Edge Preservation using Guided Image Filter Technique

Vrushali Patil[1], Dr. P. Malathi[2], Dr. Manish Sharma[3]

P.G. Student [VLSI & Embedded Systems], Dept. of E & TC, D. Y. Patil College of Engineering, Akurdi, Pune, India[1]
Professor, Dept. of E & TC, D. Y. Patil College of Engineering, Akurdi, Pune, India[2]
PG Co-ordinator, Dept. of E & TC, D. Y. Patil College of Engineering, Akurdi, Pune, India[3]
vrushpatil01@gmail.com, malathijesudason@ymail.com, manishsharma.mitm@gmail.com

Abstract- Filtering is widely used in image and video processing for various applications among which Edge- Preserving is the most popular one. Edge-preserving image smoothing has recently emerged as a valuable tool for a variety of applications such as denoising, tone mapping, non-photorealistic rendering in computer graphics and image processing. This can be achieved by a local filtering method such as bilateral filter [1]. However, this method has to face the problem of trade-off between edge-preservation abilities and smoothing abilities [2] and tends to result in staircase effect which is not acceptable for some applications. Hence, in this paper the guided filter is proposed which filters the output depending upon the information provided by the guidance image.

Keywords- Edge-preserving filtering, bilateral filter, median filter, guided filter, guidance image.

I.INTRODUCTION

Filtering is an image processing technique widely adopted in computer graphics, computer vision, computational photography, etc. It transforms the pixel intensity value to reveal certain image characteristics. More specifically, filtering can be applied in many applications such as noise reduction, colorization, edge preservation, detail enhancement, texture editing, haze/rain removal, et**c.** The most popular of this is the edge-preserving smoothening. Edge preserving smoothening refers to the image processing technique that results in smoothing of textures, while retaining the sharp edges. Since many natural images are described as a collection of gray value and oriented texture domains, a scale and orientation adaptive smoothing scheme provides a powerful noise reduction method. Edges between domains are the important features for the interpretation of images. However, smoothing operators tends to blur the edges or borders between the domains. Hence, such a filter is required to be used that not only reduce the noise but also does not degrade the edges, i.e. an edge preserving filter.

Linear translation invariant (LTI) filters such filters, all with explicit kernels have been widely used in image restoration, blurring/sharpening, edge detection, feature extraction, etc. LTI filtering also includes the method of solving Poisson's equation such as in HDR compression, image matting and image stitching, wherein the filtering kernels are implicitly defined by the inverse of a homogenous Laplacian matrix. The kernels included in LTI filters are spatially invariant and does not depend on the image content. But generally, it is desirable to incorporate additional information from a given guidance image while filtering.

One way to achieve this purpose is to optimize a quadratic function which enforces some constraints on the unknown output with the help of guidance image, where the guidance image may be the filter's input itself or another image. Thus, a large sparse matrix encoded with the information involved in the guidance image is solved. Although this approach yields the state of art quality, it results in a long computational time. Another method is to explicitly build the guidance image into the filter kernels. Bilateral filter is the most popular one of such filters.

In this paper we have introduced a novel explicit image filter called Guided filter that overcome the gradient reversal artifacts introduced while using bilateral filter.

II. LITERATURE OF RELATED WORK

Edge-preserving filtering techniques can be categorized as explicit/implicit weighted average filters and non-average filters.

2.1 Explicit Weighted- Average Filters

The bilateral filter is perhaps the simplest and most intuitive one among explicit weighted-average filters. In [1], the concept of bilateral filtering for edge preserving smoothening was introduced. It was mentioned that a common technique for preserving edges during smoothing is to compute the median in the filter’s support rather than computing the mean. Although this filter is effective in many cases such as noise removal and extraction of detail at a fine spatial scale, it has also been noticed that it may have artifacts in detail decomposition [2] and high dynamic range (HDR) compression [3]. Artifacts results from the pixels around the edge that have an unstable Gaussian Weighted Sum. Hence, the results may exhibit unwanted profiles around the edges. Fast implementation of bilateral filter also has been a challenging problem. The bilateral filter later generalized to joint bilateral filter [4], wherein the weights are computed from another guidance image rather than the filter input, is favored specifically when the image that is to be filtered is not reliable to provide the information about the edges. The reason behind it is that when a pixel on an edge has few similar pixels around it, the Gaussian weighted average becomes stable. For real time implementation [5], a bilateral filter involves histogram based approximation due to its computation efficiency and memory concern.

2.2 Implicit Weighted Average Filters

A series of approaches optimize a quadratic cost function and solve a linear system, which is equivalent to implicitly filtering an image by an inverse matrix. In image segmentation [6] and colorization [7], the affinities of this matrix are Gaussian functions of the color similarities. The weighted least squares filter in [2] adjusts the matrix affinities according to the image gradients and produces halo-free edge-preserving smoothing.

Although this optimization based approaches often generate high quality results, solving the linear system is time consuming. Direct solvers like Gaussian Elimination are not practical due to the memory-demanding “filled in” problem. The implicit weighted-average filters take at least a few seconds to process a one megapixel image either by preconditioning or by multi-grid [2].

2.3 Non-average Filters

Edge-preserving filtering can also be achieved by non-average filters. The median filter is an edge-aware operator and also a special case of local histogram filters [8], wherein histogram filters have O(N) time implementations in a way as the bilateral grid. The Total-Variation (TV) filters [9] optimize an L1-regularized cost function, and are shown equivalent to iterative median filtering [10]. The L1 cost function can also be optimized via half-quadratic split [11], alternating between a quadratic model and soft shrinkage. But it has been noticed that non-average filters are computationally expensive and complex.

III. GUIDED FILTER

In order to overcome the artifacts[2][3] introduced by bilateral filter, a new edge preserving performance known as Guided image filter is proposed that performs edge-preserving smoothing on an image, using the content of the second image i.e. the guidance image, in order to influence the filtering. The guidance image can be the image itself, a different version of the image or a completely different image. If the guidance image is same as the input image to be filtered, the structures are the same i.e. an edge in original image is the same as in the guidance image.

Guided image filtering is one of the spatial domain enhancement technique in which the filtering output is locally a linear transform of the guidance image. It takes into account the statistics of a region in the corresponding spatial neighborhood in the guidance image while calculating the value of the output pixel. Guided filter has good edge-preserving smoothing properties and does not suffer from the gradient reversal artifacts that are seen when using bilateral filter. It can perform better at the pixels near the edge when compared to bilateral filter. The guided filter is also a more generic concept beyond smoothing. By using the guidance image, it makes the filtering output more structured and less smoothed than the input. It can transfer the structures of the guidance image to the filtering output, enabling new filtering applications such as dehazing and guided feathering. Also, guided filter adopts the fast and non-approximation characteristics of linear time algorithm and provides an ideal option for real time applications in case of HD filtering. Hence, it is considered to be one of the fastest edge preserving filters.

Guided filter generally has an O(N) time (in the number of pixels N) exact algorithm for both gray scale and color images, regardless of the kernel size and the range of intensity. O(N) time represents that the time complexity is independent of the window radius(r) and hence arbitrary kernel sizes can be used in the applications.

3.1 Definition:

Here, the main idea and equations of a guided filter is reviewed. The key assumption of the guided filter defines a local linear model between the guidance image *I* and the filtered output image q, taking p as an input image as shown in fig.1 which represents an illustration of the guided filtering process.

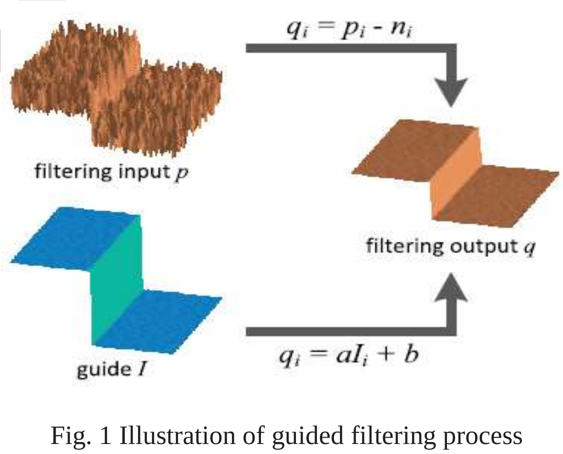


Fig. 1 Illustration of guided filtering process

It is assumed that q is a linear transform of *I* in a window w_k, that is centered at pixel k.

$$q_i = a_k I_i + b_k \forall\, i \in w_k \qquad (1)$$

where, a_k and b_k are considered to be linear coefficients that are constant in w_k. A square window of radius r is used. The relation is shown in fig.1. This local linear model ensures that q has edge only if *I* have edge.

In order to determine the linear coefficients (a_k,b_k), we need constraints from the filtering input p. We model the output q as the input p subtracting some unwanted components n like noise/textures:

$$q_i = p_i - n_i \qquad (2)$$

$$a_k = \frac{\frac{1}{|w|}\sum_{i \in w_k} I_i p_i - \mu_k \bar{p}_k}{\sigma_k^2 + \epsilon} \qquad (3)$$

$$b_k = \bar{p}_k - a_k \mu_k \qquad (4)$$

where, μ_k is the mean whereas σ_k^2 is the variance of *I* in window w_k and $|w|$ is the number of pixels in window w_k.

Also, $\bar{p}_k = \frac{1}{|w|} \sum_{i \in w_k} p_i$ is the mean of p in window w_k. After obtaining the linear coefficients (a_k,b_k), we can compute the filtering output q_i from equation 4. Since a pixel *i* is involved in all the overlapping windows w_k that will cover *I* and hence the value of q_i in (4) does not remain same when computed in different windows. A solution is to average all the possible values of q_i. Therefore, after computing the linear coefficients for all the windows w_k in the image, we can compute the filtering output by:

$$q_i = \frac{1}{|w|} \sum_{k|i \in w_k} a_k I_i + b_k \qquad (5)$$

Considering the symmetry of the box window, we rewrite (5) by

$$q_i = a_{i1} I_i + b_{i1} \qquad (6)$$

where, $a_{i1} = \frac{1}{|w|} \sum_{k \in w_i} a_k$ and $b_{i1} = \frac{1}{|w|} \sum_{k \in w_i} b_k$ are the average coefficients of all the windows overlapping *i*. As (a_{i1},b_{i1}) are the output of a mean filter, the gradients obtained from them can be expected to be very much smaller than that of the guidance image *I* near strong edges. This situation concludes that abrupt intensity changes in *I* can be preserved in q mostly. Hence, (3), (4), (6) represents the definition of the guided filter.

3.2 Algorithm of Guided Filter:

Input: Input image p, guidance image *I*, radius r, regularization parameter ϵ.

Output: Filtering output image q.

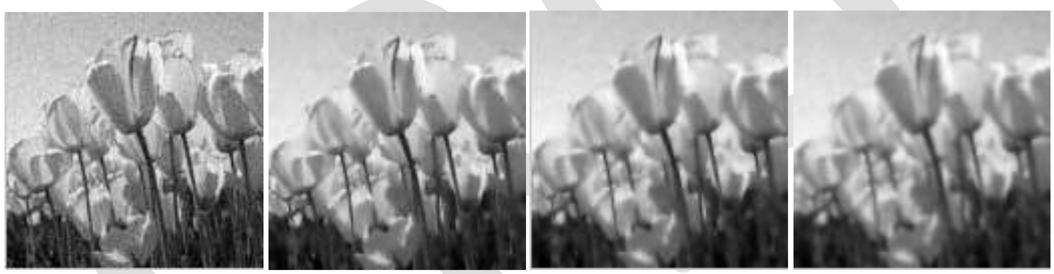
1. Read the image *I* which acts as the guidance image.
2. Make input image p equal to the guidance image *I*.
3. Enter the assumed values of r and ϵ.
4. Compute the mean
   $mean_I = f_{mean}(I)$
   $mean_p = f_{mean}(p)$
   $corr_I = f_{mean}(I.*I)$
   $corr_{Ip} = f_{mean}(I.*p)$
5. Compute the covariance and variance
   $var_I = corr_I - mean_{I.*} mean_I$
   $cov_{Ip} = corr_{Ip} - mean_{I.*} mean_p$
6. Compute the value of linear coefficients.
   $a = cov_{Ip} / (var_I + \epsilon)$
   $b = mean_p - a.* mean_I$

7. Compute the mean of a and b
   $mean_a = f_{mean}(a)$
   $mean_b = f_{mean}(a)$
8. Obtain the filtered output image using mean of a and b
   $q = mean_a .* I + mean_b$

3.3 Simulation of Guided Filter:

Guided filter involves the operation of mean filter f_{mean} within a window w_k of radius r. The input image p that is in the form of either jpeg or bmp is required to be converted into gray level. Further, Gaussian noise is added to the image. Filtering is performed using guided filter so that there will be no loss of information at the edges of the image. Then retrieving back the filtered input image from 2D to 1D, the filtered output image gets displayed.

Input **$\epsilon = 0.1^2$** **$\epsilon = 0.2^2$** **$\epsilon = 0.4^2$**



Input **$\epsilon = 0.1^2$** **$\epsilon = 0.2^2$** **$\epsilon = 0.4^2$**

Fig. 2 Results from MATLAB simulation

From the above results it is concluded that the patches with variance (σ^2)much smaller than ϵ are smoothed and those with variance much larger than ϵ are preserved. The impact of ϵ in the guided filter algorithm is to determine what an edge is or to obtain a high variance patch (when the guidance image I changes a lot within w_k), that should be preserved.

IV. CONCLUSION

Guided filter performs very well in terms of both quality and efficiency in a great variety of applications such as noise reduction, haze removal, image fusion, detail smoothing/enhancement, HDR compression and joint up-sampling. This filter has great potential in computer vision and computer graphics in order to suppress and extract information from images.

REFERENCES:

[1] C. Tomasi and R. Manduchi, "Bilateral filtering for gray and color images", in Proceedings of IEEE 6th International Conference on Computer Vision, pp. 839-846, 1998.

[2] Z. Farbman, R. Fattal, D. Lischinski, and R. Szeliski, "Edge-preserving decompositions for multiscale tone and detail manipulation", ACM Transactions on Graphics, vol. 27, no. 3, pp. 67:1–67:10, 2008.

[3] Fredo Durand and Julie Dorsey, "Fast bilateral filtering for the display of high dynamic range images", ACM Transactions on Graphics, vol. 21, no. 3, pp. 257-266, 2002.
[4] G. Petschnigg, R. Szeliski, M. Agrawal, M. Cohen, H. Hoppe and K. Toyama, "Digital photography with flash and no-flash image pairs", ACM Transactions on Graphics, vol. 23, no. 3, pp. 664-672, 2004.
[5] Y. C. Tseng, P.H. Hsu and T.S. Chang, "A 124 Mpixels/s VLSI design for histogram-based joint bilateral filtering", IEEE Transactions on Image Processing, vol. 20, no. 11, pp. 3231–3241, 2011.
[6] L. Grady, "Random Walks for Image Segmentation", IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 28, no. 11, pp. 1768-1783, 2006.
[7] Anat. Levin, Dani. Lischinski, and Yair. Weiss, "Colorization Using Optimization", ACM Transactions on Graphics, vol. 23, no. 3, pp. 689-694, 2004.
[8] M. Kass and J. Solomon, "Smoothed Local Histogram Filters", ACM Transactions on Graphics, vol. 29, pp. 101-110, 2010.
[9] L. I. Rudin, S. Osher, and E. Fatemi, "Nonlinear Total Variation Based Noise Removal Algorithms", Physica D, vol. 60, nos. 1-4, pp. 259-268, 1992.
[10] Y. Li and S. Osher, "A New Median Formula with Applications to PDE Based Denoising", Communications in Mathematical Sciences, vol. 7, no. 3, pp. 741-753, 2009.
[11] Y. Wang, J. Yang, W. Yin, and Y. Zhang, "A New Alternating Minimization Algorithm for Total Variation Image Reconstruction", Society for Industrial and Applied Mathematics Journal on Imaging Sciences, vol. 1, no. 3, pp. 248-272, 2008.
[12] CarstenGarnica, Frank Boochs and Marek Twardochlib, "A new approach to edge- preserving smoothing for edge extraction and image segmentation", International Archives of Photogrammetry and Remote Sensing, vol. 33, Part B3,2000.
[13] Li-Cheng Chiu and Chiou-ShannFuh, "A Robust Denoising Filter with Adaptive Edge Preservation", Springer-Verlag Berlin Heidelberg, pp. 923–926, 2008.
[14] P. Bakker, L. J. , Van Vliet and P. W. Verbeek, "Edge preserving orientation adaptive filtering", IEEE Computer Society Conference on Computer Vision and Pattern Recognition, pp. 1063-6919, 2009.
[15] Kaiming He, C. Rhemann, C. Rother, X. Tang, and J. Sun, "A global sampling method for alpha matting", in Proceedings of IEEE Conference on Computer Vision and Pattern Recognition, pp. 2049–2056, 2011.
[16] Y. Ding, J. Xiao, and J. Yu, "Importance filtering for image retargeting", in Proceedings of IEEE Conference on Computer Vision and Pattern Recognition, pp. 89–96, 2011.
[17] Chieh-Chi Kao, Jui-Hsin Lai and Shao-Yi Chien, "VLSI architecture design of Guided Filter for 30 Frames/s full HD video", IEEE Transactions on Circuits and Systems for Video Technology, vol. 24, no. 3, 2014.