Bi-affinity Filter: A Bilateral Type Filter for Color Images

Mithun Das Gupta and Jing Xiao

Epson Research and Development,
San Jose, California, USA
{mdasgupta,xiaoj}@erd.epson.com

Abstract. We propose a new filter called Bi-affinity filter for color images. This filter is similar in structure to the bilateral filter. The proposed filter is based on the color line model, which does not require the explicit conversion of the RGB values to perception based spaces such as CIELAB. The bi-affinity filter measures the affinity of a pixel to a small neighborhood around it and weighs the filter term accordingly. We show that this method can perform at par with standard bilateral filters for color images. The small edges of the image are usually enhanced leading to a very easy image enhancement filter.

Keywords: Bilateral filter, RGB color filtering, image matting, matting Laplacian.

1 Introduction

Bilateral filter was originally proposed by Tomasi and Manduchi [1]. The principle idea behind such a filtering operation is to combine information from spatial domain as well as feature domain. It can be represented as

$$\mathbf{h}(x) = \frac{1}{k(x)} \sum_{y \in \Omega_x} f_s(x,y) g_r(\mathbf{I}(x), \mathbf{I}(y)) \mathbf{I}(y) \tag{1}$$

where $\mathbf{I}$ and $\mathbf{h}$ are the input and output images respectively, x and y are pixel locations over the image grid, Ω_x is the neighborhood induced around the central pixel x, $f_s(x,y)$ measures the spatial affinity between pixels at x and y and $g_r(\mathbf{I}(x), \mathbf{I}(y))$ denotes the feature/measurement/photometric affinity. $k(x)$ is the normalization term given by

$$k(x) = \sum_{y \in \Omega_x} f_s(x,y) g_r(\mathbf{I}(x), \mathbf{I}(y)) \tag{2}$$

The spatial and range filters (f, g respectively), are commonly set to be Gaussian filters

$$f_s(x,y) = exp(\frac{-\|x-y\|_2^2}{2\sigma_s^2}), \qquad g_r(u,v) = exp(\frac{-\|u-v\|_2^2}{2\sigma_r^2})$$

K.N. Kutulakos (Ed.): ECCV 2010 Workshops, Part II, LNCS 6554, pp. 27–40, 2012.

parameterized by the variances σ_s, σ_r. The range filter penalizes distance in the feature space and hence the filter has an inherent edge preserving property. Due to this important property bilateral filter has been one of the most widely used filtering techniques within computer vision community.

Bilateral filter is a non-linear filter and as such many researchers have proposed techniques to decompose the non-linear filter into a sum of separable one dimensional filters or similar cascaded representations [2]. Singular value decomposition of the 2D kernel is one such approach which has been proposed by [3,4]. Paris *et al.* [5] proposed an approximation of the bilateral filter by filtering subsampled copies of the image with discrete intensity kernels, and recombining the results using linear interpolation.

Recently numerous researches have identified the run-time of the bilateral filter as the critical bottleneck and a few techniques have been proposed which render the filtering operation almost constant time, albeit with larger space requirements [6,7] and behavioral approximations. The research into improving the filter performance heavily relies on the form of the filters which are applied in the range as well as spatial domain. Porikli's method [6] can be entirely broken down to an approximation of a product of a box filter for smoothing and a polynomial or 4th order Taylor series approximation of a Gaussian kernel.

Traditionally, researchers have overlooked one of the most important shortfalls of the bilateral filter, which is a unified handling of multi-channel color images. This is due to the independence assumption within the color channels, such that the filter processes each channel on its own. As a direct consequence, bilateral filter produces color artifacts at sharp color edges. One of the remedies proposed in the original work by Tomasi *et al.* [1] was to convert from RGB space to CIELAB space. According to them, once the image is converted to the CIELAB space the channel wise bilateral filter does not produce such artifacts. We try to investigate further into this weakness and propose a new technique which works at par with the transformed domain techniques which have been the standard practices within the community so far.

2 Color Models

The deterioration of bilateral filter for RGB space seems to indicate that one constant range filter is probably not enough to capture the edge variations in all the channels, and hence a conversion to a suitable space such as CIELAB, which is perceptually more uniform than RGB, is performed. Though this transformation is very fast and can be implemented in hardware, this does not preclude the research in alleviating this necessity. This inherent shortcoming of bilateral filter to work in the RGB space can be traced back to the idea of quantifying the nearness of the color of two pixels within some spatial neighborhood. To determine whether two pixels have the same real world color, the color coordinates of a generic color model are used. Any generic color model assumes either there is no color distortion in the neighborhood, or there is an identical color distortion for all imaging conditions. In practice, when dealing with real world

images of an unknown source, these assumptions are rarely true as scene surface color is distorted differently in different images as well as different image regions, depending on the scene and camera settings.

2.1 Color Line Model

The introduction of color lines has been attributed to Omer *et al.* [8], who proposed the idea that the cluster of pixel colors in the RGB space appear to be mostly tubular regions, thereby adhering to the fact that most small regions in natural images can be decomposed into a linear combination of 2 colors. This has the obvious potential in edge preserving filtering domain, since it brings down the estimation problem of a valid range filter from 3 channels to 2.

When looking at the RGB histogram of real world images (Fig. 1), it can be clearly observed that the histogram is very sparse, and it is structured. Color line model exploits these two properties of color histograms by describing the elongated color clusters. It results in an image specific color representation that has two important properties: robustness to color distortion and a compact description of colors in an image. This idea has been used for image matting [9,10], Bayer demosaicing [11] and more recently for image de-noising and de-blurring [12,13]. The matting idea can be further utilized in edge preserving filter applications by removing the constant range filter all-together. The 2 color characteristics of a small patch can be exploited to evaluate the best range variance for the patch itself. This idea is the key intuition behind the new filter introduced in this work.

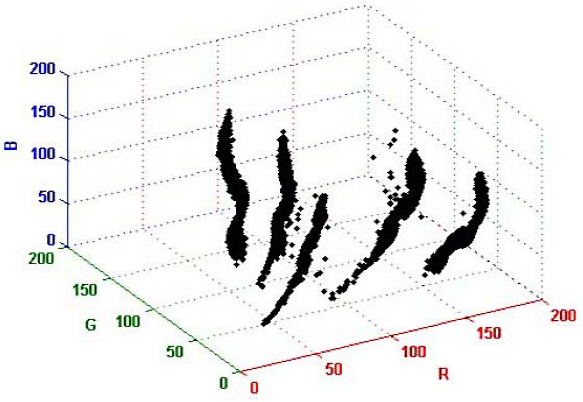

Fig. 1. RGB color histogram adapted from [8]

2.2 Closed Form Matting

The two-color model states that any pixel color I_i can be represented as a linear combination of two colors P and S, where these colors are piecewise smooth and

can be derived from local properties within a small neighborhood containing the pixel i.

$$I_i = \alpha_i P + (1 - \alpha_i) S, \quad \forall i \in w, \quad 0 \leq \alpha_i \leq 1 \tag{3}$$

where w is a small patch. α is called the *matting coefficient*. The patch size is a key parameter in this model, as it is true only for *small* neighborhoods. As the resolution and the size of the images grow, so should the window size as well, to capture a valid neighborhood. For color images, it was proved by Levin et al. [10], that if the color line property is obeyed then the 4D linear model satisfied by the matting coefficient, within a small window, at each pixel can be written as

$$\alpha_i = \sum_c a^c I_i^c + b, \quad \forall i \in w, \quad c \in \{1, 2, 3\} \tag{4}$$

where c is the index over the color channels. Given such a model we can formulate the cost function for evaluating the matting coefficient α. For an image with N pixels we define the cost as

$$J(\boldsymbol{\alpha}, \mathbf{a}, b) = \sum_{k \in N} \left(\sum_{i \in w_k} (\alpha_i - \sum_c a_k^c I_i^c - b_k)^2 + \epsilon \sum_c {a_k^c}^2 \right) \tag{5}$$

where w_k is a small window around pixel k and $\mathbf{a} = \{a_i^c\}$, for all $i = [1, N]$. ϵ is a regularization weight for uniqueness as well as smoothness of the solution.

Theorem 1. *Let $J(\boldsymbol{\alpha}) \doteq \min_{\mathbf{a},b} \; J(\boldsymbol{\alpha}, \mathbf{a}, b)$, then $J(\boldsymbol{\alpha}) = \boldsymbol{\alpha}^T L \boldsymbol{\alpha}$, where L is an $N \times N$ matrix, whose ij^{th} element is given by*

$$\sum_{k|(i,j) \in w_k} (\delta_{ij} - \frac{1}{|w_k|}(1 + (\mathbf{I}_i - \boldsymbol{\mu}_k)^T \tilde{\boldsymbol{\Sigma}}_k^{-1} (\mathbf{I}_j - \boldsymbol{\mu}_k))) \tag{6}$$

where δ_{ij} is the Kronecker delta, $\boldsymbol{\mu}_k$ is a 3×1 mean vector of colors inside the k^{th} window with both i and j as members, $\mathbf{I}_i$ and $\mathbf{I}_j$ are the color vectors at location i and j, $\tilde{\boldsymbol{\Sigma}}_k = \boldsymbol{\Sigma}_k + \frac{\epsilon}{|w_k|} I_3$, where $\boldsymbol{\Sigma}_k$ is the 3×3 covariance matrix, $|w_k|$ is the cardinality of the window and I_3 is the 3×3 identity matrix.

Proof. We would like to point out that, Levin *et al.* [10], prove the theorem based on an extension, from a gray scale case. We present the full 3-channel proof which can be readily extended to more channels if necessary.

Rewriting Eq. 5 in a matrix notation, where $\|.\|$ denotes the 2-norm,

$$J(\boldsymbol{\alpha}, \mathbf{a}, b) = \sum_k \|\mathbf{G}_k . \bar{\mathbf{a}}_k - \boldsymbol{\alpha}_k\| \tag{7}$$

where

$$\mathbf{G}_k = \begin{bmatrix} I_1^R & I_1^G & I_1^B & 1 \\ \vdots & \vdots & \vdots & \vdots \\ I_{w_k}^R & I_{w_k}^G & I_{w_k}^B & 1 \\ \sqrt{\epsilon} & 0 & 0 & 0 \\ 0 & \sqrt{\epsilon} & 0 & 0 \\ 0 & 0 & \sqrt{\epsilon} & 0 \end{bmatrix}, \quad \bar{\mathbf{a}}_k = \begin{bmatrix} a_k^R \\ a_k^G \\ a_k^B \\ b \end{bmatrix}, \quad \boldsymbol{\alpha}_k = \begin{bmatrix} \alpha_1 \\ \vdots \\ \alpha_{w_k} \\ 0 \\ 0 \\ 0 \end{bmatrix} \tag{8}$$

Note that another representation of $\mathbf{G}_k$ is possible where the last 3 rows are combined to a single row of the form $[\sqrt{\epsilon}\ \sqrt{\epsilon}\ \sqrt{\epsilon}\ 0]$, but this form leads to an unstable covariance matrix. For known $\boldsymbol{\alpha}_k$, we can solve the least square problem

$$\bar{\mathbf{a}}_k^{\star} = \arg\min \|\mathbf{G}_k.\bar{\mathbf{a}}_k - \boldsymbol{\alpha}_k\| \tag{9}$$

$$= (\mathbf{G}_k^T\mathbf{G}_k)^{-1}\mathbf{G}_k^T\boldsymbol{\alpha}_k \tag{10}$$

Substituting this solution in Eq. 7, and denoting $\mathbf{L}_k = I_{|w_k|+3} - \mathbf{G}_k(\mathbf{G}_k^T\mathbf{G}_k)^{-1}\mathbf{G}_k^T$, where $I_{|w_k|+3}$ is the identity matrix of size $(|w_k|+3)$, we obtain, $J(\boldsymbol{\alpha}) = \sum_k \|\mathbf{L}_k\boldsymbol{\alpha}_k\| = \sum_k(\boldsymbol{\alpha}_k^T\mathbf{L}_k^T\mathbf{L}_k\boldsymbol{\alpha}_k)$. Making the additional observation that

$$\begin{aligned}\mathbf{L}_k^T\mathbf{L}_k &= (I_{|w_k|+3} - \mathbf{G}_k(\mathbf{G}_k^T\mathbf{G}_k)^{-1}\mathbf{G}_k^T)^T(I_{|w_k|+3} - \mathbf{G}_k(\mathbf{G}_k^T\mathbf{G}_k)^{-1}\mathbf{G}_k^T)\\ &= I_{|w_k|+3} + \mathbf{G}_k(\mathbf{G}_k^T\mathbf{G}_k)^{-1}\mathbf{G}_k^T\mathbf{G}_k(\mathbf{G}_k^T\mathbf{G}_k)^{-1}\mathbf{G}_k^T - 2\mathbf{G}_k(\mathbf{G}_k^T\mathbf{G}_k)^{-1}\mathbf{G}_k^T\\ &= I_{|w_k|+3} - \mathbf{G}_k(\mathbf{G}_k^T\mathbf{G}_k)^{-1}\mathbf{G}_k^T = \mathbf{L}_k\end{aligned}$$

we can write $J(\boldsymbol{\alpha}) = \sum_k(\boldsymbol{\alpha}_k^T\mathbf{L}_k\boldsymbol{\alpha}_k)$. To complete the proof we need to find the expression for $\mathbf{L}_k|_{i,j}$.

Noting the identity $E[X^2] = \sigma_{XX}^2 + E[X]^2$, denoting the individual channel means $E[R]$ as R, we can write

$$\mathbf{G}_k^T\mathbf{G}_k = |w_k| \begin{bmatrix} \overbrace{\underbrace{\begin{pmatrix} \sigma_{RR}^2 + R^2 + \frac{\epsilon}{|w_k|} & \sigma_{RG}^2 + RG & \sigma_{RB}^2 + RB \\ \sigma_{GR}^2 + GR & \sigma_{GG}^2 + G^2 + \frac{\epsilon}{|w_k|} & \sigma_{GB}^2 + GB \\ \sigma_{BR}^2 + BR & \sigma_{BG}^2 + BG & \sigma_{BB}^2 + B^2 + \frac{\epsilon}{|w_k|} \end{pmatrix}}_{A} \underbrace{\begin{pmatrix} R \\ G \\ B \end{pmatrix}}_{D} \\ \underbrace{\begin{pmatrix} R & G & B \end{pmatrix}}_{D^T} \underbrace{1}_{C} \end{bmatrix} \tag{11}$$

where we have divided the matrix into 4 components. Note that $D = \boldsymbol{\mu}_k$ for the $k^{\mathtt{th}}$ window. Inverse of the above system can now be written as ([14])

$$(\mathbf{G}_k^T\mathbf{G}_k)^{-1} = \frac{1}{|w_k|}\begin{bmatrix} P & Q \\ R & S \end{bmatrix}$$

$$P = (A - DC^{-1}D^T)^{-1} = (A - DD^T)^{-1}$$

$$= \begin{bmatrix} \sigma_{RR}^2 + \frac{\epsilon}{|w_k|} & \sigma_{RG}^2 & \sigma_{RB}^2 \\ \sigma_{GR}^2 & \sigma_{GG}^2 + \frac{\epsilon}{|w_k|} & \sigma_{GB}^2 \\ \sigma_{BR}^2 & \sigma_{BG}^2 & \sigma_{BB}^2 + \frac{\epsilon}{|w_k|} \end{bmatrix}^{-1} = \tilde{\boldsymbol{\Sigma}}_k^{-1}$$

$$Q = -P(DC^{-1}) = -PD = -\tilde{\boldsymbol{\Sigma}}_k^{-1}\boldsymbol{\mu}_k$$

$$R = -(C^{-1}D^T)P = -D^TP = -\boldsymbol{\mu}_k^T\tilde{\boldsymbol{\Sigma}}_k^{-1}$$

$$S = C^{-1} - R(DC^{-1}) = 1 - RD = 1 + \boldsymbol{\mu}_k^T\tilde{\boldsymbol{\Sigma}}_k^{-1}\boldsymbol{\mu}_k$$

Putting all the terms together, we can write

$$(\mathbf{G}_k^T\mathbf{G}_k)^{-1} = \frac{1}{|w_k|}\begin{bmatrix} \tilde{\boldsymbol{\Sigma}}_k^{-1} & -\tilde{\boldsymbol{\Sigma}}_k^{-1}\boldsymbol{\mu}_k \\ -\boldsymbol{\mu}_k^T\tilde{\boldsymbol{\Sigma}}_k^{-1} & 1+\boldsymbol{\mu}_k^T\tilde{\boldsymbol{\Sigma}}_k^{-1}\boldsymbol{\mu}_k \end{bmatrix} \tag{12}$$

$$\mathbf{G}_k(\mathbf{G}_k^T\mathbf{G}_k)^{-1} = \frac{1}{|w_k|}\begin{bmatrix} (\mathbf{I}_1-\boldsymbol{\mu}_k)^T\tilde{\boldsymbol{\Sigma}}_k^{-1} & 1-(\mathbf{I}_1-\boldsymbol{\mu}_k)^T\tilde{\boldsymbol{\Sigma}}_k^{-1}\boldsymbol{\mu}_k \\ (\mathbf{I}_2-\boldsymbol{\mu}_k)^T\tilde{\boldsymbol{\Sigma}}_k^{-1} & 1-(\mathbf{I}_2-\boldsymbol{\mu}_k)^T\tilde{\boldsymbol{\Sigma}}_k^{-1}\boldsymbol{\mu}_k \\ \vdots & \vdots \\ (\mathbf{I}_{w_k}-\boldsymbol{\mu}_k)^T\tilde{\boldsymbol{\Sigma}}_k^{-1} & 1-(\mathbf{I}_{w_k}-\boldsymbol{\mu}_k)^T\tilde{\boldsymbol{\Sigma}}_k^{-1}\boldsymbol{\mu}_k \\ \sqrt{\epsilon}\tilde{\boldsymbol{\Sigma}}_k^{-1} & \sqrt{\epsilon}\tilde{\boldsymbol{\Sigma}}_k^{-1}\boldsymbol{\mu}_k \end{bmatrix} \tag{13}$$

Right multiplication by $\mathbf{G}_k^T$ yields the final symmetric form, where we show only the i^th column for conciseness and ease of understanding

$$\mathbf{G}_k(\mathbf{G}_k^T\mathbf{G}_k)^{-1}\mathbf{G}_k^T\ [:,i] = \frac{1}{|w_k|}\begin{bmatrix} 1+(\mathbf{I}_1-\boldsymbol{\mu}_k)^T\tilde{\boldsymbol{\Sigma}}_k^{-1}(\mathbf{I}_i-\boldsymbol{\mu}_k) \\ 1+(\mathbf{I}_2-\boldsymbol{\mu}_k)^T\tilde{\boldsymbol{\Sigma}}_k^{-1}(\mathbf{I}_i-\boldsymbol{\mu}_k) \\ 1+(\mathbf{I}_3-\boldsymbol{\mu}_k)^T\tilde{\boldsymbol{\Sigma}}_k^{-1}(\mathbf{I}_i-\boldsymbol{\mu}_k) \\ \vdots \\ 1+(\mathbf{I}_{w_k}-\boldsymbol{\mu}_k)^T\tilde{\boldsymbol{\Sigma}}_k^{-1}(\mathbf{I}_i-\boldsymbol{\mu}_k) \\ \epsilon\tilde{\boldsymbol{\Sigma}}_k^{-1}(\mathbf{I}_i-\boldsymbol{\mu}_k) \end{bmatrix}$$

Subtracting from $I_{|w_k|+3}$ and summing over k concludes the proof. Note that $\mathbf{G}_k$ has 3 extra rows, (or C extra rows for general case) for the regularization ϵ. These can be neglected in the final expression since they do not explicitly effect the other computations. □

3 Bi-affinity Filter

The laplacian matrix $\mathbf{L}$, whose elements are defined in Eq. 6, is called the *matting laplacian* [10]. The usual decomposition of the laplacian matrix into a diagonal matrix and a weight matrix leads to the formulation $\mathbf{L} = \mathbf{D} - \mathbf{W}$. Here $\mathbf{D}$ is a diagonal matrix with the terms $D_{ii} = \#[k|i \in w_k]$ at its diagonal, which represents the cardinality of the number of windows the pixel i is a member of. The individual terms of the weight matrix $\mathbf{W}$, called the *matting affinity*, are given by

$$W_{ij} = \sum_{k|(i,j)\in w_k} \frac{1}{w_k}(1+(\mathbf{I}_i-\boldsymbol{\mu}_k)^T(\boldsymbol{\Sigma}_k+\frac{\epsilon}{w_k}\mathbf{I}_3)^{-1}(\mathbf{I}_j-\boldsymbol{\mu}_k)) \tag{14}$$

By definition, all the rows of a laplacian matrix sum to zero, which leads to $D_{ii} = \sum_j W_{ij}$. At the local minima the solution $\boldsymbol{\alpha}^\star$ satisfies the first order optimality

condition $\mathbf{L}^T\boldsymbol{\alpha}^\star = 0$. So we can write the optimal condition for minimizing $J(\boldsymbol{\alpha})$ as

$$\mathbf{L}^T\boldsymbol{\alpha}^\star = (\mathbf{D}-\mathbf{W})^T\boldsymbol{\alpha}^\star = \begin{pmatrix} D_{11}\alpha_1^\star & -\sum_j W_{1j}\alpha_j^\star \\ D_{22}\alpha_2^\star & -\sum_j W_{2j}\alpha_j^\star \\ \vdots & \vdots \\ D_{nn}\alpha_n^\star & -\sum_j W_{Nj}\alpha_j^\star \end{pmatrix}$$

Substituting $D_{ii} = \sum_j W_{ij}$ into the above system of equations and invoking the first order optimality condition leads to

$$\begin{pmatrix} \sum_j(\alpha_1^\star - \alpha_j^\star)W_{1j} \\ \sum_j(\alpha_2^\star - \alpha_j^\star)W_{2j} \\ \vdots \\ \sum_j(\alpha_n^\star - \alpha_j^\star)W_{nj} \end{pmatrix} = 0 \tag{15}$$

The effect of this equation is that the affinity W_{ij} for two pixels with the same color (same $\alpha^\star$), is a positive quantity varying with the homogeneity of the local windows containing the pixels i and j as governed by Eqn. 14. But for pixels with different color (different $\alpha^\star$) the affinity is *zero*. In essence the rows of the laplacian matrix $\mathbf{L}$ work as a *zero-sum* filter kernel, after appropriate resizing. For our proposed filter, we replace the range filter of traditional bilateral filter with the appropriate row from the matting laplacian. This leads to the formulation of the *bi-affinity* filter

$$\mathbf{h}_{\sigma,\epsilon}(x) = \frac{\sum_{y\in\Omega_x} f_s^\sigma(x,y)L_{xy}^\epsilon \mathbf{I}(y)}{\sum_{y\in\Omega_x} f_s^\sigma(x,y)L_{xy}^\epsilon} \tag{16}$$

where we denote the dependence on the user specified parameters σ, ϵ on the filter output. The parameter σ controls the amount of spatial blurring and is same as the spatial filter variance in standard bilateral filter. The parameter ϵ works analogous to the range variance parameter in traditional bilateral filter. Note that the relative weight attributed to the regularization term ϵ, determines the smoothness of the α estimates, which in our work translates to the smoothness of the filtered image. Bilateral filter has an inherent bending effect at the edges, which can be observed in the very simple experiment shown in Fig. 2. Bi-affinity filter does not smooth the edge, due to the affinity formulation which is zero across the edge. This effect can be achieved by bilateral filtering only under infinite range variance.

The calculation of the exact affinity matrix W_{ij} as mentioned in Eqn. 14, involves evaluation over all possible overlapping windows, which contain the center pixel, which is $O(w^3)$, where w is the size of the window. The overall complexity can be reduced by evaluating the affinity over a smaller set of possible windows. In the simplest case, we can evaluate the terms of $W_{i,j}$ locally, thereby counting the contribution of only the local window centered at the current pixel

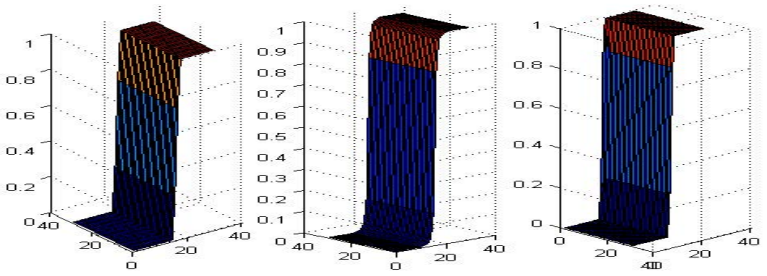

Fig. 2. Left to right: original edge, bilateral filtering, bi-affinity filtering. Note the edge curves slightly for the bilateral result.

(Fig. 3, right), and the complexity is equal to normal bilateral range filter, which is $O(w^2)$. To keep the later comparisons with bilateral filter fair, we define an approximate filter denoted by $\mathbf{h}^l(x)$

$$\mathbf{h}^l(x) = \frac{\sum_{y \in \Omega_x} f_s(x,y) L^x_{xy} \mathbf{I}(y)}{\sum_{y \in \Omega_x} f_s^(x,y) L^x_{xy}} \tag{17}$$

which considers only the local window centered around pixel x denoted by L^x. Note that we have dropped the dependence on the user specified parameters σ, ϵ for notational simplicity.

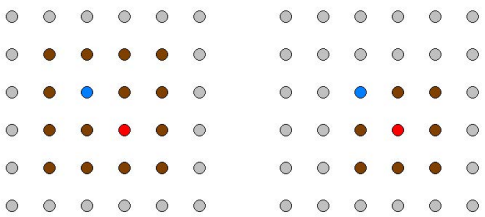

Fig. 3. Left: all possible 3×3 neighborhood windows (brown) for center pixel (red) and neighbor (blue). Right: central window only approximation.

The operations involved in computing the terms L_{ij}'s as mentioned in Eqn. 6, can be decomposed as summation of Gaussian likelihoods over window dependent parameters $\boldsymbol{\mu}_w, \boldsymbol{\Sigma}_w$. These parameters can be computed by accumulating first and second order sufficient statistics over windows. If memory complexity is not an issue then pre-computing 9 integral images can be an option. These 9 integral images correspond to 3 integral images for each of the channels `R`, `G` and `B`, 3 for `RR`, `GG` and `BB` and the remaining 3 for `RG`, `GB` and `RB`. For 3 channel color images, this is equivalent to storing 3 more images into the memory. For really large images (HDTV etc.) this option might not be the most optimal due

to the huge memory overhead. The other method is to collect sufficient statistics for the current window and then updating the statistics for each unit move from top to bottom and left to right, as proposed by the median filtering approach by Huang [15] and then improved by Weiss [16]. Both these methods can now be used to implement the bi-affinity filter.

4 Experiments

The regularization term in the affinity formulation works as an edge smoothness term. For understanding the effect of this term we vary the amount of regularization used for the process and record the PSNR with respect to the original image. We report the results with respect to the window size in Fig. 4. The PSNR degrades for larger window size, which further corroborates the two color model which is valid only for small windows. The regularization term neutralizes the effect of window size to a certain degree as seen by the band of values collecting near PSNR 96DB. This hints at a possible tradeoff between PSNR and edge smoothness. For very small regularization values, the noise across the edge can contribute to the jaggedness of the reconstruction. This effect can be countered by increasing the amount of regularization. But this increase comes at a cost, which is the increased smoothness of the overall image. Empirically, we have obtained good results for larger window sizes by keeping the regularization term relatively larger than proposed in the matting literature. The effect of regularization for fixed window size can be seen in Fig. 5. The edge reconstruction becomes increasingly jagged as the amount of regularization is decreased.

For quantitative comparisons against traditional bilateral filter, we concentrate on the range filter variance of 0.1 to 1 and vary the window size to obtain

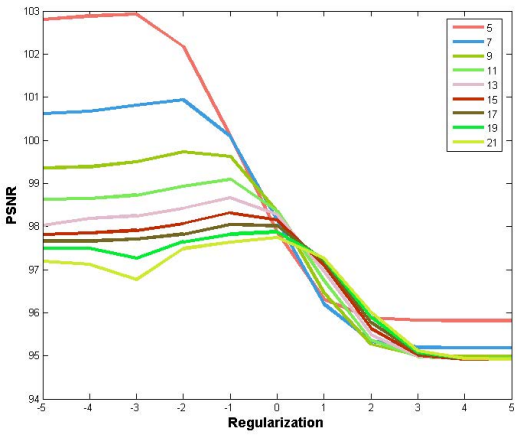

Fig. 4. PSNR with respect to ground truth. The colors depict the window size. The x axis depicts the regularization in log scale such that $\epsilon = 10^x$.

Fig. 5. Effect of regularization term ϵ. From left to right: $\epsilon = 0.0005, 0.005, 0.05, 0.5$. The edge becomes gradually smoother with increasing ϵ as can be seen at the inset images.

the curves in Fig. 6. The PSNR values obtained for our method are within acceptable deviations from those obtained for CIELAB bilateral filter, and surpass the performance at $\epsilon = 1$ and $w = 5$. Also note that Fig. 6 is a zoomed in version of Fig. 4, at x=[-1,0], approximately coinciding with the beginning of the knot.

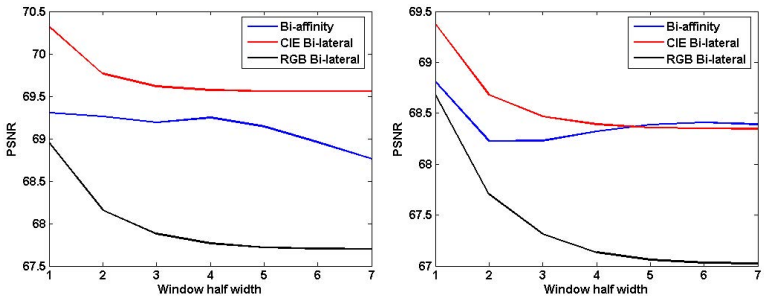

Fig. 6. PSNR comparisons. Left: $\epsilon = \sigma_r = 0.1$, right: $\epsilon = \sigma_r = 1$. $\sigma_d = 5$

In the next experiment, we present comparison of the approximate bi-affinity filter with traditional bilateral filter. This results are illustrated in Fig. 7. The response is very similar even though in our method we do not need any color conversions. For the bilateral filter, the RGB image is converted to CIELAB space and then the filter is applied individually to each channel.

4.1 Image Enhancement and Zooming

The original bi-affinity filter (Eqn. 16), has been derived from the matting laplacian formulation, which has been shown to preserve very minute details which is one of the requirements of matting [10]. In other words, our bi-affinity formulation preserves very intricate details of the image when compared to bilateral filter where only the dominant edges of the image are preserved. In this regard

Fig. 7. Top: left: original image, right: RGB bilateral filter. Bottom: left: CIELAB bilateral filter, $w = 11$, $\sigma_d = 5$, $\sigma_r = 0.1$, right: our method, $\epsilon = 0.1$.

bi-affinity filter can be thought of as preserving all edges, whereas bilateral filter only preserves strong edges. This important feature leads us to one of the most interesting applications of such a filter which is image enhancement and zooming.

Image enhancement techniques try to estimate the high-resolution data from the low-resolution data by estimating the missing information. This leads to numerous formulations, some learning based and some interpolation based. If the missing high-resolution data can be inferred, then it can be added to the interpolated input (which satisfies the data fidelity constraints) to generate the high-resolution image [17]. Given the low-resolution input in Fig. 8, we can interpolate it to the desired high-resolution size, and then add the missing high-resolution info to generate the final high-resolution result. The mean affinity at each pixel, which is the row wise normalized summation of W, contains this missing detail. This detail is shown in Fig. 8, center panel. The bi-affinity filter places a smoothed local affinity weighted kernel at each pixel. The enhancement effect is a byproduct of the filtering formulation and not the main aim of this work. We realize that existing methods, more so the iterative techniques [18], can use a formulation similar to ours to refine the estimate at each step.

Like many other *passive* filtering techniques, e.g. bilateral, bicubic, etc., our method only looks at the low-resolution observation to generate the values of the high-resolution scene. *Active* methods such as Markov random field (MRF) based

Fig. 8. Image enhancement. Left: input image is enlarged by a factor 2 (bicubic interpolation). The mean affinity map for all pixels (center), and the final enhanced image (right).

models, impose neighborhood continuity constraints. We proposed to investigate the details of such a model with the bi-affinity filter as one of its components, as a future work. Additional examples of comparisons against zooming and then post-processing with bilateral filter compared to our technique is shown in Fig. 9.

5 Conclusion and Future Work

In this paper, we have proposed a new edge preserving filter, which works on the principle of matting affinity. We present a full n-channel derivation of the matting laplacian. The formulation of matting affinity allows a better representation of the range filter term in bilateral filter class. The definition of the affinity term can be relaxed to suit different applications. We define an approximate bi-affinity filter whose output is shown to be very similar to the traditional bilateral filter. Our technique has the added advantage that no color space changes are required and hence the images can be handled in their original color space. This is a big benefit over traditional bilateral filter, which needs a conversion to perception based spaces, such as CIELAB to generate better results. The full bi-affinity filter preserves very minute details of the input image, and can be simply extended to an image enhancement application. The implementation of the filter still remains a challenge due to the small window requirement arising from the two color model constraint. We propose a diligent effort in this area, since it is evident that the kernel evaluation can be optimized in more ways than one.

Fig. 9. Image Zooming by a factor 2x. Left column: bi-cubic interpolation + bilateral filter. Right column: our method. Notice the preservation of small details in all the images.

References

1. Tomasi, C., Manduchi, R.: Bilateral filtering for gray and color images. In: ICCV 1998: Proceedings of the Sixth International Conference on Computer Vision, p. 839. IEEE Computer Society, Washington, DC (1998)
2. Wells III, W.: Efficient synthesis of gaussian filters by cascaded uniform filters. IEEE Trans. Pattern Anal. Mach. Intell. 8 (1986)
3. Geusebroek, J., Smeulders, A., van de Weijer, J.: Fast anisotropic gauss filtering. IEEE Transactions on Image Processing 12, 2003 (2002)
4. Lu, W.S., Wang, H.P., Antoniou, A.: Design of two-dimensional digital filters using the singular-value decomposition and balanced approximation method. IEEE Trans. Signal Process. 39, 2253–2262 (1991)
5. Paris, S., Durand, F.: A Fast Approximation of the Bilateral Filter Using a Signal Processing Approach. In: Leonardis, A., Bischof, H., Pinz, A. (eds.) ECCV 2006. LNCS, vol. 3954, pp. 568–580. Springer, Heidelberg (2006)
6. Porikli, F.: Constant time $o(1)$ bilateral filtering. In: IEEE Conference on Computer Vision and Pattern Recognition, CVPR (2008)
7. Yang, Q., Tan, K.H., Ahuja, N.: Real-time o(1) bilateral filtering. In: IEEE Conference on Computer Vision and Pattern Recognition, CVPR (2009)
8. Omer, I., Werman, M.: Color lines: Image specific color representation. In: CVPR (2004)
9. Bando, Y., Chen, B.Y., Nishita, T.: Extracting depth and matte using a color-filtered aperture. ACM Transactions on Graphics 27, 134:1–134:9 (2008)
10. Levin, A., Lischinski, D., Weiss, Y.: A closed form solution to natural image matting. In: CVPR (2006)
11. Bennett, E., Uyttendaele, M., Zitnick, C., Szeliski, R., Kang, S.: Video and Image Bayesian Demosaicing with a Two Color Image Prior. In: Leonardis, A., Bischof, H., Pinz, A. (eds.) ECCV 2006. LNCS, vol. 3951, pp. 508–521. Springer, Heidelberg (2006)
12. Joshi, N., Zitnick, C., Szeliski, R., Kriegman, D.: Image deblurring and denoising using color priors. In: CVPR (2009)
13. Liu, C., Szeliski, R., Kang, S.B., Zitnick, C.L., Freeman, W.T.: Automatic estimation and removal of noise from a single image. PAMI 30, 299–314 (2008)
14. Press, W.H., Teukolsky, S.A., Vetterling, W.T., Flannery, B.P.: Numerical recipes in C: the art of scientific computing, 2nd edn. Cambridge University Press, New York (1992)
15. Huang, T.S.: Transforms and median filters. In: Two-Dimensional Signal Processing II, pp. 209–211. Springer, Berlin (1981)
16. Weiss, B.: Fast median and bilateral filtering. ACM Trans. Graph. 25, 519–526 (2006)
17. Fattal, R.: Image upsampling via imposed edge statistics. In: SIGGRAPH (2007)
18. Irani, M., Peleg, S.: Motion analysis for image enhancement. In: JVCIP (1993)