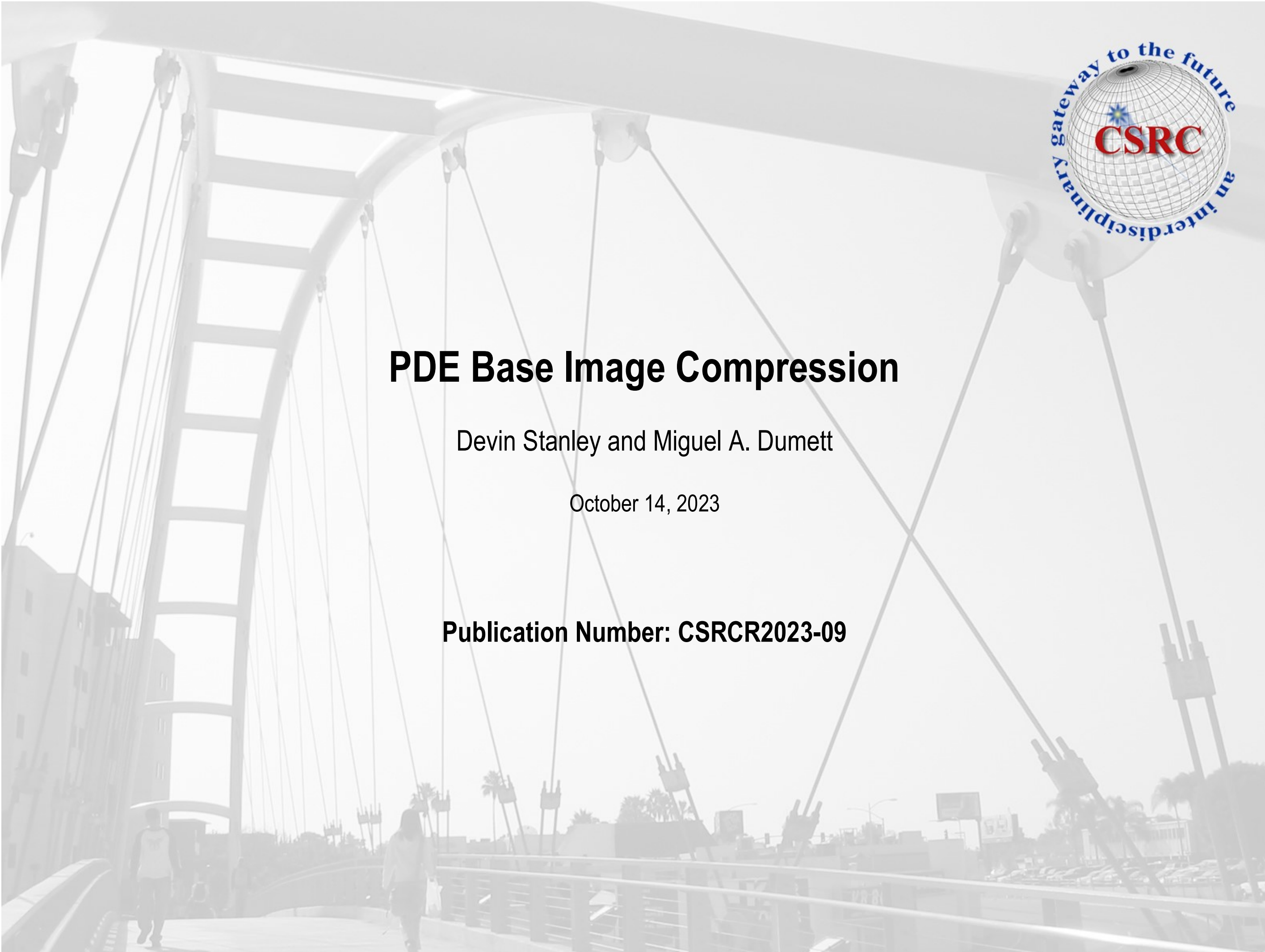


PDE Base Image Compression

Devin Stanley and Miguel A. Dumett

October 14, 2023

Publication Number: CSRCR2023-09

Computational Science &
Engineering Faculty and Students
Research Articles

Database Powered by the
Computational Science Research Center
Computing Group & Visualization Lab

COMPUTATIONAL SCIENCE & ENGINEERING

SAN DIEGO STATE UNIVERSITY

Computational Science Research Center
College of Sciences
5500 Campanile Drive
San Diego, CA 92182-1245
(619) 594-3430

PDE Base Image Compression

Devin Stanley * and Miguel A. Dumett †‡

October 14, 2023

Abstract

This paper explores the application of mimetic methods in image inpainting. It utilizes the MOLE library to develop advanced inpainting algorithms. This research investigates the integration of MOLE operators in two distinct methods: basic diffusion-based inpainting and Charbonnier diffusivity-based inpainting.

1 Introduction

In the field of image processing, inpainting techniques hold immense importance for restoring missing or corrupted parts of images. This paper explores the application of MOLE (Mimetic Operators Library Enhanced) [1] in image inpainting.

MOLE, known for its robust mathematical frameworks, is utilized to develop advanced inpainting algorithms. This research investigates the integration of MOLE operators in two distinct methods: basic diffusion-based inpainting and Charbonnier diffusivity-based inpainting. Both methods are enhanced by MOLE operators, showcasing their potential in reconstructing sparsified images.

Although the study acknowledges the broader applications of MOLE-enhanced inpainting methods, our primary focus remains on enhancing common inpainting algorithms using MOLE operators. While our exploration is not exhaustive, the implemented algorithms offer a promising foundation for further research and practical implementations specifically in inpainting sparsified images.

Restoring damaged or incomplete digital images due to corruption or missing data is a fundamental challenge in modern image processing. The technique of inpainting has emerged as a valuable tool in addressing this challenge, enabling the reconstruction of flawed images for various applications. This research delves into the exploration and integration of MOLE operators into inpainting techniques. MOLE, a powerful mathematical library focused on solving partial

*Computational Science Master's Program at San Diego State University (dstanley4099@sdsu.edu).

†Editor: Jose E. Castillo

‡Computational Science Research Center at San Diego State University (mdumett@sdsu.edu).

differential equations, holds promise for refining inpainting algorithms. By incorporating MOLE operators, our objective is to enhance inpainting not only for restoring corrupt images but also for optimizing image compression. This optimization is crucial for efficient data storage and transmission, particularly in contexts where bandwidth and storage resources are limited.

Our study centers on one of the main aspects of inpainting: reconstructing sparsified images using enhanced diffusion techniques. We explore the integration of MOLE operators into basic diffusion-based inpainting and Charbonnier diffusivity-based inpainting methods.

2 Methodology

In our inpainting approach, we leverage the principles of heat diffusion, encapsulated in the partial differential equation governing the spread of heat in a given domain. Analogous to the behavior of heat diffusing across a surface, we model the sparsified image as a grid of pixels. Each remaining pixel acts as a set of heaters, each emitting a constant heat whose intensities correspond to the pixel's value.[3]. The empty or missing pixels are akin to unlit areas, representing regions where the heat, or in our case, image information, needs to be propagated or "filled in" over a series of time intervals.

2.1 Governing Equations

Let the original, unsparsified image be f, whose domain will be defined as Ω, and let u be the inpainted image. The image domain can be partitioned known and unknown pixel values, where Ω_K represents retained data post sparsification and $\Omega \backslash \Omega_K$ represents the removed pixel data. Using the standard heat diffusion equation with reflective boundary conditions, our homogenous inpainting PDE can be defined as:

$$\nabla u = 0 \text{ on } \Omega \backslash \Omega_K \tag{1}$$

$$u = f \text{ on } \Omega_k \tag{2}$$

$$\partial u = 0 \text{ on } \partial\Omega \tag{3}$$

Whose steady-state solution can be given by:

$$c(\mathbf{x})(u - f) - (1 - c(\mathbf{x}))Lu = 0, \text{ on } \Omega \tag{4}$$

Where $c(\mathbf{x})$ is our confidence function, this prevents diffusion from occurring over the known pixel values in Ω_K. It is given as:

$$c(\mathbf{x}) = \begin{cases} 1 & \text{if } x \in \Omega_k, \\ 0 & \text{otherwise.} \end{cases} \tag{5}$$

2.2 Charbonnier Diffusivity

When Charbonnier Diffusivity[2] is used in the inpainting algorithm, the process is transformed from linear to nonlinear isotropic[1] diffusion[4]. Redefining Lu, using Charbonnier Diffusivity as a nonlinear diffusion rate, can be given as:

$$Lu := \text{ div } g(|\nabla u_\sigma|^2)\nabla u \tag{6}$$

Where $g(x^2)$ can be given as:

$$g(x^2) = \left[1 + \frac{x^2}{\lambda^2}\right]^{-\frac{1}{2}} \tag{7}$$

Where λ is a parameter that controls the diffusion rate and u_σ is a Gaussian-smoothed image of the inpainted imaged.

2.3 Algorithm Process

Our approach to image inpainting involves integrating MOLE operators into established algorithms. We begin by preprocessing the input image, converting it to grayscale, and intentionally introducing sparsity to simulate real-world scenarios resembling compressed images. In this context, sparsity represents the deliberate reduction of random image data, emulating the principles of image compression techniques.

The algorithm and its outcomes are illustrated below:

Algorithm 1 Removes a percentage of pixels from an input image

Input: *image*, *percent*
Output: *sparse*

$[h, w] = \text{size}(image)$
$key = \text{rand}(h, w)$ ▷ Create random matrix for thresholding
$key(key < percent) = -1$
$key(key \geq percent) = 1$
$sparse = (key \odot image)$ - 1
$sparse(sparse < 0) = 0$ ▷ Set all negative pixels to 0

The algorithmic process unfolds in two primary stages: basic diffusion-based inpainting and Charbonnier diffusivity-based inpainting, both enhanced by mimetic operators implemented by the MOLE library.

In the basic diffusion method, the inpainted image undergoes iterative refinement, gradually filling in the missing or sparse regions at a constant diffusive rate.

[1] Originally, the goal was to implement an anisotropic nonlinear inpainting algorithm, utilizing a diffusion tensor computed using Charbonnier Diffusivity. Unfortunately, due to the size of MOLE operators, certain computations within a memory constrained environment proved to be too large.

After each iteration, our original sparse domain Ω_K, is superimposed on the inpainted area, preventing diffusion over the known domain. MOLE-enhanced Laplacian and gradient matrices play a pivotal role here, ensuring the diffusion process is guided by mathematical precision.

The Charbonnier diffusivity method introduces dynamic adaptability, adjusting diffusivity coefficients non-linearly based on the image's evolving characteristics. This dynamic adjustment ensures a nuanced inpainting process, where the image is iteratively updated, guided by MOLE-enhanced diffusivity coefficients.

Below is a combination of both algorithms separated by a method flag as well as the implementation of the Charbonnier-diffusivity equation:

Algorithm 2 Inpaint a sparsified image

```
Input: sparse, percent, iter, λ
Output: inpainted
  dt = 0.1                                              ▷ Speed of Diffusion
  [h, w] = size(sparse)
  dx, dy = 1
  L = lap2D(2, m, dx, n, dy)                ▷ Discrete Laplacian MOLE Operator
  g = grad2D(2, m, dx, n, dy)                ▷ Discrete Gradient MOLE Operator
  inpainted = sparse
  if method == 0 then                                  ▷ Constant Diffusivity
      for i = 1 : iter do
          inpainted += inpainted * L * dt
          inpainted = superimpose(inpainted, sparse)   ▷ Confidence Function
      end for
  end if
  if method == 1 then                               ▷ Charbonnier Diffusivity
      for i = 2 : iter do
          inpainted += inpainted * L * dt
          dt = charbonnier(g, inpainted, λ)
          inpainted = superimpose(inpainted, sparse)                       ▷
      end for
  end if
```

Algorithm 3 Find the Charbonnier-diffusivity coefficient

```
Input: g, inpainted, λ
Output: dt
  U = imgaussfilt(sparse)
  dt = 1/sqrt(1+normest(g′gU))^2/λ^2
```

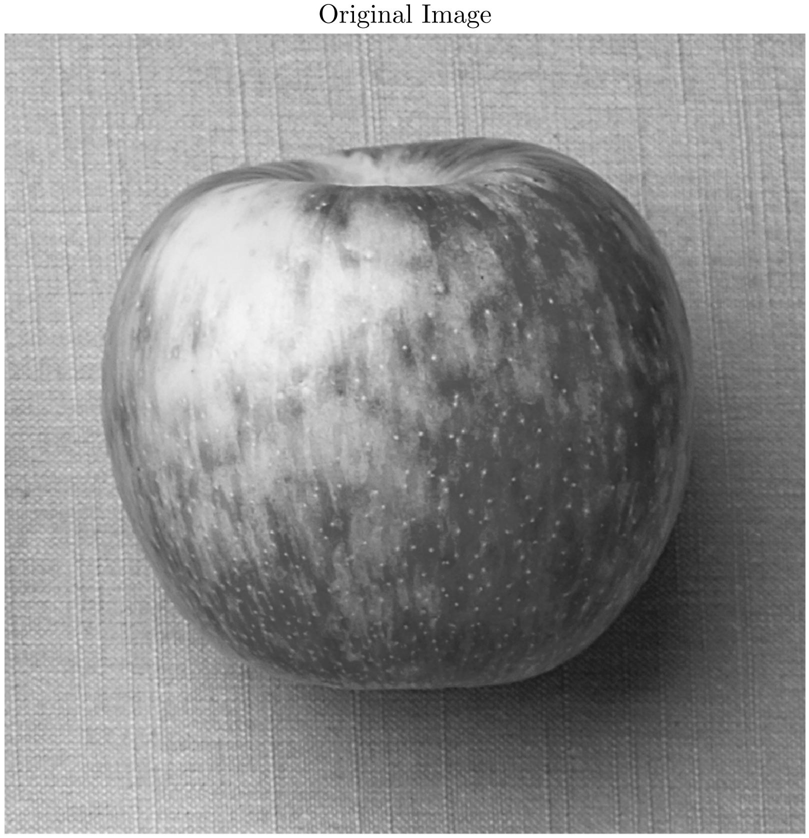
3 Results

Our findings provide a promising foundation for the development of advanced inpainting algorithms, emphasizing the pivotal role of MOLE operators. To compare the results of both the basic diffusion-based inpainting and Charbonnier diffusivity-based inpainting, we will compare their Structural Similarity Index, or SSIM. The SSIM is used to compare the similarity of an image to a reference image. This metric index is bounded by $[0, 1]$ where an SSIM of 1 indicates that the input image is identical to the reference image. The diffusion rate for all inpainted images is 0.1.

3.1 Original Image

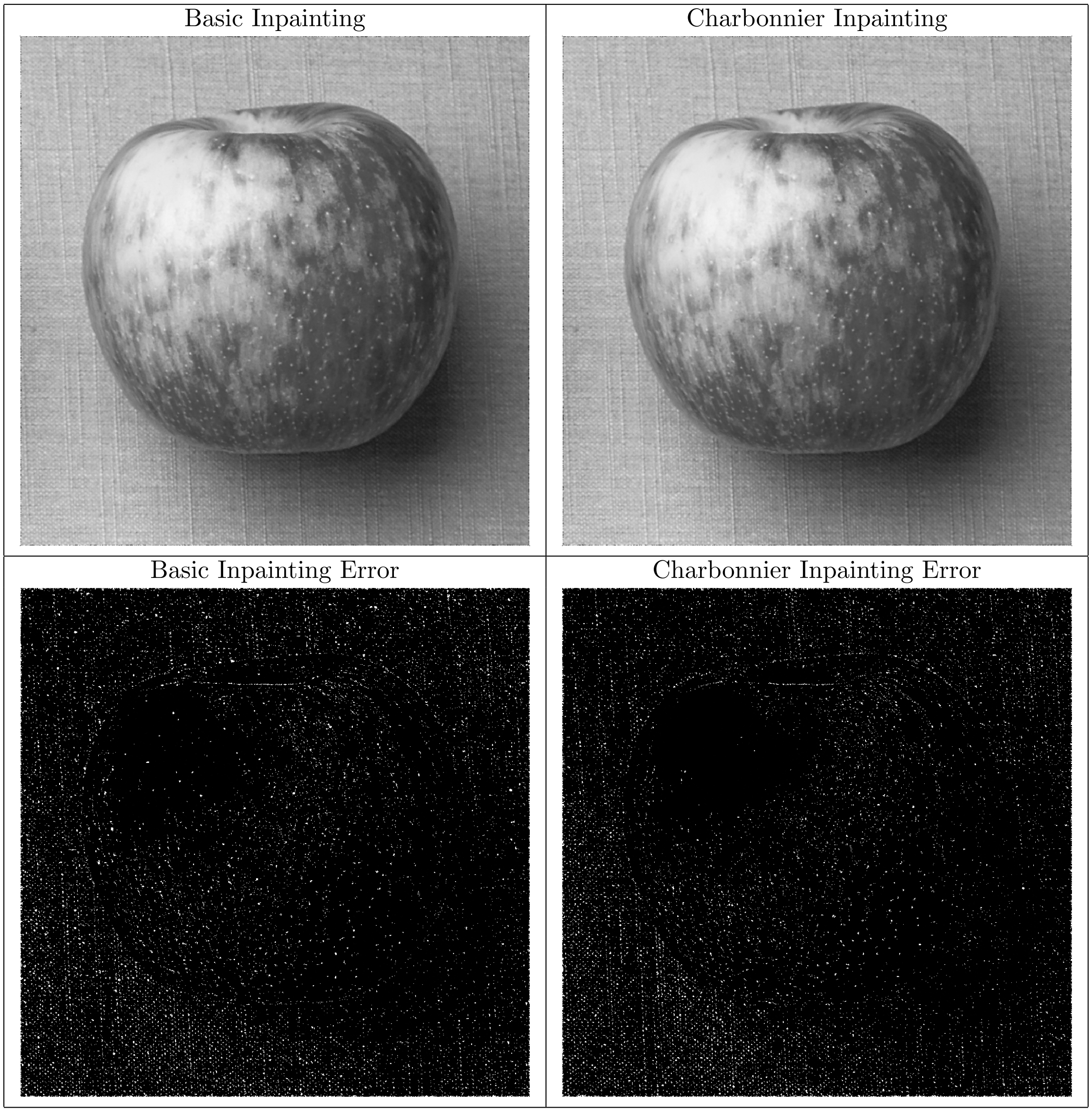
The image that we use will use for our inpainting results and metrics is shown below:

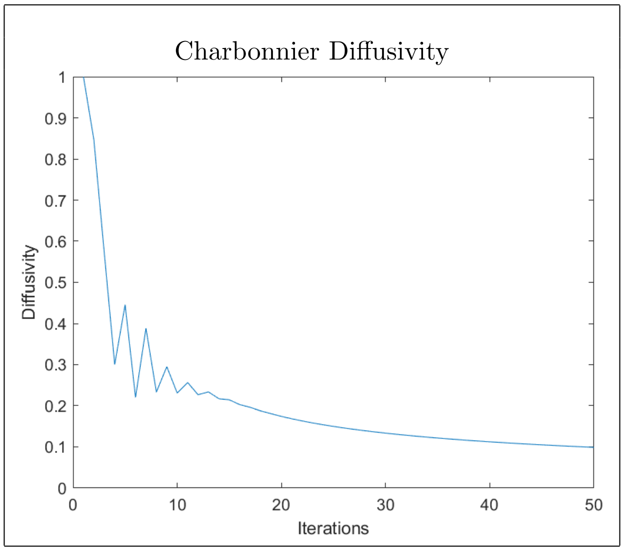
Original Image



3.2 50% Sparsification

Using 50 iterations with $\lambda = 0.001$:



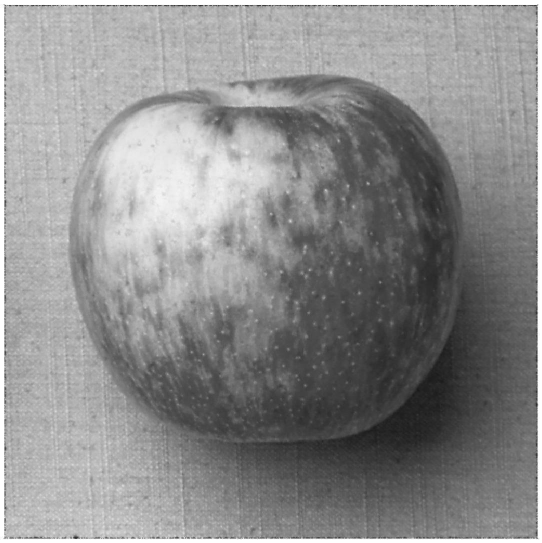
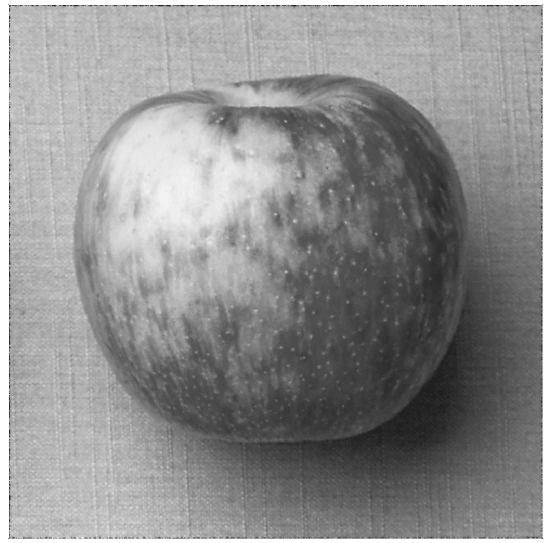
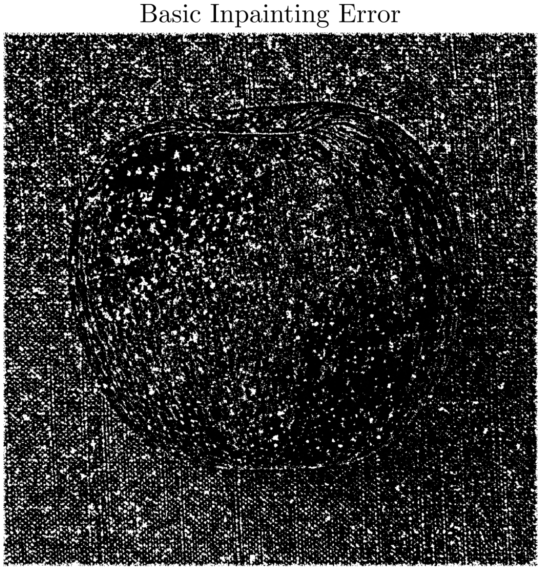
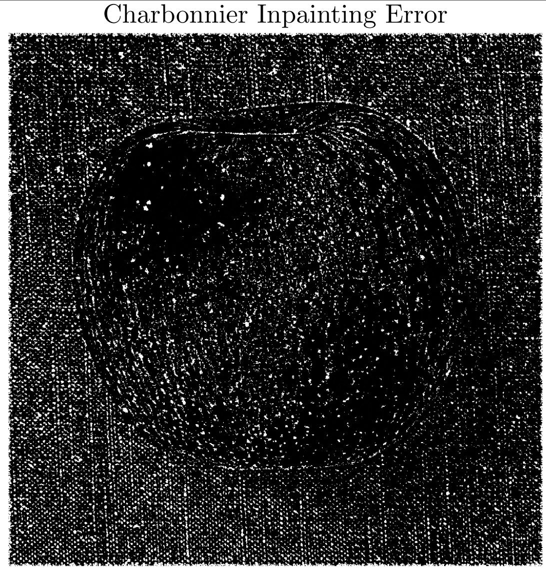


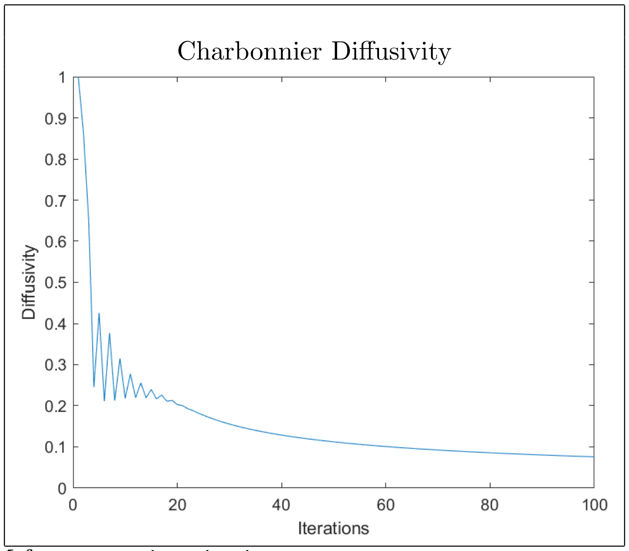
Our SSIM for comparison is given:

Base Inpainting SSIM:	0.9494
Charbonnier SSIM:	0.9505

3.3 80% Sparsification

Using 100 iterations with $\lambda = 0.001$.

Basic Inpainting	Charbonnier Inpainting
	
Basic Inpainting Error	Charbonnier Inpainting Error
	



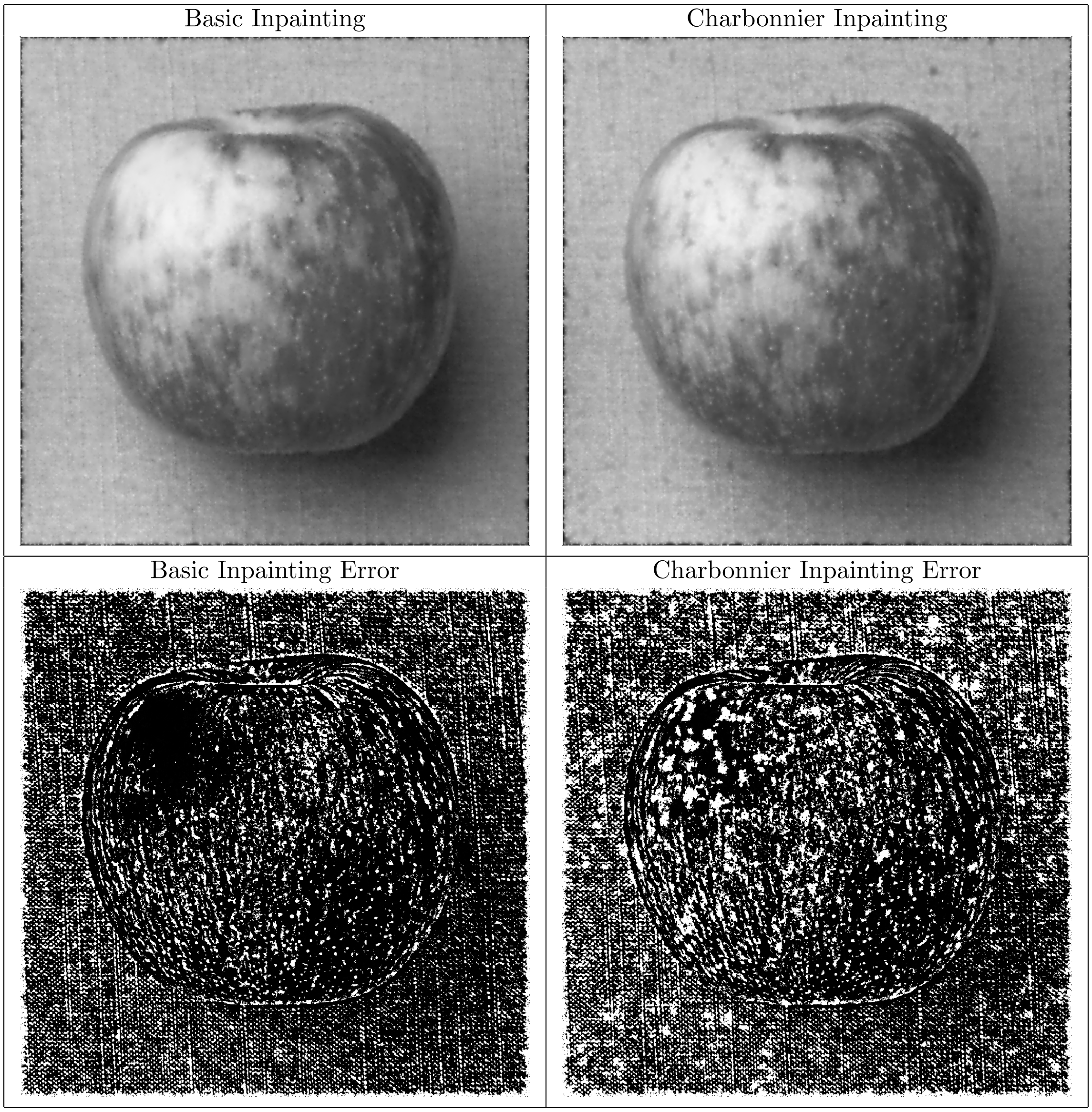
Our SSIM for comparison is given:

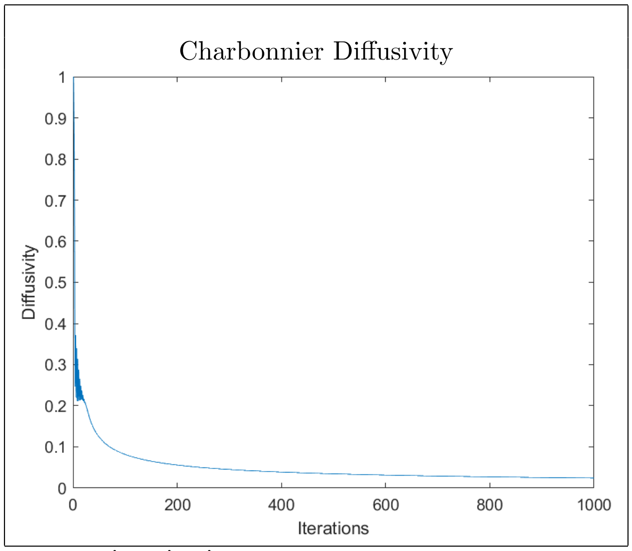
Base Inpainting SSIM:	0.8466
Charbonnier SSIM:	0.8614

3.4 95% Sparsification

Using two different amounts of iterations, with $\lambda = 0.001$.

3.4.1 1000 Iterations

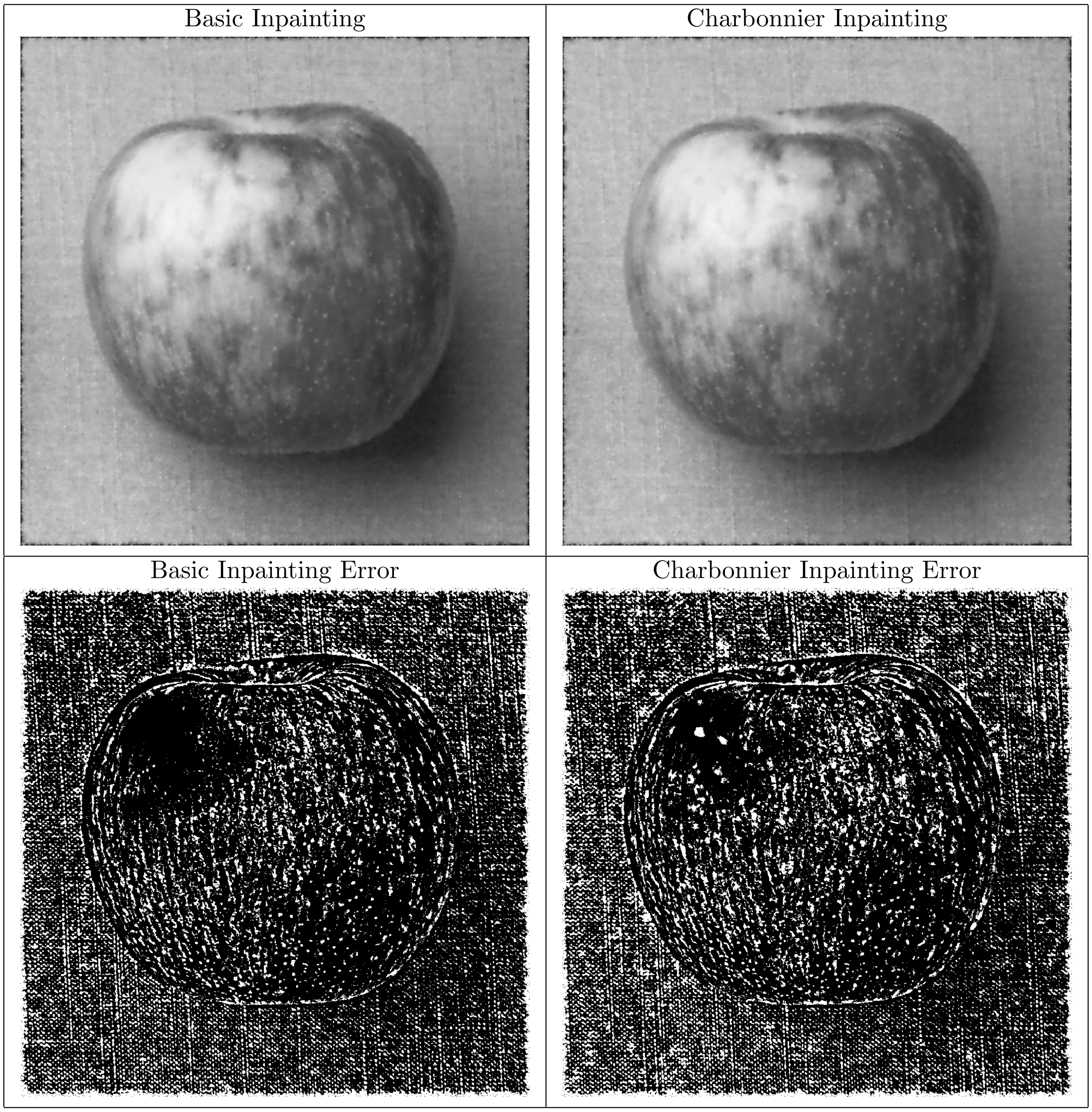


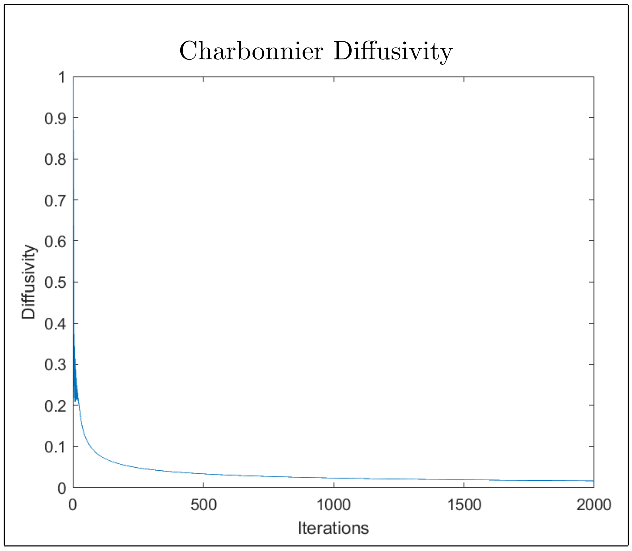


Our SSIM for comparison is given:

Base Inpainting SSIM:	0.7185
Charbonnier SSIM:	0.7129

3.4.2 2000 Iterations





Our SSIM for comparison is given:

Base Inpainting SSIM:	0.7190
Charbonnier SSIM:	0.7183

3.5 Discussion of Results

As seen in the SSIM tables, Charbonnier Diffusion shows a higher SSIM value for both 50% and 80% sparsifications, compared to basic inpainting diffusion over the same number of iterations.

At a sparsification of 95%, we see that basic diffusion has a higher SSIM. The only differing factors between all of our trials is the percent of sparsification and the number of iterations.

As shown in the Charbonnier Diffusivity figures, our diffusivity coefficient is reduced as the number of iterations increase. For trials with a higher percent of sparsification, requiring more iterations, after a certain amount of iterations we see our Charbonnier Diffusivity becomes less than the diffusivity coefficient of our base inpainting method. This implies that after a certain number of iterations, basic diffusion has a faster rate of inpainting, possibly leading to better results.

There may exist a more optimal values for both the number of iterations as well as our value, λ, which plays an integral role in the rate of change for our diffusivity in Charbonnier Diffusivity methods. Overall, more trials utilizing different parameters are required to definitively show the better of two methods, but regardless both methods show promise in their enhancement using MOLE operators.

References

[1] Corbino, J., and Castillo, J.E., High-order mimetic difference operators satisfying the extended Gauss divergence theorem, Journal of Computation and Applied Mathematics, 2020.

[2] Charbonnier, P., Blanc-Féraud, L., Aubert, G., and Barlaud, M., Deterministic edge-preserving regularization in computed imaging. IEEE Transactions on Image Processing, 6(2):298–311, 1997.

[3] Mainberger, M., PDE-Based Image Compression Based on Edges and Optimal Data. Universitat Des Saarlandes, 2014.

[4] Jumakulyyev, I., and Schultz, T., Fourth-Order Anisotropic Diffusion for Inpainting and Image Compression, SpringerLink, Springer International Publishing, 1 Jan. 1970, link.springer.com/chapter/10.1007/978-3-030-56215-1_5.