UNIVERSITY OF CALIFORNIA
SANTA CRUZ

A GENERAL FRAMEWORK FOR ITERATIVE REGULARIZATION IN IMAGE PROCESSING

A thesis submitted in partial satisfaction of the
requirements for the degree of

MASTER OF SCIENCE

in

ELECTRICAL ENGINEERING

by

Michael R. Charest Jr.

June 2006

The Thesis of Michael R. Charest Jr.
is approved:

Professor Peyman Milanfar, Chair

Professor Benjamin Friedlander

Professor Michael Elad

Dr. Sina Farsiu

Lisa C. Sloan
Vice Provost and Dean of Graduate Studies

Contents

List of Figures

List of Tables

Abstract

A General Framework for Iterative Regularization in Image Processing

by

Michael R. Charest Jr.

Many existing techniques for image restoration can be expressed in terms of minimizing a particular cost function. We address the problem of restoring images in a novel way by iteratively refining the cost function. This allows us some control over the tradeoff between the bias and variance of the image estimate. The result is an improvement in the mean-squared error as well as the visual quality of the estimate. We consider four different methods of updating the cost function and compare and contrast them. The framework presented here is extendable to a very large class of image reconstruction methods. The effectiveness of the proposed methods is illustrated on a variety of examples. Additionally, the convergence properties of the sequence of estimates produced by these iterative regularization methods lend themselves to a variety of useful applications. We discuss some of these applications and include examples to illustrate them.

Dedicated to my wife, Debra Charest,

whose love and support gets me through each day.

Acknowledgments

I would like to acknowledge the following people: my parents, Michael and Gail Charest, for raising me to have the work ethic necessary to complete this thesis; my advisor, Professor Peyman Milanfar for his direction, support, and constant feedback; my current and former lab mates in the Multi-Dimensional Signal Processing (MDSP) research group: Dr. Sina Farsiu, Davy Odom, Amyn Poonawalla, Hiro Takeda, and Dr. Morteza Shahram, for their help in all matters scholastic and beyond; and finally my thesis reading committee for taking the time out of their busy schedules give me valuable feedback on this document.

Chapter 1

Introduction

This chapter will introduce the ideas and concepts used throughout this thesis. We will begin by describing the image reconstruction problem that motivates this work, and a brief description of some of the previous work in this area will then be given.

1.1 Background

Digital imaging is used a vast array of technical fields including medical and satellite imaging. With the proliferation of cheap digital imaging equipment, digital imaging is common in a wide range of consumer applications as well. The goal in field of digital image restoration is to improve the quality of these images. Each imaging application has its own factors that contribute to degraded image quality. Medical applications are typically limited in the amount of exposure they can use. Too much exposure to X-rays, for example, can harm the patient. Satellite images often suffer from the effects of atmospheric blur. Thermal changes in the atmosphere cause slight changes in the air's refractive index, causing the image to appear out of focus. No matter the underlying cause, an always present difficulty

to overcome in any digital imaging application is the presence of noise and/or blur that degrades the image quality. This can give the image an undesirable appearance in consumer applications and can result in lost information in scientific applications.

We can represent an image $\mathbf{y}$ as:

$$\mathbf{y} = \mathbf{A}\mathbf{x} + \mathbf{v} \tag{1.1}$$

where $\mathbf{x}$ is the true image, $\mathbf{v}$ is zero-mean additive white noise that is uncorrelated to $\mathbf{x}$ and with no assumptions made on its distribution, and $\mathbf{A}$ is a blurring operator.

The goal of image reconstruction is to recover the true image $\mathbf{x}$ from the given data $\mathbf{y}$. Note that for ease of notation we carry out all of our analysis with vectors representing 1-D signals, though the treatment is valid in multiple dimensions. Also note that if $\mathbf{A} = \mathbf{I}$, where $\mathbf{I}$ is the identity matrix, then we have:

$$\mathbf{y} = \mathbf{x} + \mathbf{v} \tag{1.2}$$

The problem of recovering $\mathbf{x}$ from the data $\mathbf{y}$ in this simplified model is called "denoising." Let us consider this case first for simplicity; we will consider the more general case at the end of this chapter.

A very extensive body of work exists with many different techniques for performing image reconstruction. The Wiener filter is a well-known method for image reconstruction that considers images and noise as random processes and finds the estimate which has the minimum mean-squared error between itself and the true image [2]. The Wiener filter method assumes that the user has knowledge of the power spectrum of the true image; while this is seldom the case, it can typically be approximated. Wavelet methods process different frequency band decompositions of an image at multi-resolution scales [3]. These wavelet decomposed images have been used in an expectation maximization algorithm to

Denosing Technique	$J(\mathbf{x})$
Tikhonov [5]	$\frac{\lambda}{2}\|\mathbf{x}\|^2$
Total Variation [1], [6]	$\lambda\|\|\nabla\mathbf{x}\|\|_1$
Bilateral Filter [7], [8]	$\frac{\lambda}{2}\sum_{n=-N}^{N}\left[\mathbf{x}-\mathbf{S}^n\mathbf{x}\right]^T\mathbf{W}_{\mathbf{y},n}\left[\mathbf{x}-\mathbf{S}^n\mathbf{x}\right]$

Table 1.1: Various denoising techniques and their associated regularization terms.

estimate the true image [4]. We will focus on what are known as "regularization methods." Regularization is a very general technique for estimating $\mathbf{x}$ from the data $\mathbf{y}$ by minimizing a cost function of the form:

$$\begin{aligned}\widehat{\mathbf{x}} &= \arg\min_{\mathbf{x}} C(\mathbf{x},\mathbf{y}) \\ &= \arg\min_{\mathbf{x}} \left\{H(\mathbf{x},\mathbf{y}) + J(\mathbf{x})\right\}. \end{aligned} \tag{1.3}$$

For denoising problems the functional $H(\mathbf{x},\mathbf{y}) = \frac{1}{2}\|\mathbf{y}-\mathbf{x}\|^2$ is typically used, forcing the Euclidean distance between the measured data and the true signal to be minimized. The convex regularization functional $J(\mathbf{x})$ varies depending on the regularization method and enforces some additional constraint on the estimate. Some specific examples of cost functions of this type are given in Table **5.1**. The parameter λ in $J(\mathbf{x})$ controls the amount of regularization.

Tikhonov regularization is a classical method of this type [5] where the regularization functional requires energy of the estimate to be minimized. Total Variation (TV) regularization is a more recent image denoising method developed by Rudin, Osher, and Fatemei [9]. The TV regularization functional forces the estimate to be piecewise constant by requiring the L_1-norm of the image gradient to be minimized. The Bilateral Filter is an even more recent method first proposed as a nonlinear filter by Tomasi and Manduchi [10] and later connected to regularization by Elad [7]. For the regularization term corresponding to the Bilateral Filter in Table **5.1**, $\mathbf{S}^n$ is a matrix shift operator and $\mathbf{W}_{\mathbf{y},n}$ is a weight

matrix where the weights are a function of both the radiometric (or gray-value) and spatial distances between pixels in a local neighborhood ([7],[8]).

Figure **1.1** is an example of the types of estimates produced by using regularization methods (Bilateral Filter in this case). By looking at the estimate residual ($\mathbf{y} - \widehat{\mathbf{x}}$) we notice that we have removed some of the high frequency content of the image along with the noise. This is true more generally with other denoising techniques as well, as it is never possible to recover $\mathbf{x}$ exactly.

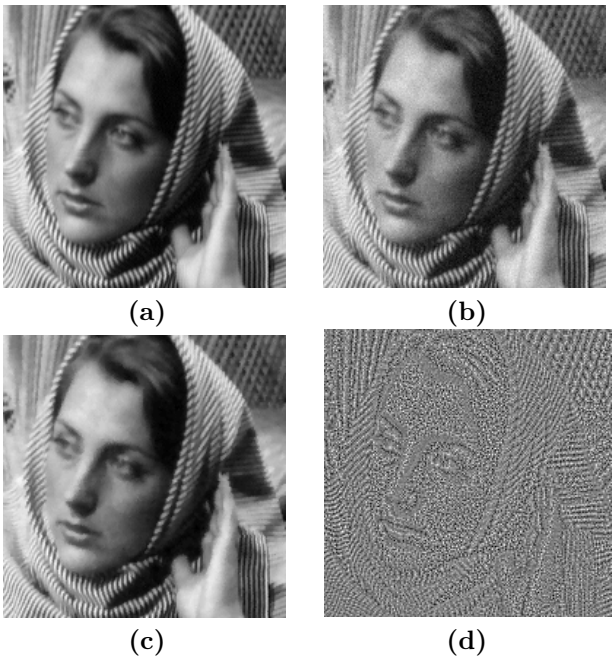


Figure 1.1: **(a)** Detail of the original 'Barbara' image **(b)** 'Barbara' with added white Gaussian noise of variance 29.5 (MSE= 29.50) **(c)** The result of minimizing the Bilateral cost function for the noisy image (b) (MSE= 19.30) **(d)** The residual (b)-(c)

1.2 Achievements of this work

In the next chapter we will turn our attention to recovering this lost detail through iterative regularization. One can think of iterative regularization in two ways. The first interpretation is that the lost detail is added back to the estimate in a controlled manner. At each iteration, the detail is extracted from the residual of the current solution. The manner is which the detail is extracted can vary, giving rise to a variety of different methods of performing iterative regularization.

The second interpretation of iterative regularization deals with the idea of modifying the initial cost function as the iterations proceed. Assuming that the parameters chosen for the initial cost function do not give the optimal solution, at each iteration, the process of iterative regularization uses the current solution to direct the cost function for the next iteration to a form that will give a better solution.

Receiving considerable attention lately (especially in [1]), is the connection between iterative regularization and the use of the Bregman distance (see Appendix A) as a penalty term. Though less intuitive and seemingly more complicated, the equivalent Bregman distance formulations are quite useful in proving the convergence properties of iterative regularization. In this work we show that there are in fact several ways to practice this idea. The work of Osher et. al ([1]) and Tukey ([11]) both fit into this general framework for iterative regularization. We herein name these methods " Osher's Method" and " Iterative Twicing Regularization" (ITR) respectively. We propose two additional iterative regularization methods that also fit this framework: " Summed Residual Regularization" (SSR) and " Iterative Unsharp Regularization" (IUR).

1.2.1 Content of this Thesis

In the coming chapters of this thesis, we will discuss the properties and applications of the different iterative regularization methods. By operating on the initial cost function in different ways, these methods offer a variety of options for improving the initial estimate. One of the methods, IUR, can be used for particular applications such as compression due to its formulation. Whereas the other methods cannot be used for these same applications. Through extensive examples we compare and contrast the different methods, and explore the benefits of each. We further describe and illustrate some of the practical application for these methods.

Additionally, we rigorously prove the convergence of these methods by relating them to the Bregman distance. Using the knowledge that these methods converge and the value to which they converge allows us to apply these methods to a variety of different applications. The statistical properties of the estimates produced by these methods are also considered.

There are still a number of major questions that are open to future work. One of these questions is regarding the relationship between the different iterative regularization methods. It is not clear what underlying property allows these very different methods to produce similar results. Another major question that has yet to be addressed is the details behind the bias-variance tradeoff that occurs in the estimates as one iterates using one of the iterative regularization methods in this framework. We experimentally show that there is a tradeoff taking place, but the underlying cause is still unknown.

Chapter 2

Iterative Regularization

2.1 Iterative Regularization Methods

The general framework that we present here seeks to improve the traditional regularized image estimate by iteratively updating the cost function of our choosing. We can express this as

$$\widehat{\mathbf{x}}_k = \arg\min_{\mathbf{x}} C_k(\mathbf{x}, \mathbf{y}). \tag{2.1}$$

The further benefit of such an approach is the level of control given to user by slowly adding the lost detail back to the initial regularized estimate. As we will see later, this will aid in the selection of cost function parameters. Normally the user must select some operating parameters for the cost function that is to be used. The iterative approach allows the initial parameter selection to be poor (that is, one that yields a poor regularized estimate either visually or in some metric such as mean-squared error) and still produce a satisfactory final estimate. As we will also see later, these methods asymptotically converge back to the initial data. This means that by iterating we can " undo" the highly non-linear function of

regularization. There are many practical applications for this property, such as compression, texture transfer, texture matching, etc.

A couple of iterative regularization methods exist separately in the regularization literature ([11], [1]). We tie these methods together as well as introduce some new methods that fit into the same framework.

We present four different algorithms for performing the cost function update. Each algorithm seeks to extract lost detail from the the residual $\mathbf{y} - \widehat{\mathbf{x}}_k$ in a unique way. Specifically:

$$
\begin{aligned}
1) \qquad \widehat{\mathbf{x}}_{k+1} &= \arg\min_{\mathbf{x}} \left\{ H\left(\mathbf{x}, \mathbf{y} + \sum_{i=1}^{k} (\mathbf{y} - \widehat{\mathbf{x}}_i)\right) + J(\mathbf{x}) \right\}, \\
2) \qquad \widehat{\mathbf{x}}_{k+1} &= \widehat{\mathbf{x}}_1 + \arg\min_{\mathbf{x}} \left\{ H\left(\mathbf{x}, \sum_{i=1}^{k} (\mathbf{y} - \widehat{\mathbf{x}}_i)\right) + J(\mathbf{x}) \right\}, \\
3) \qquad \widehat{\mathbf{x}}_{k+1} &= \widehat{\mathbf{x}}_1 + \sum_{i=1}^{k} \arg\min_{\mathbf{x}} \left\{ H\left(\mathbf{x}, \mathbf{y} - \widehat{\mathbf{x}}_i\right) + J(\mathbf{x}) \right\}, \text{ and} \\
4) \qquad \widehat{\mathbf{x}}_{k+1} &= \widehat{\mathbf{x}}_1 + \sum_{i=1}^{k} \left(\widehat{\mathbf{x}}_1 - \arg\min_{\mathbf{x}} \left\{ H\left(\mathbf{x}, \widehat{\mathbf{x}}_i\right) + J(\mathbf{x}) \right\} \right) \\
&= (k+1)\widehat{\mathbf{x}}_1 - \sum_{i=1}^{k} \arg\min_{\mathbf{x}} \left\{ H\left(\mathbf{x}, \widehat{\mathbf{x}}_i\right) + J(\mathbf{x}) \right\}.
\end{aligned}
$$

where

$$
\widehat{\mathbf{x}}_1 = \arg\min_{\mathbf{x}} \left\{ H\left(\mathbf{x}, \mathbf{y}\right) + J(\mathbf{x}) \right\}.
$$

The first method was recently proposed in [1] by Osher et al., and Method (3) is a generalization of Tukey's "twicing" idea [11] while the other two methods are ones which we introduce here.

Using $\mathcal{B}(\cdot)$ to denote the net effect of the cost function minimizations above, we

formulate these iterative regularization methods as

$$
\begin{aligned}
1) \qquad \widehat{\mathbf{x}}_{k+1} &= \mathcal{B}\left(\mathbf{y} + \sum_{i=1}^{k}(\mathbf{y} - \widehat{\mathbf{x}}_i)\right), \\
2) \qquad \widehat{\mathbf{x}}_{k+1} &= \mathcal{B}(\mathbf{y}) + \mathcal{B}\left(\sum_{i=1}^{k}(\mathbf{y} - \widehat{\mathbf{x}}_i)\right), \\
3) \qquad \widehat{\mathbf{x}}_{k+1} &= \mathcal{B}(\mathbf{y}) + \sum_{i=1}^{k}\mathcal{B}(\mathbf{y} - \widehat{\mathbf{x}}_i), \text{ and} \\
4) \qquad \widehat{\mathbf{x}}_{k+1} &= \mathcal{B}(\mathbf{y}) + \sum_{i=1}^{k}(\mathcal{B}(\mathbf{y}) - \mathcal{B}(\widehat{\mathbf{x}}_i)) \\
&= (k+1)\mathcal{B}(\mathbf{y}) - \sum_{i=1}^{k}\mathcal{B}(\widehat{\mathbf{x}}_i).
\end{aligned}
$$

which makes it clear that all four methods are related by a linear distribution of $\mathcal{B}(\cdot)$. If the minimization process themselves were linear, that is if $\mathcal{B}(a+b) = \mathcal{B}(a) + \mathcal{B}(b)$, then all four methods would be equivalent. However, for most useful regularization methods, $\mathcal{B}(a+b) \neq \mathcal{B}(a) + \mathcal{B}(b)$ and hence the methods are distinct.

We further note the framework presented here presents the possibility for infinitely many variations on these four methods, e.g.

$$
\widehat{\mathbf{x}}_{k+1} = \mathcal{B}(\mathbf{y}) + \mathcal{B}(\mathbf{y}) - \mathcal{B}(\widehat{\mathbf{x}}_k) + \mathcal{B}\left(\sum_{i=1}^{k-1}(\mathbf{y} - \widehat{\mathbf{x}}_i)\right)
$$

etc. We will only focus on the four major methods listed above since the other variations will produce estimates that are very similar to either Method 3 or Method 4. In fact, the forumlations would be exactly the same as either Method 3 or 4 (depending on the variant) for at least the first two iterations.

2.2 Description of Iterative Regularization Methods

In this section we will discuss the details of each of the iterative regularization methods introduced in the previous section.

2.2.1 Method 1: Osher et al.'s Method [1]

The work of Osher et al. improves the estimate that results from the cost function in (1.3) via the following algorithm, which we here call " Osher's Iterative Regularization Method"

$$\widehat{\mathbf{x}}_{k+1} = \mathcal{B}\left(\mathbf{y} + \sum_{i=1}^{k}(\mathbf{y} - \widehat{\mathbf{x}}_i)\right), \tag{2.2}$$

with $\widehat{\mathbf{x}}_0 = 0$.

This can also be written as

$$\widehat{\mathbf{x}}_{k+1} = \mathcal{B}\left(\mathbf{y} + \mathbf{v}_k\right), \tag{2.3}$$

with $\widehat{\mathbf{x}}_0 = 0$, $\mathbf{v}_0 = 0$, and $\mathbf{v}_{k+1} = \mathbf{v}_k + (\mathbf{y} - \widehat{\mathbf{x}}_{k+1})$. If we define $\mathbf{r}_k \equiv (\mathbf{y} - \widehat{\mathbf{x}}_k)$ we can also write:

$$\mathbf{v}_{k+1} = \mathbf{v}_k + \mathbf{r}_{k+1}.$$

The quantity $\mathbf{v}_k$ can be interpreted as a cumulative sum of the image residuals ($\mathbf{r}_k$).

We note that the sum of the residuals has been added back to the noisy image and processed again. The intuition here is that if, at each iteration, the residual contains more signal than noise, our estimate will improve. We illustrate this method in Figures (**2.1**) and (**2.2**).

Interestingly, Osher's method may be written in yet another way as:

$$\widehat{\mathbf{x}}_{k+1} = \arg\min_{\mathbf{x}} \{H(\mathbf{x}, \mathbf{y}) + J(\mathbf{x}) - J(\widehat{\mathbf{x}}_k) - \langle \mathbf{p}_k, \mathbf{x} - \widehat{\mathbf{x}}_k \rangle\} \tag{2.4}$$

where

$$\mathbf{p}_{k+1} = \mathbf{p}_k + \left(\frac{\partial H(\mathbf{x}, \mathbf{y})}{\partial \mathbf{x}} \middle| \mathbf{x} = \widehat{\mathbf{x}}_{k+1} \right) \tag{2.5}$$

with $\mathbf{p}_0 = 0$. The operator $\langle \cdot, \cdot \rangle$ denotes the duality product which is related to the standard vector inner product in the following manner: $\mathbf{a} \cdot \mathbf{b} = \langle \mathbf{a}^*, \mathbf{b} \rangle$ where $\mathbf{a}^*$ is the dual of $\mathbf{a}$. If $\mathbf{a}$

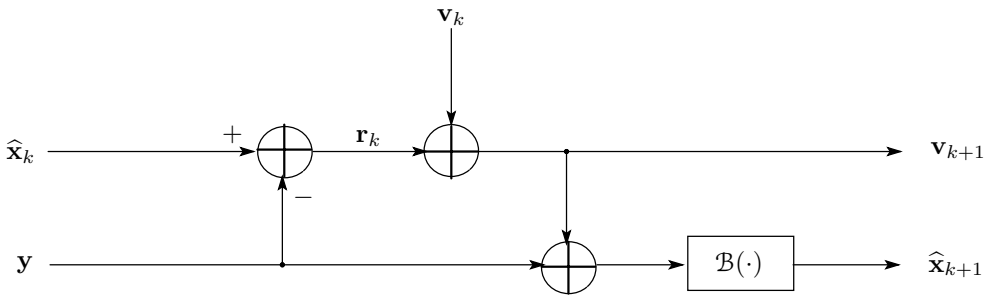


Figure 2.1: Method 1 Block Diagram

is in vector space $\mathbf{V}$, then $\mathbf{a}^*$ is in the vector space of functions that map $\mathbf{V}$ into the space of all real numbers, i.e.

$$\begin{aligned} \mathbf{a} &\in \mathbf{V} \\ \mathbf{a}^* &\in \mathbf{V}^* : \mathbf{V} \to \mathbb{R}. \end{aligned}$$

While this formulation may seem more complicated and not as intuitive as Equation (2.2), we will see later that it is vital in proving the convergence of this method.

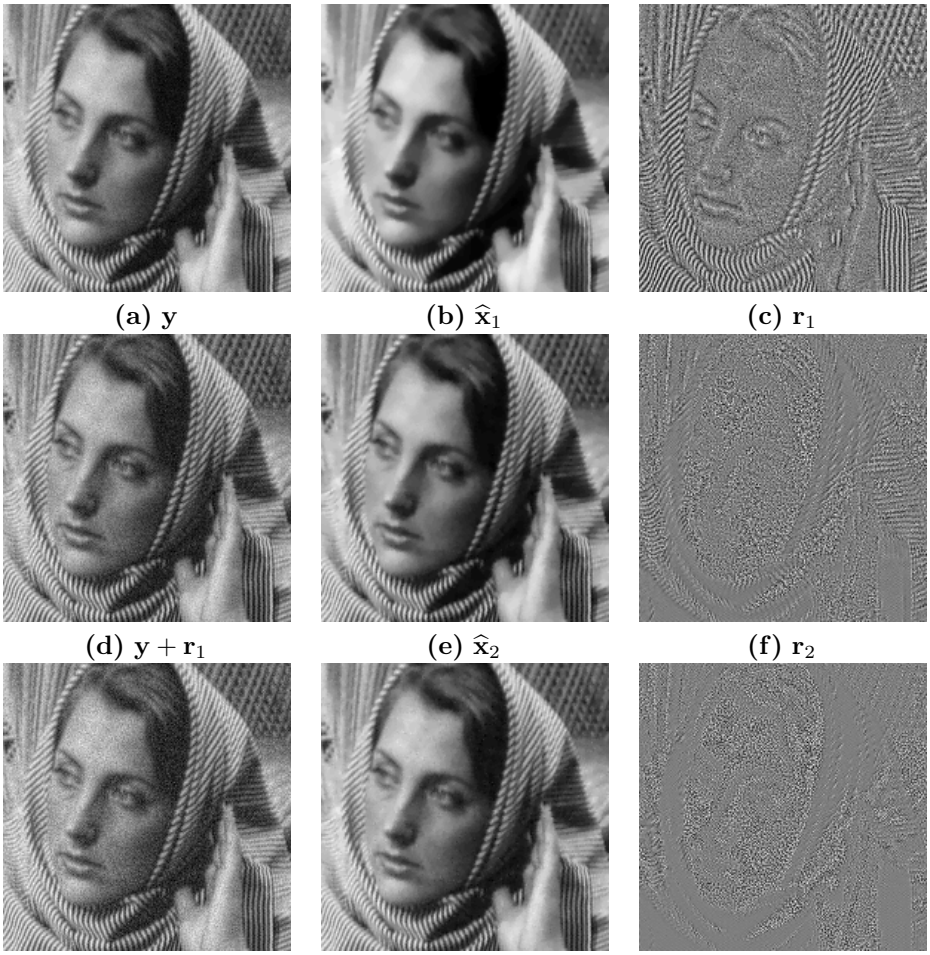
Lemma 1. *If $J(\mathbf{x})$ is non-negative and convex, Equations (2.4) and (2.2) have the same minimum, and thus produce the same estimate $\widehat{\mathbf{x}}_k$.*

Proof: We now prove this for the case when $H(\mathbf{x}, \cdot) = \frac{1}{2}\|(\cdot) - \mathbf{x}\|^2$ as an example. Define $Z_k(\mathbf{x}, \mathbf{y}) \equiv \frac{1}{2}\|\mathbf{y} + \sum_{i=1}^{k}(\mathbf{y} - \widehat{\mathbf{x}}_i) - \mathbf{x}\|^2 + J(\mathbf{x})$ and $Q_k(\mathbf{x}, \mathbf{y}) \equiv \frac{1}{2}\|\mathbf{y} - \mathbf{x}\|^2 + J(\mathbf{x}) - J(\widehat{\mathbf{x}}_k) - \langle \mathbf{p}_k, \mathbf{x} - \widehat{\mathbf{x}}_k \rangle$. Now we have

$$\frac{\partial Z_k(\mathbf{x}, \mathbf{y})}{\partial \mathbf{x}} = \mathbf{x} - \mathbf{y} - \sum_{i=1}^{k}(\mathbf{y} - \widehat{\mathbf{x}}_i) + \frac{\partial J(\mathbf{x})}{\partial \mathbf{x}} \tag{2.6}$$

and

$$\frac{\partial Q_k(\mathbf{x}, \mathbf{y})}{\partial \mathbf{x}} = \mathbf{x} - \mathbf{y} + \frac{\partial J(\mathbf{x})}{\partial \mathbf{x}} - \mathbf{p}_k. \tag{2.7}$$



(a) $\mathbf{y}$ **(b)** $\widehat{\mathbf{x}}_1$ **(c)** $\mathbf{r}_1$

(d) $\mathbf{y} + \mathbf{r}_1$ **(e)** $\widehat{\mathbf{x}}_2$ **(f)** $\mathbf{r}_2$

(g) $\mathbf{y} + \mathbf{r}_1 + \mathbf{r}_2$ **(h)** $\widehat{\mathbf{x}}_3$ **(i)** $\mathbf{r}_3$

Figure 2.2: **(a)** Noisy data **(b)** The first estimate produced by Osher's method with Total Variation regularization (MSE=53.16). **(c)** The residual from the first estimate. **(d)** The first residual is added to the noisy data. **(e)** The second estimate produced by Osher's method (MSE=17.14) from (d). **(f)** The residual from the second estimate. **(g)** The first and second residuals are added to the noisy data. **(h)** The third estimate produced by Osher's method (MSE=20.72) from (g). **(i)** The residual from the third estimate.

From 2.5 we have the following:

$$\begin{aligned}
\mathbf{p}_0 &= 0 \\
\mathbf{p}_1 &= 0 + (\mathbf{y} - \widehat{\mathbf{x}}_1) \\
\mathbf{p}_2 &= 0 + (\mathbf{y} - \widehat{\mathbf{x}}_1) + (\mathbf{y} - \widehat{\mathbf{x}}_2) \\
&\vdots \\
\mathbf{p}_k &= \sum_{i=1}^{k} (\mathbf{y} - \widehat{\mathbf{x}}_i).
\end{aligned}$$

Thus,

$$\frac{\partial Q_k(\mathbf{x}, \mathbf{y})}{\partial \mathbf{x}} = \mathbf{x} - \mathbf{y} + \frac{\partial J(\mathbf{x})}{\partial \mathbf{x}} - \sum_{i=1}^{k} (\mathbf{y} - \widehat{\mathbf{x}}_i) = \frac{\partial Z_k(\mathbf{x}, \mathbf{y})}{\partial \mathbf{x}}. \tag{2.8}$$

□

We now note that, based on the definition of the Brgeman distance (Appendix A), Equation (2.4) is equivalent to:

$$\widehat{\mathbf{x}}_{k+1} = \arg\min_{\mathbf{x}} \{H(\mathbf{x}, \mathbf{y}) + D_J^{\mathbf{p}_k}(\mathbf{x}, \widehat{\mathbf{x}}_k)\} \tag{2.9}$$

where $D_J^{\mathbf{p}_k}$ is the generalized Bregman distance (see appendix A) between $\mathbf{x}$ and $\widehat{\mathbf{x}}_k$ on functional $J(\cdot)$ for subgradient (see appendix A) $\mathbf{p}_k$.

2.2.2 Method 2: Summed Residual Regularization

This iterative regularization method is formulated as:

$$\begin{aligned}
\widehat{\mathbf{x}}_{k+1} &= \widehat{\mathbf{x}}_1 + \arg\min_{\mathbf{x}} \left\{ H\left(\mathbf{x}, \sum_{i=1}^{k} (\mathbf{y} - \widehat{\mathbf{x}}_i)\right) + J(\mathbf{x}) \right\} \\
&= \widehat{\mathbf{x}}_1 + \arg\min_{\mathbf{x}} \{H(\mathbf{x}, \mathbf{v}_k) + J(\mathbf{x})\}
\end{aligned} \tag{2.10}$$

where $\widehat{\mathbf{x}}_1$ and $\mathbf{v}_k$ are the same as in Method 1. Because the cost function operates on a sum of residuals, we give this method the name "Summed Residual Regularization" (SRR).

Using our $\mathcal{B}(\cdot)$ notation we formulate this as:

$$\widehat{\mathbf{x}}_{k+1} = \mathcal{B}(\mathbf{y}) + \mathcal{B}\left(\mathbf{v}_k\right) \tag{2.11}$$

as illustrated this in Figures (**2.3**) and (**2.4**).

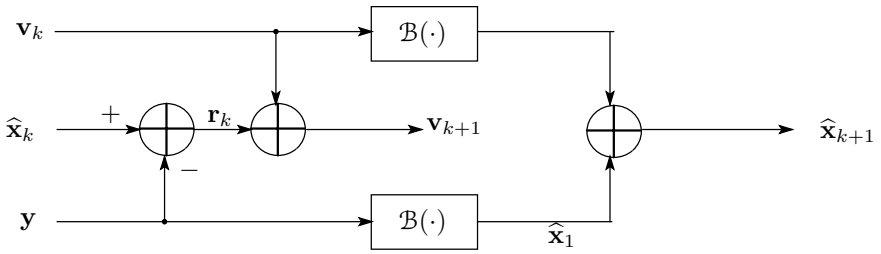


Figure 2.3: Method 2 Block Diagram

As with Osher's method, we may rewrite this method in terms of the generalized Bregman distance. Let us define $\widetilde{\mathbf{v}}_k \equiv \mathcal{B}(\mathbf{v}_k)$, then we can write this as:

$$\begin{aligned}
\widehat{\mathbf{x}}_{k+1} &= \widehat{\mathbf{x}}_1 + \arg\min_{\mathbf{x}} \left\{ H(\mathbf{x}, \mathbf{y} - \widehat{\mathbf{x}}_1) + J(\mathbf{x}) - J(\widetilde{\mathbf{v}}_{k-1}) - \langle \mathbf{p}_{k-1}, \mathbf{x} - \widetilde{\mathbf{v}}_{k-1} \rangle \right\} \\
&= \widehat{\mathbf{x}}_1 + \arg\min_{\mathbf{x}} \left\{ H(\mathbf{x}, \mathbf{y} - \widehat{\mathbf{x}}_1) + D_J^{p_{k-1}}(\mathbf{x}, \widetilde{\mathbf{v}}_{k-1}) \right\}. \qquad (2.12)
\end{aligned}$$

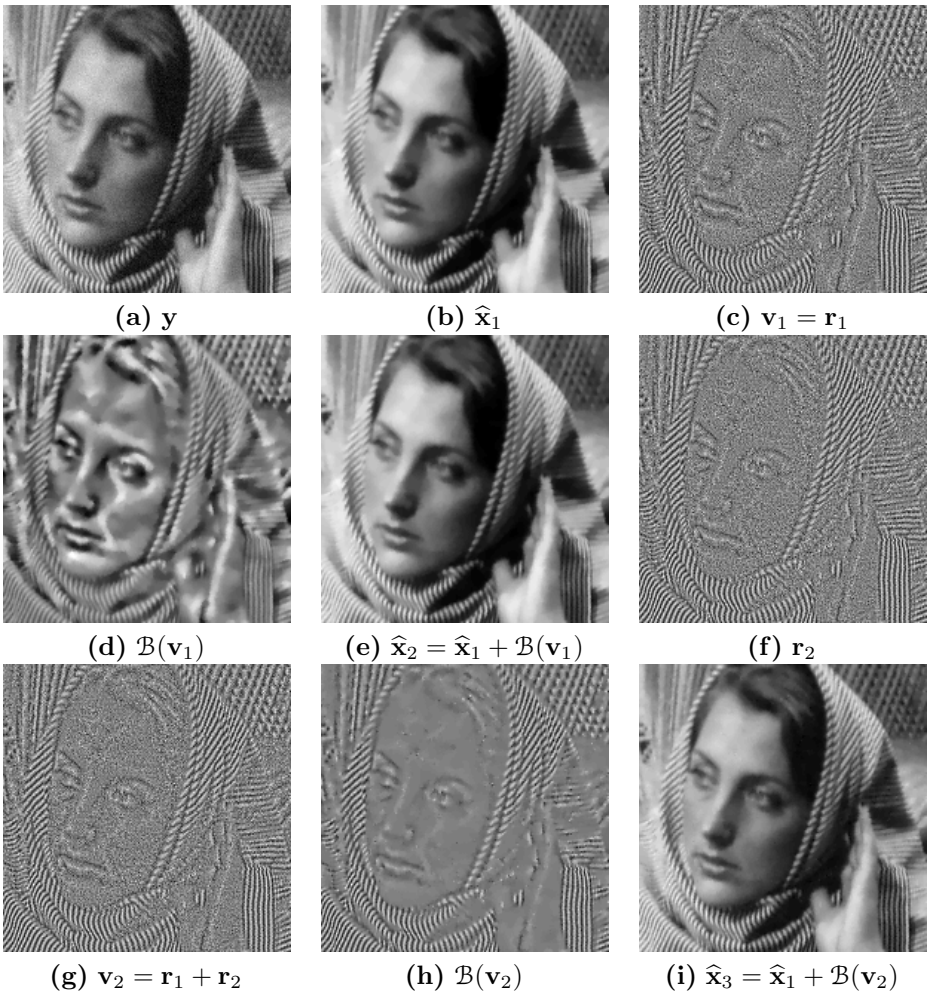
where

$$\mathbf{p}_{k+1} = \mathbf{p}_k + \left(\frac{\partial H(\mathbf{x}, \mathbf{y} - \widehat{\mathbf{x}}_1)}{\partial \mathbf{x}} \bigg| \mathbf{x} = \widetilde{\mathbf{v}}_k \right) \tag{2.13}$$

with $\mathbf{p}_0 = 0$. The first estimate, $\widehat{\mathbf{x}}_1$ is the same as in Osher's method.

Lemma 2. *If $J(\mathbf{x})$ is non-negative and convex, the gradients of the cost functions in (2.10) and (2.12) are equivalent. Thus, the cost functions differ by only a constant and have the same minimum; making the two formulations equivalent.*

Proof: We prove this for the case when $H(\mathbf{x}, \cdot) = \frac{1}{2}\|(\cdot) - \mathbf{x}\|^2$ as an example. Define $Z_k(\mathbf{x}, \mathbf{y}) \equiv \frac{1}{2}\| \sum_{i=1}^{k} (\mathbf{y} - \widehat{\mathbf{x}}_i) - \mathbf{x}\|^2 + J(\mathbf{x})$ and $Q_k(\mathbf{x}, \mathbf{y}) \equiv \frac{1}{2}\|\mathbf{y} - \widehat{\mathbf{x}}_1 - \mathbf{x}\|^2 + J(\mathbf{x}) -$



(a) $\mathbf{y}$ **(b)** $\widehat{\mathbf{x}}_1$ **(c)** $\mathbf{v}_1 = \mathbf{r}_1$

(d) $\mathcal{B}(\mathbf{v}_1)$ **(e)** $\widehat{\mathbf{x}}_2 = \widehat{\mathbf{x}}_1 + \mathcal{B}(\mathbf{v}_1)$ **(f)** $\mathbf{r}_2$

(g) $\mathbf{v}_2 = \mathbf{r}_1 + \mathbf{r}_2$ **(h)** $\mathcal{B}(\mathbf{v}_2)$ **(i)** $\widehat{\mathbf{x}}_3 = \widehat{\mathbf{x}}_1 + \mathcal{B}(\mathbf{v}_2)$

Figure 2.4: **(a)** Noisy data **(b)** The first estimate produced by SRR with Total Variation regularization (MSE=43.29). **(c)** The residual from the first estimate. **(d)** The estimate produced from the first residual. **(e)** The second estimate produced by Method 2 (MSE=40.39) from summing (b) and (d). **(f)** The residual from the second estimate. **(g)** The sum of the first and second residuals. **(h)** The estimate produced from the sum of the first and second residuals. **(i)** The third estimate produced by Method 2 (MSE=16.40) from summing (b) and (h).

$J(\widetilde{\mathbf{v}}_{k-1}) - \langle \mathbf{p}_{k-1}, \mathbf{x} - \widetilde{\mathbf{v}}_{k-1} \rangle$. Now we have

$$\frac{\partial Z_k(\mathbf{x},\mathbf{y})}{\partial \mathbf{x}} = \mathbf{x} - \sum_{i=1}^{k} (\mathbf{y} - \widehat{\mathbf{x}}_i) + \frac{\partial J(\mathbf{x})}{\partial \mathbf{x}} \tag{2.14}$$

and

$$\frac{\partial Q_k(\mathbf{x},\mathbf{y})}{\partial \mathbf{x}} = \mathbf{x} - (\mathbf{y} - \widehat{\mathbf{x}}_1) + \frac{\partial J(\mathbf{x})}{\partial \mathbf{x}} - \sum_{i=2}^{k} (\mathbf{y} - \widehat{\mathbf{x}}_i) = \frac{\partial Z_k(\mathbf{x},\mathbf{y})}{\partial \mathbf{x}}. \tag{2.15}$$

□

2.2.3 Method 3: Iterative " Twicing" Regularization

In his book [11], published in the mid 1970's, Tukey presented a method he called "twicing" where a filtered version of the data residual was added back to the inital estimate $\widehat{\mathbf{x}}_0$ as

$$\widehat{\mathbf{x}}_1 = \widehat{\mathbf{x}}_0 + \mathcal{B}(\mathbf{y} - \widehat{\mathbf{x}}_0). \tag{2.16}$$

Tukey's original motivation for this was to provide an improved method for data fitting that would go beyond a direct fit and incorporate additional " roughness" into the estimate in a controlled way. The same year, motivated by this idea, this concept was used by Kaiser and Hamming [12] as a way of sharpening the response of symmetric FIR linear filters. Both references also mentioned the possibility of iterating this process. Thus we here call the iterated version of Tukey's Twicing, " Iterative Twicing Regularization" (ITR). We can express this as:

$$\begin{aligned}
\widehat{\mathbf{x}}_{k+1} &= \widehat{\mathbf{x}}_k + \mathcal{B}(\mathbf{y} - \widehat{\mathbf{x}}_k) \\
&= \widehat{\mathbf{x}}_{k-1} + \mathcal{B}\left(\mathbf{y} - \widehat{\mathbf{x}}_{k-1}\right) + \mathcal{B}\left(\mathbf{y} - \widehat{\mathbf{x}}_k\right) \\
&\vdots \\
&= \widehat{\mathbf{x}}_1 + \sum_{i=1}^{k} \mathcal{B}\left(\mathbf{y} - \widehat{\mathbf{x}}_i\right)
\end{aligned} \tag{2.17}$$

where $\widehat{\mathbf{x}}_0 = 0$ and $\widehat{\mathbf{x}}_1 = \mathcal{B}(\mathbf{y})$. The same idea has been used in the machine learning community ([13]) under the name $L_2 Boost$.

We illustrate this algorithm in Figures (**2.5**) and (**2.6**). Note the simplified appearance of the block diagram in Figure (**2.5**) as compared to those in Figures (**2.1**) and (**2.3**). This is due to the fact that the estimates produced via Method 3 (and Method 4 as we will see shortly) can be computed from a simple update to the previous estimate whereas the estimates produced via Methods 1 and 2 must be re-calculated at each iteration based on an updated input.

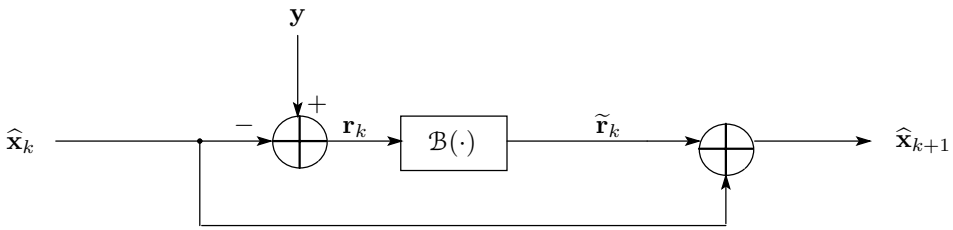


Figure 2.5: Method 3 Block Diagram

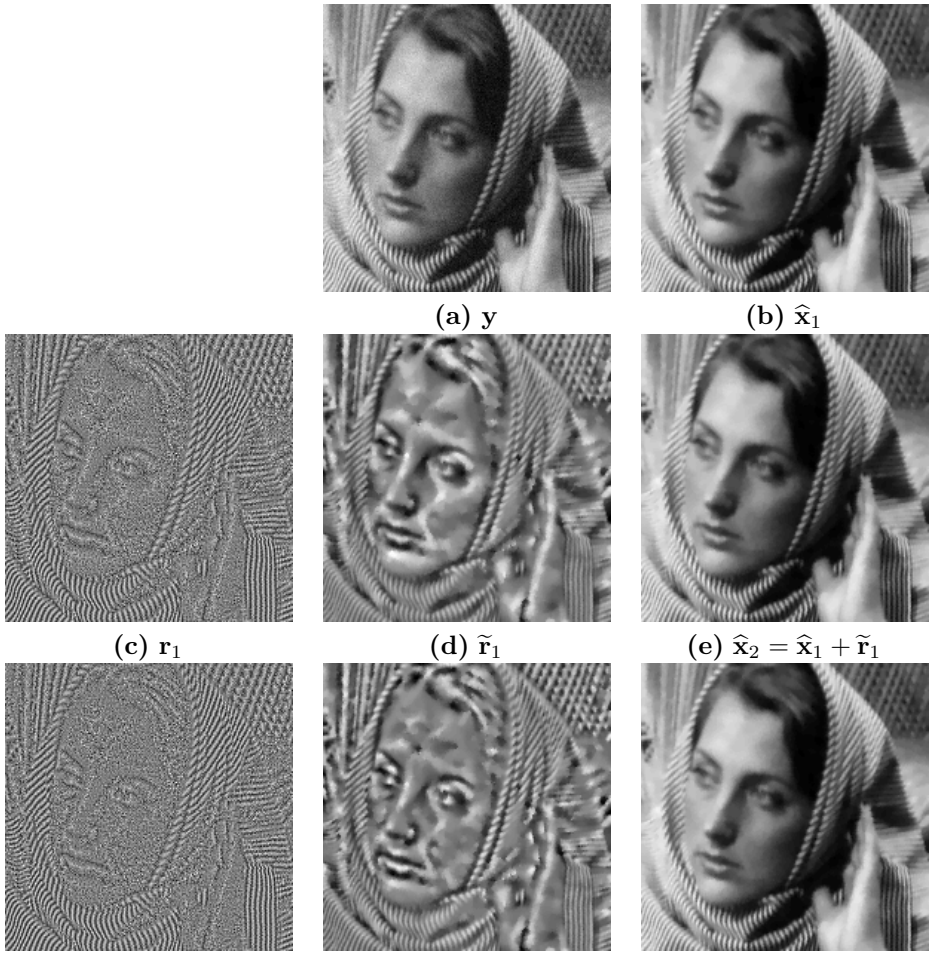
If we define $\widetilde{\mathbf{r}}_k \equiv \mathcal{B}(\mathbf{r}_k)$, we can rewrite ITR in terms of the generalized Bregman distance as:

$$\begin{aligned}
\widehat{\mathbf{x}}_{k+1} &= \widehat{\mathbf{x}}_1 + \sum_{i=1}^{k-1} + \arg\min_{\mathbf{x}} \left\{ H(\mathbf{x},0) + J(\mathbf{x}) - J(\widetilde{\mathbf{r}}_{k-1}) - \langle \mathbf{p}_{k-1}, \mathbf{x} - \widetilde{\mathbf{r}}_{k-1} \rangle \right\} \\
&= \widehat{\mathbf{x}}_1 + \sum_{i=1}^{k-1} + \arg\min_{\mathbf{x}} \left\{ H(\mathbf{x},0) + D_J^{\mathbf{p}_{k-1}}(\mathbf{x}, \widetilde{\mathbf{r}}_{k-1}) \right\} \qquad (2.18)
\end{aligned}$$

where

$$\mathbf{p}_{k+1} = \mathbf{p}_k + \left(\frac{\partial H(\mathbf{x},0)}{\partial \mathbf{x}} \middle| \mathbf{x} = \widetilde{\mathbf{r}}_{k+1} \right) \qquad (2.19)$$

with $\mathbf{p}_0 = \mathbf{y} - \widehat{\mathbf{x}}_1$ and $\widetilde{\mathbf{r}}_0 = 0$. The first estimate, $\widehat{\mathbf{x}}_1$ is the same as in both Methods 1 and 2.



(a) $\mathbf{y}$ **(b)** $\widehat{\mathbf{x}}_1$

(c) $\mathbf{r}_1$ **(d)** $\widetilde{\mathbf{r}}_1$ **(e)** $\widehat{\mathbf{x}}_2 = \widehat{\mathbf{x}}_1 + \widetilde{\mathbf{r}}_1$

(f) $\mathbf{r}_2$ **(g)** $\widetilde{\mathbf{r}}_2$ **(h)** $\widehat{\mathbf{x}}_3 = \widehat{\mathbf{x}}_1 + \widetilde{\mathbf{r}}_1 + \widetilde{\mathbf{r}}_2$

Figure 2.6: **(a)** Noisy data **(b)** The first estimate produced by ITR with Total Variation regularization (MSE=28.52). **(c)** The residual from the first estimate. **(d)** The estimate produced from the first residual. **(e)** The second estimate produced by ITR (MSE=27.11) from summing (b) and (d). **(f)** The residual from the second estimate. **(g)** The estimate produced from the second residual. **(h)** The third estimate produced by ITR method (MSE=26.48) from summing (b), (d), and (g).

Lemma 3. *If $J(\mathbf{x})$ is non-negative and convex, the gradients of the cost functions in (2.17) and (2.18) are equivalent. Thus, the cost functions differ by only a constant and have the same minimum; making the two formulations equivalent.*

Proof: We prove this for the case when $H(\mathbf{x},\cdot) = \frac{1}{2}\|(\cdot)-\mathbf{x}\|^2$ as an example. Define $Z_k(\mathbf{x},\mathbf{y}) \equiv \frac{1}{2}\|\mathbf{y}-\widehat{\mathbf{x}}_k-\mathbf{x}\|^2 + J(\mathbf{x})$ and $Q_k(\mathbf{x}) \equiv \frac{1}{2}\|\mathbf{x}\|^2 + J(\mathbf{x}) - J(\widetilde{\mathbf{r}}_{k-1}) - \langle \mathbf{p}_{k-1}, \mathbf{x}-\widetilde{\mathbf{r}}_{k-1}\rangle$. Now we have

$$\frac{\partial Z_k(\mathbf{x},\mathbf{y})}{\partial \mathbf{x}} = \mathbf{x} - (\mathbf{y}-\widehat{\mathbf{x}}_k) + \frac{\partial J(\mathbf{x})}{\partial \mathbf{x}} \tag{2.20}$$

and

$$\frac{\partial Q_k(\mathbf{x})}{\partial \mathbf{x}} = \mathbf{x} + \frac{\partial J(\mathbf{x})}{\partial \mathbf{x}} - \left(\mathbf{y} - \widehat{\mathbf{x}}_1 - \sum_{i=1}^{k-1}\widetilde{\mathbf{r}}_i\right) = \frac{\partial Z_k(\mathbf{x},\mathbf{y})}{\partial \mathbf{x}}. \tag{2.21}$$

□

2.2.4 Method 4: Iterative Unsharp Regularization

The process of unsharp masking is a well-known technique by which edges are enhanced by subtracting a blurred version of an image from the image itself [2]. The fourth algorithm for iterative regularization that we present is very similar in spirit to unsharp masking, thus we call this method "Iterative Unsharp Regularization" (IUR). We formulate it as:

$$\begin{aligned}
\widehat{\mathbf{x}}_{k+1} &= \widehat{\mathbf{x}}_k + \widehat{\mathbf{x}}_1 - \mathcal{B}(\widehat{\mathbf{x}}_k) \\
&= \widehat{\mathbf{x}}_{k-1} + (\widehat{\mathbf{x}}_1 - \mathcal{B}(\widehat{\mathbf{x}}_{k-1})) + (\mathcal{B}(\mathbf{y}) - \mathcal{B}(\widehat{\mathbf{x}}_k)) \\
&\quad\vdots \\
&= \widehat{\mathbf{x}}_1 + \sum_{i=1}^{k}(\widehat{\mathbf{x}}_1 - \mathcal{B}(\widehat{\mathbf{x}}_i)) \;=\; (k+1)\widehat{\mathbf{x}}_1 - \sum_{i=1}^{k}\mathcal{B}(\widehat{\mathbf{x}}_i)
\end{aligned} \tag{2.22}$$

where $\widehat{\mathbf{x}}_0 = 0$ and $\widehat{\mathbf{x}}_1 = \mathcal{B}(\mathbf{y})$.

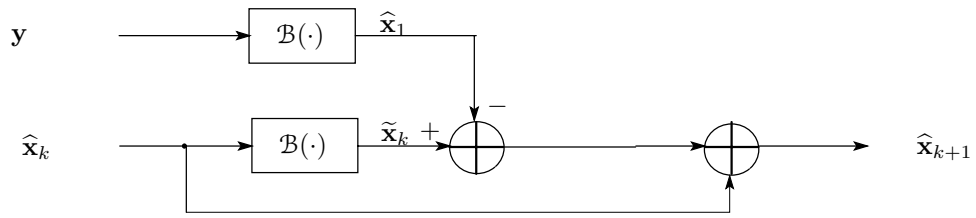


Figure 2.7: Method 4 Block Diagram

We illustrate this method in Figures (**2.7**) and (**2.8**).

If we define $\widetilde{\mathbf{x}}_k \equiv \mathcal{B}(\widehat{\mathbf{x}}_k)$, this can be rewritten in terms of the generalized Bregman distance as:

$$\begin{aligned}\widehat{\mathbf{x}}_{k+1} &= (k+1)\widehat{\mathbf{x}}_1 - \sum_{i=1}^{k-1}\widetilde{\mathbf{x}}_i - \arg\min_{\mathbf{x}}\left\{H(\mathbf{x},\widehat{\mathbf{x}}_1) + J(\mathbf{x}) - J(\widetilde{\mathbf{x}}_{k-1}) - \langle \mathbf{p}_{k-1}, \mathbf{x} - \widetilde{\mathbf{x}}_{k-1}\rangle\right\} \\ &= (k+1)\widehat{\mathbf{x}}_1 - \sum_{i=1}^{k-1}\widetilde{\mathbf{x}}_i - \arg\min_{\mathbf{x}}\left\{H(\mathbf{x},\widehat{\mathbf{x}}_1) + D_J^{p_{k-1}}(\mathbf{x},\widetilde{\mathbf{x}}_{k-1})\right\} \qquad (2.23)\end{aligned}$$

where

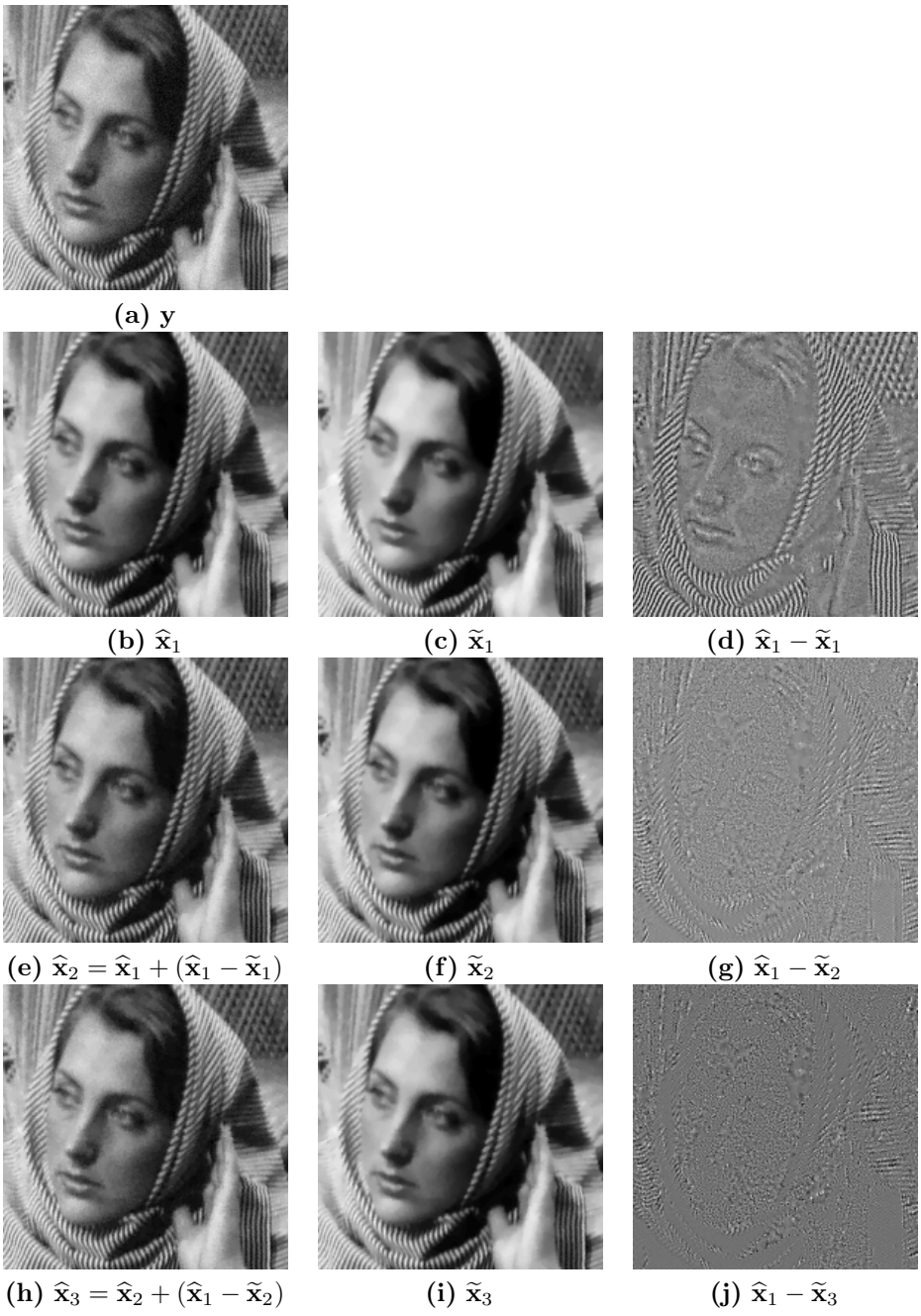
$$\mathbf{p}_{k+1} = \mathbf{p}_k + \left(\frac{\partial H(\mathbf{x},\widehat{\mathbf{x}}_1)}{\partial \mathbf{x}}\bigg|\, \mathbf{x} = \widetilde{\mathbf{x}}_{k+1}\right) \qquad (2.24)$$

with $\mathbf{p}_0 = 0$ and $\widetilde{\mathbf{x}}_0 = 0$.

Lemma 4. *If $J(\mathbf{x})$ is non-negative and convex, the gradients of the cost functions in (2.22) and (2.23) are equivalent. Thus, the cost functions differ by only a constant and have the same minimum; making the two formulations equivalent.*

Proof: We prove this for the case when $H(\mathbf{x},\cdot) = \frac{1}{2}\|(\cdot) - \mathbf{x}\|^2$ as an example. Define $Z_k(\mathbf{x},\widehat{\mathbf{x}}_k) \equiv \frac{1}{2}\|\widehat{\mathbf{x}}_k - \mathbf{x}\|^2 + J(\mathbf{x})$ and $Q_k(\mathbf{x},\widehat{\mathbf{x}}_1) \equiv \frac{1}{2}\|\widehat{\mathbf{x}}_1 - \mathbf{x}\|^2 + J(\mathbf{x}) - J(\widetilde{\mathbf{x}}_{k-1}) - \langle \mathbf{p}_{k-1}, \mathbf{x} - \widetilde{\mathbf{x}}_{k-1}\rangle$. Now we have

$$\frac{\partial Z_k(\mathbf{x},\widehat{\mathbf{x}}_k)}{\partial \mathbf{x}} = \mathbf{x} - \widehat{\mathbf{x}}_k + \frac{\partial J(\mathbf{x})}{\partial \mathbf{x}} \qquad (2.25)$$



(a) $\mathbf{y}$ **(b)** $\widehat{\mathbf{x}}_1$ **(c)** $\widetilde{\mathbf{x}}_1$ **(d)** $\widehat{\mathbf{x}}_1 - \widetilde{\mathbf{x}}_1$ **(e)** $\widehat{\mathbf{x}}_2 = \widehat{\mathbf{x}}_1 + (\widehat{\mathbf{x}}_1 - \widetilde{\mathbf{x}}_1)$ **(f)** $\widetilde{\mathbf{x}}_2$ **(g)** $\widehat{\mathbf{x}}_1 - \widetilde{\mathbf{x}}_2$ **(h)** $\widehat{\mathbf{x}}_3 = \widehat{\mathbf{x}}_2 + (\widehat{\mathbf{x}}_1 - \widetilde{\mathbf{x}}_2)$ **(i)** $\widetilde{\mathbf{x}}_3$ **(j)** $\widehat{\mathbf{x}}_1 - \widetilde{\mathbf{x}}_3$

Figure 2.8: **(a)** Noisy data **(b)** The first estimate produced by IUR with Total Variation regularization (MSE=26.76). **(c)** The estimate produced from (b). **(d)** The residual between (b) and (c). **(e)** The second estimate produced by IUR (MSE=18.09) from summing (b) and (d). **(f)** The estimate produced from (e). **(g)** The residual between (b) and (f). **(h)** The third estimate produced by IUR method (MSE=20.22) from summing (b), (d), and (g). **(i)** The estimate produced from (h). **(j)** The residual between (b) and (i).

and

$$\frac{\partial Q_k(\mathbf{x}, \widehat{\mathbf{x}}_1)}{\partial \mathbf{x}} = \mathbf{x} - \widehat{\mathbf{x}}_1 + \frac{\partial J(\mathbf{x})}{\partial \mathbf{x}} - \sum_{i=1}^{k-1} (\widehat{\mathbf{x}}_1 - \widetilde{\mathbf{x}}_i) = \frac{\partial Z_k(\mathbf{x}, \widehat{\mathbf{x}}_k)}{\partial \mathbf{x}}. \tag{2.26}$$

□

Now that we have established a general iterative regularization framework for image denoising, one may wonder how this framework is generalized to the case of image restoration. One may further wonder if these iterative methods converge, and if so, to what value? These questions, and others, will be addressed in the following chapters.

Chapter 3

Generalized Image Reconstruction

3.1 Generalized Image Reconstruction

We have looked at the case when our data model includes additive noise only. We now wish to consider the more general case when the data model includes some blur as well as additive noise. This is a more realistic model. Blurring in images can be cause by many factors including imperfect optics, motion due to long exposure times, motion of the imager itself, or atmospheric turbulence. To take these factors into account, we now consider the more general measurement model:

$$\mathbf{y} = \mathbf{A}\mathbf{x} + \mathbf{v}, \tag{3.1}$$

where $\mathbf{A}$ is a convolution operator.

A typical cost function used to estimate $\mathbf{x}$ from $\mathbf{y}$, in this case, is of the form $\frac{1}{2}\|\mathbf{y} - \mathbf{A}\mathbf{x}\|^2 + J(\mathbf{x})$.

By a simple substitution of $H(\mathbf{x}, \cdot) = \frac{1}{2}\|(\cdot) - \mathbf{A}\mathbf{x}\|^2$ into the generalized Bregman distance forms of the four iterative regularization methods presented in Chapter 2, we can

improve the estimate produced by minimizing this cost function as:

$$
\begin{aligned}
1) \qquad \widehat{\mathbf{x}}_{k+1} &= \arg\min_{\mathbf{x}} \left\{ \frac{1}{2}\|\mathbf{y}-\mathbf{A}\mathbf{x}\|^2 + D_J^{\mathbf{P}_k}(\mathbf{A}\mathbf{x}, \widehat{\mathbf{x}}_k) \right\}, && (3.2)\\
2) \qquad \widehat{\mathbf{x}}_{k+1} &= \widehat{\mathbf{x}}_1 + \arg\min_{\mathbf{x}} \left\{ \frac{1}{2}\|\mathbf{y}-\widehat{\mathbf{x}}_1-\mathbf{A}\mathbf{x}\|^2 + D_J^{\mathbf{P}_{k-1}}(\mathbf{A}\mathbf{x}, \widetilde{\mathbf{v}}_{k-1}) \right\}, && (3.3)\\
3) \qquad \widehat{\mathbf{x}}_{k+1} &= \widehat{\mathbf{x}}_k + \arg\min_{\mathbf{x}} \left\{ \frac{1}{2}\|\mathbf{A}\mathbf{x}\|^2 + D_J^{\mathbf{P}_{k-1}}(\mathbf{A}\mathbf{x}, \widetilde{\mathbf{r}}_{k-1}) \right\}, \text{ and} && (3.4)\\
4) \qquad \widehat{\mathbf{x}}_{k+1} &= \widehat{\mathbf{x}}_k + \widehat{\mathbf{x}}_1 - \arg\min_{\mathbf{x}} \left\{ \frac{1}{2}\|\widehat{\mathbf{x}}_1-\mathbf{A}\mathbf{x}\|^2 + D_J^{\mathbf{P}_{k-1}}(\mathbf{A}\mathbf{x}, \widetilde{\mathbf{x}}_{k-1}) \right\}. && (3.5)\\
& && (3.6)
\end{aligned}
$$

where

$$\widehat{\mathbf{x}}_1 = \arg\min_{\mathbf{x}} \left\{ \frac{1}{2}\|\mathbf{y}-\mathbf{A}\mathbf{x}\|^2 + J(\mathbf{x}) \right\}. \tag{3.7}$$

Which can be rewritten in the following more easily understood forms:

$$
\begin{aligned}
1) \qquad \widehat{\mathbf{x}}_{k+1} &= \mathcal{B}\left(\mathbf{y} + \sum_{i=1}^{k} \mathbf{A}^T(\mathbf{y}-\mathbf{A}\widehat{\mathbf{x}}_i) \right),\\
2) \qquad \widehat{\mathbf{x}}_{k+1} &= \widehat{\mathbf{x}}_1 + \mathbf{A}\mathcal{B}\left(\mathbf{y} - \widehat{\mathbf{x}}_1 + \sum_{i=2}^{k} \mathbf{A}^T(\mathbf{y}-\widehat{\mathbf{x}}_i) \right),\\
3) \qquad \widehat{\mathbf{x}}_{k+1} &= \widehat{\mathbf{x}}_1 + \sum_{i=1}^{k} \mathbf{A}^T\mathbf{A}\mathcal{B}\left(\mathbf{y}-\widehat{\mathbf{x}}_i\right)\\
&= \widehat{\mathbf{x}}_k + \mathbf{A}^T\mathbf{A}\mathcal{B}\left(\mathbf{y}-\widehat{\mathbf{x}}_k\right), \text{ and}\\
4) \qquad \widehat{\mathbf{x}}_{k+1} &= \widehat{\mathbf{x}}_1 + \sum_{i=1}^{k} \left(\mathbf{A}^T\widehat{\mathbf{x}}_1 - \mathbf{A}^T\mathbf{A}\mathcal{B}(\widehat{\mathbf{x}}_k)\right)\\
&= \widehat{\mathbf{x}}_k + \mathbf{A}^T\widehat{\mathbf{x}}_1 - \mathbf{A}^T\mathbf{A}\mathcal{B}\left(\widehat{\mathbf{x}}_k\right).
\end{aligned}
$$

where $\widehat{\mathbf{x}}_1$ is the same as in (3.7).

Lemma 5. *If $J(\mathbf{x})$ is non-negative and convex, the gradients of the cost functions shown above are equivalent to the gradients of the cost functions in 3.6; meaning that the cost functions have the same minima and thus are equivalent.*

Proof: As an example, we will look at Method 4. Define $Z_k(\mathbf{x}, \widehat{\mathbf{x}}_1) \equiv \frac{1}{2}\|\widehat{\mathbf{x}}_1 - \mathbf{A}\mathbf{x}\|^2 + D_J^{p_{k-1}}(\mathbf{A}\mathbf{x}, \widetilde{\mathbf{x}}_{k-1})$ and $Q_k(\mathbf{x}, \widehat{\mathbf{x}}_k) \equiv \frac{1}{2}\|\widehat{\mathbf{x}}_k - \mathbf{A}\mathbf{x}\|^2 + J(\mathbf{x})$. We have

$$\mathbf{p}_{k+1} = \mathbf{p}_k + \left(\frac{\partial \frac{1}{2}\|\widehat{\mathbf{x}}_1 - \mathbf{A}\mathbf{x}\|^2}{\partial \mathbf{x}} \middle| \mathbf{x} = \widetilde{\mathbf{x}}_{k+1} \right) \tag{3.8}$$

with $\mathbf{p}_0 = 0$ and $\widetilde{\mathbf{x}}_0 = 0$, thus

$$\mathbf{p}_k = \sum_{i=1}^{k} \mathbf{A}^T \left(\widehat{\mathbf{x}}_i - \mathbf{A}\widetilde{\mathbf{x}}_i \right). \tag{3.9}$$

Therefore,

$$\begin{aligned} \frac{\partial Z_k(\mathbf{x}, \widehat{\mathbf{x}}_1)}{\partial \mathbf{x}} &= \mathbf{A}^T (\mathbf{A}\mathbf{x} - \widehat{\mathbf{x}}_1) + \frac{\partial J(\mathbf{x})}{\partial \mathbf{x}} - \mathbf{A}^T \left(\sum_{i=1}^{k} \mathbf{A}^T \left(\widehat{\mathbf{x}}_i - \mathbf{A}\widetilde{\mathbf{x}}_i \right) \right) \\ &= \mathbf{A}^T \left(\mathbf{A}\mathbf{x} - \widehat{\mathbf{x}}_1 - \sum_{i=1}^{k} \mathbf{A}^T \left(\widehat{\mathbf{x}}_i - \mathbf{A}\widetilde{\mathbf{x}}_i \right) \right) + \frac{\partial J(\mathbf{x})}{\partial \mathbf{x}} \end{aligned} \tag{3.10}$$

and

$$\frac{\partial Q_k(\mathbf{x}, \widehat{\mathbf{x}}_1)}{\partial \mathbf{x}} = \mathbf{A}^T (\mathbf{A}\mathbf{x} - \widehat{\mathbf{x}}_k) + \frac{\partial J(\mathbf{x})}{\partial \mathbf{x}}. \tag{3.11}$$

Recall that the formulation that we are verifying is:

$$\begin{aligned} \widehat{\mathbf{x}}_{k+1} &= \widehat{\mathbf{x}}_1 + \sum_{i=1}^{k} \left(\mathbf{A}^T \widehat{\mathbf{x}}_1 - \mathbf{A}^T \mathbf{A} \mathcal{B}(\widehat{\mathbf{x}}_k) \right) \\ &= \widehat{\mathbf{x}}_1 + \sum_{i=1}^{k} \left(\mathbf{A}^T \widehat{\mathbf{x}}_1 - \mathbf{A}^T \mathbf{A} \widetilde{\mathbf{x}}_k \right), \end{aligned}$$

thus

$$\begin{aligned} \frac{\partial Q_k(\mathbf{x}, \widehat{\mathbf{x}}_1)}{\partial \mathbf{x}} &= \mathbf{A}^T \left(\mathbf{A}\mathbf{x} - \widehat{\mathbf{x}}_1 + \sum_{i=1}^{k} \left(\mathbf{A}^T \widehat{\mathbf{x}}_1 - \mathbf{A}^T \mathbf{A} \widetilde{\mathbf{x}}_k \right) \right) + \frac{\partial J(\mathbf{x})}{\partial \mathbf{x}} \\ &= \frac{\partial Z_k(\mathbf{x}, \widehat{\mathbf{x}}_1)}{\partial \mathbf{x}}. \end{aligned} \tag{3.12}$$

□

Chapter 4

Properties

We will show that the sequence of estimates produced by each of the four iterative regularization methods converges to the noisy data $\mathbf{y}$ as the number of iterations increases. Additionally we will look at the bias-variance tradeoff of the estimates produced by these methods.

The convergence proofs are similar to the one provided by Osher in [1] but different enough to warrant showing the explicit details for each of the four iterative regularization methods.

The only constraints on the functional $H(\mathbf{x}, \cdot)$ are that it must be nonnegative and convex. We will therefore use $H(\mathbf{x}, \cdot) = \frac{1}{2}\|(\cdot) - \mathbf{x}\|^2$ in the following proofs for simplicity and since this is a common functional choice for denoising problems. Note that the Bilateral Filter ([10], [7]), Total Variation ([9], [6]), Bilateral Total Variation ([8]), and Tikhonov regularization ([5]) all use this functional for denoising.

4.1 Convergence of Osher's Method

Though Osher's method is thoroughly analyzed in [1], we will repeat it here, using our own notation.

Recall that Osher's method is formulated in terms of the generalized Bregman distance as:

$$\begin{aligned} \widehat{\mathbf{x}}_{k+1} &= \arg\min_{\mathbf{x}} \{H(\mathbf{x},\mathbf{y}) + J(\mathbf{x}) - J(\widehat{\mathbf{x}}_k) - \langle \mathbf{p}_k, \mathbf{x} - \widehat{\mathbf{x}}_k \rangle\} \\ &= \arg\min_{\mathbf{x}} \{H(\mathbf{x},\mathbf{y}) + D_J^{\mathbf{p}_k}(\mathbf{x}, \widehat{\mathbf{x}}_k)\} \end{aligned} \tag{4.1}$$

where

$$\mathbf{p}_{k+1} = \mathbf{p}_k + \left(\frac{\partial H(\mathbf{x},\mathbf{y})}{\partial \mathbf{x}} \bigg| \mathbf{x} = \widehat{\mathbf{x}}_{k+1} \right) \tag{4.2}$$

with $\mathbf{p}_0 = 0$ and $\widehat{\mathbf{x}}_0 = 0$.

Lemma 6. *For non-negative $J(\mathbf{x})$, the sequence $\frac{1}{2}\|\mathbf{y} - \widehat{\mathbf{x}}_i\|^2$ is monotonically non-increasing for $i = 1, 2, \ldots, k$.*

Proof: We prove the monotonicity as follows:

$$\frac{1}{2}\|\mathbf{y} - \widehat{\mathbf{x}}_k\|^2 \leq \frac{1}{2}\|\mathbf{y} - \widehat{\mathbf{x}}_k\|^2 + D_J^{p_k}(\widehat{\mathbf{x}}_k, \widehat{\mathbf{x}}_{k-1})$$

due to the non-negativity of the Bregman distance. Note that $\mathbf{x} = \widehat{\mathbf{x}}_k$ minimizes $\frac{1}{2}\|\mathbf{y} - \mathbf{x}\|^2 + D_J^{p_k}(\mathbf{x}, \widehat{\mathbf{x}}_k)$ by definition. Hence,

$$\begin{aligned} \frac{1}{2}\|\mathbf{y} - \widehat{\mathbf{x}}_k\|^2 \leq \frac{1}{2}\|\mathbf{y} - \widehat{\mathbf{x}}_k\|^2 + D_J^{p_k}(\widehat{\mathbf{x}}_k, \widehat{\mathbf{x}}_{k-1}) &\leq \frac{1}{2}\|\mathbf{y} - \widehat{\mathbf{x}}_{k-1}\|^2 + D_J^{p_k}(\widehat{\mathbf{x}}_{k-1}, \widehat{\mathbf{x}}_{k-1}) \\ &\leq \frac{1}{2}\|\mathbf{y} - \widehat{\mathbf{x}}_{k-1}\|^2. \end{aligned} \tag{4.3}$$

□

Theorem 1. *For non-negative, convex $J(\mathbf{x})$, Osher's method converges as:*

$$\lim_{k\to\infty} \widehat{\mathbf{x}}_k = \mathbf{y}, \tag{4.4}$$

with rate of convergence

$$\|\mathbf{y} - \widehat{\mathbf{x}}_k\|^2 \leq \frac{2J(\mathbf{y})}{k} = \mathcal{O}\left((k)^{-1}\right) \tag{4.5}$$

Proof: We begin the proof with the identity:

$$\begin{aligned} D_J^{p_k}(\mathbf{x}, \widehat{\mathbf{x}}_k) - D_J^{p_{k-1}}(\mathbf{x}, \widehat{\mathbf{x}}_{k-1}) + D_J^{p_{k-1}}(\widehat{\mathbf{x}}_k, \widehat{\mathbf{x}}_{k-1}) &= \langle \widehat{\mathbf{x}}_k - \mathbf{x}, \mathbf{p}_k - \mathbf{p}_{k-1} \rangle \\ &= \langle \widehat{\mathbf{x}}_k - \mathbf{x}, \left(\frac{\partial H(\mathbf{x}, \mathbf{y})}{\partial \mathbf{x}} \middle| \mathbf{x} = \widehat{\mathbf{x}}_{k-1} \right) \rangle. \end{aligned} \tag{4.6}$$

Note that $\left(\frac{\partial H(\mathbf{x}, \mathbf{y})}{\partial \mathbf{x}} \middle| \mathbf{x} = \widehat{\mathbf{x}}_{k+1} \right)$ is a subgradient of $H(\mathbf{x}, \mathbf{y})$. Hence, from the definition of the subgradient (A.1), we have the following relation:

$$D_J^{p_k}(\mathbf{x}, \widehat{\mathbf{x}}_k) - D_J^{p_{k-1}}(\mathbf{x}, \widehat{\mathbf{x}}_{k-1}) + D_J^{p_{k-1}}(\widehat{\mathbf{x}}_k, \widehat{\mathbf{x}}_{k-1}) \leq H(\mathbf{x}, \mathbf{y}) - H(\widehat{\mathbf{x}}_k, \mathbf{y}) \tag{4.7}$$

If there exists a minimizer x of $H(\mathbf{x}, \mathbf{y})$ such that $J(x) < \infty$, from the particular choice of $\mathbf{x} = x$, the following is true:

$$D_J^{p_k}(x, \widehat{\mathbf{x}}_k) \leq D_J^{p_k}(x, \widehat{\mathbf{x}}_k) + D_J^{p_{k-1}}(\widehat{\mathbf{x}}_k, \widehat{\mathbf{x}}_{k-1}) + H(x, \mathbf{y}) \tag{4.8}$$

due to the non-negativity of the Bregman distance and H. Since we are considering $H(\mathbf{x}, \mathbf{y}) = \frac{1}{2}\|\mathbf{y} - \mathbf{x}\|^2$, we have $x = \mathbf{y}$.

Now we can combine (4.7) with (4.8) to show that:

$$D_J^{p_k}(\mathbf{x}, \widehat{\mathbf{x}}_k) \leq D_J^{p_k}(\mathbf{x}, \widehat{\mathbf{x}}_k) + D_J^{p_{k-1}}(\widehat{\mathbf{x}}_k, \widehat{\mathbf{x}}_{k-1}) + H(\widehat{\mathbf{x}}_k, \mathbf{y}) \leq H(\mathbf{x}, \mathbf{y}) + D_J^{p_{k-1}}(\mathbf{x}, \widehat{\mathbf{x}}_{k-1})$$

For the particular choice $\mathbf{x} = x = \mathbf{y}$, we have $D_J^{p_k}(x, \widehat{\mathbf{x}}_k) \leq D_J^{p_{k-1}}(x, \widehat{\mathbf{x}}_{k-1})$ which allows us to conclude a general convergence theorem. We would like to know the final value that Osher's method converges to as well, so we shall continue.

Rearranging (4.7) and substituting $H(\mathbf{x}, \mathbf{y}) = \frac{1}{2}\|\mathbf{y} - \mathbf{x}\|^2$ and $k = i$, we can write:

$$D_J^{p_i}(\mathbf{x}, \widehat{\mathbf{x}}_i) + D_J^{p_{i-1}}(\widehat{\mathbf{x}}_i, \widehat{\mathbf{x}}_{i-1}) + \frac{1}{2}\|\mathbf{y} - \widehat{\mathbf{x}}_i\|^2 - \frac{1}{2}\|\mathbf{y} - \mathbf{x}\|^2 \leq D_J^{p_{i-1}}(\mathbf{x}, \widehat{\mathbf{x}}_{i-1}).$$

We can then repeatedly substitute $i = 1, 2, \ldots, k$ into the above formulation to obtain:

$$D_J^{p_k}(\mathbf{x}, \widehat{\mathbf{x}}_k) + \sum_{i=1}^{k} \left[D_J^{p_{i-1}}(\widehat{\mathbf{x}}_i, \widehat{\mathbf{x}}_{i-1}) + \frac{1}{2}\|\mathbf{y} - \widehat{\mathbf{x}}_i\|^2 - \frac{1}{2}\|\mathbf{y} - \mathbf{x}\|^2 \right] \leq D_J^{p_0}(\mathbf{x}, \widehat{\mathbf{x}}_0) = J(\mathbf{x}). \quad (4.9)$$

For example: when $i = 1$ we have

$$D_J^{p_1}(\mathbf{x}, \widehat{\mathbf{x}}_1) + D_J^{p_0}(\widehat{\mathbf{x}}_1, \widehat{\mathbf{x}}_0) + \frac{1}{2}\|\mathbf{y} - \widehat{\mathbf{x}}_1\|^2 - \frac{1}{2}\|\mathbf{y} - \mathbf{x}\|^2 \leq D_J^{p_0}(\mathbf{x}, \widehat{\mathbf{x}}_0) = J(\mathbf{x}), \quad (4.10)$$

and when $i = 2$ we have

$$D_J^{p_2}(\mathbf{x}, \widehat{\mathbf{x}}_2) + D_J^{p_1}(\widehat{\mathbf{x}}_2, \widehat{\mathbf{x}}_1) + \frac{1}{2}\|\mathbf{y} - \widehat{\mathbf{x}}_2\|^2 - \frac{1}{2}\|\mathbf{y} - \mathbf{x}\|^2 \leq D_J^{p_1}(\mathbf{x}, \widehat{\mathbf{x}}_1). \quad (4.11)$$

Substituting (4.11) into (4.10) yields:

$$D_J^{p_2}(\mathbf{x}, \widehat{\mathbf{x}}_2) + \sum_{i=1}^{2} \left[D_J^{p_{i-1}}(\widehat{\mathbf{x}}_i, \widehat{\mathbf{x}}_{i-1}) + \frac{1}{2}\|\mathbf{y} - \widehat{\mathbf{x}}_i\|^2 - \frac{1}{2}\|\mathbf{y} - \mathbf{x}\|^2 \right] \leq J(\mathbf{x}).$$

Now, using the fact that $D_J^{p_{i-1}}(\widehat{\mathbf{x}}_i, \widehat{\mathbf{x}}_{i-1}) \geq 0$ we can rewrite (4.9) as:

$$D_J^{p_k}(\mathbf{x}, \widehat{\mathbf{x}}_k) + \sum_{i=1}^{k} \left[\frac{1}{2}\|\mathbf{y} - \widehat{\mathbf{x}}_i\|^2 \right] - k \left[\frac{1}{2}\|\mathbf{y} - \mathbf{x}\|^2 \right] \leq J(\mathbf{x}). \quad (4.12)$$

We can then substitute $\mathbf{x} = x = \mathbf{y}$ into (4.12), (since this is the value of $\mathbf{x}$ that will guarantee convergence) to obtain:

$$D_J^{p_k}(\mathbf{y}, \widehat{\mathbf{x}}_k) + \sum_{i=1}^{k} \left[\frac{1}{2}\|\mathbf{y} - \widehat{\mathbf{x}}_i\|^2 \right] \leq J(\mathbf{y}) \quad (4.13)$$

Since the sequence $\frac{1}{2}\|\mathbf{y} - \widehat{\mathbf{x}}_i\|^2$ is monotonically non-increasing for $i = 1, 2, \ldots, k$ we can write:

$$D_J^{p_k}(\mathbf{y}, \widehat{\mathbf{x}}_k) + k \left[\frac{1}{2}\|\mathbf{y} - \widehat{\mathbf{x}}_k\|^2 \right] \leq J(\mathbf{y}). \quad (4.14)$$

Equation (4.14) can be rearranged as:

$$\frac{1}{2}\|\mathbf{y} - \widehat{\mathbf{x}}_k\|^2 \leq \frac{J(\mathbf{y}) - D_J^{p_k}(\mathbf{y}, \widehat{\mathbf{x}}_k)}{k} \leq \frac{J(\mathbf{y})}{k}. \quad (4.15)$$

This implies that

$$\lim_{k\to\infty} \widehat{\mathbf{x}}_k = \mathbf{y}. \tag{4.16}$$

This also conveniently gives us a rate of convergence.

$$\|\mathbf{y} - \widehat{\mathbf{x}}_k\|^2 \leq \frac{2J(\mathbf{y})}{k} = \mathcal{O}\left((k)^{-1}\right) \tag{4.17}$$

□

We verify this rate of convergence by plotting $\|\mathbf{y} - \widehat{\mathbf{x}}_k\|^2$ vs k in Figure (**4.1**). To generate this curve, $\widehat{\mathbf{x}}_k$ was calculated from the image in Figure (**4.5** b) using Osher's method. We note that $\|\mathbf{y} - \widehat{\mathbf{x}}_k\|^2$ does in fact approach 0 as $k \to \infty$. The rate at which it approaches 0 appears to be on the order of k^{-3}, which indicates that (4.17) may not be a tight bound.

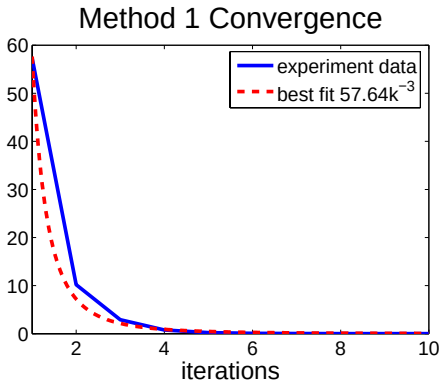


Figure 4.1: Plot of $\|\mathbf{y} - \widehat{\mathbf{x}}_k\|^2$ vs k verifying the convergence properties of method 1.

4.2 Convergence of SRR

Recall that SRR is formulated in terms of the generalized Bregman distance as:

$$\begin{aligned}\widehat{\mathbf{x}}_{k+1} &= \arg\min_{\mathbf{x}} \{H(\mathbf{x}, \mathbf{y} - \widehat{\mathbf{x}}_1) + J(\mathbf{x}) - J(\widetilde{\mathbf{v}}_{k-1}) - \langle \mathbf{p}_{k-1}, \mathbf{x} - \widetilde{\mathbf{v}}_k \rangle\} \\ &= \arg\min_{\mathbf{x}} \left\{H(\mathbf{x}, \mathbf{y} - \widehat{\mathbf{x}}_1) + D_J^{\mathbf{p}_{k-1}}(\mathbf{x}, \widetilde{\mathbf{v}}_{k-1})\right\} \qquad (4.18)\end{aligned}$$

where

$$\mathbf{p}_{k+1} = \mathbf{p}_k + \left(\frac{\partial H(\mathbf{x}, \mathbf{y} - \widehat{\mathbf{x}}_1)}{\partial \mathbf{x}} \bigg| \mathbf{x} = \widetilde{\mathbf{v}}_k \right) \qquad (4.19)$$

with $\mathbf{p}_0 = 0$ and $\widetilde{\mathbf{v}}_0 = 0$.

Lemma 7. *For non-negative $J(\mathbf{x})$, the sequence $\frac{1}{2}\|\mathbf{y} - \widehat{\mathbf{x}}_1 - \widetilde{\mathbf{v}}_{i-1}\|^2$ is monotonically non-increasing for $i = 2, 3, \ldots, k$.*

Proof: We prove the monotonicity as follows:

$$\frac{1}{2}\|\mathbf{y} - \widehat{\mathbf{x}}_1 - \widetilde{\mathbf{v}}_k\|^2 \leq \frac{1}{2}\|\mathbf{y} - \widehat{\mathbf{x}}_1 - \widetilde{\mathbf{v}}_k\|^2 + D_J^{p_{k-1}}(\widetilde{\mathbf{v}}_k, \widetilde{\mathbf{v}}_{k-1})$$

due to the non-negativity of the Bregman distance. Note that $\mathbf{x} = \widetilde{\mathbf{v}}_k$ minimizes $\frac{1}{2}\|\mathbf{y} - \widehat{\mathbf{x}}_1 - \mathbf{x}\|^2 + D_J^{p_{k-1}}(\mathbf{x}, \widetilde{\mathbf{v}}_{k-1})$ by definition. Hence,

$$\begin{aligned}\frac{1}{2}\|\mathbf{y} - \widehat{\mathbf{x}}_1 - \widetilde{\mathbf{v}}_k\|^2 &\leq \frac{1}{2}\|\mathbf{y} - \widehat{\mathbf{x}}_1 - \widetilde{\mathbf{v}}_k\|^2 + D_J^{p_{k-1}}(\widetilde{\mathbf{v}}_k, \widetilde{\mathbf{v}}_{k-1}) \\ &\leq \frac{1}{2}\|\mathbf{y} - \widehat{\mathbf{x}}_1 - \widetilde{\mathbf{v}}_{k-1}\|^2 + D_J^{p_{k-1}}(\widetilde{\mathbf{v}}_{k-1}, \widetilde{\mathbf{v}}_{k-1}) = \frac{1}{2}\|\mathbf{y} - \widehat{\mathbf{x}}_1 - \widetilde{\mathbf{v}}_{k-1}\|^2. \qquad (4.20)\end{aligned}$$

□

Theorem 2. *For non-negative, convex $J(\mathbf{x})$, as $k \to \infty$, SRR converges to*

$$\begin{aligned}\mathbf{y} - \widehat{\mathbf{x}}_1 &= \widetilde{\mathbf{v}}_{k-1} \\ \widehat{\mathbf{x}}_1 + \widetilde{\mathbf{v}}_{k-1} &= \mathbf{y} \\ \widehat{\mathbf{x}}_{k-1} &= \mathbf{y} \qquad (4.21)\end{aligned}$$

with rate of convergence

$$\|\mathbf{y} - \widehat{\mathbf{x}}_1 - \widetilde{\mathbf{v}}_{k-1}\|^2 \leq \frac{2J(\mathbf{y} - \widehat{\mathbf{x}}_1)}{k-1} = \mathcal{O}\left((k)^{-1}\right) \tag{4.22}$$

Proof: We begin the proof with the identity:

$$\begin{aligned} D_J^{p_{k-1}}(\mathbf{x}, \widetilde{\mathbf{v}}_{k-1}) - D_J^{p_{k-2}}(\mathbf{x}, \widetilde{\mathbf{v}}_{k-2}) + D_J^{p_{k-2}}(\widetilde{\mathbf{v}}_{k-1}, \widetilde{\mathbf{v}}_{k-2}) &= \langle \widetilde{\mathbf{v}}_{k-1} - \mathbf{x}, \mathbf{p}_{k-1} - \mathbf{p}_{k-2} \rangle \\ &= \langle \widetilde{\mathbf{v}}_{k-1} - \mathbf{x}, \left(\frac{\partial H(\mathbf{x}, \mathbf{y} - \widehat{\mathbf{x}}_1)}{\partial \mathbf{x}} \middle| \mathbf{x} = \widetilde{\mathbf{v}}_{k-2} \right) \rangle. \end{aligned} \tag{4.23}$$

Note that $\left(\frac{\partial H(\mathbf{x}, \mathbf{y} - \widehat{\mathbf{x}}_1)}{\partial \mathbf{x}} \middle| \mathbf{x} = \widetilde{\mathbf{v}}_{k-2} \right)$ is a subgradient of $H(\mathbf{x}, \mathbf{y} - \widehat{\mathbf{x}}_1)$. Hence, from the definition of the subgradient (A.1), we have the following relation:

$$\begin{aligned} &D_J^{p_{k-1}}(\mathbf{x}, \widetilde{\mathbf{v}}_{k-1}) - D_J^{p_{k-2}}(\mathbf{x}, \widetilde{\mathbf{v}}_{k-2}) + D_J^{p_{k-2}}(\widetilde{\mathbf{v}}_{k-1}, \widetilde{\mathbf{v}}_{k-2}) \\ &\qquad \leq H(\mathbf{x}, \mathbf{y} - \widehat{\mathbf{x}}_1) - H(\widetilde{\mathbf{v}}_{k-1}, \mathbf{y} - \widehat{\mathbf{x}}_1). \end{aligned} \tag{4.24}$$

If there exists a minimizer x of $H(\mathbf{x}, \mathbf{y} - \widehat{\mathbf{x}}_1)$ such that $J(x) < \infty$, from the particular choice of $\mathbf{x} = x$, the following is true:

$$D_J^{p_{k-1}}(x, \widetilde{\mathbf{v}}_{k-1}) \leq D_J^{p_{k-1}}(x, \widetilde{\mathbf{v}}_{k-1}) + D_J^{p_{k-2}}(\widetilde{\mathbf{v}}_{k-1}, \widetilde{\mathbf{v}}_{k-2}) + H(x, \mathbf{y} - \widehat{\mathbf{x}}_1) \tag{4.25}$$

due to the non-negativity of the Bregman distance and H. Since we are considering $H(\mathbf{x}, \mathbf{y} - \widehat{\mathbf{x}}_1) = \frac{1}{2}\|\mathbf{y} - \widehat{\mathbf{x}}_1 - \mathbf{x}\|^2$, we have $x = \mathbf{y} - \widehat{\mathbf{x}}_1$.

Now we can combine (4.24) with (4.25) to show that:

$$\begin{aligned} D_J^{p_{k-1}}(\mathbf{x}, \widetilde{\mathbf{v}}_{k-1}) &\leq D_J^{p_{k-1}}(\mathbf{x}, \widetilde{\mathbf{v}}_{k-1}) + D_J^{p_{k-2}}(\widetilde{\mathbf{v}}_{k-1}, \widetilde{\mathbf{v}}_{k-2}) + H(\widetilde{\mathbf{v}}_{k-1}, \mathbf{y} - \widehat{\mathbf{x}}_1) \\ &\leq H(\mathbf{x}, \mathbf{y} - \widehat{\mathbf{x}}_1) + D_J^{p_{k-2}}(\mathbf{x}, \widetilde{\mathbf{v}}_{k-2}). \end{aligned}$$

For the particular choice $\mathbf{x} = x = \mathbf{y} - \widehat{\mathbf{x}}_1$, we have $D_J^{p_{k-1}}(x, \widetilde{\mathbf{v}}_{k-1}) \leq D_J^{p_{k-2}}(x, \widetilde{\mathbf{v}}_{k-2})$ which allows us to conclude a general convergence theorem. We would like to know the final value that SRR converges to as well, so we shall continue.

Rearranging (4.24) and substituting $H(\mathbf{x}, \mathbf{y} - \widehat{\mathbf{x}}_1) = \frac{1}{2}\|\mathbf{y} - \widehat{\mathbf{x}}_1 - \mathbf{x}\|^2$ and $k = i$, we can write:

$$D_J^{p_{i-1}}(\mathbf{x}, \widetilde{\mathbf{v}}_{i-1}) + D_J^{p_{i-2}}(\widetilde{\mathbf{v}}_{i-1}, \widetilde{\mathbf{v}}_{i-2}) + \frac{1}{2}\|\mathbf{y} - \widehat{\mathbf{x}}_1 - \widetilde{\mathbf{v}}_{i-1}\|^2 - \frac{1}{2}\|\mathbf{y} - \widehat{\mathbf{x}}_1 - \mathbf{x}\|^2 \leq D_J^{p_{i-2}}(\mathbf{x}, \widetilde{\mathbf{v}}_{i-2}).$$

We can then repeatedly substitute $i = 2, 3, \ldots, k$ into the above formulation to obtain:

$$D_J^{p_{k-1}}(\mathbf{x}, \widetilde{\mathbf{v}}_{k-1}) + \sum_{i=2}^{k} \left[D_J^{p_{i-2}}(\widetilde{\mathbf{v}}_{i-1}, \widetilde{\mathbf{v}}_{i-2}) + \frac{1}{2}\|\mathbf{y} - \widehat{\mathbf{x}}_1 - \widetilde{\mathbf{v}}_{i-1}\|^2 - \frac{1}{2}\|\mathbf{y} - \widehat{\mathbf{x}}_1 - \mathbf{x}\|^2 \right] \leq D_J^{p_0}(\mathbf{x}, \widetilde{\mathbf{v}}_0) = J(\mathbf{x}). \quad (4.26)$$

For example: when $i = 2$ we have

$$D_J^{p_1}(\mathbf{x}, \widetilde{\mathbf{v}}_1) + D_J^{p_0}(\widetilde{\mathbf{v}}_1, \widetilde{\mathbf{v}}_0) + \frac{1}{2}\|\mathbf{y} - \widehat{\mathbf{x}}_1 - \widetilde{\mathbf{v}}_1\|^2 - \frac{1}{2}\|\mathbf{y} - \widehat{\mathbf{x}}_1 - \mathbf{x}\|^2 \leq D_J^{p_0}(\mathbf{x}, \widetilde{\mathbf{v}}_0) = J(\mathbf{x}) \quad (4.27)$$

and when $i = 3$ we have

$$D_J^{p_2}(\mathbf{x}, \widetilde{\mathbf{v}}_2) + D_J^{p_1}(\widetilde{\mathbf{v}}_2, \widetilde{\mathbf{v}}_1) + \frac{1}{2}\|\mathbf{y} - \widehat{\mathbf{x}}_1 - \widetilde{\mathbf{v}}_2\|^2 - \frac{1}{2}\|\mathbf{y} - \widehat{\mathbf{x}}_1 - \mathbf{x}\|^2 \leq D_J^{p_1}(\mathbf{x}, \widetilde{\mathbf{v}}_1). \quad (4.28)$$

Substituting (4.28) into (4.27) yields:

$$D_J^{p_2}(\mathbf{x}, \widetilde{\mathbf{v}}_{k-1}) + \sum_{i=2}^{3} \left[D_J^{p_{i-2}}(\widetilde{\mathbf{v}}_{i-1}, \widetilde{\mathbf{v}}_{i-2}) + \frac{1}{2}\|\mathbf{y} - \widehat{\mathbf{x}}_1 - \widetilde{\mathbf{v}}_{i-1}\|^2 - \frac{1}{2}\|\mathbf{y} - \widehat{\mathbf{x}}_1 - \mathbf{x}\|^2 \right] \leq J(\mathbf{x}).$$

Now, using the fact that $D_J^{p_{i-2}}(\widetilde{\mathbf{v}}_{i-1}, \widetilde{\mathbf{v}}_{i-2}) \geq 0$ we can rewrite (4.26) as:

$$D_J^{p_{k-1}}(\mathbf{x}, \widetilde{\mathbf{v}}_{k-1}) + \sum_{i=2}^{k} \left[\frac{1}{2}\|\mathbf{y} - \widehat{\mathbf{x}}_1 - \widetilde{\mathbf{v}}_{i-1}\|^2 \right] - (k-1) \left[\frac{1}{2}\|\mathbf{y} - \widehat{\mathbf{x}}_1 - \mathbf{x}\|^2 \right] \leq J(\mathbf{x}). \quad (4.29)$$

We can then substitute $\mathbf{x} = x = \mathbf{y} - \widehat{\mathbf{x}}_1$ into (4.29), (since this is the value of $\mathbf{x}$ that will guarantee convergence) to obtain:

$$D_J^{p_{k-1}}(\mathbf{y} - \widehat{\mathbf{x}}_1, \widetilde{\mathbf{v}}_{k-1}) + \sum_{i=2}^{k} \left[\frac{1}{2}\|\mathbf{y} - \widehat{\mathbf{x}}_1 - \widetilde{\mathbf{v}}_{i-1}\|^2 \right] \leq J(\mathbf{y} - \widehat{\mathbf{x}}_1) \quad (4.30)$$

Since the sequence $\frac{1}{2}\|\mathbf{y} - \widehat{\mathbf{x}}_1 - \widetilde{\mathbf{v}}_{i-1}\|^2$ is monotonically non-increasing for $i = 2, 3, \ldots, k$ we can write:

$$D_J^{p_{k-1}}(\mathbf{y} - \widehat{\mathbf{x}}_1, \widetilde{\mathbf{v}}_{k-1}) + (k-1) \left[\frac{1}{2}\|\mathbf{y} - \widehat{\mathbf{x}}_1 - \widetilde{\mathbf{v}}_{k-1}\|^2 \right] \leq J(\mathbf{y} - \widehat{\mathbf{x}}_1). \quad (4.31)$$

Equation (4.31) can be rearranged as:

$$\frac{1}{2}\|\mathbf{y}-\widehat{\mathbf{x}}_1-\widetilde{\mathbf{v}}_{k-1}\|^2 \leq \frac{J(\mathbf{y}-\widehat{\mathbf{x}}_1)-D_J^{p_{k-1}}(\mathbf{y}-\widehat{\mathbf{x}}_1,\widetilde{\mathbf{v}}_{k-1})}{k-1} \leq \frac{J(\mathbf{y}-\widehat{\mathbf{x}}_1)}{k-1} \tag{4.32}$$

This implies that as $k \to \infty$

$$\begin{aligned} \mathbf{y}-\widehat{\mathbf{x}}_1 &= \widetilde{\mathbf{v}}_{k-1} \\ \widehat{\mathbf{x}}_1+\widetilde{\mathbf{v}}_{k-1} &= \mathbf{y} \\ \widehat{\mathbf{x}}_{k-1} &= \mathbf{y}. \end{aligned} \tag{4.33}$$

This also conveniently gives us a rate of convergence.

$$\|\mathbf{y}-\widehat{\mathbf{x}}_1-\widetilde{\mathbf{v}}_{k-1}\|^2 \leq \frac{2J(\mathbf{y}-\widehat{\mathbf{x}}_1)}{k-1} = \mathcal{O}\left((k)^{-1}\right) \tag{4.34}$$

□

We verify this rate of convergence by plotting $\|\mathbf{y}-\widehat{\mathbf{x}}_1-\widehat{\mathbf{x}}_k\|^2$ vs k in Figure (**4.2**). To generate this curve, $\widehat{\mathbf{x}}_k$ was calculated from the image in Figure (**4.5** b) using SRR. Because we don't compute $\widetilde{\mathbf{v}}_k$ until the second iteration, our plot begins at $k=2$. We note that $\|\mathbf{y}-\widehat{\mathbf{x}}_1-\widehat{\mathbf{x}}_k\|^2$ does in fact approach 0 as $k \to \infty$. The rate at which it approaches 0 appears to be on the order of k^{-3}, which indicates that (4.34) may not be a tight bound.

4.3 Convergence of ITR

Recall that ITR can be formulated such that each estimate can be calculated as a simple update to the previous estimate:

$$\widehat{\mathbf{x}}_{k+1} = \widehat{\mathbf{x}}_k + \mathcal{B}(\mathbf{y}-\widehat{\mathbf{x}}_k). \tag{4.35}$$

If we assume that the sequence of estimates produced by this method are convergent, then as $k \to \infty$, $\widehat{\mathbf{x}}_{k+1} = \widehat{\mathbf{x}}_k = \widehat{\mathbf{x}}_\infty$, which implies that $\mathcal{B}(\mathbf{y}-\widehat{\mathbf{x}}_\infty) = 0$. Remember that $\mathcal{B}(\cdot)$ here

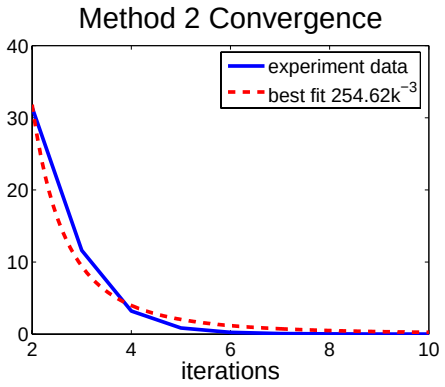


Figure 4.2: Plot of $\|\mathbf{y} - \widehat{\mathbf{x}}_1 - \widehat{\mathbf{x}}_k\|^2$ vs k verifying the convergence properties of SRR

denotes the minimization of the cost function $Q_k(\mathbf{x}, \cdot) = \frac{1}{2}\| \cdot - \mathbf{x}\|^2 + J(\mathbf{x})$. Thus, if the regularization term $J(\mathbf{x})$ is non-negative and convex with a minimum at $\mathbf{x} = 0$, then

$$\frac{\partial Q_k(\mathbf{x}, 0)}{\partial \mathbf{x}} = \mathbf{x} + \frac{\partial J(\mathbf{x})}{\partial \mathbf{x}} = 0 \iff \mathbf{x} = 0. \tag{4.36}$$

Thus $\mathcal{B}(\mathbf{y} - \widehat{\mathbf{x}}_\infty)$ implies that $\mathbf{y} = \widehat{\mathbf{x}}_\infty$ once convergence has occurred. We shall now thoroughly prove that the series of estimates produced by ITR is convergent.

Recall that ITR is formulated in terms of the generalized Bregman distance as:

$$\begin{aligned} \widehat{\mathbf{x}}_{k+1} &= \widehat{\mathbf{x}}_1 + \sum_{i=1}^{k-1} \widetilde{\mathbf{r}}_i + \arg\min_{\mathbf{x}} \{H(\mathbf{x}, 0) + J(\mathbf{x}) - J(\widetilde{\mathbf{r}}_{k-1}) - \langle \mathbf{p}_{k-1}, \mathbf{x} - \widetilde{\mathbf{r}}_{k-1} \rangle\} \\ &= \widehat{\mathbf{x}}_1 + \sum_{i=1}^{k-1} \widetilde{\mathbf{r}}_i + \arg\min_{\mathbf{x}} \left\{H(\mathbf{x}, 0) + D_J^{p_{k-1}}(\mathbf{x}, \widetilde{\mathbf{r}}_{k-1})\right\} \end{aligned} \tag{4.37}$$

where

$$\mathbf{p}_{k+1} = \mathbf{p}_k + \left(\frac{\partial H(\mathbf{x}, 0)}{\partial \mathbf{x}} \middle| \mathbf{x} = \widetilde{\mathbf{r}}_{k+1} \right) \tag{4.38}$$

with $\mathbf{p}_0 = \mathbf{y} - \widehat{\mathbf{x}}_1$.

Lemma 8. *For non-negative $J(\mathbf{x})$, the sequence $\frac{1}{2}\|\widetilde{\mathbf{r}}_{i-1}\|^2$ is monotonically non-increasing for $i = 2, 3, \ldots, k$.*

Proof: We prove the monotonicity as follows:

$$\frac{1}{2}\|\widetilde{\mathbf{r}}_k\|^2 \leq \frac{1}{2}\|\widetilde{\mathbf{r}}_k\|^2 + D_J^{p_{k-1}}(\widetilde{\mathbf{r}}_k, \widetilde{\mathbf{r}}_{k-1})$$

due to the non-negativity of the Bregman distance. Note that $\mathbf{x} = \widetilde{\mathbf{r}}_k$ minimizes $\frac{1}{2}\|\mathbf{x}\|^2 + D_J^{p_{k-1}}(\mathbf{x}, \widetilde{\mathbf{r}}_{k-1})$ by definition. Hence,

$$\frac{1}{2}\|\widetilde{\mathbf{r}}_k\|^2 \leq \frac{1}{2}\|\widetilde{\mathbf{r}}_k\|^2 + D_J^{p_{k-1}}(\widetilde{\mathbf{r}}_k, \widetilde{\mathbf{r}}_{k-1}) \leq \frac{1}{2}\|\widetilde{\mathbf{r}}_{k-1}\|^2 + D_J^{p_{k-1}}(\widetilde{\mathbf{r}}_{k-1}, \widetilde{\mathbf{r}}_{k-1}) = \frac{1}{2}\|\widetilde{\mathbf{r}}_{k-1}\|^2. \quad (4.39)$$

□

Theorem 3. *For non-negative, convex $J(\mathbf{x})$ with a minimum at $\mathbf{x} = 0$, ITR converges as:*

$$\lim_{k\to\infty} \widehat{\mathbf{x}}_{k-1} = \mathbf{y}, \quad (4.40)$$

with rate of convergence

$$\|\widetilde{\mathbf{r}}_{k-1}\|^2 \leq \frac{2\left(J(0) - J(\widehat{\mathbf{x}}_1) + \langle \mathbf{y} - \widehat{\mathbf{x}}_1, \widehat{\mathbf{x}}_1\rangle\right)}{k-1} = \mathcal{O}\left((k)^{-1}\right). \quad (4.41)$$

Proof: We begin the proof with the identity:

$$\begin{aligned} D_J^{p_{k-1}}(\mathbf{x}, \widetilde{\mathbf{r}}_{k-1}) - D_J^{p_{k-2}}(\mathbf{x}, \widetilde{\mathbf{r}}_{k-2}) + D_J^{p_{k-2}}(\widetilde{\mathbf{r}}_{k-1}, \widetilde{\mathbf{r}}_{k-2}) &= \langle \widetilde{\mathbf{r}}_{k-1} - \mathbf{x}, \mathbf{p}_{k-1} - \mathbf{p}_{k-2}\rangle \\ &= \langle \widetilde{\mathbf{r}}_{k-1} - \mathbf{x}, \left(\frac{\partial H(\mathbf{x}, 0)}{\partial \mathbf{x}} \middle| \mathbf{x} = \widetilde{\mathbf{r}}_{k-1} \right)\rangle. \quad (4.42) \end{aligned}$$

Note that $\left(\frac{\partial H(\mathbf{x},0)}{\partial \mathbf{x}} \middle| \mathbf{x} = \widetilde{\mathbf{r}}_{k-1} \right)$ is a subgradient of $H(\mathbf{x}, 0)$. Hence, from the definition of the subgradient (A.1), we have the following relation:

$$D_J^{p_{k-1}}(\mathbf{x}, \widetilde{\mathbf{r}}_{k-1}) - D_J^{p_{k-2}}(\mathbf{x}, \widetilde{\mathbf{r}}_{k-2}) + D_J^{p_{k-2}}(\widetilde{\mathbf{r}}_{k-1}, \widetilde{\mathbf{r}}_{k-2}) \leq H(\mathbf{x}, 0) - H(\widetilde{\mathbf{r}}_{k-1}, 0) \quad (4.43)$$

If there exists a minimizer x of $H(\mathbf{x}, 0)$ such that $J(x) < \infty$, from the particular choice of $\mathbf{x} = x$, the following is true:

$$D_J^{p_{k-1}}(x, \widetilde{\mathbf{r}}_{k-1}) \leq D_J^{p_{k-1}}(x, \widetilde{\mathbf{r}}_{k-1}) + D_J^{p_{k-2}}(\widetilde{\mathbf{r}}_{k-1}, \widetilde{\mathbf{r}}_{k-2}) + H(x, 0) \quad (4.44)$$

due to the non-negativity of the Bregman distance and H. Since we are considering $H(\mathbf{x},0)=\frac{1}{2}\|\mathbf{x}\|^2$, we have $x=0$.

Now we can combine (4.43) with (4.44) to show that:

$$D_J^{p_{k-1}}(\mathbf{x},\widetilde{\mathbf{r}}_{k-1}) \leq D_J^{p_{k-1}}(\mathbf{x},\widetilde{\mathbf{r}}_{k-1})+D_J^{p_{k-2}}(\widetilde{\mathbf{r}}_{k-1},\widetilde{\mathbf{r}}_{k-2})+H(\widetilde{\mathbf{r}}_{k-1},0) \leq H(\mathbf{x},0)+D_J^{p_{k-2}}(\mathbf{x},\widetilde{\mathbf{r}}_{k-2}).$$

For the particular choice $\mathbf{x}=x=0$, we have $D_J^{p_{k-1}}(x,\widetilde{\mathbf{r}}_{k-1}) \leq D_J^{p_{k-2}}(x,\widetilde{\mathbf{r}}_{k-2})$ which allows us to conclude a general convergence theorem.

Rearranging (4.43) and substituting $H(\mathbf{x},0)=\frac{1}{2}\|\mathbf{x}\|^2$ and $k=i$, we can write:

$$D_J^{p_{i-1}}(\mathbf{x},\widetilde{\mathbf{r}}_{i-1})+D_J^{p_{i-2}}(\widetilde{\mathbf{r}}_{i-1},\widetilde{\mathbf{r}}_{i-2})+\frac{1}{2}\|\widetilde{\mathbf{r}}_{i-1}\|^2-\frac{1}{2}\|\mathbf{x}\|^2 \leq D_J^{p_{i-2}}(\mathbf{x},\widetilde{\mathbf{r}}_{i-2}).$$

We can then repeatedly substitute $i=2,3,\ldots,k$ into the above formulation to obtain:

$$\begin{aligned} D_J^{p_{k-1}}(\mathbf{x},\widetilde{\mathbf{r}}_{k-1})+\sum_{i=2}^{k}\left[D_J^{p_{i-2}}(\widetilde{\mathbf{r}}_{i-1},\widetilde{\mathbf{r}}_{i-2})+\frac{1}{2}\|\widetilde{\mathbf{x}}_{i-1}\|^2-\frac{1}{2}\|\mathbf{x}\|^2\right] &\leq D_J^{p_0}(\mathbf{x},\widetilde{\mathbf{r}}_0) \\ &\leq J(\mathbf{x})-J(\widehat{\mathbf{x}}_1)-\langle \mathbf{y}-\widehat{\mathbf{x}}_1,\mathbf{x}-\widehat{\mathbf{x}}_1\rangle. \quad (4.45) \end{aligned}$$

For example: when $i=2$ we have

$$D_J^{p_1}(\mathbf{x},\widetilde{\mathbf{r}}_1)+D_J^{p_0}(\widetilde{\mathbf{r}}_1,\widetilde{\mathbf{r}}_0)+\frac{1}{2}\|\widetilde{\mathbf{r}}_1\|^2-\frac{1}{2}\|\mathbf{x}\|^2 \leq D_J^{p_0}(\mathbf{x},\widetilde{\mathbf{r}}_0) \quad (4.46)$$

and when $i=3$ we have

$$D_J^{p_2}(\mathbf{x},\widetilde{\mathbf{r}}_2)+D_J^{p_1}(\widetilde{\mathbf{r}}_2,\widetilde{\mathbf{r}}_1)+\frac{1}{2}\|\widetilde{\mathbf{r}}_2\|^2-\frac{1}{2}\|\mathbf{x}\|^2 \leq D_J^{p_1}(\mathbf{x},\widetilde{\mathbf{r}}_1). \quad (4.47)$$

Substituting (4.47) into (4.46) yields:

$$D_J^{p_2}(\mathbf{x},\widetilde{\mathbf{r}}_{k-1})+\sum_{i=2}^{3}\left[D_J^{p_{i-2}}(\widetilde{\mathbf{r}}_{i-1},\widetilde{\mathbf{r}}_{i-2})+\frac{1}{2}\|\widetilde{\mathbf{r}}_{i-1}\|^2-\frac{1}{2}\|\mathbf{x}\|^2\right] \leq D_J^{p_0}(\mathbf{x},\widetilde{\mathbf{r}}_0).$$

Now, using the fact that $D_J^{p_{i-2}}(\widetilde{\mathbf{r}}_{i-1},\widetilde{\mathbf{r}}_{i-2}) \geq 0$ we can rewrite (4.45) as:

$$D_J^{p_{k-1}}(\mathbf{x},\widetilde{\mathbf{r}}_{k-1})+\sum_{i=2}^{k}\left[\frac{1}{2}\|\widetilde{\mathbf{r}}_{i-1}\|^2\right]-(k-1)\left[\frac{1}{2}\|\mathbf{x}\|^2\right] \leq J(\mathbf{x})-J(\widehat{\mathbf{x}}_1)-\langle \mathbf{y}-\widehat{\mathbf{x}}_1,\mathbf{x}-\widehat{\mathbf{x}}_1\rangle. \quad (4.48)$$

We can then substitute $\mathbf{x} = x = 0$ into (4.48), (since this is the value of $\mathbf{x}$ that will guarantee convergence) to obtain:

$$D_J^{p_{k-1}}(0, \widetilde{\mathbf{r}}_{k-1}) + \sum_{i=2}^{k} \left[\frac{1}{2} \|\widetilde{\mathbf{r}}_{i-1}\|^2 \right] \leq J(0) - J(\widehat{\mathbf{x}}_1) + \langle \mathbf{y} - \widehat{\mathbf{x}}_1, \widehat{\mathbf{x}}_1 \rangle. \tag{4.49}$$

Since the sequence $\frac{1}{2}\|\widetilde{\mathbf{r}}_{i-1}\|^2$ is monotonically non-increasing for $i = 2, 3, \ldots, k$ we can write:

$$D_J^{p_{k-1}}(0, \widetilde{\mathbf{r}}_{k-1}) + (k-1) \left[\frac{1}{2} \|\widetilde{\mathbf{r}}_{k-1}\|^2 \right] \leq J(0) - J(\widehat{\mathbf{x}}_1) + \langle \mathbf{y} - \widehat{\mathbf{x}}_1, \widehat{\mathbf{x}}_1 \rangle. \tag{4.50}$$

Equation (4.50) can be rearranged as:

$$\begin{aligned} \frac{1}{2} \|\widetilde{\mathbf{r}}_{k-1}\|^2 &\leq \frac{J(0) - J(\widehat{\mathbf{x}}_1) + \langle \mathbf{y} - \widehat{\mathbf{x}}_1, \widehat{\mathbf{x}}_1 \rangle - D_J^{p_{k-1}}(0, \widetilde{\mathbf{r}}_{k-1})}{k-1} \\ &\leq \frac{J(0) - J(\widehat{\mathbf{x}}_1) + \langle \mathbf{y} - \widehat{\mathbf{x}}_1, \widehat{\mathbf{x}}_1 \rangle}{k-1} \end{aligned} \tag{4.51}$$

This implies that as $k \to \infty$,

$$\begin{aligned} \widetilde{\mathbf{r}}_{k-1} &= 0 \\ \mathcal{B}(\mathbf{y} - \widehat{\mathbf{x}}_{k-1}) &= 0. \end{aligned} \tag{4.52}$$

As we stated earlier, because the regularization function $J(\mathbf{x})$ is non-negative and convex with a minimum at $\mathbf{x} = 0$, this implies that

$$\lim_{k \to \infty} \widehat{\mathbf{x}}_{k-1} = \mathbf{y}. \tag{4.53}$$

This also conveniently gives us a rate of convergence.

$$\|\widetilde{\mathbf{r}}_{k-1}\|^2 \leq \frac{2\left(J(0) - J(\widehat{\mathbf{x}}_1) + \langle \mathbf{y} - \widehat{\mathbf{x}}_1, \widehat{\mathbf{x}}_1 \rangle\right)}{k-1} = \mathcal{O}\left((k)^{-1}\right) \tag{4.54}$$

□

We verify this rate of convergence by plotting $\|\widetilde{\mathbf{r}}_{k-1}\|^2$ vs k in Figure (**4.3**). To generate this curve, $\widehat{\mathbf{x}}_k$ was calculated from the image in Figure (**4.5** b) using ITR. Because

we don't compute $\widetilde{\mathbf{r}}_k$ until the second iteration, our plot begins at $k = 2$. We note that $\|\widetilde{\mathbf{r}}_{k-1}\|^2$ does in fact approach 0 as $k \to \infty$. The rate at which it approaches 0 appears to be on the order of k^{-3}, which indicates that (4.53) may not be a tight bound.

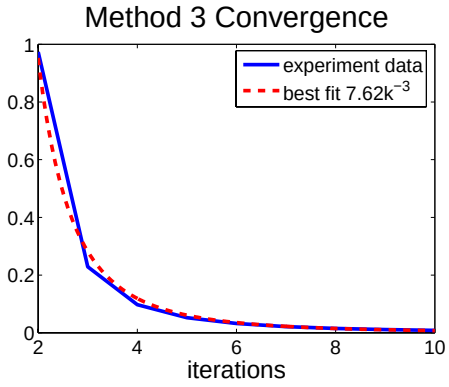


Figure 4.3: Plot of $\|\widetilde{\mathbf{r}}_{k-1}\|^2$ vs k verifying the convergence properties of method 3.

4.4 Convergence of IUR

Recall that IUR can be formulated such that each estimate can be calculated as a simple update to the previous estimate:

$$\widehat{\mathbf{x}}_{k+1} = \widehat{\mathbf{x}}_k + \widehat{\mathbf{x}}_1 - \mathcal{B}(\widehat{\mathbf{x}}_k). \tag{4.55}$$

If we assume that the sequence of estimates produced by this method are convergent, then as $k \to \infty$, $\widehat{\mathbf{x}}_{k+1} = \widehat{\mathbf{x}}_k = \widehat{\mathbf{x}}_\infty$, which implies that

$$\begin{aligned} \widehat{\mathbf{x}}_1 - \mathcal{B}(\widehat{\mathbf{x}}_\infty) &= 0 \\ \mathcal{B}(\mathbf{y}) - \mathcal{B}(\widehat{\mathbf{x}}_\infty) &= 0 \\ \mathcal{B}(\mathbf{y}) &= \mathcal{B}(\widehat{\mathbf{x}}_\infty) \end{aligned}$$

Remember that $\mathcal{B}(\cdot)$ denotes the minimization of a cost function. Thus, if the cost function that is used is convex, there will only be one minimum. Thus $\mathcal{B}(\mathbf{y}) = \mathcal{B}(\widehat{\mathbf{x}}_\infty)$, implies that $\mathbf{y} = \widehat{\mathbf{x}}_\infty$.

Notice in Equation (4.55) that IUR does not require direct knowledge of the noisy data $\mathbf{y}$ to compute the estimate $\widehat{\mathbf{x}}_k$. For $k > 1$, $\widehat{\mathbf{x}}_k$ can be computed just from $\widehat{\mathbf{x}}_1$, the filtered version of $\mathbf{y}$. The other three methods do however require direct knowledge of $\mathbf{y}$ at each iteration to compute their estimates (see Equations (2.2) (2.11) and (2.17)). We can take advantage of this property (unique to IUR) along with IUR's convergence property, to "undo" the regularization in the first iteration by continuing to iterate. Since regularization methods are typically highly non-linear operations, this is rather impressive. We give this method of recovering $\mathbf{y}$ from $\widehat{\mathbf{x}}_1$ the name "inverse regularization." The ability to be used for inverse regularization is a clear advantage of IUR over the other iterative regularization methods in this framework. We will look at a number of the applications that make use of this interesting property in the next chapter.

We shall now thoroughly prove that the series of estimates produced by IUR is convergent.

Recall that IUR is formulated in terms of the generalized Bregman distance as:

$$\begin{aligned}\widehat{\mathbf{x}}_{k+1} &= (k+1)\widehat{\mathbf{x}}_1 - \sum_{i=1}^{k-1}\widetilde{\mathbf{x}}_i - \arg\min_{\mathbf{x}}\left\{H(\mathbf{x},\widehat{\mathbf{x}}_1) + J(\mathbf{x}) - J(\widetilde{\mathbf{x}}_{k-1}) - \langle \mathbf{p}_{k-1}, \mathbf{x} - \widetilde{\mathbf{x}}_{k-1}\rangle\right\} \\ &= (k+1)\widehat{\mathbf{x}}_1 - \sum_{i=1}^{k-1}\widetilde{\mathbf{x}}_i - \arg\min_{\mathbf{x}}\left\{H(\mathbf{x},\widehat{\mathbf{x}}_1) + D_J^{p_{k-1}}(\mathbf{x},\widetilde{\mathbf{x}}_{k-1})\right\} \qquad (4.56)\end{aligned}$$

where

$$\mathbf{p}_{k+1} = \mathbf{p}_k + \left(\frac{\partial H(\mathbf{x},\widehat{\mathbf{x}}_1)}{\partial \mathbf{x}}\bigg|\, \mathbf{x} = \widetilde{\mathbf{x}}_{k+1}\right) \qquad (4.57)$$

with $\mathbf{p}_0 = 0$ and $\widetilde{\mathbf{x}}_0 = 0$.

Lemma 9. *For non-negative $J(\mathbf{x})$, the sequence $\frac{1}{2}\|\widehat{\mathbf{x}}_1 - \widetilde{\mathbf{x}}_{i-1}\|^2$ is monotonically non-*

increasing for $i = 2, 3, \ldots, k$.

Proof: We prove the monotonicity as follows:

$$\frac{1}{2}\|\widehat{\mathbf{x}}_1 - \widetilde{\mathbf{x}}_k\|^2 \leq \frac{1}{2}\|\widehat{\mathbf{x}}_1 - \widetilde{\mathbf{x}}_k\|^2 + D_J^{p_{k-1}}(\widetilde{\mathbf{x}}_k, \widetilde{\mathbf{x}}_{k-1})$$

due to the non-negativity of the Bregman distance. Note that $\mathbf{x} = \widetilde{\mathbf{x}}_k$ minimizes $\frac{1}{2}\|\widehat{\mathbf{x}}_1 - \mathbf{x}\|^2 + D_J^{p_{k-1}}(\mathbf{x}, \widetilde{\mathbf{x}}_{k-1})$ by definition. Hence,

$$\begin{aligned}\frac{1}{2}\|\widehat{\mathbf{x}}_1 - \widetilde{\mathbf{x}}_k\|^2 \leq \frac{1}{2}\|\widehat{\mathbf{x}}_1 - \widetilde{\mathbf{x}}_k\|^2 + D_J^{p_{k-1}}(\widetilde{\mathbf{x}}_k, \widetilde{\mathbf{x}}_{k-1}) &\leq \frac{1}{2}\|\widehat{\mathbf{x}}_1 - \widetilde{\mathbf{x}}_{k-1}\|^2 + D_J^{p_{k-1}}(\widetilde{\mathbf{x}}_{k-1}, \widetilde{\mathbf{x}}_{k-1}) \\ &\leq \frac{1}{2}\|\widehat{\mathbf{x}}_1 - \widetilde{\mathbf{x}}_{k-1}\|^2. \qquad (4.58)\end{aligned}$$

□

Theorem 4. *For non-negative, convex* $J(\mathbf{x})$*, IUR converges as:*

$$\lim_{k\to\infty} \widehat{\mathbf{x}}_k = \mathbf{y}, \qquad (4.59)$$

with rate of convergence

$$\|\widehat{\mathbf{x}}_1 - \widetilde{\mathbf{x}}_{k-1}\|^2 \leq \frac{2J(\widehat{\mathbf{x}}_1)}{k-1} = \mathcal{O}\left((k)^{-1}\right). \qquad (4.60)$$

Proof: We begin the proof with the identity:

$$\begin{aligned}D_J^{p_{k-1}}(\mathbf{x}, \widetilde{\mathbf{x}}_{k-1}) - D_J^{p_{k-2}}(\mathbf{x}, \widetilde{\mathbf{x}}_{k-2}) + D_J^{p_{k-2}}(\widetilde{\mathbf{x}}_{k-1}, \widetilde{\mathbf{x}}_{k-2}) &= \langle \widetilde{\mathbf{x}}_{k-1} - \mathbf{x}, \mathbf{p}_{k-1} - \mathbf{p}_{k-2}\rangle \\ &= \langle \widetilde{\mathbf{x}}_{k-1} - \mathbf{x}, \left(\frac{\partial H(\mathbf{x}, \widehat{\mathbf{x}}_1)}{\partial \mathbf{x}} \middle| \mathbf{x} = \widetilde{\mathbf{x}}_{k-1} \right)\rangle. \qquad (4.61)\end{aligned}$$

Note that $\left(\frac{\partial H(\mathbf{x}, \widehat{\mathbf{x}}_1)}{\partial \mathbf{x}} \middle| \mathbf{x} = \widetilde{\mathbf{x}}_{k-1} \right)$ is a subgradient of $H(\mathbf{x}, \widehat{\mathbf{x}}_1)$. Hence, from the definition of the subgradient (A.1), we have the following relation:

$$D_J^{p_{k-1}}(\mathbf{x}, \widetilde{\mathbf{x}}_{k-1}) - D_J^{p_{k-2}}(\mathbf{x}, \widetilde{\mathbf{x}}_{k-2}) + D_J^{p_{k-2}}(\widetilde{\mathbf{x}}_{k-1}, \widetilde{\mathbf{x}}_{k-2}) \leq H(\mathbf{x}, \widehat{\mathbf{x}}_1) - H(\widetilde{\mathbf{x}}_{k-1}, \widehat{\mathbf{x}}_1). \qquad (4.62)$$

If there exists a minimizer x of $H(\mathbf{x}, \widehat{\mathbf{x}}_1)$ such that $J(x) < \infty$, from the particular choice of $\mathbf{x} = x$, the following is true:

$$D_J^{p_{k-1}}(x, \widetilde{\mathbf{x}}_{k-1}) \leq D_J^{p_{k-1}}(x, \widetilde{\mathbf{x}}_{k-1}) + D_J^{p_{k-2}}(\widetilde{\mathbf{x}}_{k-1}, \widetilde{\mathbf{x}}_{k-2}) + H(x, \widehat{\mathbf{x}}_1) \tag{4.63}$$

due to the non-negativity of the Bregman distance and H. Since we are considering $H(\mathbf{x}, \widehat{\mathbf{x}}_1) = \frac{1}{2}\|\widehat{\mathbf{x}}_1 - \mathbf{x}\|^2$, we have $x = \widehat{\mathbf{x}}_1$.

Now we can combine (4.62) with (4.63) to show that:

$$\begin{aligned} D_J^{p_{k-1}}(\mathbf{x}, \widetilde{\mathbf{x}}_{k-1}) &\leq D_J^{p_{k-1}}(\mathbf{x}, \widetilde{\mathbf{x}}_{k-1}) + D_J^{p_{k-2}}(\widetilde{\mathbf{x}}_{k-1}, \widetilde{\mathbf{x}}_{k-2}) + H(\widetilde{\mathbf{x}}_{k-1}, \widehat{\mathbf{x}}_1) \\ &\leq H(\mathbf{x}, \widehat{\mathbf{x}}_1) + D_J^{p_{k-2}}(\mathbf{x}, \widetilde{\mathbf{x}}_{k-2}) \end{aligned}$$

For the particular choice $\mathbf{x} = x = \widehat{\mathbf{x}}_1$, we have $D_J^{p_{k-1}}(x, \widetilde{\mathbf{x}}_{k-1}) \leq D_J^{p_{k-2}}(x, \widetilde{\mathbf{x}}_{k-2})$ which allows us to conclude a general convergence theorem.

Rearranging (4.62) and substituting $H(\mathbf{x}, \widehat{\mathbf{x}}_1) = \frac{1}{2}\|\widehat{\mathbf{x}}_1 - \mathbf{x}\|^2$ and $k = i$, we can write:

$$D_J^{p_{i-1}}(\mathbf{x}, \widetilde{\mathbf{x}}_{i-1}) + D_J^{p_{i-2}}(\widetilde{\mathbf{x}}_{i-1}, \widetilde{\mathbf{x}}_{i-2}) + \frac{1}{2}\|\widehat{\mathbf{x}}_1 - \widetilde{\mathbf{x}}_{i-1}\|^2 - \frac{1}{2}\|\widehat{\mathbf{x}}_1 - \mathbf{x}\|^2 \leq +D_J^{p_{i-2}}(\mathbf{x}, \widetilde{\mathbf{x}}_{i-2}).$$

We can then repeatedly substitute $i = 2, 3, \ldots, k$ into the above formulation to obtain:

$$\begin{aligned} D_J^{p_{k-1}}(\mathbf{x}, \widetilde{\mathbf{x}}_{k-1}) + \sum_{i=2}^{k} \left[D_J^{p_{i-2}}(\widetilde{\mathbf{x}}_{i-1}, \widetilde{\mathbf{x}}_{i-2}) + \frac{1}{2}\|\widehat{\mathbf{x}}_1 - \widetilde{\mathbf{x}}_{i-1}\|^2 - \frac{1}{2}\|\widehat{\mathbf{x}}_1 - \mathbf{x}\|^2 \right] \\ \leq D_J^{p_0}(\mathbf{x}, \widetilde{\mathbf{x}}_0) = J(\mathbf{x}). \end{aligned} \tag{4.64}$$

For example: when $i = 2$ we have

$$D_J^{p_1}(\mathbf{x}, \widetilde{\mathbf{x}}_1) + D_J^{p_0}(\widetilde{\mathbf{x}}_1, \widetilde{\mathbf{x}}_0) + \frac{1}{2}\|\widehat{\mathbf{x}}_1 - \widetilde{\mathbf{x}}_1\|^2 - \frac{1}{2}\|\widehat{\mathbf{x}}_1 - \mathbf{x}\|^2 \leq D_J^{p_0}(\mathbf{x}, \widetilde{\mathbf{x}}_0) = J(\mathbf{x}) \tag{4.65}$$

and when $i = 3$ we have

$$D_J^{p_2}(\mathbf{x}, \widetilde{\mathbf{x}}_2) + D_J^{p_1}(\widetilde{\mathbf{x}}_2, \widetilde{\mathbf{x}}_1) + \frac{1}{2}\|\widehat{\mathbf{x}}_1 - \widetilde{\mathbf{x}}_2\|^2 - \frac{1}{2}\|\widehat{\mathbf{x}}_1 - \mathbf{x}\|^2 \leq D_J^{p_1}(\mathbf{x}, \widetilde{\mathbf{x}}_1). \tag{4.66}$$

Substituting (4.66) into (4.65) yields:

$$D_J^{p_2}(\mathbf{x}, \widetilde{\mathbf{x}}_{k-1}) + \sum_{i=2}^{3} \left[D_J^{p_{i-2}}(\widetilde{\mathbf{x}}_{i-1}, \widetilde{\mathbf{x}}_{i-2}) + \frac{1}{2}\|\widehat{\mathbf{x}}_1 - \widetilde{\mathbf{x}}_{i-1}\|^2 - \frac{1}{2}\|\widehat{\mathbf{x}}_1 - \mathbf{x}\|^2 \right] \leq J(\mathbf{x}).$$

Now, using the fact that $D_J^{p_{i-2}}(\widetilde{\mathbf{x}}_{i-1}, \widetilde{\mathbf{x}}_{i-2}) \geq 0$ we can rewrite (4.64) as:

$$D_J^{p_{k-1}}(\mathbf{x}, \widetilde{\mathbf{x}}_{k-1}) + \sum_{i=2}^{k} \left[\frac{1}{2}\|\widehat{\mathbf{x}}_1 - \widetilde{\mathbf{x}}_{i-1}\|^2 \right] - (k-1) \left[\frac{1}{2}\|\widehat{\mathbf{x}}_1 - \mathbf{x}\|^2 \right] \leq J(\mathbf{x}). \tag{4.67}$$

We can then substitute $\mathbf{x} = x = \widehat{\mathbf{x}}_1$ into (4.67), (since this is the value of $\mathbf{x}$ that will guarantee convergence) to obtain:

$$D_J^{p_{k-1}}(\widehat{\mathbf{x}}_1, \widetilde{\mathbf{x}}_{k-1}) + \sum_{i=2}^{k} \left[\frac{1}{2}\|\widehat{\mathbf{x}}_1 - \widetilde{\mathbf{x}}_{i-1}\|^2 \right] \leq J(\widehat{\mathbf{x}}_1) \tag{4.68}$$

Since the sequence $\frac{1}{2}\|\widehat{\mathbf{x}}_1 - \widetilde{\mathbf{x}}_{i-1}\|^2$ is monotonically non-increasing for $i = 2, 3, \ldots, k$ we can write:

$$D_J^{p_{k-1}}(\widehat{\mathbf{x}}_1, \widetilde{\mathbf{x}}_{k-1}) + (k-1) \left[\frac{1}{2}\|\widehat{\mathbf{x}}_1 - \widetilde{\mathbf{x}}_{k-1}\|^2 \right] \leq J(\widehat{\mathbf{x}}_1). \tag{4.69}$$

Equation (4.69) can be rearranged as:

$$\frac{1}{2}\|\widehat{\mathbf{x}}_1 - \widetilde{\mathbf{x}}_{k-1}\|^2 \leq \frac{J(\widehat{\mathbf{x}}_1) - D_J^{p_{k-1}}(\widehat{\mathbf{x}}_1, \widetilde{\mathbf{x}}_{k-1})}{k-1} \leq \frac{J(\widehat{\mathbf{x}}_1)}{k-1} \tag{4.70}$$

This implies that $\lim_{k\to\infty} \widetilde{\mathbf{x}}_{k-1} = \widehat{\mathbf{x}}_1$. Recalling that

$$\begin{aligned} \widehat{\mathbf{x}}_1 &= \arg\min_{\mathbf{x}} \left\{ \frac{1}{2}\|\mathbf{y} - \mathbf{x}\|^2 + J(\mathbf{x}) \right\} \\ &= \mathcal{B}(\mathbf{y}) \end{aligned} \tag{4.71}$$

and that

$$\begin{aligned} \widetilde{\mathbf{x}}_{k-1} &= \arg\min_{\mathbf{x}} \left\{ \frac{1}{2}\|\widehat{\mathbf{x}}_{k-1} - \mathbf{x}\|^2 + J(\mathbf{x}) \right\} \\ &= \mathcal{B}(\widehat{\mathbf{x}}_k), \end{aligned} \tag{4.72}$$

this further implies that

$$\lim_{k\to\infty} \widehat{\mathbf{x}}_k = \mathbf{y} \tag{4.73}$$

if, as discussed earlier, $\mathcal{B}(\cdot)$ minimizes a convex cost function.

This also conveniently gives us a rate of convergence.

$$\|\widehat{\mathbf{x}}_1 - \widetilde{\mathbf{x}}_{k-1}\|^2 \leq \frac{2J(\widehat{\mathbf{x}}_1)}{k-1} = \mathcal{O}\left((k)^{-1}\right) \tag{4.74}$$

□

We verify this rate of convergence by plotting $\|\widehat{\mathbf{x}}_1 - \widetilde{\mathbf{x}}_{k-1}\|^2$ vs k in Figure (**4.4**). To generate this curve, $\widehat{\mathbf{x}}_k$ was calculated from the image in Figure (**4.5** b) using IUR. Because we don't compute $\widetilde{\mathbf{x}}_k$ until the second iteration, our plot begins at $k = 2$. We note that $\|\widehat{\mathbf{x}}_1 - \widetilde{\mathbf{x}}_{k-1}\|^2$ does in fact approach 0 as $k \to \infty$. The rate at which it approaches 0 appears to be on the order of k^{-6}, which indicates that (4.74) may not be a tight bound.

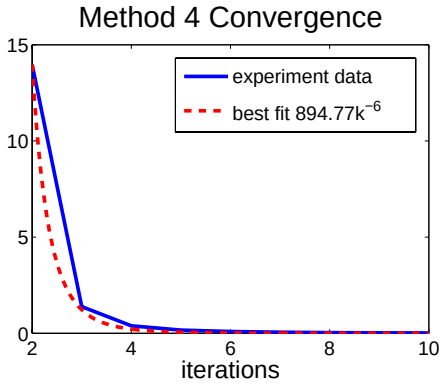


Figure 4.4: Plot of $\|\widehat{\mathbf{x}}_1 - \widetilde{\mathbf{x}}_{k-1}\|^2$ vs k verifying the convergence properties of method 4.

As a side note, recall that the general image restoration formulation of IUR is:

$$\widehat{\mathbf{x}}_{k+1} = \widehat{\mathbf{x}}_k + \mathbf{A}^T\widehat{\mathbf{x}}_1 - \mathbf{A}^T\mathbf{A}\mathcal{B}\left(\widehat{\mathbf{x}}_k\right). \tag{4.75}$$

Interestingly, if we assume that this generalized version of Method 4 converges, then it's

final value is a least-squares solution. That is, as $k \to \infty$, $\widehat{\mathbf{x}}_{k+1} = \widehat{\mathbf{x}}_k = \widehat{\mathbf{x}}_\infty$, and we have:

$$\begin{aligned} \mathbf{A}^T\widehat{\mathbf{x}}_1 - \mathbf{A}^T\mathbf{A}\mathcal{B}\left(\widehat{\mathbf{x}}_\infty\right) &= 0 \\ \mathbf{A}^T\mathbf{A}\mathcal{B}\left(\widehat{\mathbf{x}}_\infty\right) &= \mathbf{A}^T\mathcal{B}(\mathbf{y}) \\ \mathcal{B}\left(\widehat{\mathbf{x}}_\infty\right) &= (\mathbf{A}^T\mathbf{A})^{-1}\mathbf{A}^T\mathcal{B}(\mathbf{y}) \end{aligned} \tag{4.76}$$

4.5 Bias-Variance Tradeoff

To measure the effectiveness of the iterative regularization algorithms in recovering the true signal $\mathbf{x}$ from the data $\mathbf{y}$, the mean-squared error (MSE) is a natural choice. The MSE is defined as

$$mse(\widehat{\theta}) = E\left[(\widehat{\theta} - \theta)^2\right] \tag{4.77}$$

where $\widehat{\theta}$ is the estimate and θ is the underlying signal. We can rewrite the MSE as

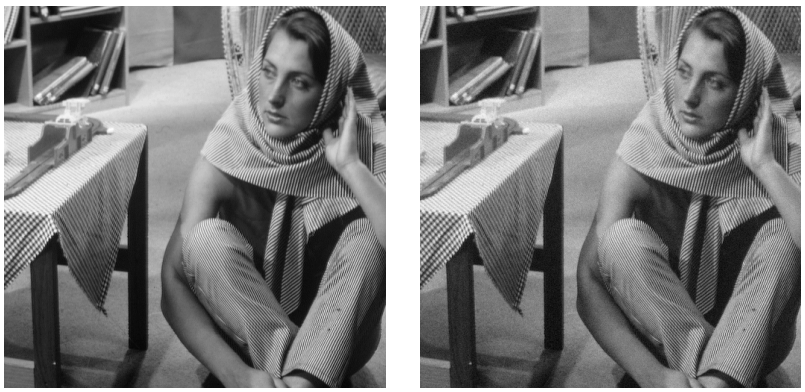
$$mse(\widehat{\theta}) = E\left(\left[\left(\widehat{\theta} - E(\widehat{\theta})\right) + \left(E(\widehat{\theta}) - \theta\right)\right]^2\right)$$

$$\begin{aligned} &= E\left[\left(\widehat{\theta} - E(\widehat{\theta})\right)^2\right] + 2\,E\left[\left(\widehat{\theta} - E(\widehat{\theta})\right)\left(E(\widehat{\theta}) - \theta\right)\right] \\ &\quad + E\left[\left(E(\widehat{\theta}) - \theta\right)^2\right] \\ &= var(\widehat{\theta}) + 0 + \left(E(\widehat{\theta}) - \theta\right)^2 = var(\widehat{\theta}) + bias^2(\theta). \end{aligned}$$

Thus, as is well-known, MSE is the sum of the estimate variance and squared-bias [14].

Ref. [13] provides a bias-variance tradeoff analysis for L_2 boosting (which is equivalent to the ITR method), however, some of the key assumptions made in that analysis do not apply to the iterative regularization methods that we present here (namely, in our general analysis $\mathcal{B}(\cdot)$ is a *non-linear* estimator). Because of this nonlinearity, the statistics of

the estimate $\widehat{\mathbf{x}}_k$ are difficult to compute analytically. We will show the bias-variance trade-off properties of the four iterative regularization methods experimentally using Monte-Carlo simulations.



(a) **(b)** MSE=29.54

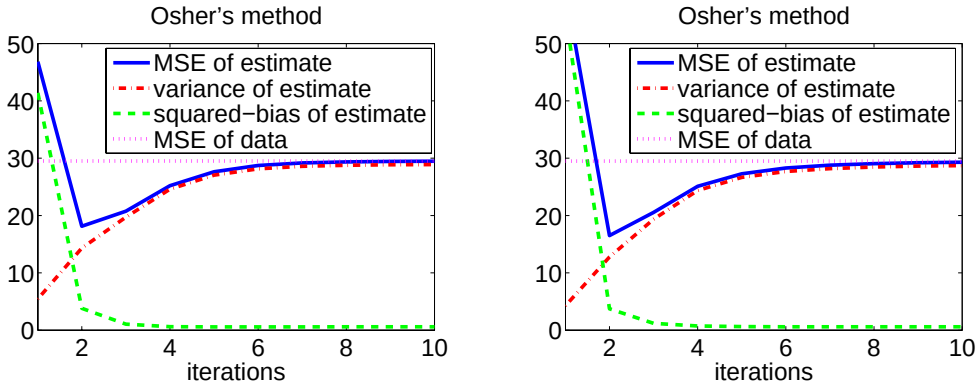
Figure 4.5: **(a)** The original 'Barbara' image **(b)** 'Barbara' with added while Gaussian noise of variance 29.5 (PSNR= 33.43dB)

For these Monte-Carlo simulations, we add 1 realization of random (Gaussian) noise with a variance of 29.5 to the image in Figure (**4.5** a) resulting in 1 noisy image. A series of estimates ($\widehat{\mathbf{x}}_1$ through $\widehat{\mathbf{x}}_{10}$) is computed from this noisy image using one of the iterative regularization methods. This process is repeated for a total of 50 different noise realizations. Next, the average variance, bias, and MSE between each of the 50 versions of each of the estimates ($\widehat{\mathbf{x}}_1$ through $\widehat{\mathbf{x}}_{10}$)are calculated. The entire simulation is then repeated for a different iterative regularization method.

In Figure (**4.6**) we have plotted the average MSE, variance, and squared-bias of Osher's iterative regularization method as a function of iteration number. Two different regularization functionals (Total Variation and Bilateral Filter) were used for comparison. The operating parameters for each of the regularization functionals were selected to yield the overall lowest MSE. For the Bilateral Filter regularization we used kernel size $N = 2$

(5), spatial blur parameter $\sigma_d = 1.1$, and radiometric blur parameter $\sigma_r = 35$. For the Total Variation Regularization we used regularization parameter $\lambda = .18$ and 50 steepest descent iterations. For details on the implementation of either of these regularization methods, please see [10] and [6] respectively.

Notice, in Figure (**4.6**), that the squared-bias decreases as we iterate but the variance increases. The mean-square error optimal estimate occurs where the sum of these two values is at a minimum, which occurs at the second iteration of Osher's method in this case.



(a) Bilateral **(b)** Total Variation

Figure 4.6: Average MSE, variance, and squared-bias of the estimates $\widehat{\mathbf{x}}_k$ of the noisy versions of the image shown in Figure (**4.5** a), (Figure (**4.5** b) is an example of one such noisy version) using Osher's iterative regularization method (with (a) Bilateral and (b) Total Variation regularization functionals).

Interestingly, the best overall MSE estimate is arrived at by applying a large amount of regularization such that the first estimate has a low variance (due to the removal of noise) but a high bias (due to the loss of high frequency image details). The bias decreases as we iterate and more of the "lost detail" is returned to the estimate. Some noise is also returned along with the detail causing the variance to increase. The MSE, being the sum of these two statistics is optimal at the point in the iterative process where we get

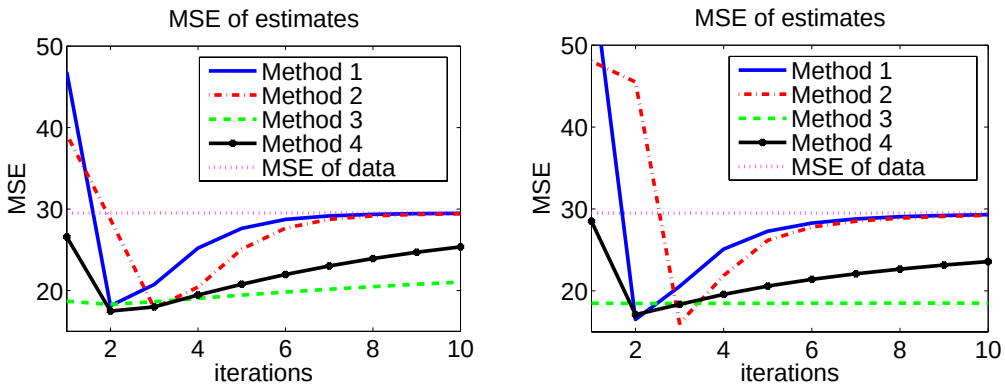
	Bilateral Filter Parameters	**Total Variation Parameters**
Method 1	$N = 2, \sigma_d = 1.1, \sigma_r = 35$	$\lambda = .18, 50$ steepest descent iterations
Method 2	$N = 2, \sigma_d = 1.1, \sigma_r = 31$	$\lambda = .21, 50$ steepest descent iterations
Method 3	$N = 2, \sigma_d = 1.1, \sigma_r = 14$	$\lambda = .7, 50$ steepest descent iterations
Method 4	$N = 2, \sigma_d = 1.1, \sigma_r = 23$	$\lambda = .32, 50$ steepest descent iterations

Table 4.1: Regularization operating parameters used in Figures (**4.6**),(**4.7**),(**4.9**), and (**4.8**).

the best tradeoff between restored texture and suppressed noise. Had we chosen too small an amount of regularization, the variance of the first estimate would have been too high to yield much of an improvement in MSE.

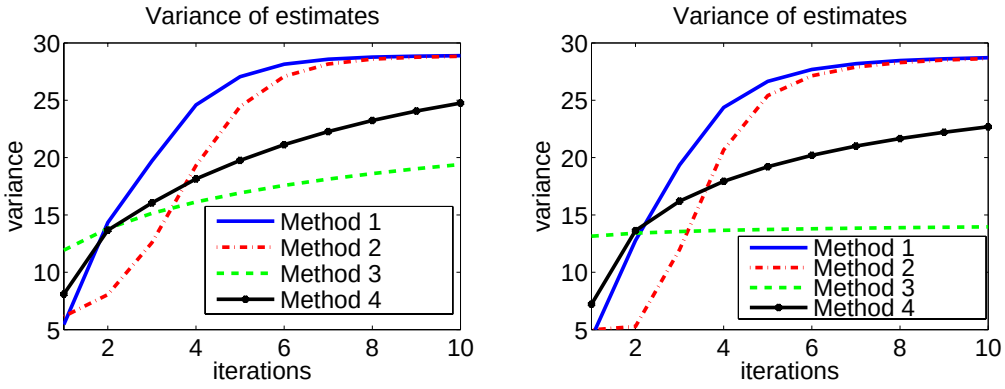
We compare the average MSE, squared-bias, and variance of the estimated produced by all four iterative regularization methods in Figures (**4.7**),(**4.9**), and (**4.8**) respectively. Again, both Bilateral Filter and Total Variation regularization are represented for comparison (see Table **4.1** for the operating parameters used). The general behavior of the variance, squared-bias, and MSE of the estimates produced by each of the methods appears to be very similar, though the MSE of ITR's estimates appears to converge much more slowly than the other methods.

Note that the minimum MSE estimate produced by each of the four iterative regularization methods (using a given regularization functional) have approximately the same MSE. This allows us to compare how each of the methods trades off variance for bias to produce the optimal MSE estimate. Methods 1,3,and 4 have minimum MSE at $k = 2$, while method 2 has its minimum MSE at $k = 3$. By looking at the variance of methods 1,3, and 4 at $k = 2$ and the variance of method 2 at $k = 3$ we can see that all four of the iterative regularization methods actually trade off the bias and variance of the estimates in the same way. That is to say that the optimal MSE estimate of produced by each of the methods has the same proportion of variance and squared-bias.



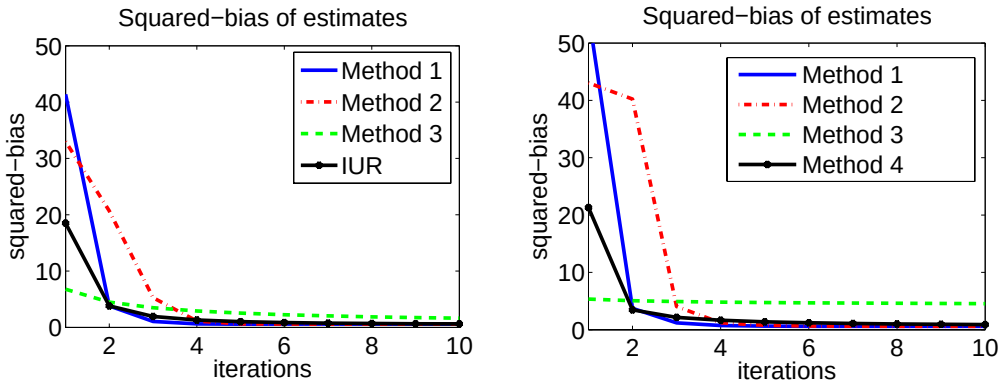
(a) Bilateral **(b)** Total Variation

Figure 4.7: Average MSE of the estimates $\widehat{\mathbf{x}}_k$ of the noisy versions of the image shown in Figure (**4.5** a), (Figure (**4.5** b) is an example of one such noisy version) using iterative regularization methods 1-4 (with (a) Bilateral and (b) Total Variation regularization functionals).



(a) Bilateral **(b)** Total Variation

Figure 4.8: Average variance of the estimates $\widehat{\mathbf{x}}_k$ of the noisy versions of the image shown in Figure (**4.5** a), (Figure (**4.5** b) is an example of one such noisy version) using iterative regularization methods 1-4 (with (a) Bilateral and (b) Total Variation regularization functionals).



(a) Bilateral **(b)** Total Variation

Figure 4.9: Average squared-bias of the estimates $\widehat{\mathbf{x}}_k$ of the noisy versions of the image shown in Figure (**4.5** a), (Figure (**4.5** b) is an example of one such noisy version) using iterative regularization methods 1-4 (with (a) Bilateral and (b) Total Variation regularization functionals).

Chapter 5

Applications Beyond Image Reconstruction

Iterative regularization methods take an initially over-regularized image and slowly add the texture that was removed by the initial regularization back to the estimate in a controlled manner. As we have seen in the previous chapters, for simple denoising problems, there is an optimal tradeoff between the amount of noise removed and the amount of texture retained at which the iterations would be stopped. Another interpretation of this is that the optimal tradeoff between bias and variance results in the smallest MSE (recall that we investigated the bias-variance tradeoff properties of the estimates produced by our iterative regularization framework in Chapter 4). Continuing to iterate using the iterative regularization framework causes the estimate to asymptotically converge back to the initial data. Essentially, the iterative regularization framework allows us to invert the highly non-linear initial regularization. However, IUR is the only method in the framework that does not require direct knowledge of $\mathbf{y}$ in order to compute it's estimates, but rather requires

only $\widehat{\mathbf{x}}_1$, the filtered version of $\mathbf{y}$. When used in this modality we refer to this iterative regularization method as " Inverse Regularization."

Osher's method was originally motivated in [1] by the desire to reduce the piecewise " staircase" effect that Total Variation can sometimes produce in its estimates. The other methods in our iterative regularization framework were similarly motivated by the desire to reduce the bias in the estimate produced by any general denoising method. In addition to the image restoration capabilities that initially motivated this framework, there are a wide variety of applications where inverse regularization is highly useful. We shall explore these applications in this chapter.

5.1 Compression

As movie theaters switch over to digital projectors, the method of transporting movies is changing as well. Instead of physically moving large reels of film around, movies are now being transmitted to digital theaters via satellite. In order to speed up transmission of these high definition movies, some form of compression is often used. To facilitate compression, some high frequency texture, including grain noise, may be removed in this process. It would be highly desirable to be able to reproduce the grain noise in the received movie frames and thus have the benefits of increased transmission rates and the desired appearance of grain noise in the movies.

Total Variation Regularization has been discussed as a method of decomposing an image into two parts: high frequency texture and piecewise constant cartoon ([6], [1], [15], [16]). To a lesser degree, there has been a similar analysis of the Bilateral Filter ([10], [7]).

Both techniques were originally intended to provide an estimate of the underlying

Denosing Technique	$J(\mathbf{x})$
Total Variation [1], [6]	$\lambda \|\|\|\nabla \mathbf{x}\|\|\|_1$
Bilateral [7], [8]	$\frac{\lambda}{2} \sum_{n=-N}^{N} \left[\mathbf{x} - \mathbf{S}^n \mathbf{x}\right]^T \mathbf{W}_{\mathbf{y},n} \left[\mathbf{x} - \mathbf{S}^n \mathbf{x}\right]$

Table 5.1: Total Variation and Bilateral Filter Regularization methods and their associated regularization terms.

signal in the classic denoising problem:

$$\mathbf{y} = \mathbf{x} + \mathbf{v}. \tag{5.1}$$

where $\mathbf{x}$ is the true image and $\mathbf{v}$ is zero-mean additive white noise that is uncorrelated to $\mathbf{x}$ and with no assumptions made on its distribution. The image decomposition property of these techniques were a later discovery.

Recall that both Total Variation Regularization and the Bilateral Filter can be implemented in the regularization framework as:

$$\widehat{\mathbf{x}} = \arg\min_{\mathbf{x}} \left\{ \frac{1}{2} \|\mathbf{y} - \mathbf{x}\|^2 + J(\mathbf{x}) \right\} \tag{5.2}$$

where $J(\mathbf{x})$ is a convex regularization functional as summarized in Table **5.1**. The parameter λ controls the amount of regularization. The signal $\widehat{\mathbf{x}}$ is an estimate of the the true signal $\mathbf{x}$ produced by minimizing the cost function associated with either Total Variation Regularization or Bilateral Filter Regularization (as we shall call the regularization interpretation of the Bilateral Filter).

For the regularization term corresponding to the Bilateral Filter, $\mathbf{S}^n$ is a matrix shift operator and $\mathbf{W}_{\mathbf{y},n}$ is a weight matrix where the weights are a function of both the radiometric and spatial distances between pixels in a local neighborhood.

Inverse regularization is perfectly suited for providing bandwidth savings in compression applications. The " cartoon" version of the signal $\widehat{\mathbf{x}}_1$ is generated by simply applying Total Variation or the Bilateral Filter to the signal $\mathbf{y}$ which may contain high frequency texture that is desirable to keep but difficult to compress. The " cartoon" is much easier

Parameters	**Bandwidth (bits/pixel)**	**% reduction**	**PSNR (dB)**
$\lambda = .2$, $k = 400$	7.49	6.94	35.67
$\lambda = .4$, $k = 46$	7.83	2.73	43.01
$\lambda = .5$, $k = 20$	7.94	1.31	45.95
$\lambda = .6$, $k = 11$	8.01	0.47	48.43

Table 5.2: Lossless (packbits) compression results using IUR with Total Variation regularization.

to compress in a lossless compression scheme such as 'run-length coding' ([17]) due to its piecewise-constant nature.

By transmitting just the " cartoon" and the regularization parameters used to generate it, we can exactly reconstruct the original signal $\mathbf{y}$ by using (IUR) inverse regularization at the receiver (iterating until the estimate $\widehat{\mathbf{x}}_k$ converges). Table **5.2** shows the results from using 'packbits' compression (a variant of run-length coding) on the JFK image shown in Figure **5.1** (a). The digital Total Variation Filter ([6]) was used to implement Total Variation Regularization. Note that as the amount of regularization is increased (making the transmitted image $\widehat{\mathbf{x}}_1$ more piecewise constant), the bandwidth is decreased but at a cost of increased number of iterations at the receiver. For these results the image $\widehat{\mathbf{x}}_k$ was considered to have converged when its mean-squared error (MSE) was less than or equal to the arbitrary value 0.085.

It is well known that the result with the best PSNR is not necessarily the best visual quality result. Furthermore, for the applications that we are considering, visual results are all that we care about. We are simply trying to reduce the amount of bandwidth necessary to transmit a grainy image and have the received image appear to be just as grainy as the original image. For this reason, lossy compression schemes such as JPEG may be used as well. However, they are not as well suited for compressing images containing texture since the artifacts introduced into the received version of the "cartoon" can change the appearance of the received data. These artifacts can then be amplified in the iterative process at the

decoder.

5.1.1 Approximate Bilateral Filter Version of Inverse Regularization

We now introduce an alternate method of creating a visually similar result using the bilateral filter to generate $\widehat{\mathbf{x}}_1$, the "cartoon" version of the data at the transmitter.

We can use just one iteration of IUR to return a majority of the lost detail back to the "cartoon" $\widehat{\mathbf{x}}_1$. Then we can add zero-mean white noise with approximately the same statistics as the grainy noise in $\mathbf{y}$ to the estimate $\widehat{\mathbf{x}}_2$. The resulting image which we shall denote as $\widehat{\mathbf{y}}$ will closely match the first and second order statistics of $\mathbf{y}$ and thus be visually similar to the desired output $\mathbf{y}$. Earlier work in film grain synthesis can be found in [18], [19], and [20]. Here we consider an entirely different approach.

The Bilateral Filter is a locally adaptive filter which, in the $1-$D case, can be written as

$$\widehat{\mathbf{x}}_1(j) = \sum_{n=-N}^{N} \mathbf{w}(j,n)\mathbf{y}(j-n) \tag{5.3}$$

or more concisely as $\widehat{\mathbf{x}}_1 = \mathcal{B}(\mathbf{y}, N)$, where the integer N defines a $2N+1$ symmetric window about any given value of $\mathbf{y}(j)$ in the signal $\mathbf{y}$ over which the filter operates. The locally adaptive weights $\mathbf{w}(j,n)$ are related to both spatial and radiometric (e.g. pixel intensity) distance between the value $\mathbf{y}(j)$ and its N pixels to each side. The details of how these coefficients are chosen can be found in [10].

The error image $\mathbf{e} = \mathbf{y} - \widehat{\mathbf{x}}_1$ contains the lost texture that we wish to replicate at the receiver in our IUR inverse regularization scheme. Let us now study the statistics of this error image.

We calculate the point-wise expected value of $\mathbf{e}$ as follows:

$$\begin{aligned} E\left[\mathbf{e}(j)\right] &= E\left[\mathbf{y}(j) - \widehat{\mathbf{x}}_1(j)\right] \\ &= E\left[\mathbf{x}(j) + \mathbf{v}(j) - \widehat{\mathbf{x}}_1(j)\right] \\ &= \mathbf{x}(j) - E\left[\widehat{\mathbf{x}}_1(j)\right]. \end{aligned} \tag{5.4}$$

We note that the lost detail restored by a single iteration of IUR (e.g $\widehat{\mathbf{x}}_1 - \mathcal{B}(\widehat{\mathbf{x}}_1)$) will have approximately the same expected value as the error $\mathbf{e}$ under the condition that

$$\mathbf{x} \approx \widehat{\mathbf{x}}_1 \tag{5.5}$$

which is true in general when the noise variance is small (remember that the Bilateral Filter was designed to estimate the true signal $\mathbf{x}$). Thus $\widehat{\mathbf{x}}_2 = \widehat{\mathbf{x}}_1 + (\widehat{\mathbf{x}}_1 - \mathcal{B}(\widehat{\mathbf{x}}_1))$ matches (approximately) the expected value of $\mathbf{y}$ at each pixel.

We can now compute the point-wise variance of $\mathbf{e}$ as:

$$\begin{aligned} Var\left[\mathbf{e}(j)\right] &= E\left[\left(\mathbf{e}(j) - E\left[\mathbf{e}(j)\right]\right)^2\right] \\ &= E\left[\left(\mathbf{y}(j) - \sum_{n=-N}^{N} \mathbf{w}(j,n)\mathbf{y}(j-n) - E\left[\mathbf{y}(j) - \sum_{n=-N}^{N} \mathbf{w}(j,n)\mathbf{y}(j-n)\right]\right)^2\right]. \end{aligned}$$

If we naively assume $\mathcal{B}(\mathbf{y}) \approx \mathcal{B}(\mathbf{x}) + \mathcal{B}(\mathbf{v})$ and that $E\left[\mathcal{B}(\mathbf{v})\right] \approx 0$, then we have:

$$\begin{aligned} Var\left[\mathbf{e}(j)\right] &\approx E\left[\left(\mathbf{y}(j) - \sum_{n=-N}^{N} \mathbf{w}(j,n)\mathbf{y}(j-n) - \mathbf{x}(j) + \sum_{n=-N}^{N} \mathbf{w}(j,n)\mathbf{x}(j-n)\right)^2\right] \\ &= E\left[\left(\mathbf{v}(j) - \sum_{n=-N}^{N} \mathbf{w}(j,n)\mathbf{v}(j-n)\right)^2\right] \\ &= E\left[\mathbf{v}^2(j)\right] - 2E\left[\mathbf{v}(j) \sum_{n=-N}^{N} \mathbf{w}(j,n)\mathbf{v}(j-n)\right] + E\left[\left(\sum_{n=-N}^{N} \mathbf{w}(j,n)\mathbf{v}(j-n)\right)^2\right]. \end{aligned} \tag{5.6}$$

Let's look more closely at the second term in the above expression. Since we assuming that

our noise is uncorrelated and zero-mean, the expected value

$$E\left[\mathbf{v}(j)\mathbf{v}(j-n)\right] \tag{5.7}$$

where $n \neq 0$ will be zero, and the only non-zero term will be

$$\begin{aligned} E\left[\mathbf{v}(j)\mathbf{w}(j,0)\mathbf{v}(j-0)\right] &= E\left[\mathbf{w}(j,0)\mathbf{v}^2(j)\right] \\ &= \sigma^2\mathbf{w}(j,0) \end{aligned} \tag{5.8}$$

where σ^2 is the variance of the noise $\mathbf{v}$.

Similarly, in the third term of (5.6), the squared summation can be expanded, and all of the terms of the form

$$E\left[\mathbf{w}(j,n)\mathbf{v}(j-n)\mathbf{w}(j,m)\mathbf{v}(j-m)\right] \tag{5.9}$$

with $n \neq m$ will be zero. The remaining non-zero terms will be:

$$\begin{aligned} E\left[\sum_{n=-N}^{N}\mathbf{w}^2(j,n)\mathbf{v}^2(j,n)\right] &= \sum_{n=-N}^{N}\mathbf{w}^2(j,n)E\left[\mathbf{v}^2(j,n)\right] \\ &= \sigma^2\sum_{n=-N}^{N}\mathbf{w}^2(j,n). \end{aligned} \tag{5.10}$$

Now, putting together Equations (5.6), (5.8) and (5.10), we can write the approximate point-wise variance of $\mathbf{e}$ as:

$$Var\left[\mathbf{e}(j)\right] = \left(1 - 2\mathbf{w}(j,0) + \sum_{n=-N}^{N}\mathbf{w}^2(j,n)\right)\sigma^2 \tag{5.11}$$

From (5.11) and (5.4) we conclude that, in order for our estimate to match both the variance as well as the mean of $\mathbf{y}$, we must add zero-mean white noise with the same distribution and variance as the grainy noise in $\mathbf{y}$ to the estimate $\widehat{\mathbf{x}}_2$. Thus, if we define $\mathbf{w}_T \equiv \left(1 - 2\mathbf{w}(j,0) + \sum_{n=-N}^{N}\mathbf{w}^2(j,n)\right)$, we can estimate

$$\begin{aligned} \widehat{\mathbf{y}} &= \widehat{\mathbf{x}}_1 + (\widehat{\mathbf{x}}_1 - \mathcal{B}(\widehat{\mathbf{x}}_1)) + \left(1 - 2\mathbf{w}(j,0) + \sum_{n=-N}^{N}\mathbf{w}^2(j,n)\right)^{\frac{1}{2}}\sigma\,\epsilon(j) \\ &= \widehat{\mathbf{x}}_1 + (\widehat{\mathbf{x}}_1 - \mathcal{B}(\widehat{\mathbf{x}}_1)) + \mathbf{w}_T^{\frac{1}{2}}\,\sigma\,\epsilon(j) \end{aligned} \tag{5.12}$$

where ϵ is a zero-mean white noise process with the same distribution as $\mathbf{v}$ but with unit variance. The last term in the above expression (5.12) returns the stochastic component of $\mathbf{y}$ (a spatially varying variance), whereas the first two terms in effect " undo" (at least partially) the deterministic action of the Bilateral Filter on the true structure of the image.

Note we must transmit two images $\widehat{\mathbf{x}}_1$ and $\mathbf{w}_T$ (as well as the Bilateral Filter parameters and the variance of the grain noise σ^2) in order to create $\widehat{\mathbf{y}}$ at the receiver. However, $\mathbf{w}_T$ is highly sparse and therefore easy to compress, and, as mentioned earlier, the piecewise nature of $\widehat{\mathbf{x}}_1$ makes it easy to compress as well. As a result we get a visually similar result at the output of the receiver that used less bandwidth than a non-compressed transmission and required less work (iterations) to produce. An example of the estimates produced by this method are shown in Figure (**5.1**). The Bilateral Filter parameters used in this example were $N = 2$, $\sigma_d = 1.1$ and $\sigma_r = 44$, and the variance of the texture $\mathbf{v}$ was estimated to be $\sigma^2 \approx 55$. The compression scheme used was JPEG compression with a quality factor of 100 (which is still lossy compression) for the grainy image (Figure (**5.1** a)) and " cartoon" (Figure (**5.1** b)), and JPEG compression with a quality factor of 80 for the Bilateral Filter coefficient image (Figure (**5.1** c)). Note that we can tolerate a larger degree of data loss in the compressed version of $\mathbf{w}_T$ since its effect of the reconstructed image $\widehat{\mathbf{y}}$ is more subtle than that of the compressed version of $\widehat{\mathbf{x}}_1$. Also not that the pixels values of $\mathbf{w}_T$ are typically small and will likely be rounded to zero at the quantizing stage of whatever compression algorithm is being used, unless the dynamic range is scaled up. In this example we has scaled to the dynamic range of $\mathbf{w}_T$ to be in the range $[0, 255]$ before compression, then scaled it back to its original range at the receiver. This requires that the dynamic range scale also be transmitted, but since this is only two numbers, the effect on the total bandwidth is negligible.

The total number of bytes transmitted using the approximate Bilateral Filter version of inverse regularization (19620+2962=22582 bytes) is less than the number of bytes (28871 bytes) required if we were to simply compress the the grainy image.

(a) 28871 bytes **(b)** 19620 bytes **(c)** 2962 bytes

(d) **(e)**

Figure 5.1: **(a)** The original grainy image $\mathbf{y}$, compressed using JPEG with a quality factor of 100. **(b)** The Bilateral Filtered image $\widehat{\mathbf{x}}_1 = \mathcal{B}(\mathbf{y})$ compressed using JPEG with a quality factor of 100. **(c)** The compressed Bilateral Filter coefficient image $\mathbf{w}_T$, shown here with gray levels reversed for ease of viewing, compressed using JPEG with a quality factor of 80. **(d)** The result of using one iteration of IUR on (b). **(d)** The resulting image $\widehat{\mathbf{y}}$ with reconstructed grain texture.

5.2 Texture Transfer

The ability to transfer the texture from one image to another finds uses in a variety of areas. For artistic effect, it may be desirable to add grainy texture to images captured on digital media. Composite images combine elements from different photographs into one; or sometimes a computer generated image is combined with a real one. It is often desirable that

the texture of the composite image be consistent. Also, high frequency textures such as hair or sand can be difficult or labor intensive to implement in computer generated images. In some instances it would be advantageous to simply extract the texture from a real image and transfer it to the computer generated image. Commercial products such as 'Grain Surgery' (*http://www.visinf.com/gs/ps/*) exist solely to accomplish these types of tasks. Our method offers an alternative to these products.

The simplest way to accomplish the texture transfer is to apply either Total Variation or Bilateral Filter Regularization to the texture source $\mathbf{y}$ resulting in the piecewise-constant image $\widehat{\mathbf{x}}_1$. The texture information $(\mathbf{y} - \widehat{\mathbf{x}}_1)$ can then be added to the textureless image $\mathbf{z}$. We can express this process as:

$$\widehat{\mathbf{z}} = \mathbf{z} + \mathcal{N}(\mathbf{y} - \widehat{\mathbf{x}}_1), \tag{5.13}$$

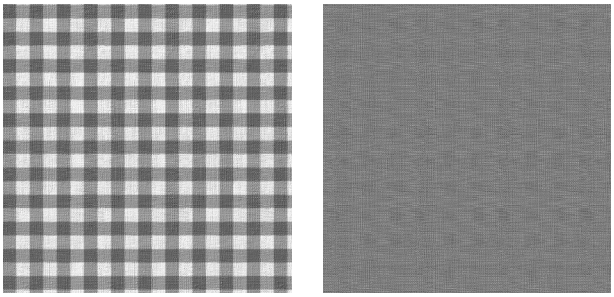
where $\mathcal{N}(\cdot)$ is a normalization of the image gray values such that the texture and target images are in the same dynamic range. However, there is an advantage to using IUR to transfer the texture; namely more control. By transferring texture from image $\mathbf{y}$ to image $\mathbf{z}$ as:

$$\widehat{\mathbf{z}}_k = \mathbf{z} + \mathcal{N}\left(\sum_{i=1}^{k}(\widehat{\mathbf{x}}_i - \widetilde{\mathbf{x}}_i)\right) \tag{5.14}$$

we can control the amount of texture that is transferred, easily optimizing the appearance of the resulting image.

In Figure (**5.3**) we show an example of this technique. We use Bilateral Filter regularization with kernel size $N = 2$, spatial parameter $\sigma_d = 1.1$, and radiometric parameter $\sigma_r = 50$ to iteratively add the texture from the source image (Figure(**5.2**)) to the target image in Figure (**5.3**) (a). A simple intensity threshold was used to mask the portion of the image that we did not want to add texture to. Any kind of image segmentation may be used

to achieve similar results. Notice how the transferred texture becomes more pronounced as the number of iterations increases. The image in Figure (**5.3**) (e) looks very similar to the image in Figure (**5.3**) (f) because the IUR process has stabilized.



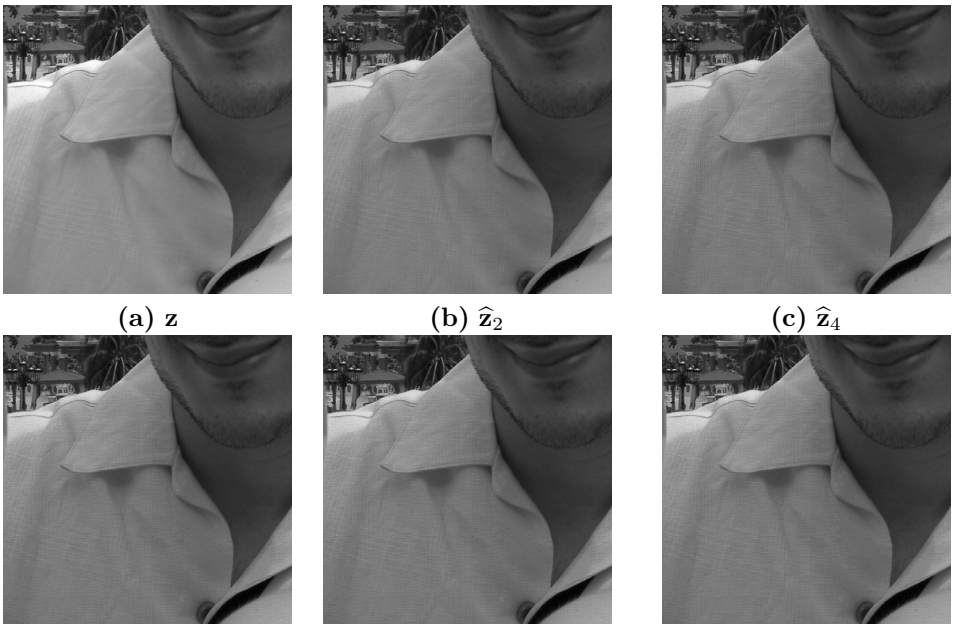
(a) $\mathbf{y}$ **(b)** $(\mathbf{y} - \widehat{\mathbf{x}}_1)$

Figure 5.2: **(a)** The texture source. **(b)** The texture extracted from (a) by applying Bilateral Filter Regularization with $N = 2$, $\sigma_d = 1.1$, $\sigma_r = 50$.

5.3 Conclusions

In Chapter 4 we were able to show that the estimates produced by each of the iterative regularization methods in our framework eventually converge back to the initial data. One of the the methods in particular, IUR, is unique in the fact that the iterations do not require knowledge of the initial data. Thus, it can be thought of as inverse regularization; undoing the effects of regularization through iterations.

We have demonstrated the bandwidth savings that are possible by regularizing an image before transmission and using IUR at the receiver to restore the original image. We have also illustrated an approximate version of inverse regularization using the bilateral filter that is very useful for restoring grain texture to images using only a single iteration at the receiver. Finally, we showed an example of the controlled manner by which inverse regularization can be used to transfer texture from a texture source to texture-less image.



(a) $\mathbf{z}$ **(b)** $\widehat{\mathbf{z}}_2$ **(c)** $\widehat{\mathbf{z}}_4$

(d) $\widehat{\mathbf{z}}_6$ **(e)** $\widehat{\mathbf{z}}_{10}$ **(f)** $\mathbf{z} + \mathcal{N}(\mathbf{y} - \widehat{\mathbf{x}}_1)$

Figure 5.3: **(a)** The original image without any texture added. **(b)** The result of one IUR iteration (with Bilateral Filter Regularization) to transfer the texture of the image in Figure **5.2** (a) to the image in (a). **(c)** The result of three IUR iterations. **(d)** The result of five IUR iterations. **(e)** The result of nine IUR iterations. **(f)** The result of adding the full texture of Figure **5.2** (b) to (a).

In addition to the many applications for image restoration for which our iterative regularization framework was initially motivated, these other applications further exemplify the utility of our framework.

Chapter 6

Image Reconstruction Experimental Results

While we show experiment results throughout this thesis, we further expand upon them in this chapter to illustrate a number of key points.

In Chapters 2 and 3 we discussed our iterative regularization framework in the context of image restoration: namely, deblurring and denoising. In this chapter, we illustrate this thoroughly with extensive examples. We use several different regularization methods to help illustrate the generality of the iterative regularization framework. These methods are briefly described below.

The Bilateral Filter ([10]) is a spatially adaptive filter that replaces each pixel of the noisy image with a weighted average of the pixels in some surrounding neighborhood. These weights depend on both the spatial distance (far away pixels contribute less to the estimate) and the radiometric (gray value) distance (pixels with a very different gray level contribute less to the estimate). The Bilateral Filter requires some control parameters to

compute its estimate. These parameters are: N, which determines the $(2N+1) \times (2N+1)$ window over which the weighted averages are computed; σ_d, which controls how strongly the spatial distances affect the weights; and σ_r, which controls how strongly the radiometric distances affect the weights (see Appendix B for more details).

Total Variation Regularization ([9]) may be implemented as a steepest descent problem, as described in Appendix C. Based on this implementation, Total Variation regularization requires the following control parameters: λ, which controls the amount of regularization; γ, the gradient descent step size; and i, the number of gradient descent steps.

Bilateral Total Variation regularization was formulated in [8] as an extension to Total Variation regularization based upon the Bilateral Filter idea of using both spatial and radiometric information to calculate the estimate. It is formulated as:

$$\widehat{\mathbf{x}} = \arg\min_{\mathbf{x}} \left\{ \frac{1}{2}\|\mathbf{y} - \mathbf{x}\|^2 + \lambda \sum_{n=-N}^{N} \alpha^{|n|} \|\mathbf{x} - \mathbf{S}^n \mathbf{x}\|_1 \right\} \tag{6.1}$$

where $0 \leq \alpha \leq 1$ to reflect the decreased confidence in pixels far away from the one being estimated. Much like the Bilateral Filter, this can be interpreted as a spatially adaptive filter where each pixel of the noisy image is replaced with the weighted average of the pixels in a local neighborhood. The weights are dependant on spatial and radiometric distances. However, unlike the Bilateral Filter, the distance metric used is the L_1 norm, not the L_2 norm. This is solved via gradient descent as:

$$\widehat{\mathbf{x}}_{i+1} = \widehat{\mathbf{x}}_i - \gamma \left[\widehat{\mathbf{x}}_i - \mathbf{y} + \lambda \left(\sum_{n=-N}^{N} \alpha^{|n|} [\mathbf{I} - \mathbf{S}^n] \, sign[\widehat{\mathbf{x}}_i - \mathbf{S}^n \widehat{\mathbf{x}}_i] \right) \right]. \tag{6.2}$$

Based on this implementation, Bilateral Total Variation regularization requires the following control parameters: N, which determines the $(2N+1) \times (2N+1)$ window over which the weighted averages are computed; α, which controls how strongly the spatial distances affect the weights; λ, which controls the amount of regularization; γ, the gradient descent step

size; and i, the number of gradient descent steps.

Tikhonov regularization ([5]) is perhaps the most widely used regularization method. It is formulated as:

$$\widehat{\mathbf{x}} = \arg\min_{\mathbf{x}} \left\{ \frac{1}{2}\|\mathbf{y}-\mathbf{x}\|^2 + \frac{\lambda}{2}\|\Gamma\mathbf{x}\|^2 \right\}. \tag{6.3}$$

where Γ is usually a highpass operator such as the Laplacian. For our examples we in fact use the Laplacian operator version. We use gradient descent to carry out the cost function minimization as follows:

$$\widehat{\mathbf{x}}_{i+1} = \widehat{\mathbf{x}}_i - \gamma \left[\widehat{\mathbf{x}}_i - \mathbf{y} + \lambda\Gamma^T\Gamma\widehat{\mathbf{x}}_i \right]. \tag{6.4}$$

Based on this implementation, Tikhonov regularization requires the following control parameters: λ, which controls the amount of regularization, γ, the gradient descent step size, and i, the number of gradient descent steps.

Classic kernel regression ([21], [22]) is rather well known and frequently used in the machine learning community. However, its use in the image processing field for image estimation has been limited (though it receives a very thorough treatment in ([23]). The goal of classic kernel regression is to consider the image pixels as samples of an unknown regression function for which we wish to estimate the Q-th order Taylor series approximation coefficients. In other words, the denoising problem model for classic kernel regression is:

$$\mathbf{y} = z(\mathbf{x}) + \mathbf{v} \tag{6.5}$$

where $z(\mathbf{x}) \approx (\beta_0 - \beta_1\mathbf{x} + \beta_2\mathbf{x}^2 + ... + \beta_Q\mathbf{x}^Q)$ and $\mathbf{v}$ is zero-mean additive white noise that is uncorrelated to $\mathbf{x}$. If we let $g\left(\mathbf{x}, \{\beta_i\}_{i=0}^{Q}\right) = \beta_0 - \beta_1\mathbf{x} + \beta_2\mathbf{x}^2 + ... + \beta_Q\mathbf{x}^Q$, we can formulate classic kernel regression as:

$$\widehat{\mathbf{x}} = \arg\min_{g\left(\mathbf{x},\{\beta_i\}_{i=0}^{Q}\right)} \left\{ \left[\mathbf{y} - g\left(\mathbf{x}, \{\beta_i\}_{i=0}^{Q}\right)\right]^T \mathbf{W}_h \left[\mathbf{y} - g\left(\mathbf{x}, \{\beta_i\}_{i=0}^{Q}\right)\right] \right\} \tag{6.6}$$

where $\mathbf{W}_h$ is a (space dependant) weight matrix. The weights are computed as a decaying function of spatial distance (pixels further from the pixel to be estimated will be weighted less than those closer to the pixel to be estimated). For our experiments we will use a Gaussian function for this purpose. The details of how this method is implemented can be found in ([23]). Based on this implementation, classic kernel regression has the following control parameters: h, which controls how strongly the spatial distances affect the weights; N, which determines the $(2N+1) \times (2N+1)$ window over which the weighted averages are computed; and Q, the regression order. We also note that Di Marzio and Taylor consider iterating kernel regression via boosting (which is closely related to IUR) in [24] and [25].

6.1 Experiment 1: Best MSE Method

As we have seen in Chapter 5, IUR clearly has an advantage over the other methods in the sense that it can be used for inverse regularization (reconstruction of $\mathbf{y}$ from the regularized estimate $\widehat{\mathbf{x}}_1$) while the other methods cannot. However, we would like to investigate whether any of the methods is superior to the others in terms of producing the lowest MSE estimate. Therefore, we have applied all four methods to the same image. Additionally, we have used several regularization functionals for each method as well. Table **6.1** summarizes the comparisons made in this first experiment.

The image that we are trying to denoise is shown in Figure **6.1** (b). This image was generated by adding white Gaussian noise of variance $\sigma^2 = 29.5$ to the image shown in Figure **6.1** (a). We have produced each estimate for 20 different noise realizations in order to compute the average MSE in this and all remaining experiments in this chapter.

As we mentioned in the introduction of this chapter, all of the regularization methods that we compare here require control parameters to produce their estimates. In order to

	Method 1	**Method 2**	**Method 3**	**Method 4**
BF	Figure **6.6** (a)	Figure **6.6** (b)	Figure **6.6** (c)	*Figure **6.6** (d)*
TV	Figure **6.7** (a)	*Figure **6.7** (b)*	Figure **6.7** (c)	Figure **6.7** (d)
BTV	Figure **6.8** (a)	*Figure **6.8** (b)*	Figure **6.8** (c)	Figure **6.8** (d)
Tik	*Figure **6.9** (a)*	*Figure **6.9** (b)*	*Figure **6.9** (c)*	*Figure **6.9** (d)*
CKR	*Figure **6.10** (a)*	*Figure **6.10** (b)*	*Figure **6.10** (c)*	*Figure **6.10** (d)*

Table 6.1: The different denoising techniques performed on Figure **6.1** (b) in Experiment 1. The lowest MSE result in each row is italicized. BF stands for Bilateral Filter, TV for Total Variation regularization, BTV for Bilateral Total Variation regularization, Tik for Tikhonov regularization, and CKR for classic kernel regression.

(a) **(b)**

Figure 6.1: **(a)** The original 'Barbara' image **(b)** 'Barbara' with added white Gaussian noise of variance 29.5 (MSE= 29.50)

find the lowest MSE estimate, each of these parameters was tuned by hand. In many cases a "reasonable" initial estimate helps to reduce the amount of trial-and-error testing needed to tune a particular parameter. For instance, Total Variation Regularization, Bilateral Total Variation Regularization, and Tikhonov Regularization all use gradient descent to minimize their associated cost functions. It makes sense to choose the gradient descent step size to be " small" and the number of gradient descent steps to be" large" so that the gradient descent algorithm will converge to the minimum solution. The parameters, the number of iterations used for each method, and the estimates themselves are shown in Figures (**6.6**), (**6.7**), (**6.8**), (**6.9**), and (**6.10**). The best way to see the subtle differences between the methods is to look at the residual $|\mathbf{y} - \hat{\mathbf{x}}_k|$, thus these are shown as well. A residual that contains less structure

and looks more like pure noise is one indication of better denoising. We have reversed the gray levels of these residuals for easier viewing and scaled their dynamic ranges such that all of the residuals in a particular figure have the same dynamic range. We shall similarly scale the dynamic ranges and reverse the gray levels of all the residuals that we will show in subsequent experiments as well.

Recall that all of the methods in our iterative regularization framework are related by a linear distribution of the $\mathcal{B}(\cdot)$ operator:

$$
\begin{aligned}
1) \qquad \widehat{\mathbf{x}}_{k+1} &= \mathcal{B}\left(\mathbf{y} + \sum_{i=1}^{k}(\mathbf{y} - \widehat{\mathbf{x}}_i)\right), \\
2) \qquad \widehat{\mathbf{x}}_{k+1} &= \mathcal{B}(\mathbf{y}) + \mathcal{B}\left(\sum_{i=1}^{k}(\mathbf{y} - \widehat{\mathbf{x}}_i)\right), \\
3) \qquad \widehat{\mathbf{x}}_{k+1} &= \mathcal{B}(\mathbf{y}) + \sum_{i=1}^{k}\mathcal{B}(\mathbf{y} - \widehat{\mathbf{x}}_i), \text{ and} \\
4) \qquad \widehat{\mathbf{x}}_{k+1} &= \mathcal{B}(\mathbf{y}) + \sum_{i=1}^{k}(\mathcal{B}(\mathbf{y}) - \mathcal{B}(\widehat{\mathbf{x}}_i)) \\
&= (k+1)\mathcal{B}(\mathbf{y}) - \sum_{i=1}^{k}\mathcal{B}(\widehat{\mathbf{x}}_i).
\end{aligned}
$$

The net effect of both the Tikhonov Regularization and Classic Kernel Regression cost functions are that they produce linear operators ($\mathcal{B}$). Thus, all four iterative regularization methods are equivalent when Tikhonov Regularization or Classic Kernel Regression are used.

The best MSE estimate for each regularization type is shown in italics in Table (**6.1**). Method 2 produces the best estimate in the Total Variation and Bilateral Total Variation regularization cases, while method 4 produces the best estimate in the Bilateral Filter case, and all four methods perform the same in the Tikhonov regularization and classic kernel regression cases. In each case, the MSE values are an average of 20 Monte-Carlo experiments. Based on this experiment it seems that there is not one iterative regularization

method in particular that performs better than the others. The best method to use can depend on the type of regularization functional and on the image itself. Also note that the mean-squared error of the estimates for any particular choice of regularization functional are generally not very different from one another. Though one method may yield a slightly lower MSE estimate, the other three methods will produce an estimate of similar quality.

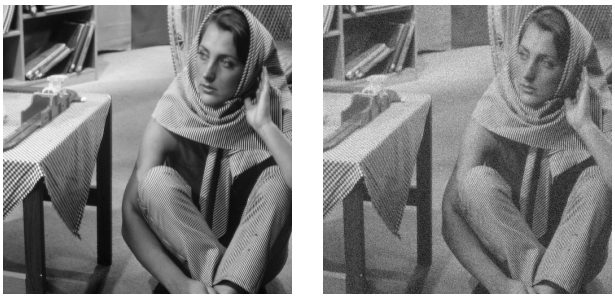
6.2 Experiment 2: Low SNR Denoising

The signal-to-noise ratio (SNR) is a good indicator of the amount of noise in an image. We define it here (in units of dB) as:

$$SNR(\mathbf{y}) = 10 \log_{10} \left(\frac{Var[\mathbf{x}]}{\sigma^2} \right) \tag{6.7}$$

where σ^2 is the variance of the noise, $\mathbf{v}$. The image used in Experiment 1 has an SNR of 20dB, which is reasonably high. Thus we expect to produce very good (low MSE) estimates. The task of estimating the true image from the noisy data becomes more difficult as the SNR is decreased. We now examine how the iterative regularization methods perform when the SNR is much lower than in experiment 1.

We will denoise two different images in this experiment. The first image is shown in Figure **6.2** (b). This image was generated by adding white Gaussian noise of variance $\sigma^2 = 298$ to the image shown in Figure **6.2** (a) yielding a SNR of 10dB. The second image is shown in Figure **6.3** (b), and was generated by adding white Gaussian noise of variance $\sigma^2 = 229$ to the image shown in Figure **6.3** (a) yielding a SNR of 10dB. We use our iterative regularization framework with Bilateral Total Variation regularization to find the best MSE estimates in each case. The operating parameters were tuned by hand and are shown, along with the number of iterations and the estimates themselves, in Figures (**6.11**) and (**6.12**).



(a) **(b)**

Figure 6.2: **(a)** The original 'Barbara' image **(b)** 'Barbara' with added white Gaussian noise of variance 298 (MSE= 298.0)

(a) **(b)**

Figure 6.3: **(a)** The original 'Lena' image **(b)** 'Lena' with added white Gaussian noise of variance 229 (MSE= 229.0)

Notice that, although ITR did not produce the best MSE estimate in experiment 1 for any regularization functional (except for Tikhonov regularization and classic kernel regression which produce the same estimate for all four methods), it does produce the best MSE estimate in this experiment. Once again, the MSE values are an average of 20 Monte-Carlo experiments. This suggests that ITR may be the best iterative regularization method to use when the noisy image has a low SNR. Also note that the best MSE estimates produced via ITR appear to have a higher variance than the estimates from the other three methods, suggesting that the ITR trades off a higher variance for a lower bias in the presence of more noise.

6.3 Experiment 3: Removing Grain Noise From a Color Image

In this experiment we will illustrate two facets of the versatility of our iterative regularization framework: it can be applied to images with various kinds of noise, and it can be applied to color images as well. The image that we are trying to denoise is shown in Figure (**6.4**). This is a color image and the high frequency texture throughout the image is film grain noise.

We used Total Variation regularization as our regularization functional in this experiment. Following [26], we transferred this **RGB** image into **YCrCb** representation in order to achieve a better quality color image estimate. We then hand tuned the Total Variation regularization operating parameters in order to achieve the best visual quality estimate of the luminance channel (**Y**). These same parameters were then used on the chrominance channels (**Cr** and **Cb**). These parameters, the number of iterations, and the estimates are shown in Figure (**6.13**).

Figure 6.4: The grainy 'JFK' image

We cannot compute the MSE in this case because we do not have access to the true image **x**. However, we note that we have produced estimates of a high visual quality for this

color image with film grain noise. This attests to the ability of the iterative regularization framework to work on various types of noise. The residual of the estimate produced by method 2 appears to be the most 'noise-like,' indicating that this estimate is the best representation of the true signal, $\mathbf{x}$, among the four iterative regularization methods that we show.

6.4 Experiment 4: Deblurring

All of the experiments so far have focused on denoising. However, recall that we have more general formulations of all the iterative regularization methods in our framework as well. These formulations are as follows:

$$
\begin{aligned}
1) \qquad \widehat{\mathbf{x}}_{k+1} &= \mathcal{B}\left(\mathbf{y} + \sum_{i=1}^{k} \mathbf{A}^T(\mathbf{y} - \mathbf{A}\widehat{\mathbf{x}}_i)\right), \\
2) \qquad \widehat{\mathbf{x}}_{k+1} &= \widehat{\mathbf{x}}_1 + \mathbf{A}\mathcal{B}\left(\mathbf{y} - \widehat{\mathbf{x}}_1 + \sum_{i=2}^{k} \mathbf{A}^T(\mathbf{y} - \widehat{\mathbf{x}}_i)\right), \\
3) \qquad \widehat{\mathbf{x}}_{k+1} &= \widehat{\mathbf{x}}_1 + \sum_{i=1}^{k} \mathbf{A}^T\mathbf{A}\mathcal{B}\left(\mathbf{y} - \widehat{\mathbf{x}}_i\right) \\
&= \widehat{\mathbf{x}}_k + \mathbf{A}^T\mathbf{A}\mathcal{B}\left(\mathbf{y} - \widehat{\mathbf{x}}_k\right), \text{ and} \\
4) \qquad \widehat{\mathbf{x}}_{k+1} &= \widehat{\mathbf{x}}_1 + \sum_{i=1}^{k} \left(\mathbf{A}^T\widehat{\mathbf{x}}_1 - \mathbf{A}^T\mathbf{A}\mathcal{B}(\widehat{\mathbf{x}}_k)\right) \\
&= \widehat{\mathbf{x}}_k + \mathbf{A}^T\widehat{\mathbf{x}}_1 - \mathbf{A}^T\mathbf{A}\mathcal{B}\left(\widehat{\mathbf{x}}_k\right).
\end{aligned}
$$

where $\mathbf{A}$ is a matrix blur operator and

$$
\widehat{\mathbf{x}}_1 = \arg\min_{\mathbf{x}} \left\{ \frac{1}{2}\|\mathbf{y} - \mathbf{A}\mathbf{x}\|^2 + J(\mathbf{x}) \right\}. \tag{6.8}
$$

The image that we are trying to deblur is shown in Figure (**6.5** b). This image was generated by convolving the image in Figure (**6.5** a) with a 3×3 averaging kernel and then adding

white Gaussian noise of variance $\sigma^2 = 28.46$. We use the generalized version of our iterative regularization framework with the Bilateral Filter to find the best MSE estimates. The operating parameters were tuned by hand and are shown, along with the number of iterations and the estimates themselves in Figure (**6.14**). For comparison we also show the estimate obtained from the standard least-squares estimator ($\widehat{\mathbf{x}} = (\mathbf{A}^T\mathbf{A})^{-1}\mathbf{A}^T\mathbf{y}$) in Figure (**6.15**). The least squares estimator was implemented using steepest descent; 400 iterations with a step size of $\gamma = .01$ were used.

(a) **(b)**

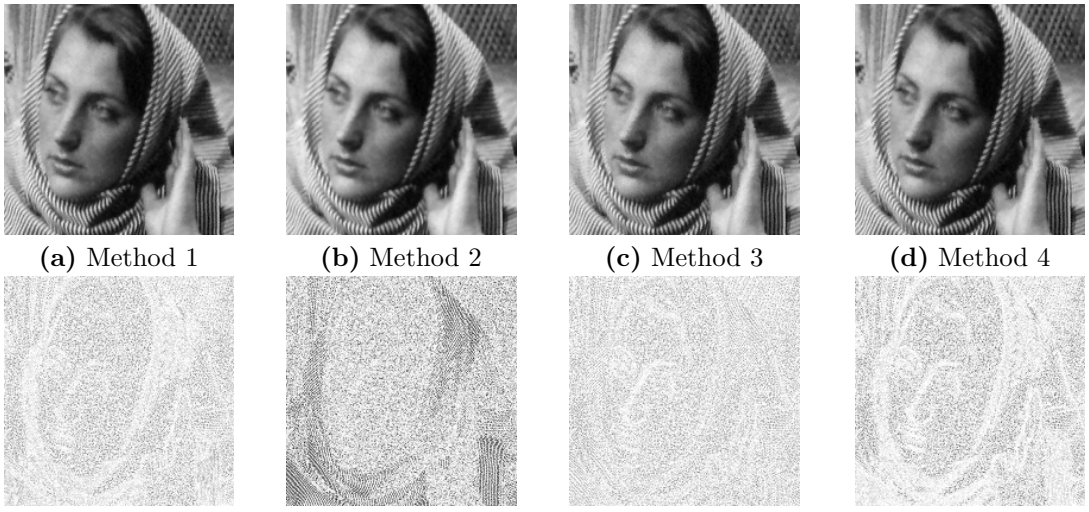
Figure 6.5: **(a)** The original 'Peppers' image **(b)** The noisy, blurred version of 'Peppers.'

We note that Method 1 and 4 appear to do the best job of deblurring the image, though all four methods achieve a lower MSE than the least-squares estimate. In each case, the MSE values are an average of 20 Monte-Carlo experiments.

We have now thoroughly illustrated the image restoration capabilities of our iterative regularization framework and will move on to some conclusions about what we have presented in this thesis and a short discussion of some areas where this work may be expanded in the future.

	N	σ_d	σ_r	**iterations**
Method 1	2	1.1	35	2
Method 2	2	1.1	31	3
Method 3	2	1.1	14	2
Method 4	2	1.1	23	2

Bilateral Filter parameters used to generate the estimates below



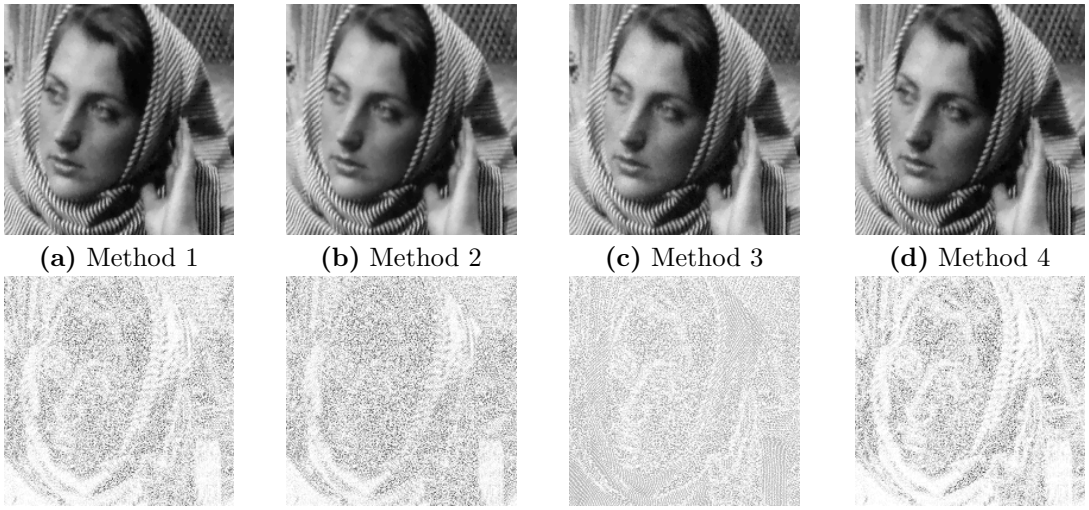
(a) Method 1 **(b)** Method 2 **(c)** Method 3 **(d)** Method 4

(e) Avg. MSE=18.12 **(f)** Avg. MSE=17.84 **(g)** Avg. MSE=18.27 **(h)** Avg. MSE=17.48

Figure 6.6: Detail of the best MSE estimates of the image in Figure (**6.1** b) using the Bilateral Filter and iterating via: **(a)** Osher's iterative regularization method, **(b)** SRR, **(c)** ITR, and **(d)** IUR. **(e)** The residual of (a) **(f)** The residual of (b) **(g)** The residual of (c) **(h)** The residual of (d)

	λ	γ	i	**iterations**
Method 1	.18	0.1	50	2
Method 2	.21	0.1	50	3
Method 3	.70	0.1	50	2
Method 4	.32	0.1	50	2

Total Variation Regularization parameters used to generate the estimates below



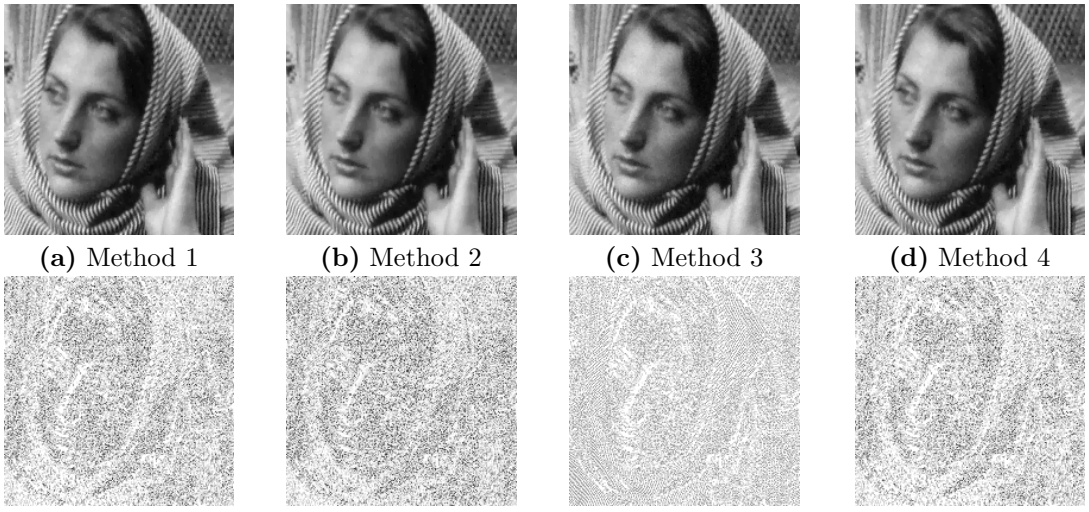
(a) Method 1 **(b)** Method 2 **(c)** Method 3 **(d)** Method 4

(e) Avg. MSE=16.48 **(f)** Avg. MSE=15.99 **(g)** Avg. MSE=18.46 **(h)** Avg. MSE=17.08

Figure 6.7: Detail of the best MSE estimates of the image in Figure (**6.1** b) using Total Variation regularization and iterating via: **(a)** Osher's iterative regularization method, **(b)** SRR, **(c)** ITR, and **(d)** IUR. **(e)** The residual of (a) **(f)** The residual of (b) **(g)** The residual of (c) **(h)** The residual of (d)

	N	α	λ	γ	i	**iterations**
Method 1	2	0.008	788	0.01	50	2
Method 2	2	0.008	789	0.01	50	3
Method 3	2	0.008	207	0.01	50	2
Method 4	2	0.008	406	0.01	50	2

Bilateral Total Variation Regularization parameters used to generate the estimates below



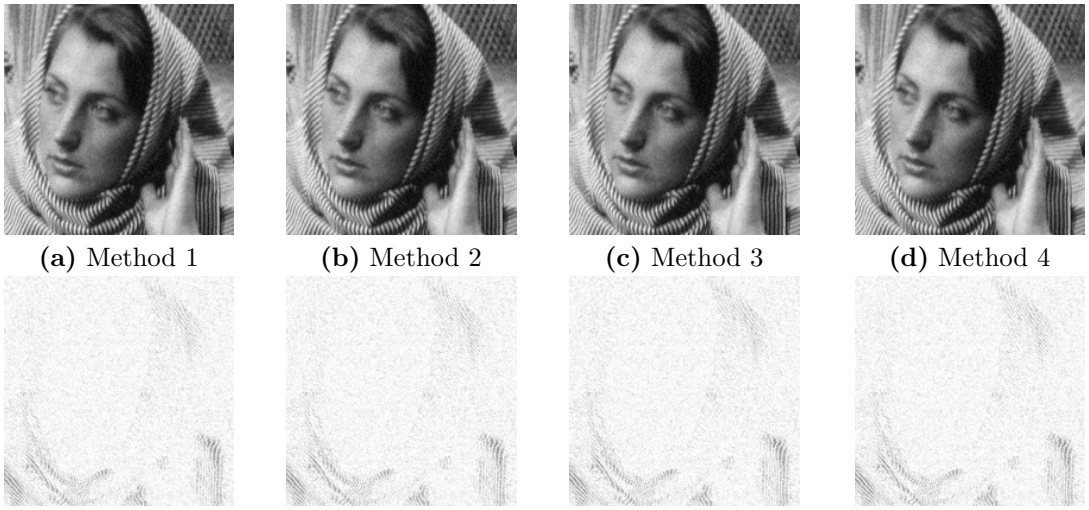
(a) Method 1 **(b)** Method 2 **(c)** Method 3 **(d)** Method 4

(e) Avg. MSE=19.42 **(f)** Avg. MSE=18.88 **(g)** Avg. MSE=19.37 **(h)** Avg. MSE=19.44

Figure 6.8: Detail of the best MSE estimates of the image in Figure (**6.1** b) using Bilateral Total Variation regularization and iterating via: **(a)** Osher's iterative regularization method, **(b)** SRR, **(c)** ITR, and **(d)** IUR. **(e)** The residual of (a) **(f)** The residual of (b) **(g)** The residual of (c) **(h)** The residual of (d)

	λ	γ	i	**iterations**
Method 1	2.34	0.1	60	4
Method 2	2.34	0.1	60	4
Method 3	2.34	0.1	60	4
Method 4	2.34	0.1	60	4

Tikhonov Regularization parameters used to generate the estimates below



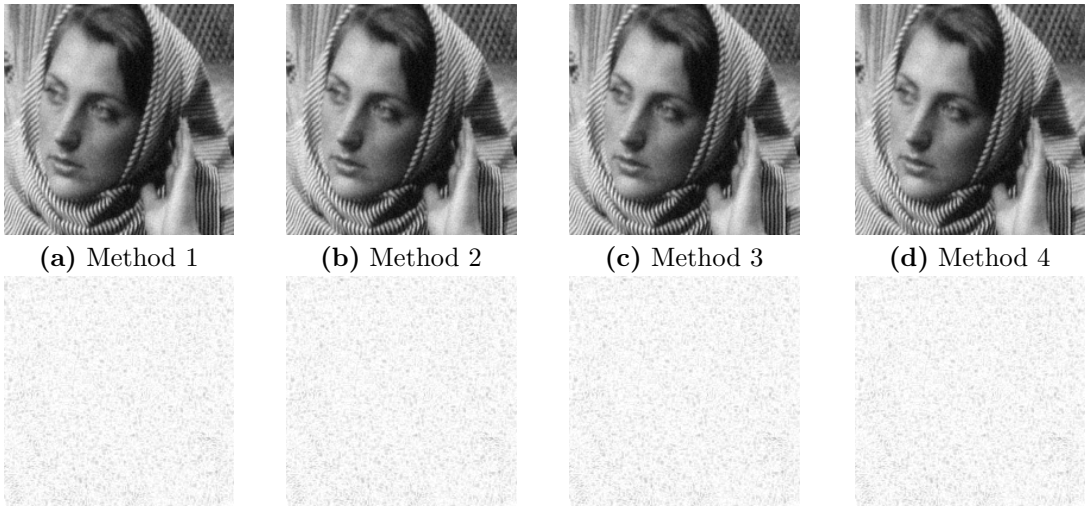
(a) Method 1 **(b)** Method 2 **(c)** Method 3 **(d)** Method 4

(e) Avg. MSE=26.09 **(f)** Avg. MSE=26.09 **(g)** Avg. MSE=26.09 **(h)** Avg. MSE=26.09

Figure 6.9: Detail of the best MSE estimates of the image in Figure (**6.1** b) using Tikhonov regularization and iterating via: **(a)** Osher's iterative regularization method, **(b)** SRR, **(c)** ITR, and **(d)** IUR. **(e)** The residual of (a) **(f)** The residual of (b) **(g)** The residual of (c) **(h)** The residual of (d)

	N	h	Q	**iterations**
Method 1	7	0.84	2	5
Method 2	7	0.84	2	5
Method 3	7	0.84	2	5
Method 4	7	0.84	2	5

Classic kernel regression parameters used to generate the estimates below



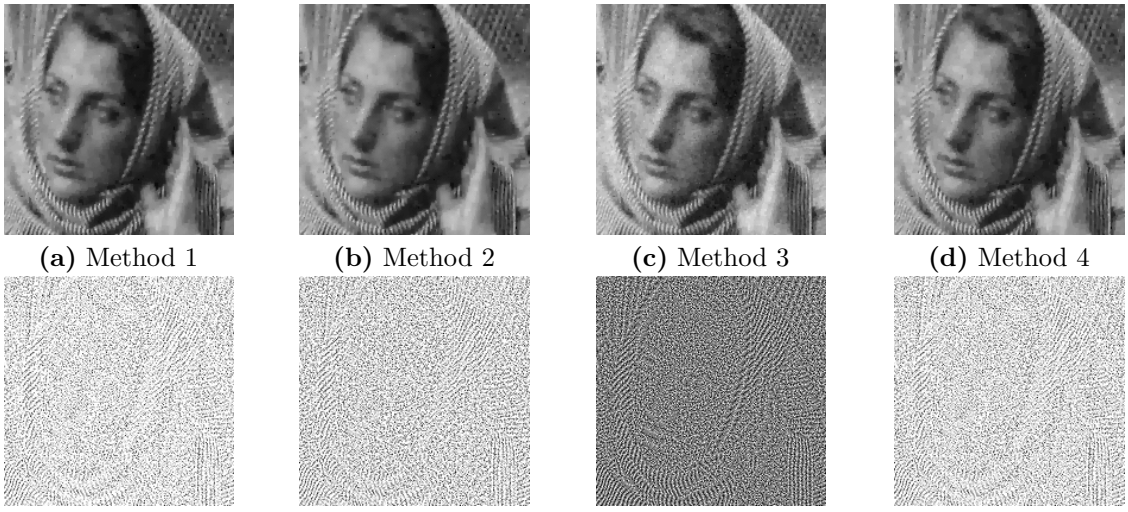
(a) Method 1 **(b)** Method 2 **(c)** Method 3 **(d)** Method 4

(e) Avg. MSE=23.57 **(f)** Avg. MSE=23.57 **(g)** Avg. MSE=23.57 **(h)** Avg. MSE=23.57

Figure 6.10: Detail of the best MSE estimates of the image in Figure (**6.1** b) using classic kernel regression and iterating via: **(a)** Osher's iterative regularization method, **(b)** SRR, **(c)** ITR, and **(d)** IUR. **(e)** The residual of (a) **(f)** The residual of (b) **(g)** The residual of (c) **(h)** The residual of (d)

	N	α	λ	γ	i	**iterations**
Method 1	2	0.008	2903	0.01	50	2
Method 2	2	0.008	2790	0.01	50	3
Method 3	2	0.008	900	0.01	50	2
Method 4	2	0.008	1410	0.01	50	2

Bilateral Total Variation Regularization parameters used to generate the estimates below



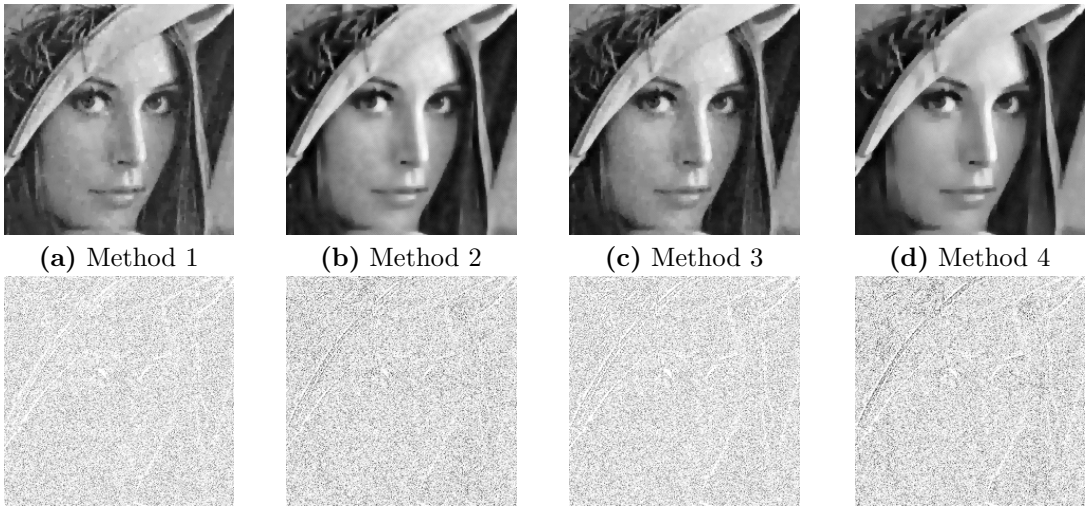
(a) Method 1 **(b)** Method 2 **(c)** Method 3 **(d)** Method 4

(e) Avg. MSE=123.83 **(f)** Avg. MSE=116.12 **(g)** Avg. MSE=111.04 **(h)** Avg. MSE=123.33

Figure 6.11: Detail of the best MSE estimates of the image in Figure (**6.2** b) using Bilateral Total Variation regularization and iterating via: **(a)** Osher's iterative regularization method, **(b)** SRR, **(c)** ITR, and **(d)** IUR. **(e)** The residual of (a) **(f)** The residual of (b) **(g)** The residual of (c) **(h)** The residual of (d)

	N	α	λ	γ	i	**iterations**
Method 1	2	0.008	2750	0.01	50	2
Method 2	2	0.008	4267	0.01	50	3
Method 3	2	0.008	1334	0.01	50	2
Method 4	2	0.008	2949	0.01	50	2

Bilateral Total Variation Regularization parameters used to generate the estimates below



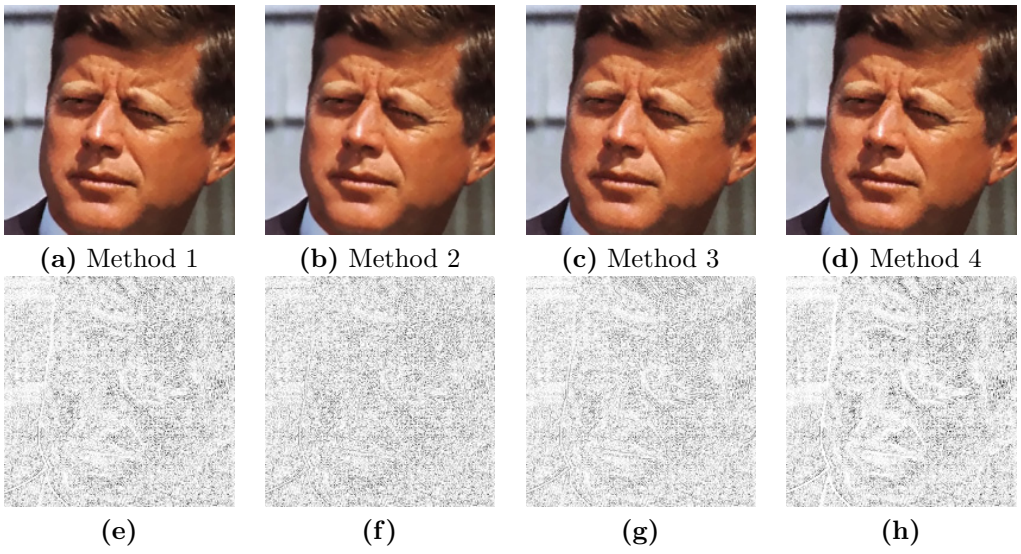
(a) Method 1 **(b)** Method 2 **(c)** Method 3 **(d)** Method 4

(e) Avg. MSE=41.67 **(f)** Avg. MSE=49.31 **(g)** Avg. MSE=39.16 **(h)** Avg. MSE=52.12

Figure 6.12: Detail of the best MSE estimates of the image in Figure (**6.3** b) using Bilateral Total Variation regularization and iterating via: **(a)** Osher's iterative regularization method, **(b)** SRR, **(c)** ITR, and **(d)** IUR. **(e)** The residual of (a) **(f)** The residual of (b) **(g)** The residual of (c) **(h)** The residual of (d)

	λ	γ	i	**iterations**
Method 1	.11	0.1	50	2
Method 2	.14	0.1	50	3
Method 3	.31	0.1	50	4
Method 4	.18	0.1	50	2

Total Variation Regularization parameters used to generate the estimates below



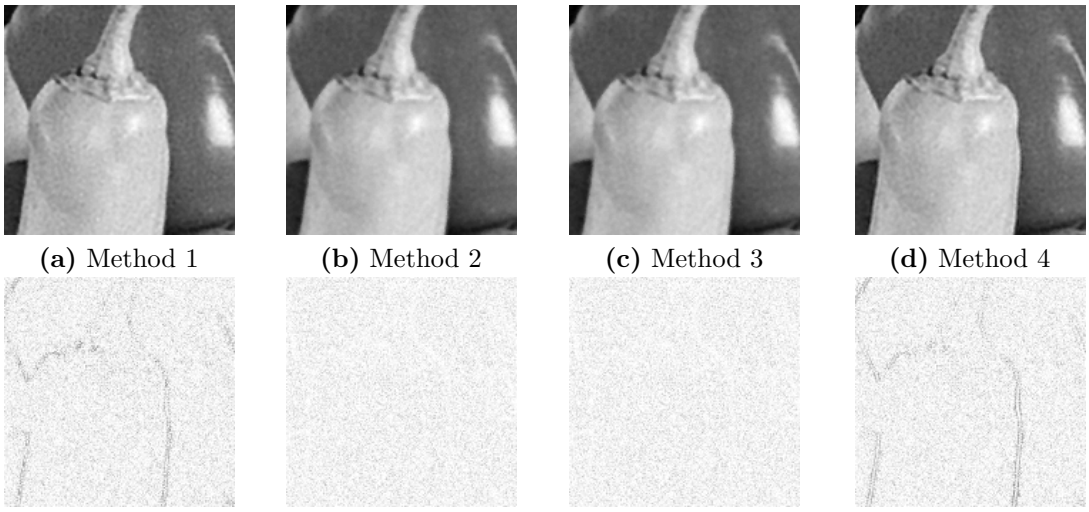
(a) Method 1 **(b)** Method 2 **(c)** Method 3 **(d)** Method 4

(e) **(f)** **(g)** **(h)**

Figure 6.13: Detail of the best MSE estimates of the image in Figure (**6.4**) using Total Variation regularization and iterating via: **(a)** Osher's iterative regularization method, **(b)** SRR, **(c)** ITR, and **(d)** IUR. **(e)** The residual of (a) **(f)** The residual of (b) **(g)** The residual of (c) **(h)** The residual of (d)

	N	σ_d	σ_r	**iterations**
Method 1	2	1.1	56	12
Method 2	2	1.1	23	10
Method 3	2	1.1	24	15
Method 4	2	1.1	23	8

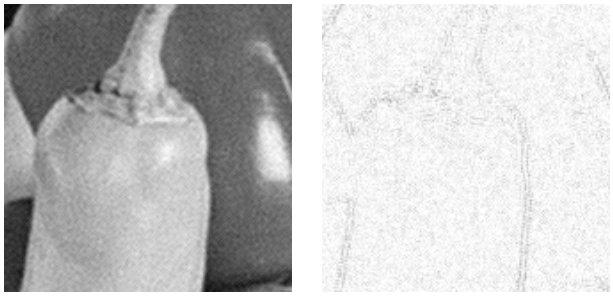
Bilateral Filter parameters used to generate the estimates below



(a) Method 1 **(b)** Method 2 **(c)** Method 3 **(d)** Method 4

(e) Avg. MSE=31.68 **(f)** Avg. MSE=59.59 **(g)** Avg. MSE=58.72 **(h)** Avg. MSE=41.70

Figure 6.14: Detail of the best MSE estimates of the image in Figure (**6.5** b) using the Bilateral Filter and iterating via the general (deblurring) formulations of: **(a)** Osher's iterative regularization method, **(b)** SRR, **(c)** ITR, and **(d)** IUR. **(e)** The residual of (a) **(f)** The residual of (b) **(g)** The residual of (c) **(h)** The residual of (d)



(a) Least-squares **(b)** Avg. MSE=66.01

Figure 6.15: **(a)** Detail of the best MSE estimates of the image in Figure (**6.5** b) using the standard least-squares estimator **(b)** The residual of (a)

Chapter 7

Conclusions and Future Work

Our iterative regularization framework presents a very general methodology for image reconstruction. Under this framework we have related the independently derived methods of Osher ([1]) and Tukey ([11]) and shown that infinitely many variants on these methods are possible. We proposed two new major variants as methods in their own right; making a total of four major iterative regularization methods which we studied in this thesis. Each of these methods describes a technique for iteratively returning the lost detail to an initially over-regularized estimate. The true generality of our framework lies in the fact that any form of regularization may be used. One can think of any image restoration method as simply a black box. Our framework presents several ways to feedback the estimate into this black box and improve the overall reconstruction quality.

Essentially all image restoration methods require some control parameters to be picked by the user. A typical technique to achieve a good quality estimate for these methods is as follows: begin with a guess for the parameter values, compute the estimate, slightly change one or more of the parameter values, compute the estimate again, and finally compare

it to the last one. This process is continued until the best estimate is found. A more analytical alternative is to use the cross-validation framework ([21], [22]). Either method is usually quite tedious and time consuming. We have seen in our numerous experiments that by alternatively using the iterative regularization framework, these control parameters may be picked naively and the iterations will act as a controlled manner of correcting for an initially bad guess.

We have also seen that all four major iterative regularization methods are useful; there is not one method that is always more advantageous to use. From our experiments it would appear that methods 2 and 4 are most useful for denoising images with a moderate to high SNR; method 3 is most useful for denoising images with a low SNR, and methods 1 and 4 are most useful for deblurring images. Additionally, iterative Total Variation regularization appears to yield the best denoising results among all the experiments shown. We note that though one method may yield a slightly lower MSE estimate for a particular image, the other three methods typically produce an estimate of similar quality. Also, in Chapter 5 we saw that only method 4 can be used for inverse regularization, allowing it to be used for a larger variety of applications such as compression and texture transfer.

In the future, the bias-variance tradeoff analysis of the estimates produced by our framework could be studied further. We currently use experimental methods to compare the bias and variance of the estimates from the different methods; we could further our understanding of the tradeoff if we could do this analytically as well. Additionally, we currently use only one application of the Bilateral Filter as a regularization method. We know from ([7]) that multiple applications of the Bilateral Filter remove more noise but tend to also increase the bias of the estimate. Another future direction of this research could be to study what effect this has when used in an iterative regularization scheme.

We have presented a generalized version of our iterative regularization framework for deblurring when the blur function is known. A possible future direction for this work is to investigate a blind deconvolution approach to iterative regularization.

Finally, we have seen that iterative regularization can improve an estimate when the initial operating parameters were chosen naively. A further area to investigate in the future is whether or not iterative regularization can produce better estimates than one iteration of the normal regularization estimator (e.g. Bilateral Filter, Total Variation regularization, etc.) with optimal operating parameters.

Appendix A

The Generalized Bregman Distance

The details behind the subgradient and the generalized Bregman distance (both used to prove the convergence properties of the iterative regularization methods) are given herein.

A.1 The Subgradient

The gradient is a vector operator that computes the directional first derivatives of a given function. When a function contains discontinuities or is otherwise non-differentiable, the subgradient is defined. The vector $\mathbf{p}$ is defined as a subgradient of the convex functional $J(\cdot)$ at $\mathbf{x}$ if it satisfies the inequality

$$J(\mathbf{q}) \geq J(\mathbf{x}) + \langle \mathbf{p}, \mathbf{q} - \mathbf{x} \rangle \tag{A.1}$$

for all $\mathbf{q}$. The operator $\langle \cdot, \cdot \rangle$ denotes the duality product which is related to the standard vector inner product in the following manner: $\mathbf{a} \cdot \mathbf{b} = \langle \mathbf{a}^*, \mathbf{b} \rangle$ where $\mathbf{a}^*$ is the dual of $\mathbf{a}$. Note that when $J(\cdot)$ is in fact differentiable, the only value of $\mathbf{p}$ that will satisfy this inequality is

$\mathbf{p} = \nabla J(\mathbf{x})$. Thus, the subgradient reduces to the regular gradient when the function that it is applied to is differentiable.

We will now prove that the values used for $\mathbf{p}_k$ in Equations (2.4, 2.12, 2.18, and 2.23) are in fact subgradients of the functional $J(\mathbf{x})$ in their associated cost functions. This is done by induction using $H(\mathbf{x}, \cdot) = \frac{1}{2}\|(\cdot) - \mathbf{x}\|^2$ for illustrative purposes.

Recall that method 1 is formulated in terms of the generalized Bregman distance as:

$$\begin{aligned} \hat{\mathbf{x}}_{k+1} &= \arg\min_{\mathbf{x}} \{Q_k\} \\ &= \arg\min_{\mathbf{x}} \{H(\mathbf{x}, \mathbf{y}) + J(\mathbf{x}) - J(\hat{\mathbf{x}}_k) - \langle \mathbf{p}_k, \mathbf{x} - \hat{\mathbf{x}}_k \rangle\} \end{aligned} \tag{A.2}$$

where

$$\mathbf{p}_{k+1} = \mathbf{p}_k + \left(\frac{\partial H(\mathbf{x}, \mathbf{y})}{\partial \mathbf{x}} \middle| \mathbf{x} = \hat{\mathbf{x}}_{k+1} \right) \tag{A.3}$$

with $\mathbf{p}_0 = 0$.

When $k = 0$ we have:

$$\hat{\mathbf{x}}_1 = \arg\min_{\mathbf{x}} \{Q_0(\mathbf{x}, \mathbf{y})\} = \arg\min_{\mathbf{x}} \left\{ \frac{1}{2}\|\mathbf{y} - \mathbf{x}\|^2 + J(\mathbf{x}) \right\} \tag{A.4}$$

which is a well studied regularization problem known to have an optimal solution [27]. Because $\hat{\mathbf{x}}_1$ minimizes $Q_0(\cdot, \mathbf{y})$ and because $J(\mathbf{x})$ is non-negative and therefore $\mathbf{p}_0 = 0$ is a subgradient of $J(\mathbf{x})$, we have:

$$\frac{\partial Q_0(\hat{\mathbf{x}}_1, \mathbf{y})}{\partial \mathbf{x}} = \hat{\mathbf{x}}_1 - \mathbf{y} + \frac{\partial J(\hat{\mathbf{x}}_1)}{\partial \mathbf{x}} - \mathbf{p}_0 = 0 \tag{A.5}$$

hence

$$\frac{\partial J(\hat{\mathbf{x}}_1)}{\partial \mathbf{x}} = \mathbf{p}_0 + (\mathbf{y} - \hat{\mathbf{x}}_1) = \mathbf{p}_1. \tag{A.6}$$

When k=1:

$$\hat{\mathbf{x}}_2 = \arg\min_{\mathbf{x}} \{Q_1(\mathbf{x}, \mathbf{y})\} = \arg\min_{\mathbf{x}} \left\{ \frac{1}{2}\|\mathbf{y} - \mathbf{x}\|^2 + J(\mathbf{x}) - \langle \mathbf{x}, \mathbf{p}_1 \rangle \right\} \tag{A.7}$$

$$\frac{\partial Q_1(\hat{\mathbf{x}}_2, \mathbf{y})}{\partial \mathbf{x}} = \hat{\mathbf{x}}_2 - \mathbf{y} + \frac{\partial J(\hat{\mathbf{x}}_2)}{\partial \mathbf{x}} - \mathbf{p}_1 = 0 \tag{A.8}$$

$$\frac{\partial J(\hat{\mathbf{x}}_2)}{\partial \mathbf{x}} = \mathbf{p}_1 + (\mathbf{y} - \hat{\mathbf{x}}_2) = \mathbf{p}_2. \tag{A.9}$$

etc.

This same process can be used to verify that $\mathbf{p}_k$ is in fact a subgradient of $J(\mathbf{x})$ in the cost functions of the other three iterative regularization methods as well.

A.2 The Generalized Bregman Distance

The Bregman distance associated with functional J is defined as

$$D_J(\mathbf{u}, \mathbf{q}) \equiv J(\mathbf{u}) - J(\mathbf{q}) - \left\langle \left. \frac{\partial J(\mathbf{u})}{\partial \mathbf{u}} \right|_{\mathbf{u}=\mathbf{q}}, \mathbf{u} - \mathbf{q} \right\rangle \tag{A.10}$$

When J is continuously differentiable, the Bregman distance is the difference at point $\mathbf{u}$ between $J(\cdot)$ and the first-order Taylor series approximation to $J(\cdot)$ at $\mathbf{q}$. It defines a notion of distance between $\mathbf{u}$ and $\mathbf{q}$ in the sense that it satisfies the condition $D_J(\mathbf{u}, \mathbf{q}) \geq 0$ for all $\mathbf{u}, \mathbf{q}$ and $D_J(\mathbf{u}, \mathbf{u}) = 0$. However, since it is not necessarily symmetric nor does it necessarily satisfy the triangle inequality, it cannot be considered as a metric.

A graphical representation of Bregman distance as a measure of the convexity of J is shown in Figure (**A.1**).

For the generalized Bregman distance introduced in [1], the gradient $\frac{\partial J(\mathbf{u})}{\partial \mathbf{u}}$ is instead replaced by $\mathbf{p}$ a subgradient of J. This extends the concept of the Bregman distance to convex but non-continuously differentiable functionals.

The generalized Bregman distance is written as:

$$D_J^p(\mathbf{u}, \mathbf{q}) \equiv J(\mathbf{u}) - J(\mathbf{q}) - \langle \mathbf{p}, \mathbf{u} - \mathbf{q} \rangle . \tag{A.11}$$

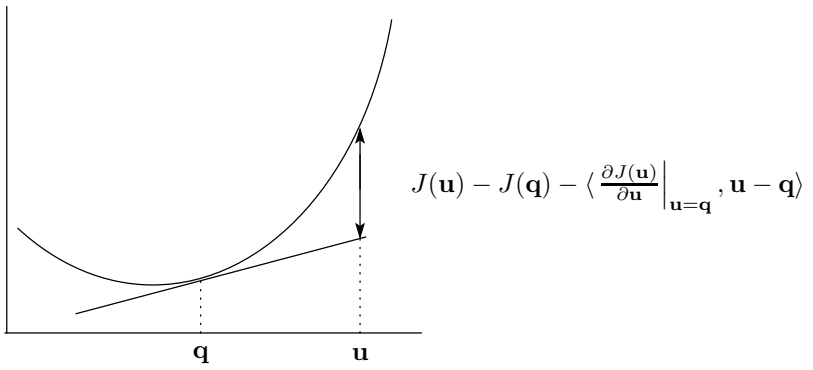


Figure A.1: The Bregman distance $D_J(\mathbf{u}, \mathbf{q})$

Appendix B

The Bilateral Filter

The Bilateral Filter (first proposed in [10]) is a simple one-pass, neighborhood filter that is very effective at removing noise while preserving edges. This is because, unlike conventional low-pass filters, the Bilateral Filter uses both the spatial distance and radiometric (gray level) differences of pixels in a neighborhood to determine the weight that each pixel will have in the weighted average that will replace the noisy pixel value at the center of the neighborhood. Conventional low-pass filters use only spatial distances.

We can express the point-by-point application of the Bilateral Filter to the noisy image $\mathbf{y}$ as:

$$\widehat{\mathbf{x}}(i) = \frac{\sum_{n=-N}^{N} \mathbf{W}(i,n)\mathbf{y}(i)}{\sum_{n=-N}^{N} \mathbf{W}(i,n)} \tag{B.1}$$

where $\widehat{\mathbf{x}}$ is the Bilateral Filtered image, N defines the neighborhood size, and $\mathbf{W}(i,n)$ is the weight matrix. This weight matrix is computed as the product of the spatial and radiometric weights $\mathbf{W}(i,n) = \mathbf{W}_r(i,n)\mathbf{W}_d(i,n)$. While both the radiometric and spatial weights may be any decaying function of the spatial and radiometric distance respectively, we use the

original formulations laid out in [10]:

$$\begin{aligned} \mathbf{W}_d(i,n) &= \exp\left(-\frac{n^2}{2\sigma_d}\right) \\ \mathbf{W}_r(i,n) &= \exp\left(-\frac{[\mathbf{y}(i)-\mathbf{y}(i-n)]^2}{2\sigma_r}\right). \end{aligned} \tag{B.2}$$

The parameters σ_r and σ_d control the strength of the radiometric and spatial filtering (respectively).

In [7], it was shown that the Bilateral Filter could equivalently be formulated as a regularized cost function of the form:

$$\widehat{\mathbf{x}} = \arg\min_{\mathbf{x}} \left\{ \frac{1}{2}\|\mathbf{y}-\mathbf{x}\|^2 + \frac{\lambda}{2}\sum_{n=-N}^{N} \left[\mathbf{x}-\mathbf{S}^n\mathbf{x}\right]^T \mathbf{W}_{\mathbf{y},n} \left[\mathbf{x}-\mathbf{S}^n\mathbf{x}\right] \right\} \tag{B.3}$$

where $\mathbf{S}^n$ is a matrix shift operator, λ controls the amount of regularization, and $\mathbf{W}_{\mathbf{y},n}$ is a (spatial and radiometric dependant) weight matrix. In Chapter 6, we don't solve this cost function minimization problem directly, but rather we simply apply the adaptive filtering technique described in [10].

Appendix C

Total Variation Regularization

Total Variation regularization is a regularization cost function for image denoising developed by Rudin, Osher, and Fatemei [9]. The TV regularization functional forces the estimate to be piecewise constant by requiring the L_1-norm of the image gradient to be minimized. This tends to preserve the edges and smooth out other areas in an image.

Total Variation Regularization ([9]) is formulated as:

$$\widehat{\mathbf{x}} = \arg\min_{\mathbf{x}} \left\{ \frac{1}{2}\|\mathbf{y} - \mathbf{x}\|^2 + \lambda \|\nabla \mathbf{x}\|_1 \right\} \tag{C.1}$$

where λ controls the amount of regularization. This can be solved in practice by using the gradient descent technique, which can be written as:

$$\begin{aligned} \widehat{\mathbf{x}}_{i+1} &= \widehat{\mathbf{x}}_i - \gamma \left[\frac{\partial}{\partial \widehat{\mathbf{x}}_i} \left(\frac{1}{2}\|\mathbf{y} - \widehat{\mathbf{x}}_i\|^2 + \lambda \|\nabla \widehat{\mathbf{x}}_i\|_1 \right) \right] \\ &\approx \widehat{\mathbf{x}}_i - \gamma \left[\widehat{\mathbf{x}}_i - \mathbf{y} + \lambda \left(\sum_{n=-1}^{1} [\mathbf{I} - \mathbf{S}^n] \, sign[\widehat{\mathbf{x}}_i - \mathbf{S}^n \widehat{\mathbf{x}}_i] \right) \right] \end{aligned} \tag{C.2}$$

where γ is the gradient descent step size, i is the iteration number, and

$$sign[z] = \begin{cases} -1 & \text{if } z < 0 \\ 1 & \text{otherwise.} \end{cases} \tag{C.3}$$

We have approximated $\nabla\widehat{\mathbf{x}}_i \approx \sum_{n=-1}^{1}[\widehat{\mathbf{x}}_i - \mathbf{S}^n\widehat{\mathbf{x}}_i]$, where $\mathbf{S}^n$ is a matrix shift operator.

Bibliography

[1] S. Osher, M. Burger, D. Goldfarb, J. Xu, and W. Yin, "An iterative regularization method for total variation-based image restoration," *SIAM Multiscale Model. and Simu.*, vol. 4, pp. 460–489, 2005.

[2] R. Gonzalez and R. Woods, *Digital Image Processing, second edition.* Pearson, 2002.

[3] D. Donoho and I. Johnstone, "An ideal spatial adaptation by wavelet shrinkage," *Biometrika*, vol. 81, no. 3, pp. 425–455, 1994.

[4] M. Figueiredo and R. Nowak, "An EM algorithm for wavelet-based image restoration," *IEEE Trans. on Image Processing*, vol. 12, no. 8, pp. 906–916, 2003.

[5] A. Tikhonov, "On the stability of inverse problems," *Dokl. Akad. Nauk SSSR*, vol. 39, no. 5, pp. 195–198, 1943.

[6] T. Chan, S. Osher, and J. Shen, "The digital TV filter and nonlinear denoising," *IEEE Trans. on Image Processing*, vol. 10, no. 2, pp. 231–241, 2001.

[7] M. Elad, "On the origin of the bilateral filter and ways to improve it," *IEEE Trans. on Image Processing*, vol. 11, no. 10, pp. 1141–1151, 2002.

[8] S. Farsiu, M. Robinson, M. Elad, and P. Milanfar, "Fast and robust mulitframe super resolution," *IEEE Trans. on Image Processing*, vol. 13, no. 10, pp. 1327–1344, 2004.

[9] L. Rudin, S. Osher, and E. Fatemi, "Nonlinear total variation based noise removal algorithms," *Phys. D*, vol. 60, pp. 259–268, 1992.

[10] C. Tomasi and R. Manduchi, "Bilateral filtering for gray and color images," *Proceedings of the IEEE International Conference on Computer Vision*, pp. 836–846, 1998.

[11] J. Tukey, *Exploratory Data Analysis.* Addison-Wesley, 1977.

[12] J. Kaiser and R. Hamming, "Sharpening the response of a symmetric nonrecursive filter by multiple use of the same filter," *IEEE Trans. on Acoustic Speech and Signal Processing*, vol. ASSP-25, no. 5, pp. 415–422, Oct. 1977.

[13] P. Buhlmann and B. Yu, "Boosting with the L_2 loss: Regression and classification," *Journal of the American Statistical Association*, vol. 98, no. 462, pp. 324–339, 2003.

[14] S. Kay, *Fundamentals of Statistical Signal Processing: Estimation Theory*, A. Oppenheim, Ed. Prentice Hall PTR, 1993.

[15] J. Aujol, G. Aubert, L. Blanc-Feraud, and A. Chambolle, "Image decomposition into a bounded variation component and an oscillating component," *Journal of Mathematical Imaging and Vision*, vol. 22, no. 1, pp. 71–88, 2005.

[16] L. Vese and S. Osher, "Modeling textures with total variation minimization and oscillatory patterns in image processing," *Journal of Scientific Computing*, vol. 19, pp. 553–572, 2003.

[17] V. Bhaskaran and K. Konstantinides, *Image and Video Compression Standards - Algorithms and Architectures.* Kluwer, 1995.

[18] P. Campisi, J. Yan, and D. Hatzinakos, "Signal-dependent film grain noise generation using homomorphic adaptive filtering," *IEE Proc. Vis. Image Signal Process.*, vol. 147, no. 3, pp. 283–287, 2000.

[19] J. Yan, P. Campisi, and D. Hatzinakos, "Film grain noise removal and generation for color images," *IEEE International Conference on Acoustics Speech and Signal Processing*, 1998.

[20] J. Yan and D. Hatzinakos, "Signal-dependent film grain noise removal and generation based on higher-order statistics," *IEEE Signal Processing Workshop on Higher-Order Statistics*, pp. 77–81, 1997.

[21] M. Wand and M. Jones, *Kernel Smoothing*. Chapman and Hall, 1995.

[22] W. Hardle, M. Muller, S. Sperlich, and A. Werwatz, *Nonparametric and Semiparametric Models*. Springer, 2004.

[23] H. Takeda, S. Farsiu, and P. Milanfar, "Kernel regression for image processing and reconstruction," *to appear in IEEE Trans. on Image Processing*.

[24] M. Di Marzio and C. Taylor, "Kernel density classification and boosting: an L_2 analysis," *to appear in Statistics and Computing*.

[25] ——, "Boosting kernel density estimates: a bias reduction technique?" *Biometrika*, vol. 91, no. 1, pp. 226–233, 2004.

[26] S. Farsiu, M. Elad, and P. Milanfar, "Multiframe demosaicing and super-resolution of color images," *IEEE Trans. on Image Processing*, vol. 15, no. 1, pp. 141–159, 2006.

[27] R. Acar and C. Vogel, "Analysis of total variation penalty methods for ill-posed problems," *Inverse Problems*, vol. 10, pp. 1217–1229, 1994.