcal{A}}\,\mathbf{f} \qquad (13)\\ &= (\mathbf{I} - \mathcal{A})(\mathbf{I} - \mathcal{A}\mathcal{P})^{-1}\mathbf{f} = \mathcal{S}\mathbf{f},\end{aligned}$$

where $\mathcal{P} = \mathcal{D}^{\star-1}\mathcal{W}^\star$ and

$$\mathcal{S} = (\mathbf{I} - \mathcal{A})(\mathbf{I} - \mathcal{A}\mathcal{P})^{-1} = (\mathbf{I} - \mathcal{A})\sum\nolimits_{n=0}^{\infty} \mathcal{A}^n\mathcal{P}^n. \tag{14}$$

That is, $\mathcal{S}$ is defined as a weighted average of all matrices $\mathcal{P}^n$, $n = 0, ..., \infty$. Note that $\mathcal{P}^n$ is the $n^{th}$ order transition probability of the random walker, and its element $p_{ij}^n$ represents the probability that the random walker at the vertex $j$ arrives at the vertex $i$ after $n$ time transitions [24]. As $n$ increases, the random walker can travel far away, and we expect to see coarser structures. Thus, $\sum_{n=0}^{\infty} \mathcal{A}^n\mathcal{P}^n$ considers all paths between two vertices at all scales ($n = 0, ..., \infty$), and each $\mathcal{P}^n$ is modulated by the weight $\mathcal{A}^n$ that is controlled by the regularization parameter $\lambda$ as in (11). By increasing $\lambda$, the random walker can travel to a distant vertex more easily. This indicates that the regularization parameter has a similar role to the standard deviation in the Gaussian kernel.

# 4. Applications

Our model is applied to depth super-resolution, scale-space filtering, texture removal, flash/non-flash denoising, and RGB/NIR denoising. Additional results are available in the supplementary material.

## 4.1. Depth Super-Resolution

**Parameter Settings.** In our model, input and guidance images, $\mathbf{f}$ and $\mathbf{g}$, are set to sparse depth and HR images, respectively, where $c_i = 1$ if the pixel $i$ of the sparse depth map $\mathbf{f}$ has valid data, and otherwise, 0. The constant initialization is used, and the bandwidths and the step index are fixed to all experiments ($\mathbf{u}^0 = \mathbb{1}$, $\mu = 60$, $\nu = 30$, and $k = 10$). The regularization parameter $\lambda$ is set to 0.1 for synthetic examples, and set to 5 for real-world examples due to huge amounts of noise. Other results for the comparison have been obtained from source codes provided by the authors, and all the parameters have been carefully set through intensive experiments for the best performance. For the quantitative comparison, the bad matching errors (BMEs) for all regions and regions near depth discontinuities are measured as $\mathcal{O}_{all} = \sum\left(\left|u_i^\star - u_i^{gt}\right| > \delta\right)/N$ and $\mathcal{O}_{disc} = \sum\left(m_i\left|u_i^\star - u_i^{gt}\right| > \delta\right)/M$, respectively, where $\delta$ is a depth error tolerance [29]. $u_i^\star \in \mathbf{u}^\star$ and $u_i^{gt} \in \mathbf{u}^{gt}$ represent estimated and ground truth depth maps, respectively. $\mathbf{m}$ is a binary mask where $m_i = 1$ if the pixel $i$ belongs to the regions near depth discontinuities, and otherwise, 0, and $M = \|\mathbf{m}\|_1$.

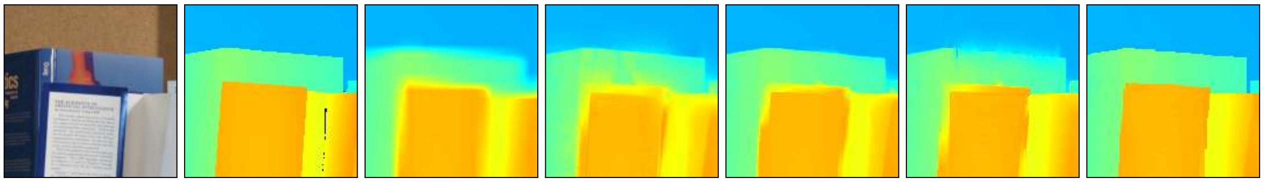

(a) HR image (b) Ground truth (c) Bilinear Int. (d) GF [10] (e) WMF [20] (f) TGV [8] (g) Ours

Figure 5. Visual comparison of upsampled depth maps on a snippet of the *books* sequence in the Middlebury test bed [29]. In contrast to static guidance regularization such as (d) GF [10] and (e) WMF [20], (g) joint static and dynamic guidance model interpolates LR depth maps by considering structures of color and depth images both, preserving sharp depth transitions.

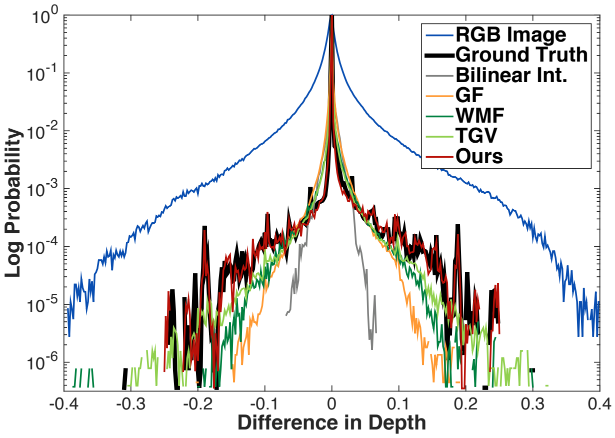


Figure 6. Log 10 of the normalized histograms of relative depth and intensity values (gradients along x- and y-axis) from the Middlebury test bed [29]. (Best viewed in color.)

**Synthetic Examples.** We have synthesized the LR depth map by (×8) downsampling a ground truth image from the Middlebury benchmark data set [29]: *Tsukuba*, *Venus*, *Teddy*, *Cones*, *Art*, *Books*, *Dolls*, *Laundry*, *Moebius*, and *Reindeer*, and used the corresponding color image as the HR intensity image. Table 1 summarizes average BMEs with the tolerance $\delta = 1$. $\mathcal{O}_{all}$ has been measured only, except when the ground truth index map of discontinuous regions $\mathbf{m}$ is available. Our model outperforms other regularization methods, especially around depth discontinuities. Figure 5 gives quantitative results, and clearly shows the different behavior between static guidance and joint static and dynamic guidance models. For example, the gradient of the depth map becomes similar to that of the color image in static guidance methods [8, 10, 20], which tends to eliminate or smooth depth boundaries, and causes jagged artifacts. This can be further verified by observing statistical distributions of upsampled depth maps as shown in Fig. 6. Table. 2 compares average BMEs and processing time of the constant and $l_1$ initializations[6] by varying the number of steps. This table shows that 1) our solver with the $l_1$ initialization convergences faster than that with the constant one. For example, the $l_1$ and constant initializations converge with 5 and 30 steps, respectively, 2) both initializations give almost the same error at the convergence, and 3) the $l_1$ initialization takes less time than the constant initialization for the convergence.

Table 1. Average BMEs of Upsampled Depth Maps on the Middlebury Test Bed [29]

| $\mathbf{u}^0 = \mathbb{1}$ | $\mathcal{O}_{all} \pm std.$ | $\mathcal{O}_{disc} \pm std.$ |
|---|---|---|
| Bilinear Int. | 15.98±8.29 | 38.63±5.17 |
| GF [10] | 19.85±11.2 | 35.40±6.96 |
| Park et al. [25] | 14.81±5.97 | 22.65±3.89 |
| TGV [8] | 12.34±6.40 | 22.73±6.08 |
| WMF [20] | 9.84±5.48 | 19.88±7.47 |
| Ours | **7.08±3.42** | **12.05±6.30** |

Table 2. Quantitative Comparison of Upsampled Depth Maps from Constant and $l_1$ Initializations on the Middlebury Test Bed [29]

| | $\mathbf{u}^0 = \mathbb{1}$ | | $\mathbf{u}^0 = \mathbf{u}_{l_1}$ | |
|---|---|---|---|---|
| $k$ | $\mathcal{O}_{all} \pm std.$ | *time (s)* | $\mathcal{O}_{all} \pm std.$ | *time (s)* |
| 1 | 10.05±4.76 | 0.60 | 7.55±3.54 | 4.78 |
| 3 | 7.60±3.64 | 1.44 | 7.14±3.37 | 5.70 |
| 5 | 7.23±3.52 | 2.33 | **7.07±3.36** | **6.61** |
| 10 | 7.08±3.42 | 4.39 | 7.07±3.37 | 8.68 |
| 20 | 6.68±3.86 | 8.48 | 7.07±3.38 | 12.71 |
| 30 | **7.05±3.41** | **12.47** | 7.08±3.38 | 16.67 |
| 40 | 7.05±3.41 | 16.88 | 7.07±3.38 | 20.93 |
| 50 | 7.05±3.41 | 21.06 | 7.07±3.38 | 25.39 |

Table 3. BMEs of Upsampled Depth Maps on the Graz Data Set [8]

| $\mathbf{u}^0 = \mathbb{1}$ | *Books* | *Devil* | *Shark* | $\mathcal{O}_{all} \pm std.$ |
|---|---|---|---|---|
| Bilinear Int. | 16.21 | 13.68 | 17.60 | 15.83±1.99 |
| GF [10] | 19.65 | 13.12 | 20.68 | 17.82±4.10 |
| TGV [8] | 11.83 | 9.70 | 13.98 | 11.84±2.14 |
| WMF [20] | 13.33 | 9.81 | 15.77 | 12.97±3.00 |
| Ours | **9.91** | **8.09** | **12.71** | **10.24±2.33** |

**Real-World Examples.** Recently, Ferstl *et al.* [8] have introduced a benchmark data set where they provide both LR depth maps captured by ToF sensor and highly accurate ground truth depth maps acquired from using structured light. We have performed a quantitative evaluation using this data set [8] in Table 3. The BMEs are computed by setting the error tolerance to 5 % of a pre-defined depth range (0∼255). This experiment demonstrates that the proposed method outperforms the state of the art.

## 4.2. Scale-Space Filtering and Texture Removal

**Parameter Settings.** For scale-space filtering, the input image $\mathbf{f}$ is guided by itself ($\mathbf{g} = \mathbf{f}$). In texture removal, the

[6] An average BME of $\mathbf{u}_{l_1}$ itself is 12.43.

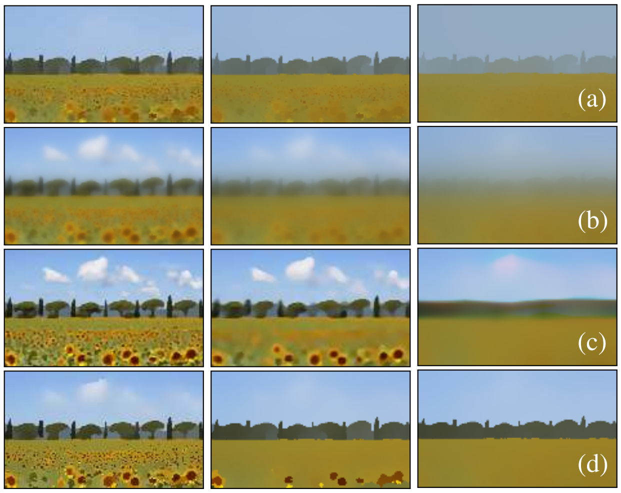


Figure 7. Examples of the scale-space representation obtained by (a) WLS [7] [(from left to right) $\lambda = 5 \times 10^3, 3 \times 10^4, 2 \times 10^5$, $\mu = 40$], (b) WLS [7] [(from left to right) $\lambda = 50, 300, 2000$, $\mu = 5$], (c) RGF [35] [(from left to right) $\sigma_s = 4, 10, 50$, $\sigma_r = 0.05$, $k = 5$], (d) our model [$\mathbf{u}^0 = \mathbf{u}_{l_1}$, (from left to right) $\lambda = 200, 1200, 3000$].

static guidance image is set to a Gaussian-filtered version of the input image, $\mathbf{g} = \mathbf{G}_\sigma \mathbf{f}$ where $\mathbf{G}_\sigma$ is the Gaussian kernel with standard deviation $\sigma$. The regularization parameter $\lambda$ and $\sigma$ vary according to the scale. The bandwidths and the step index are fixed to all experiments ($\mu = 5$, $\nu = 40$, and $k = 5$) in both applications.

**Scale-Space Filtering.** A scale-space representation can be obtained by repeatedly applying the regularization method while varying the regularization parameter $\lambda$. Figure 7 shows examples of the scale-space constructed by (a) and (b) WLS [7], (c) RGF [35], and (d) our model. The WLS [7], a representative of static guidance regularization, alters the scale of structures by varying the regularization parameter. It suffers from global intensity shifting [10] (Fig. 7(a)) or does not preserve structural information at coarse scales (Fig. 7(b)). This could be alleviated by dynamic guidance regularization as in the RGF [35]. However, the RGF does not use the structure of the input image, and the scale is controlled by isotropic Gaussian kernels, which leads to poor boundary localization at coarse scales (Fig. 7(c)). In contrast, our model uses the structures of input and desired output images, and the scale depends on the regularization parameter, providing well localized boundaries even at coarse scales. Moreover, it is robust to global intensity shifting (Fig. 7(d)). The scale-space representation meets two criteria: *causality* and *immediate localization*. Causality means that any feature at a coarse scale must possess a cause at a finer scale [26]. Immediate localization means that object boundaries should be sharp and coincide well with the meaningful boundaries at each scale. We have empirically found that our model meets the causality condition, $\min\{u^\star_{j,\lambda}\} \leq u^\star_{i,\lambda+\tau} \leq \max\{u^\star_{j,\lambda}\}$ where $\tau > 0$, and $u^\star_{i,\lambda} \in \mathbf{u}^\star_\lambda$ is the steady-state solution for $\lambda$. The accuracy of boundary localization is evaluated on the BSDS300 [23].

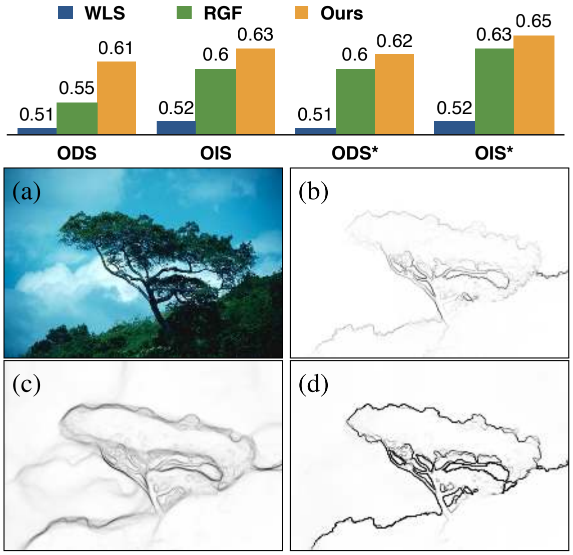
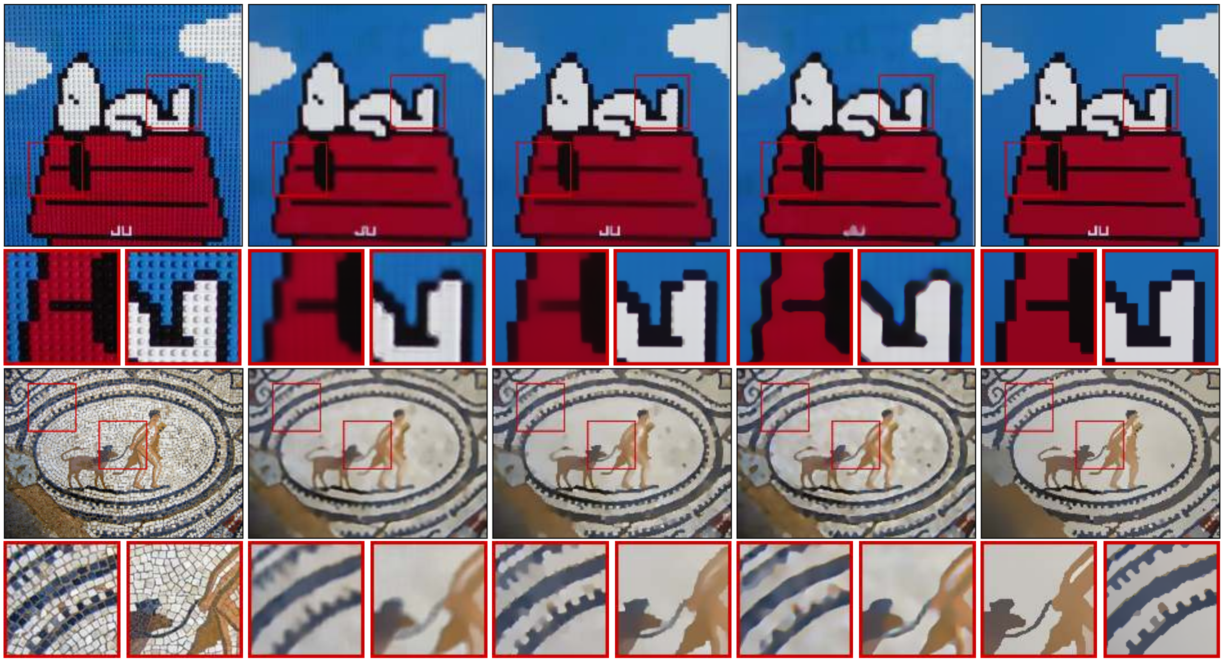


Figure 8. Evaluation of edge localization on the BSDS300 [23] (top), and examples of the gradient magnitude averaged over the scale-space (bottom). Given (a) input image, the scale-space is constructed by (b) WLS [7] [$\mu = 40$], (c) RGF [35] [$\sigma_r = 0.05$, $k = 5$], and (d) our model [$\mathbf{u}^0 = \mathbb{1}$], by varying scale parameters, i.e., $\lambda \leq 2 \times 10^5$ in WLS [7], $\sigma_s \leq 50$ in RGF [35], and $\lambda \leq 2 \times 10^3$ in our model.

For all images in the data set, average ODS and OIS [1] are measured by using gradient magnitudes of regularized images, as shown in Fig. 8. These images are obtained by varying scale parameters, i.e., $\lambda$ in WLS [7] and our model, and $\sigma_s$ in RGF [35]. ODS is the F-measure at a fixed contour threshold across the entire data set, while OIS refers to the per-image best F-measure. In the histograms of Fig. 8, average ODS (OIS) is evaluated with gradient images, each of which is averaged over the scale-space, e.g., Fig. 8 (d). Average ODS$^*$ (OIS$^*$) is evaluated with gradient images at the fixed scale that provides maximum ODS (OIS) for each image. In both cases, our model outperforms other regularization methods, showing sharper boundary transitions.

**Texture Removal.** For removing textures while maintaining other high-frequency structures, we need a guidance image that does not have textures, but contains large structures, e.g., edges. Since it is hard to get such an image, we set the static guidance image to a Gaussian-filtered version of the original image $\mathbf{f}$, $\mathbf{g} = \mathbf{G}_\sigma \mathbf{f}$. This removes the textures of scale $\sigma$, but it also smoothes structural edges, e.g., boundaries. Our dynamic guidance and fidelity term reconstruct smoothed boundaries, similar to [35]. Figure 9 shows regularization examples of (top) regular and (bottom) irregular textures. Our model completely removes textures without artifacts, and maintains small, high-frequency, but impor-



(a) Input (b) Cov. M1 [14] (c) RTV [32] (d) RGF [35] (e) Ours

Figure 9. Examples of the texture removal for (top) regular and (bottom) irregular textures. (a) Input image, (b) Cov. M1 [14] [$\sigma = 0.3$, $r = 10$], (c) RTV [32] [$\lambda = 0.01$, $\sigma = 6$], (d) RGF [35] [$\sigma_s = 5$, (from top to bottom) $\sigma_r = 0.1, 0.05$, $k = 5$], (e) ours [$\mathbf{u}^0 = \mathbf{u}_{l_1}$, (from top to bottom) $\lambda = 1000, 100$, $\sigma = 2$].

tant structures to be preserved, e.g., corners.

### 4.3. Other Applications

Our model can be applied to joint image restoration tasks. We have applied it to RGB/NIR and flash/non-flash denoising problems as shown in Figs. 10 and 11. In RGB/NIR denoising, color image $\mathbf{f}$ is regularized with the flash NIR image $\mathbf{g}$. Similarly, the non-flash image $\mathbf{f}$ is regularized with flash image $\mathbf{g}$. Since there exist structural dissimilarities between static guidance and input images ($\mathbf{g}$ and $\mathbf{f}$), the results might have artifacts and unnatural appearance. For example, static guidance regularization such as GF [10] cannot deal with a gradient reversal in flash NIR images [33], resulting in smoothed edges. Our model handles the structural differences between images, and shows performance comparable to the state of the art [33].

## 5. Discussion

We have presented a joint filtering framework that is widely applicable to computer vision and computational photography. Contrary to static guidance methods, we leverage dynamic guidance images as well, and they can exploit the structural information of the input image. Although our model does not have a closed-form solution, it converges rapidly to a local minimum. The simple and flexible formulation of our framework makes it applicable to a great variety of applications.

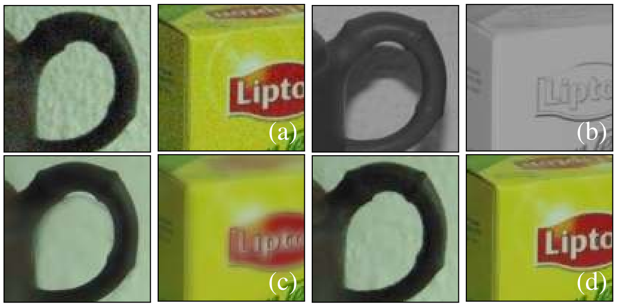


Figure 10. RGB and flash NIR image restoration. (a) RGB images, (b) NIR images, (c) GF [10] [$r = 3$, $\varepsilon = 4^{-4}$], (d) ours [$\mathbf{u}^0 = \mathbb{1}$, $\lambda = 15$, $\mu = 60$, $\nu = 30$, $k = 5$].

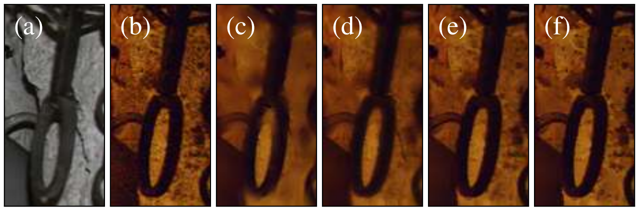


Figure 11. Flash and non-flash image restoration. (a) Flash image, (b) non-flash image, (c) GF [10] [$r = 3$, $\varepsilon = 4^{-4}$], (d) result of [27], (e) result of [33], (f) ours [$\mathbf{u}^0 = \mathbb{1}$, $\lambda = 15$, $\mu = 60$, $\nu = 30$, $k = 5$]. The results of (d) and (e) are from their project webpages.

**Acknowledgments.** The authors would like to thank Francis Bach for helpful discussions. This work was supported in part by the ERC grant VideoWorld and the Institut Universitaire de France.

## References

[1] P. Arbelaez, M. Maire, C. Fowlkes, and J. Malik. Contour detection and hierarchical image segmentation. *IEEE TPAMI*, 2011. 7

[2] T. Brox, O. Kleinschmidt, and D. Cremers. Efficient nonlocal means for denoising of textural patterns. *IEEE TIP*, 2008. 1, 2

[3] D. Chan, H. Buisman, C. Theobalt, S. Thrun, et al. A noise-aware filter for real-time depth upsampling. *in Proc. CVPRW*, 2008. 2

[4] P. Charbonnier, L. Blanc-Féraud, G. Aubert, and M. Barlaud. Deterministic edge-preserving regularization in computed imaging. *IEEE TIP*, 1997. 1

[5] I. Daubechies, R. DeVore, M. Fornasier, and C. S. Güntürk. Iteratively reweighted least squares minimization for sparse recovery. *Communications on Pure and Applied Mathematics*, 2010. 2

[6] F. Durand and J. Dorsey. Fast bilateral filtering for the display of high-dynamic-range images. 2002. 3

[7] Z. Farbman, R. Fattal, D. Lischinski, and R. Szeliski. Edge-preserving decompositions for multi-scale tone and detail manipulation. *ACM TOG*, 2008. 2, 3, 4, 5, 7

[8] D. Ferstl, C. Reinbacher, R. Ranftl, M. Rüther, and H. Bischof. Image guided depth upsampling using anisotropic total generalized variation. *in Proc. ICCV*, 2013. 1, 2, 6

[9] F. R. Hampel, E. M. Ronchetti, P. J. Rousseeuw, and W. A. Stahel. *Robust statistics: the approach based on influence functions*, volume 114. John Wiley & Sons, 2011. 3

[10] K. He, J. Sun, and X. Tang. Guided image filtering. *in Proc. ECCV*, 2010. 1, 2, 6, 7, 8

[11] P. W. Holland and R. E. Welsch. Robust regression using iteratively reweighted least-squares. *Communications in Statistics-Theory and Methods*, 1977. 3

[12] D. R. Hunter and K. Lange. A tutorial on mm algorithms. *The American Statistician*, 2004. 4

[13] P. Isola, D. Zoran, D. Krishnan, and E. H. Adelson. Crisp boundary detection using pointwise mutual information. *in Proc. ECCV*, 2014. 3

[14] L. Karacan, E. Erdem, and A. Erdem. Structure-preserving image smoothing via region covariances. *ACM TOG*, 2013. 8

[15] J. Kopf, M. Cohen, D. Lischinski, and M. Uyttendaele. Joint bilateral upsampling. *ACM TOG*, 2007. 1, 2

[16] G. R. Lanckriet and B. K. Sriperumbudur. On the convergence of the concave-convex procedure. *NIPS*, 2009. 4

[17] M. Lang, O. Wang, T. Aydin, A. Smolic, and M. H. Gross. Practical temporal consistency for image-based graphics applications. *ACM TOG*, 2012. 2

[18] A. Levin, D. Lischinski, and Y. Weiss. Colorization using optimization. *ACM TOG*, 2004. 2

[19] M.-Y. Liu, O. Tuzel, and Y. Taguchi. Joint geodesic upsampling of depth images. *in Proc. CVPR*, 2013. 3

[20] Z. Ma, K. He, Y. Wei, J. Sun, and E. Wu. Constant time weighted median filtering for stereo matching and beyond. *in Proc. ICCV*, 2013. 1, 2, 6

[21] O. Mac Aodha, N. D. Campbell, A. Nair, and G. J. Brostow. Patch based synthesis for single depth image super-resolution. *in Proc. ECCV*, 2012. 3

[22] J. Mairal. Incremental majorization-minimization optimization with application to large-scale machine learning. *arXiv preprint arXiv:1402.4419*, 2014. 4

[23] D. Martin, C. Fowlkes, D. Tal, and J. Malik. A database of human segmented natural images and its application to evaluating segmentation algorithms and measuring ecological statistics. *in Proc. ICCV*, 2001. 7

[24] M. Meila and J. Shi. A random walks view of spectral segmentation. *in Proc. AISTATS*, 2001. 5

[25] J. Park, H. Kim, Y.-W. Tai, M. S. Brown, and I. Kweon. High quality depth map upsampling for 3d-tof cameras. *in Proc. ICCV*, 2011. 1, 2, 3, 6

[26] P. Perona and J. Malik. Scale-space and edge detection using anisotropic diffusion. *IEEE TPAMI*, 1990. 1, 2, 3, 4, 5, 7

[27] G. Petschnigg, R. Szeliski, M. Agrawala, M. Cohen, H. Hoppe, and K. Toyama. Digital photography with flash and no-flash image pairs. *ACM TOG*, 2004. 1, 8

[28] C. Rhemann, A. Hosni, M. Bleyer, C. Rother, and M. Gelautz. Fast cost-volume filtering for visual correspondence and beyond. *in Proc. CVPR*, 2011. 1

[29] D. Scharstein and R. Szeliski. A taxonomy and evaluation of dense two-frame stereo correspondence algorithms. *IJCV*, 2002. 5, 6

[30] C. Tomasi and R. Manduchi. Bilateral filtering for gray and color images. *in Proc. ICCV*, 1998. 2, 4

[31] C. J. Wu. On the convergence properties of the em algorithm. *The Annals of statistics*, 1983. 4

[32] L. Xu, Q. Yan, Y. Xia, and J. Jia. Structure extraction from texture via relative total variation. *ACM TOG*, 2012. 8

[33] Q. Yan, X. Shen, L. Xu, S. Zhuo, X. Zhang, L. Shen, and J. Jia. Cross-field joint image restoration via scale map. *in Proc. ICCV*, 2013. 1, 8

[34] K.-J. Yoon and I. S. Kweon. Adaptive support-weight approach for correspondence search. *IEEE TPAMI*, 2006. 1

[35] Q. Zhang, X. Shen, L. Xu, and J. Jia. Rolling guidance filter. *in Proc. ECCV*, 2014. 1, 2, 4, 5, 7, 8

[36] Z. Zhang, J. T. Kwok, and D.-Y. Yeung. Surrogate maximization/minimization algorithms for adaboost and the logistic regression model. *ICML*, 2004. 4

[37] H. Zimmer, A. Bruhn, and J. Weickert. Optic flow in harmony. *IJCV*, 2011. 3