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

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.

Model. 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. The influence of the guidance on the input image varies spatially, and is controlled by affinity functions that measure similarities between adjacent vertices: $\phi_\mu(g_i - g_j)$ and $\phi_\nu(u_i - u_j)$, $i, j \in \mathcal{N}$ where $\phi_\sigma(x) = \exp(-\sigma x^2)$. μ and ν are the bandwidths of static and dynamic guidance, respectively. $\mathcal{N}$ is a set of neighborhoods that can be defined in a local and/or nonlocal manner. Let $\mathcal{W}_g = \left[\phi_\mu(g_i - g_j)\right]_{N\times N}$, $\mathcal{W}_u = \left[\phi_\nu(u_i - u_j)\right]_{N\times N}$, and $\mathcal{C} = diag\left([c_1,\ldots,c_N]\right)$. We minimize an objective function of the form:

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

where $\mathcal{W} = \mathcal{W}_g \circ \mathcal{W}_u$, and $\circ$ denotes the Hadamard product of the matrices. $\mathbb{1}$ is a $N\times1$ vector, where all the entries are 1. The diagonal entries c_i of $\mathcal{C}$ are confidence values for the pixels i of the input image.

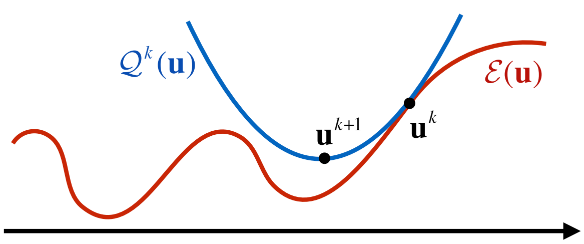

Figure 1: Sketch of the majorization-minimization algorithm. Given some estimate $\mathbf{u}^k$ of the minimum of $\mathcal{E}$, a surrogate function $\mathcal{Q}^k(\mathbf{u})$ is constructed. The next estimate $\mathbf{u}^{k+1}$ is then computed by minimizing $\mathcal{Q}^k$

Solver. We solve this nonconvex optimization problem by the majorization-minimization algorithm (Fig. 1) [1]:

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

$$\begin{cases} \mathcal{E}(\mathbf{u}) \le \mathcal{Q}^k(\mathbf{u}), \forall \mathbf{u}, \mathbf{u}^k \in \Theta \\ \mathcal{E}(\mathbf{u}^k) = \mathcal{Q}^k(\mathbf{u}^k), \forall \mathbf{u}^k \in \Theta \end{cases}, \tag{2}$$

where $\Theta \subset [0,1]^N$, as follows:

$$\begin{aligned} \mathcal{Q}^k(\mathbf{u}) &= \mathbf{u}^T\left[\mathcal{C} + \lambda\mathcal{L}^k\right]\mathbf{u} - 2\mathbf{f}^T\mathcal{C}\mathbf{u} + \mathbf{f}^T\mathcal{C}\mathbf{f} \\ &\quad - \lambda\mathbf{u}^{kT}\mathcal{L}^k\mathbf{u}^k + \frac{\lambda}{\nu}\mathbb{1}^T\left(\mathcal{W}_g - \mathcal{W}^k\right)\mathbb{1}. \end{aligned} \tag{3}$$

$\mathcal{L}^k = \mathcal{D}^k - \mathcal{W}^k$ is a *dynamic* Laplacian matrix at the step k, where $\mathcal{W}^k = \mathcal{W}_g \circ \mathcal{W}_{u^k}$ and $\mathcal{D}^k = diag\left([d_1^k,\ldots,d_N^k]\right)$ 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 function 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:

$$\mathbf{u}^{k+1} = \underset{\mathbf{u}\in\Theta}{\arg\min}\,\mathcal{Q}^k(\mathbf{u}) = (\mathcal{C} + \lambda\mathcal{L}^k)^{-1}\mathcal{C}\mathbf{f}. \tag{4}$$

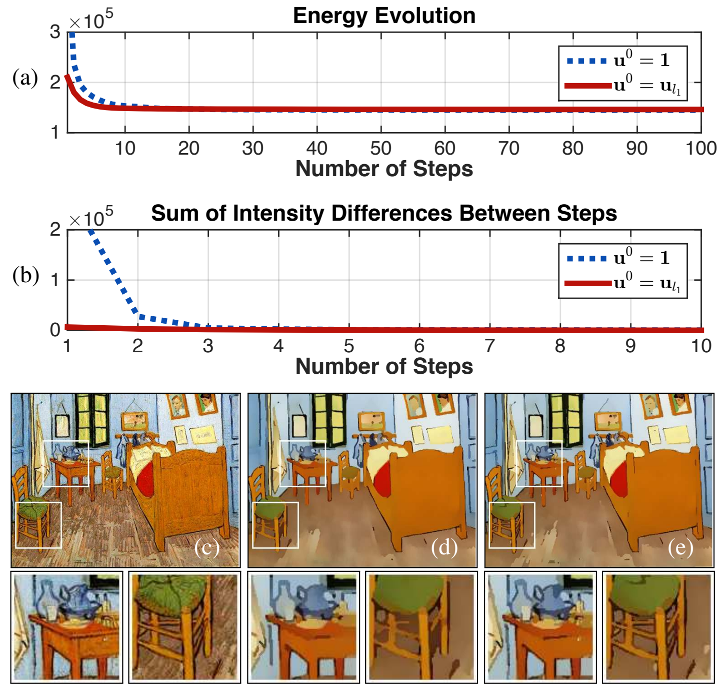

Figure 2: An example of (a) energy evolution and (b) a sum of intensity difference between successive steps, i.e., $\|\mathbf{u}^k - \mathbf{u}^{k+1}\|_1$, given (c) the input image. Our model monotonically converges, and guarantees a meaningful solution in the steady-state: (d) $\mathbf{u}^0 = \mathbb{1}$, $k = 30$, 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 [$\lambda = 50$, $\mu = 5$, $\nu = 40$].

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

$$\mathcal{E}(\mathbf{u}^{k+1}) \le \mathcal{Q}^k(\mathbf{u}^{k+1}) \le \mathcal{Q}^k(\mathbf{u}^k) = \mathcal{E}(\mathbf{u}^k), \tag{5}$$

and it can be shown to converge to a local minimum of $\mathcal{E}$ [2].

Convergence. Figure 2 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. 2 (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[1].

Applications. We demonstrate the flexibility and effectiveness of our model in several applications including depth super-resolution, scale-space filtering, texture-aware smoothing, flash/non-flash denoising, and RGB/NIR denoising.

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

[2] CF Jeff Wu. On the convergence properties of the em algorithm. *The Annals of statistics*, 1983.

*WILLOW project-team, Département d'Informatique de l'Ecole Normale Supérieure, ENS/Inria/CNRS UMR 8548. This is an extended abstract. The full paper is available at the Computer Vision Foundation webpage.

[1]After 5 steps, an average (maximum) value of the per-pixel intensity difference is 9.4×10^{-5} (1.7×10^{-3}) with $\mathbf{u}^0 = \mathbb{1}$ and 4.3×10^{-5} (8.7×10^{-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 $k = 5$.