Implementation and Analysis of the Bilateral and Joint Bilateral Filter

S.Ellershaw

Image Processing Coursework, Software Methodologies

Submitted: December 14, 2018

A review of the literature found the bilateral filter to be a non-iterative noise reducing, edge preserving image filter. Modification of the underlying algorithm gives the joint bilateral filter which additionally merges high frequency information from one image to another. The properties and effect of the filters' parameters were shown via their implementation. Applications of the joint bilateral filter in filtering Microsoft's Kinect depth maps and upsampling are also discussed.

I. INTRODUCTION

The bilateral filter was originally described by Aurich and Weule in 1995 but not given it's current name for a further 3 years in the work by Tomasi and Manduchi [1]. At the time of this work an area of active research was in finding alternatives to the Gaussian low-pass filter.

Gaussian Filter

The Gaussian filter relates the Gaussian blur intensity, $GB[I]$, of a central pixel, p, in a neighbourhood, S, to the intensities, I_q, of the pixels in the neighbourhood, q, and their distances from p.

$$GB[I]_p = \sum_{q \epsilon S} G_\sigma(|p-q|) I_q, \tag{1}$$

where the G_σ is the Gaussian function defined around $\mu = 0$ and σ is varied to optimise the filters effect [1]

$$G_\sigma = \frac{1}{2\pi\sigma^2} e^{-\frac{x^2}{2\sigma^2}}. \tag{2}$$

The Gaussian's noise reduction properties can be shown to be quantitatively effective [2]. However the underlying assumption that the spatial variation of pixel intensity is low and so can be averaged in a local neighbourhood fails at edges. Therefore efforts were made to improve the Gaussian filter.

Bilateral Filter

At the time of Tomasi and Manduchi's work a popular alternative to the Gaussian filter was that of anisotropic diffusion. This however required the iterative solving of partial differential equations which led to issues of stability and efficiency. The bilateral filter presented a non-iterative solution to the Gaussian edge blurring problem, via modification to Eq.1 [3]. A further Gaussian term relating the differing pixel intensities, $I_p - I_q$, was added mirroring that of the distance term.

$$BF[I]_p = \frac{1}{W_p} \sum_{q \epsilon S} G_{\sigma_s}(|p-q|) G_{\sigma_r}(|I_p - I_q|) I_q, \tag{3}$$

where W_p is the normalisation factor ensuring there is no overall change in intensity between the original and filtered image

$$W_p = \sum_{q \epsilon S} G_{\sigma_s}(|p-q|) G_{\sigma_r}(|I_p - I_q|). \tag{4}$$

It is of note that the filter now depends on two parameters σ_s and σ_r. These can be altered to optimise the effect of the filter depending on the application [1].

As previously discussed the Gaussian filter does not preserve edges. For grey scale images this causes a blurred effect. However for coloured images, if the RGB channels are smoothed independently auras of unrelated colours can appear as the RGB channel intensities are not necessarily related. The bilateral filter however combines the RGB channels into a single intensity. This was originally done using the CIE-Lab color space which maps colors by human perception of the closeness of colours rather than the traditional RGB map[3]. However OpenCV in its implementation, which was used in this investigation, uses L1 norm to compute differences between colors [4]. This is a simpler measurement of the average of the absolute values of the colours [5].

Joint Bilateral Filter

The bilateral filter was further modified for a number of applications, such as the joint bilateral filter. This was independently developed by teams at Microsoft [6] and MIT [7]. The aim of this altered filter was to combine the best features of flash and non flash images in a low light environment. Flash photography has a greater signal to noise ratio allowing finer high frequency details, which would be normally hidden by noise, such as texture of objects to be recorded. Also by illuminating surfaces with a point light surface details can be enhanced. Furthermore, photography techniques such as using a long exposure time, leading to blur, or a larger aperture, decreasing depth of field, are no longer required. Flash can destroy the ambient lighting in low light environments though [7].

A popular approach in image processing in the early 2000's was to combine multiple images to produce a single improved image, for example combining multiple images at different camera exposures to produce a single high dynamic range image [8]. Building upon this idea the bilateral filter formula, Eqn.3, was altered to use the flash image, F, to compute the edge-stopping function due to its better estimate of the high frequency details of the scene.

$$JBF[I]_p = \frac{1}{W_p}\sum_{q\epsilon S} G_{\sigma_s}(|p-q|)G_{\sigma_r}(|F_p - F_q|)I_q, \quad (5)$$

W_p is adjusted accordingly to normalise the function. This approach theoretically maintains the edge preserving and noise reducing properties of the bilateral filter and enhances this with the high frequency information from the flash image [6].

II. METHODS

An investigation into the effect of the parameters σ_s and σ_r on the bilateral filter was performed, with a neighbourhood of 50 pixels. This was done using the OpenCV implementation of the filter [9] in python and using the two test images, see Appendix A Fig.4

The the joint bilateral filter was then implemented using Eqn.5. Two different approaches were taken in the original papers when calculating the colour intensity of a pixel. The Microsoft team analysed each RGB channel independently [6], whilst the MIT combined the RGB channels as shown in Eq.6, where R, G, B represent the intensity values for the red, green and blue channels respectively [7].

$$I_p = \frac{R}{G+B+R} + \frac{G}{G+B+R} + \frac{B}{G+B+R}. \quad (6)$$

Reflecting on the work on the original bilateral filter and the beneficial effect that combining the color channels was found to have on edges when filtering coloured photos [7] it was decided to follow the MIT approach in this implementation

The implementation was tested on the original test images of the Microsoft paper [6], see Appendix A Fig.5A and B, with a pixel neighbourhood of 9 pixels and the σ parameters were adjusted in order to maximise the beneficial effect of the filter.

III. RESULTS

Grids, mimicking the results from Tomasi and Manduchi's original paper [3], were produced from Fig.4B. These grids show the effect of independently increasing σ_d values across rows and σ_s down columns. These grids are shown in Appendix A Fig.6. Panels of special interest are shown in Fig.1.

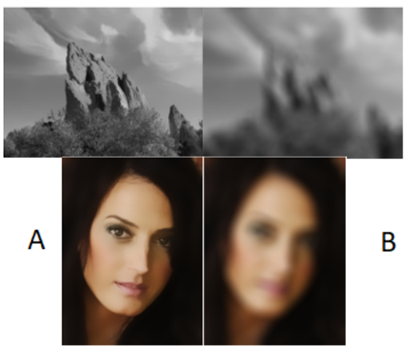


FIG. 1: Selected panels from Fig.3 showing; A the 'cartoon' effect of large σ_r and B the Gaussian blur of large σ_s

A close up section of the color image of Fig.1A is also shown in Fig.2. This shows the effect of the CV2 bilateral filter on coloured edges.

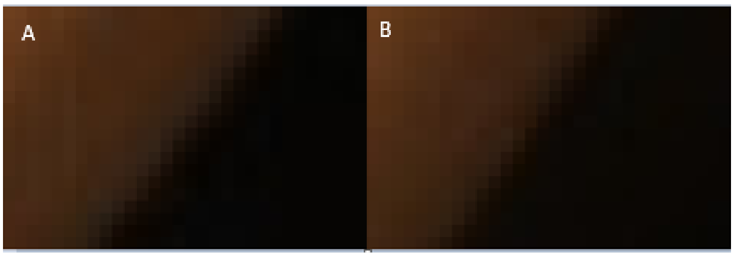


FIG. 2: Zoomed in section of right cheekbone and hair of Fig.4B. A shows before filtering and B after filtering, $\sigma_s = 60$ and $\sigma_r = 200$

The output from the implementation of the joint bilateral filter is shown in Fig.3, it also shown in comparison with the test images in Appendix A Fig.5. Extensive variation of the filter's parameters was conducted in order to optimise the output. It was found for the computation time to be reasonable the maximum pixel neighbourhood was 9. Under this condition $\sigma_s = 1.5$ and $\sigma_r = 1$ optimised the output image.

FIG. 3: Optimised output of joint bilateral filter implementation, $\sigma_s = 1.5$ and $\sigma_r = 1$

IV. DISCUSSION

A. Bilateral Filter

A number of properties of the bilateral filter are shown in Fig.6. Firstly that increasing the σ_d parameter makes the filter act in a Gaussian fashion and the filtered image to tend towards Gaussian blur. This can be seen in the bottom right panels of both grids.

Secondly increasing the σ_s leads to a loss of fine texture and a 'cartoonish' image produced. However the original shading is maintained. This effect can be be seen in the sky of the bottom left panel of Fig.6A or the face in the central panel of the $\sigma_s = 200$ row.

It can also be seen that if either parameter has a low value the effect of the other parameter is mitigated, this is due to the bilateral filter being a product of the both the distance and intensity functions, as shown in Eq.(3).

A further feature of the bilateral filter on coloured images is it's preservation of edge colouring. Fig.2A shows the unfiltered image with an aura of light colour at the edge between the cheek and hair. This can be caused phenomena such as lens blurring and pixel averaging [3]. In Fig.2A this aura is removed and no extraneous colors are added. This shows that the implementation of the bilateral filter with L1 norm color measurement still maintains the color edge preservation of the original implementation using the CIE-Lab space.

B. Joint Bilateral Filter

Fig.5 shows a number of the properties of the joint bilateral filter. Firstly the lighting and colour tones of the ambient atmosphere from the non-flash image are conserved. Secondly, the noise from the non-flash image has been removed and textural details such as that of the leather sofa and stone wall has been added. Finally, the edges of objects such as the jugs and edge of the sofa are well preserved. However, in this implementation the glare from the lights or camera flash was not handled. Therefore further processing would be advantageous.

The implementation of the filter took a naive approach calculating the intensity of each pixel individually. Although a significant increase in efficiency was gained by calculating a fixed the Gaussian mask as this was invariant all pixels as σ_d was held constant. The consequence of this naive approach was the computation of the filtered picture even for a small neighbourhood of 9 pixels took approximately 2 minutes.

In reviewing the literature for the bilateral filter this brute force implementation has a complexity of $O(|S^2|)$ where S is the size of the neighbourhood. A number of more advanced techniques have been shown to have much lower complexities such as separable kernels with a complexity of $O(|S|\sigma_s)$ though can lead to artefacts such as streaks [10].

Development of faster algorithms has enabled the joint bilateral filter to be used in real time. For example, the Microsoft Kinect a low cost consumer sensor produces depth maps of environments however these are noisy and can have holes in. Therefore these maps must be filtered before the they can be used for for advanced tasks such as object recognition or multi-view rendering. A novel algorithm was proposed in which the joint bilateral filter was used to couple the depth map to an aligned colour photo to fill the holes in the map. The joint bilateral filter's properties accounting for both colour and spacial separation made it ideal for this application. This algorithm was shown to return higher quality maps, in particular at edges, than the filters used at the time [11].

The joint bilateral filter can also be used in the process of upsampling. Many filtering process such as tone mapping of HDR images require a smooth global function describing a parameter to be found resulting in their complexity being linear with space. As digital images increased in size, due to advances in camera technology, the only reasonable way to continue these computations was to first down scale the image then perform the operation and finally upscale the image to its original resolution. Commonly interpolation kernels were used, but like the Gaussian filter, these assumed smoothness prior to interpolation and so caused blurring at edges. It has been shown that applying the joint bilateral filter to the downscaled image using the original high resolution image in the Gaussian intensity function could preserve these edges citepkopf2007joint.

V. CONCLUSIONS

In conclusion, the bilateral filter has noise reducing and edge preserving properties and can be optimised through variation of the σ parameters. The joint bilateral filter was successfully implemented and used to merge the high frequency features from a flash image to a non-flash image test whilst again preserving edges and reducing noise. Applications of the joint bilateral filter were found to include upsampling and filtering Kinetic depth maps.

References

[1] Sylvain Paris, Pierre Kornprobst, Jack Tumblin, and Frédo Durand. A gentle introduction to bilateral filtering and its applications. In *ACM SIGGRAPH 2007 courses*. ACM, 2007.

[2] Athanasios Papoulis. *Probability, random variables, and stochastic processes*. McGraw-Hill, 1991.

[3] Carlo Tomasi and Roberto Manduchi. Bilateral filtering for gray and color images. In *Computer Vision, 1998. Sixth International Conference on*. IEEE, 1998.

[4] OpenCV Dev Team. Domain transform filter. https://docs.opencv.org/3.0-beta/modules/ximgproc/doc/edge_aware_filters.html#jointbilateralfilter, 2018. [Online; accessed 20-November-2018].

[5] Stephen A Langer, Andrew CE Reid, SI Haan, RE Garcia, RC Lua, and VR Coffman. The oof2 manual. *NIST, US*, 2008.

[6] Georg Petschnigg, Richard Szeliski, Maneesh Agrawala, Michael Cohen, Hugues Hoppe, and Kentaro Toyama. Digital photography with flash and no-flash image pairs. In *ACM transactions on graphics (TOG)*. ACM, 2004.

[7] Elmar Eisemann and Frédo Durand. Flash photography enhancement via intrinsic relighting. In *ACM transactions on graphics (TOG)*. ACM, 2004.

[8] Paul E Debevec and Jitendra Malik. Recovering high dynamic range radiance maps from photographs. In *ACM SIGGRAPH 2008 classes*. ACM, 2008.

[9] OpenCV Dev Team. Image filtering. https://docs.opencv.org/2.4/modules/imgproc/doc/filtering.html?highlight=bilateralfilter#bilateralfilter, 2018. [Online; accessed 20-November-2018].

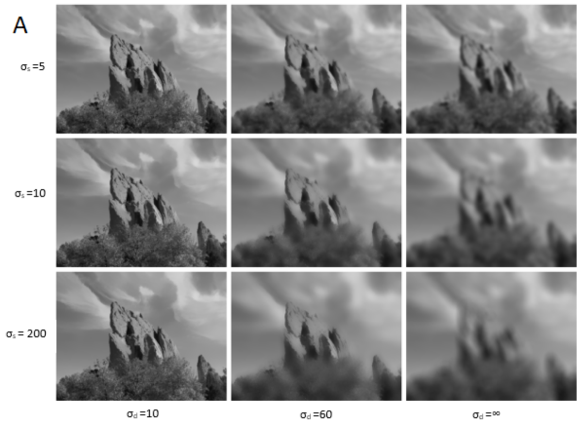
[10] Tuan Q Pham and Lucas J Van Vliet. Separable bilateral filtering for fast video preprocessing. In *Multimedia and Expo, 2005. ICME 2005. IEEE International Conference on*. IEEE, 2005.

[11] Anh Vu Le, Seung-Won Jung, and Chee Sun Won. Directional joint bilateral filter for depth images. *Sensors*, 2014.

APPENDIX A: IMAGE

FIG. 4: Grayscale (A) and color (B) test images used for bilateral filter tests

FIG. 5: A and B are the non flash and flash test images respectively. C is the output image from the joint bilateral filter implementation.



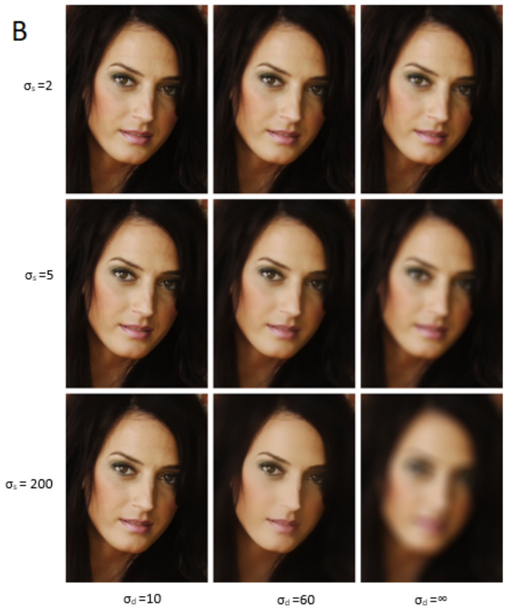


FIG. 6: Results from varying the σ parameters of openCv's bilateral filter function for both grey scale (A) and color (B) images