IOSR Journal of Computer Engineering (IOSR-JCE)
e-ISSN: 2278-0661, p- ISSN: 2278-8727Volume 16, Issue 1, Ver. VI (Feb. 2014), PP 62-65
www.iosrjournals.org

Use Map EstimationFor Image Sharpening

Mrs. Kshirsagar P. G., Dr. Ashok Gaikwad

Abstract: *This paper presents a various paradigms for estimating a single latent sharp image given multiple blurry and/or noisy observations. Whether employing it to make an unusable image good, a good image better or giving a great image that extra edge, it produces unparalleled sharpening and deblurring results that add distinction and definition. From a blurred image to recover a sharp version is a long-standing inverse problem. We point out the weaknesses of the deterministic filter and unify the limitation. Theoretically and experimentally we analyze image deblurring through three paradigms are: 1) The filter determination 2) Estimation by Bayesian 3) Alpha tonal correction methods. The resulting paradigms, which require no essential tuning parameters, can recover a high quality image from a set of observations containing potentially both blurry and noisy examples, without knowing a priori the degradation type of each observation. Our goal is to reveal the limitations and potentials of recent methods when dealing with quite large blurs and severe noise. Experimental results on both synthetic and real-world test images clearly demonstrate the ability to produce a desired or intended result of the proposed method.*

Keywords: *Step-Edge Based Filter (SEBF), Sample Number (SN), Bounded-Input Bounded-Output (BIBO).*

I. Introduction

Image blur and noise are the very famous factors arises in photography and damage the photograph, which causes especially because of limited amount of light. Image denoising produce same problem because of duplication of noisefree image and noise images.Image in the presence of noise is an inherently ill-posed problem. The observed blurred image only provides a partial constraint on the solution—there exist many "sharp" images that when convolved with the blur kernel can match the observed blurred and noisy image. Image denoising presents a similar problem due to the ambiguity between the high-frequencies of the unobserved noisefree image and those of the noise. Thus, the central challenge in deconvolution and denoising is to develop methods to disambiguate solutions and bias the processes toward more likely results given some prior information.

Image filtering allows you to apply various effects on photos. The type of image filtering described here uses a 2D filter similar to the one included in Paint Shop Pro as User Defined Filter and in Photoshop as Custom Filter.The trick of image filtering is that you have a 2D filter matrix, and the 2D image. Then, for every pixel of the image, take the sum of products. Each product is the color value of the current pixel or a neighbor of it, with the corresponding value of the filter matrix. The center of the filter matrix has to be multiplied with the current pixel, the other elements of the filter matrix with corresponding neighbor pixels.This operation where you take the sum of products of elements from two 2D functions, where you let one of the two functions move over every element of the other function, is called Convolution or Correlation. The difference between Convolution and Correlation is that for Convolution you have to mirror the filter matrix, but usually it's symmetrical anyway so there's no difference.

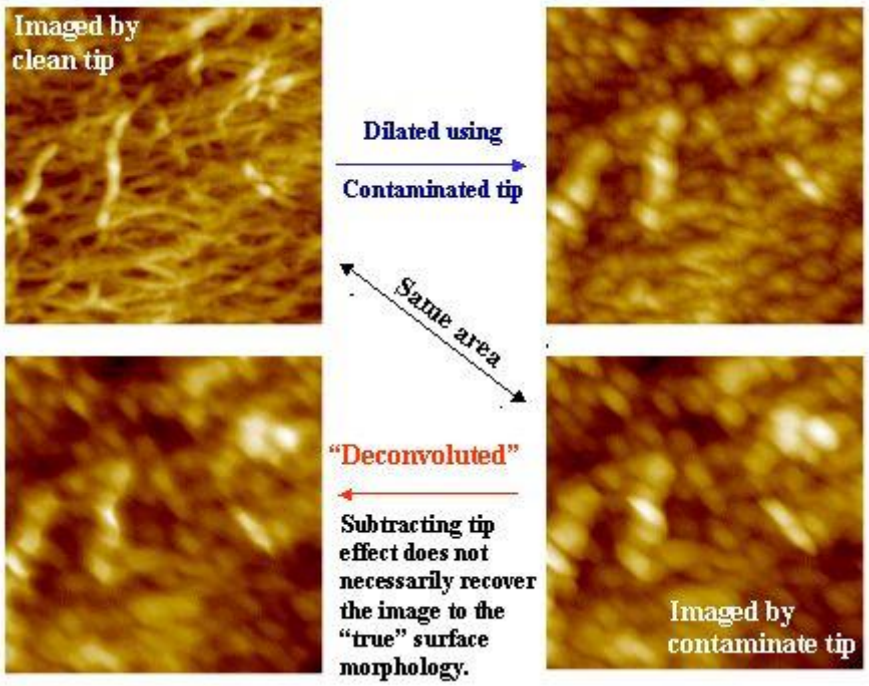
The filters with convolution are relatively simple. More complex filters, that can use more fancy functions, exist as well, and can do much more complex things (for example the Colored Pencil filter in Photoshop), but such filters aren't discussed here. The 2D convolution operation requires a 4-double loop, so it isn't extremely fast, unless you use small filters. Here we'll usually be using 3x3 or 5x5 filters.

II. Method Of Paradigm

The deterministic filter can be modeled as deterministic function F of the input blurred image I:F(I)=L,with L denoting the output sharp image. One of the most well-known approaches in this paradigm is unsharp masking, of which the basic idea is to reduce the low frequency first, and then high-lights the high-frequency components. The performance may be different according to the adopted high-pass filters and the adaptive edge weights.

This approach assumes that the blurred edges do not shift too far away from the latent sharp edges; thus, it can handle only the defocus blurs and very small motion blurs. For very large blurs, the image narrow edges or details are severely damaged and very difficult to restore. A practical solution is to detect and restore large step edges explicitly or implicitly, which we call the Step-Edge Based Filter (SEBF).Explicit SEBF rest locates the step edge and then propagates the local intensity extreme toward the edge. Implicit SEBF performs edge detection and restoration in a single step, based on zero crossings of high- pass filters. Commonly used

implicit SEBFs include the shock filter, the backward diffusion and many other adapted versions .There are following advantages in SEBF as compared with Bayesian estimation i.e. our second paradigm 1) The SEBF can handle various blurs without adaptation because it is independent of the blurring processes (blur models), and 2)The performance of the SEBF is not constrained by the sample number(SN) because it depends on image local features rather than sufficient samples.



BUILT IN FILTER

Core Image comes with dozens of built-in filters ready to support image processing in your app. their characteristics, their iOS and OS X availability, and shows a sample image produced by the filter. The list of built-in filters can change, so for that reason, Core Image provides methods that let you query the system for the available filters .A filter category specifies the type of effect—blur, distortion, generator, and so forth—or its intended use—still images, video, non square pixels, and so on. A filter can be a member of more than one category. A filter also has a display name, which is the name to show to users and a filter name, which is the name you must use to access the filter programmatically. Most filters have one or more input parameters that let you control how processing is done. Each input parameter has an attribute class that specifies its data type, such as NS Number. An input parameter can optionally have other attributes, such as its default value, the allowable minimum and maximum values, the display name for the parameter, and any other attributes .For example, the CI Color Monochrome filter has three input parameters—the image to process, a monochrome color, and the color intensity. You supply the image and have the option to set a color and its intensity. Most filters, including the CI Color Monochrome filter, have default values for each non image input parameter. Core Image uses the default values to process your image if you choose not to supply your own values .

Using Transition Effects

Transitions are typically used between images in a slide show or to switch from one scene to another in video. These effects are rendered over time and require that you set up a timer. The purpose of this section is to show how to set up the timer. You'll learn how to do this by setting up and applying the copy machine transition filter to two still images. The copy machine transition creates a light bar similar to what you see in a copy machine or image scanner. The light bar sweeps from left to right across the initial image to reveal the target image.This filter looks like before, partway through, and after the transition from an image of ski boots to an image of a skier. (To learn more about specific input parameter of the CI Copy Machine.

Linear Filters

To review and compare the two types of filtering, the first step is to briefly describe the attributes that comprise linear filtering.Several principles define a linear system. The first two are the basic definitions of linearity. If a system is defined to have an input as x[n] = ax[n1] + bx[n2], then the linear system response is y[n] = ay[n1] + by[n2]. This is known as the superposition property, and is fundamental to linear system design.

The second property is shift invariance. If y[n] is the response to a linear, shift-invariant system with input x[n], then y[n-n0] is the response to the system with input x[n-n0].In addition, two extra conditions are imposed, causal and stable. The causal condition is needed when considering systems in which future values are not known (for example, in video streaming). It is possible to consider a system that is not causal when looking at captured images with samples before and after the target location (for example, in a buffered version of an image frame). Stability is imposed to keep a filter's output from exceeding a finite limit, given an input that also does not exceed a finite limit. This is called the Bounded-Input Bounded-Output (BIBO) condition.For most cases, a system is evaluated in the spatial frequency domain. To accomplish this, the convolution theorem is used, providing the necessary tools to evaluate frequency domain information. If x[n] and h[n] are two sequences, their convolution is defined as shown in Equation 1.

$$y[n] = \sum_{k=-\infty}^{\infty} x[k]h[n-k]$$

Equation 1

The corresponding frequency response is shown in Equation 2.

$$Y\left(e^{-i\omega}\right) = X\left(e^{-i\omega}\right)H(e^{-i\omega})$$

Equation 2

In Equation 2, e-**iw**is the frequency domain representation and *w* is the frequency variable from -p to p. This fundamental relationship describes the response of a filter in terms of frequency – low pass, high pass, band pass, and so on. Depending on the nature of the filter kernel h[n], a wide variety of responses can be realized for any image data set.

A typical low-pass filter with 25 .The idea of a low-pass filter is to preserve low-frequency information and reduce or eliminate high-frequency information in an image. It blurs edges but keeps smooth areas of an image intact. In a similar manner, high-pass filters preserve edges and other high-frequency information but filter low-frequency regions of an image.

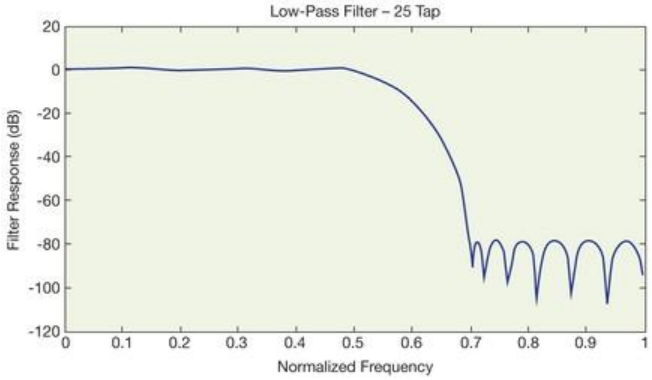


Figure 1: A typical low-pass filter maintains low-frequency elements and reduces or removes high-frequency elements in an image.

NONLINEAR FILTERS

Nonlinear filters have quite different behavior compared to linear filters. For nonlinear filters, the filter output or response of the filter does not obey the principles outlined earlier, particularly scaling and shift invariance. Moreover, a nonlinear filter can produce results that vary in a non-intuitive manner.The simplest nonlinear filter to consider is the median or rank-order filter. In the median filter, filter output depends on the ordering of input values, usually ranked from smallest to largest or vice versa. A filter support range with an odd number of values is used, making it easy to select the output.For example, suppose a filter was based on five values. In the region of interest, x0..x4, the values are ordered from smallest to largest. The value at position 2 is selected as the output. Consider the case at low frequency; all the values are the same or close to it. In this case, the value selected will be the original value ± some small error. In the case of high frequency, such as an edge,

the values on one side of the edge will be low and the values on the other side will be high. When the ordering is done, the low values will still be in the low position and the high values will still be in the high position. A selection of the middle value will either be on the low side or the high side, but not in the middle, as would be the case using a linear low-pass filter. The median filter is sometimes called an edge-preserving filter due to this property. It is useful in removing outliers such as impulse noise.

SELECTING THE RIGHT FILTER

Both filter types have their place in image processing functions. In a typical pipeline for real-time image processing, it is not uncommon to have dozens of both types included to form, shape, detect, and manipulate image information. Moreover, each of these filter types can be parameterized to work one way under certain circumstances and another way under a different set of circumstances using adaptive filter rule generation.Filtering image data is a standard process used in almost all image processing systems. The goals vary from noise removal to feature abstraction. Linear and nonlinear filters are the two most utilized forms of filter construction. Knowing which type of filter to select depends on the goals and nature of the image data. In cases where the input data contains a large amount of noise but the magnitude is low, a linear low-pass filter may suffice. Conversely, if an image contains a low amount of noise but with relatively high magnitude, then a median filter may be more appropriate. In either case, the filter process changes the overall frequency content of the image.

III. Conclusion

Recovery of the sharp image from a blurred one is an important and long- standing problem for many applications. In thispaper, we have rest analyzed the potentials and limitationslatent in recent methods when handling quite large blurs andsignificant noise. While our method. out performs state-of-theart methods both in robustness to noise and the capability of handling quite large blurs, it is still limited by the images dominated by narrow edges. Recovering the totally damaged narrow edges is still a very challenging problem faced by state-of- THE-ART METHODS AND SHOULD ATTRACT FURTHER RESEARCH EFFORTS

ACKNOWLEDGMENT

The authors wish to thank all the respective author.

References

[1]. Gonzalez and Woods, Digital image processing, 2nd edition, Prentice Hall, 2002. Chap 4 Sec 4.3, 4.4; Chap 5 Sec 5.1 – 5.3(pages 167-184 and 220-243)
[2]. C. Liu,W. T. Freeman, R. Szeliski, and S. B. Kang. Noise estimation from a single image. In Proc.IEEE Conf. Computer Vision and Patter Recognition, pages 901–908, Jun 2006.
[3]. BEN-EZRA, M., AND NAYAR, S. K. 2004. Motion-based motion deblurring. TPAMI 26, 6, 689–698.
[4]. JIA, J. 2007. Single image motion deblurring using transparency. In CVPR.
[5]. LUCY, L. 1974. Bayesian-based iterative method of image restoration. Journal of Astronomy 79, 745–754.
[6]. RASKAR, R., AGRAWAL, A., AND TUMBLIN, J. 2006. Coded exposure photography: Motion deblurring using fluttered shutter.ACM Transactions on Graphics 25, 3, 795–804.
[7]. YUAN, L., SUN, J., QUAN, L., AND SHUM, H.-Y. 2007. Image Deblurring with Blurred/Noisy Image Pairs. In SIGGRAPH