Reduced noise and ringing for image processing

(Sam) Yu Wang, Patrick Kano, Jonathan Cain, and Dave Douglas

The problem - Noise of image processing

Using image processing to enhance image resolution has many commercial and defense applications. Popular image processing algorithms approaches are inverse filters[1,2] and iterative algorithms. Specific examples include Wiener deconvolution, Regularized Least Squares, Linear Least Squares, and the Nearest Neighbor Pixel Deconvolution (NNPD) [3,4]. In this project, the focus is on inverse filters and how to accommodate the major issue of image noise.

An inverse filter works by taking the Fourier transform (FT) of an image and dividing it by the FT of the point spread function (PSF). The point spread function is specific to the imaging device and describes the response to a perfect point source. However, during division in Fourier space, some denominators may become zero, therefore we must introduce thresholds. Proper selection of the threshold value is key, if too low it may increase noise and introduce ringing.

Figure 1. Image from Hubble's HUDF (left) with selected Region Of Interest – ROI (right)

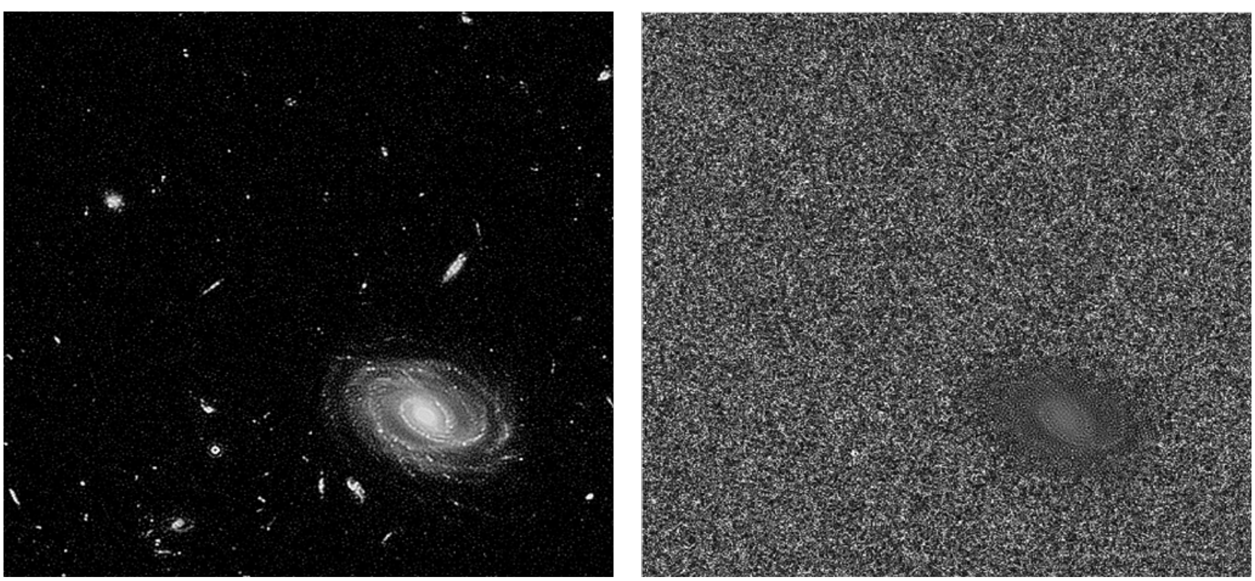


Figure 2. ROI processed by NNPD with larger threshold (left) and smaller threshold (right)

In our discussion, we'll use NNPD as an example to address the problem, though it can be applied to other image processing methods. Figure 1 shows an example of an HUDF image taken by Hubble Space Telescope (Credit: NASA/ESA/STSCI); we select a Region of Interest (ROI) from its green channel.

The ROI is then processed by NNPD with a larger threshold, the result is shown in Figure 2. The image at left is with a proper threshold selection and we can see that the processed image has much higher resolution and shows more hidden details than the Hubble image. If, however we want to further increase the resolution by reducing the threshold, the noise becomes overwhelmed, as shown on the right.

Mathematical description of the problem

In mathematics, deconvolution is the inverse of convolution[5]. Both operations are used in signal processing and image processing. For example, it may be possible to recover the original signal after a filter (convolution) by using a deconvolution method with a certain degree of accuracy. Due to the measurement error of the recorded signal or image, it can be demonstrated that the worse the signal-to-noise ratio (SNR), the worse the reversing of a filter will be hence, inverting a filter is not always a good solution as the error amplifies. Deconvolution offers a solution to this problem.

The point spread function (PSF) describes the response of a focused optical imaging system to a point source or point object[6]. A more general term for the PSF is the system's impulse response; the PSF is the impulse response or impulse response function (IRF) of a focused optical imaging system. The PSF in many contexts can be thought of as the extended blob in an image that represents a single point object, that is considered as a spatial impulse. In functional terms, it is the spatial domain version (i.e., the inverse Fourier transform) of the optical transfer function (OTF) of an imaging system.

In general, a detected image I^d can be written as convolution of the object image I^o and the PSF without noise:

$$I^d = I^o \ast PSF \qquad (1).$$

Now we want to recover the original object image I^o. Using the FT and the Convolution theorem, we have:

$$F^d = F^o \cdot \Delta \qquad (2),$$

and $$F^o = F^d / \Delta \qquad (2a).$$

Here F^o , F^d , and Δ are the FT of I^o , I^d, *and PSF* respectively. Formula (2a) shows how we can enhance the detected image to the original object image in the Fourier domain. Since Δ is the denominator in (2a), if Δ goes to zero, we have a problem. A common method to avoid this problem is to replace Δ with some threshold value, that may generate noise.

In the following discussion, we will use the NNPD as an example to show how we can use contour integral method to solve the zero-denominator problem, though apparently the discussion is suitable for any inverse filter type of image processing method.

The NNPD assumes the PSF is symmetric and uses nearest neighbor pixel model to write the PSF according to the distance between any pixel to the PSF center, as shown in Figure 3: the PSF's center pixel has a value of a_0; it's 1st order of nearest 4 neighbor pixels (left, right, up and down) have value of a_1; it's 2nd order of 4 nearest neighbors have value of a_2, … etc. Now this PSF can be written as:

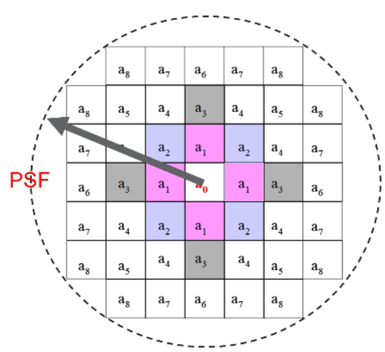


Figure 3. The nearest neighbor pixel model.

$$PSF = a_0\delta_{i,j} + a_1\,(\delta_{i,j+1} + \delta_{i,j-1} + \delta_{i+1,j} + \delta_{i-1,j}) + a_2(\delta_{i+1,j+1} + \delta_{i+1,j-1} + \delta_{i-1,j=1} + \delta_{i-1,j-1}) + \ldots \qquad (3)$$

Here the $\delta_{i,j}$, $\delta_{i,j+1}$, … are the Kronecker delta functions.

For a NxN image, using the Shift theorem of FT, the equation (2) becomes:

$$F^d_{lm} = F^0_{lm}\left[a_0 + a_1\left(e^{\frac{2\pi i m}{N}} + e^{\frac{-2\pi i m}{N}} + e^{\frac{2\pi i l}{N}} + e^{\frac{-2\pi i l}{N}}\right) + a_2\left(e^{2\pi i\frac{(l+m)}{N}} + e^{-2\pi i\frac{(l+m)}{N}} + e^{2\pi i\frac{(l-m)}{N}} + e^{-2\pi i\frac{(l-m)}{N}}\right) + \cdots\right] \quad (4)$$

0r $$F^0_{lm} = F^d_{lm}/[a_0 + 2a_1(\cos\frac{2\pi m}{N} + \cos\frac{2\pi l}{N}) + 2a_2[\cos\frac{2\pi(l+m)}{N} + \cos\frac{2\pi(l-m)}{N}] + \cdots] \quad (4a)$$

Finally apply the inversed FT to (4a), a pixel of a recovered object image I^0_{jk} can be written as:

$$I^0_{jk} = \frac{1}{N}\sum_{l=0}^{N-1}\sum_{m=0}^{N-1}\frac{F^d_{lm}}{\Delta_{lm}} e^{2\pi i\frac{(jl+km)}{N}} \quad (5)$$

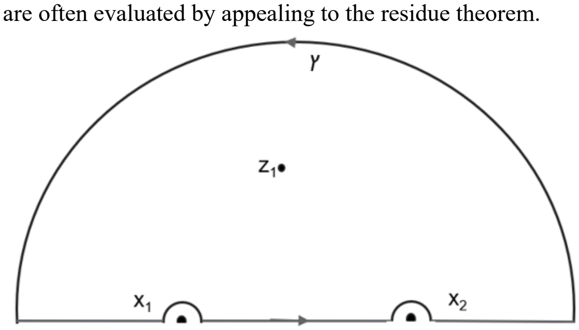
Here the F^d_{lm} is the Fourier component of the detected image, and

$$\Delta_{lm} = a_0 + a_1\cdot 2[\cos(\frac{2\pi l}{N}) + \cos(\frac{2\pi m}{N})] + a_2\cdot 2[\cos(2\pi\frac{l+m}{N}) + \cos(2\pi\frac{l-m}{N})]+\ldots \quad (6)$$

However, the denominator Δ_{lm} may also become zero, so we must set a threshold for Δ_{lm} if evaluating this sum directly   Under certain conditions, a discrete Fourier transform can be approximated by a continuous Fourier transform, allowing the possibility of continuing the integrand into the complex plane and using contour integration to evaluate the enhanced image. Poles on the real line produced by zeroes of Δ_{lm} require careful treatment.

Our mathematical solution of the problem

In complex analysis, contour integration is a method to calculate integrals along a path in the complex plane. Contour integrals are often evaluated by appealing to the residue theorem.



In complex analysis, the residue theorem, sometimes called Cauchy's residue theorem, is a powerful tool to evaluate line integrals of analytic functions.

https://en.wikipedia.org/wiki/Residue_theorem

$$\oint_\Gamma dz\, f(z) = \pm 2\pi i \sum_n \mathrm{Res}\left[\, f(z)\,\right]_{z=z_n}$$

Fig 4. A typical path of a contour.

As an example, let's look at our formula (5). In the formula (5), assuming the summations are well-approximated as integrals, we can consider the natural continuation of the Fourier components F^d_{lm} into the complex plane, where the zero points of the denominator are poles lying along the real axis in the complex domain. By carefully choosing an appropriate closed contour, we should be able to calculate the inverse Fourier transform.

Since we are dealing with finite 2d images, the Fourier transform has the form of a double summation. Contour integration does not have a 2d equivalent; we therefore only approximate the first summation with an integral while keeping the second sumation unchanged.

For example, in formula (5), we treat the summation over *m* as an integral, make a contour path to calculate it, but keep the summation over *l* as a summation to add up all the integration values over *m*. If for each *l*, when *m=m'*, there is a pole, we then draw a semi-circle contour path in a way that the contour goes along the real axis, makes a small semi-circle around each pole, and finally close the contour by making a large semi-circle to go back to the start point on the real axis. By computing the contour integral, which is a combination of the integral along the real axis, the integral of the large semi-circles, and integrals of these small semi-circles around the poles, we can finally obtain the enhanced image pixel value I^0_{jk} without assuming any threshold. Any poles lying within the closed contour contribute their residues to the computation via the residue theorem.

By applying the contour integrals to NNPD, we can enhance image resolution without introducing any artificial threshold. Rather, it may be possible to use the values of the PSF to properly select the contour integral. This approach could then in principle be applied to other image processing methods to reduce noise and ringing.

Workshop Goals:

The objective of this workshop problem will be to devise a process that recovers an image corrupted by noise. The team should explore promising approaches which are robust against the division-by-zero issue described above, such as the contour integration approach described or other approaches. While we have shown that image processing noise and ringing can be reduced by applying the contour integrals to image processing methods, there are still some details and unsolved problems. For example: how do we locate poles in a discrete Fourier domain? How do we deal with poles falling in between two pixels; should we generate a new sub-pixel image for better localization of the poles? How do we make an optimized choice between image resolution and noise level? How do we recover a corrupted image with fast speed? Are there any constraints of using this method? What does the noise structure look like?

References:

1. J.L. Starck, E. Patin and F. Murtagh, “Deconvolution in Astronomy: A Review”, The Astronomy Society of Pacific., 114, 1051-1169 (2002).
2. Alan C. Bovik and Scott T. Acton, “The Essential Guide to Image Processing” Academic Press, 2nd ed., p225-239 (2009).
3. Yu Wang and Yujing Lu "Enhance image resolution with nearest neighbor pixel deconvolution", Proc. SPIE 10998, Advanced Optics for Imaging Applications: UV through LWIR IV, 109980S (14 May 2019).
4. Yu Wang, United State Patent, US7912307 (2011).
5. https://en.wikipedia.org/wiki/Deconvolution
6. https://en.wikipedia.org/wiki/Point_spread_function