CS 736 Programming Assignment 2

Image Denoising with MRF Priors

Submitted By

Krishna Wadhwani: 160010031

Vipul Ramtekkar: 16D110013

Contents

1 Objective **2**

2 Code Structure **2**

3 Analysis And Results: Brain MRI Images **2**

3.1 Part A: Original RRMSE Values . . . 2
3.1.1 Low Noise Image . . . 2
3.1.2 Medium Noise Image . . . 2
3.1.3 High Noise Image . . . 2
3.2 Part B: RRMSE values for different parameter values . . . 2
3.2.1 Low Noise Level . . . 2
3.2.2 Medium Noise Level . . . 3
3.2.3 High Noise Level . . . 4
3.3 Part C: Noiseless, Noisy and Denoised Images . . . 5
3.3.1 Low Noise Image . . . 5
3.3.2 Medium Noise Image . . . 7
3.3.3 High Noise Image . . . 10
3.4 Part D: Objective Function Plots . . . 12
3.4.1 Low Noise Image . . . 12
3.4.2 Medium Noise Image . . . 14
3.4.3 High Noise Image . . . 15

4 Analysis And Results: Coloured Digital Pathology Images **16**

4.1 Proposed Strategy . . . 16
4.2 Original RRMSE Values . . . 17
4.2.1 H channel . . . 17
4.2.2 S channel . . . 17
4.2.3 V channel . . . 17
4.3 RRMSE Values for different parameter values . . . 17
4.3.1 H channel . . . 17
4.3.2 S channel . . . 18
4.3.3 V channel . . . 19
4.4 Noiseless, Noisy and Denoised Images . . . 19
4.4.1 H channel . . . 22
4.4.2 S channel . . . 25
4.4.3 V channel . . . 27
4.5 Objective Function Plots . . . 30
4.5.1 H channel . . . 30
4.5.2 S channel . . . 31
4.5.3 V channel . . . 33

1 Objective

The objective of this assignment was to perform image denoising using MRF priors. We used three different MRF priors and fined tuned the hyper parameters to produce the best denoised image. We used gradient descent to minimise the cost function and dynamic step change for faster convergence.

2 Code Structure

We have used python (Version 3.7.4) for coding.

1. python scripts:
   i `functions.py`: Implementation of different MRF priors and RRMSE function calculation
   ii `denoise.py`: ICM Denoising algorithm for finding the MAP estimate of denoised image
   iii `main.py`: Runs the Denoising algorithm and reports and plots all the results for questions 1 and 2
   iv `grid_run.py`: Runs denoising algorithm over a grid of γ and α to find the optimum values of these parameters
2. Other files:
   i `requirements.txt`: Contains the libraries used along with their corresponding versions

   Instructions to run (will generate results for both the questions with respective optimum parameter values):

   `$ python main.py`

3 Analysis And Results: Brain MRI Images

3.1 Part A: Original RRMSE Values

3.1.1 Low Noise Image

Original RRMSE Values between Noiseless and Low Noise Image: 0.0519

3.1.2 Medium Noise Image

Original RRMSE Values between Noiseless and Medium Noise Image: 0.13125

3.1.3 High Noise Image

Original RRMSE Values between Noiseless and High Noise Image: 0.15553

3.2 Part B: RRMSE values for different parameter values

3.2.1 Low Noise Level

Quadratic Prior: Parameters: α

1. $\alpha^* = 0.0875$

2. RRMSE(α^*) = 0.04673
3. RRMSE($1.2\alpha^*$) = 0.04689
4. RRMSE($0.8\alpha^*$) = 0.0469

Discontinuity Adaptive Huber Prior: Parameters: α, γ

1. $\alpha^* = 0.8748, \gamma^* = 0.00211$
2. RRMSE(α^*, γ^*) = 0.04295
3. RRMSE($1.2\alpha^*, \gamma^*$) = 0.16522
4. RRMSE($0.8\alpha^*, \gamma^*$) = 0.14732
5. RRMSE($\alpha^*, 1.2\gamma^*$) = 0.13232
6. RRMSE($\alpha^*, 0.8\gamma^*$) = 0.13818

Discontinuity Adaptive Prior: Parameters: α, γ

1. $\alpha^* = 0.8748, \gamma^* = 0.0023$
2. RRMSE(α^*, γ^*) = 0.04300
3. RRMSE($1.2\alpha^*, \gamma^*$) = 0.15544
4. RRMSE($0.8\alpha^*, \gamma^*$) = 0.14697
5. RRMSE($\alpha^*, 1.2\gamma^*$) = 0.13188
6. RRMSE($\alpha^*, 0.8\gamma^*$) = 0.13749

3.2.2 Medium Noise Level

Quadratic Prior: Parameters: α

1. $\alpha^* = 0.1773$
2. RRMSE(α^*) = 0.11610
3. RRMSE($1.2\alpha^*$) = 0.11644
4. RRMSE($0.8\alpha^*$) = 0.11649

Discontinuity Adaptive Huber Prior: Parameters: α, γ

1. $\alpha^* = 0.8663, \gamma^* = 0.0048$
2. RRMSE(α^*, γ^*) = 0.11186
3. RRMSE($1.2\alpha^*, \gamma^*$) = 0.407
4. RRMSE($0.8\alpha^*, \gamma^*$) = 0.1393
5. RRMSE($\alpha^*, 1.2\gamma^*$) = 0.12315

6. RRMSE$(\alpha^*, 0.8\gamma^*) = 0.12736$

Discontinuity Adaptive Prior: Parameters: α, γ

1. $\alpha^* = 0.8862, \gamma^* = 0.00438$
2. RRMSE$(\alpha^*, \gamma^*) = 0.11200$
3. RRMSE$(1.2\alpha^*, \gamma^*) = 0.17137$
4. RRMSE$(0.8\alpha^*, \gamma^*) = 0.14047$
5. RRMSE$(\alpha^*, 1.2\gamma^*) = 0.12336$
6. RRMSE$(\alpha^*, 0.8\gamma^*) = 0.12716$

3.2.3 High Noise Level

Quadratic Prior: Parameters: α

1. $\alpha^* = 0.2407$
2. RRMSE$(\alpha^*) = 0.12708$
3. RRMSE$(1.2\alpha^*) = 0.12767$
4. RRMSE$(0.8\alpha^*) = 0.12775$

Discontinuity Adaptive Huber Prior: Parameters: α, γ

1. $\alpha^* = 0.7729, \gamma^* = 0.01389$
2. RRMSE$(\alpha^*, \gamma^*) = 0.12247$
3. RRMSE$(1.2\alpha^*, \gamma^*) = 0.15104$
4. RRMSE$(0.8\alpha^*, \gamma^*) = 0.12897$
5. RRMSE$(\alpha^*, 1.2\gamma^*) = 0.12339$
6. RRMSE$(\alpha^*, 0.8\gamma^*) = 0.12314$

Discontinuity Adaptive Prior: Parameters: α, γ

1. $\alpha^* = 0.9449, \gamma^* = 0.003$
2. RRMSE$(\alpha^*, \gamma^*) = 0.12238$
3. RRMSE$(1.2\alpha^*, \gamma^*) = 0.14183$
4. RRMSE$(0.8\alpha^*, \gamma^*) = 0.12775$
5. RRMSE$(\alpha^*, 1.2\gamma^*) = 0.12336$
6. RRMSE$(\alpha^*, 0.8\gamma^*) = 0.1229$

3.3 Part C: Noiseless, Noisy and Denoised Images

3.3.1 Low Noise Image

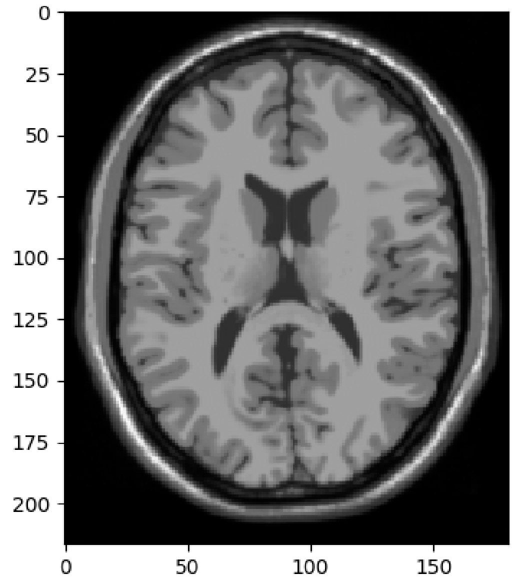

Figure 1: Ground Truth Noiseless Image

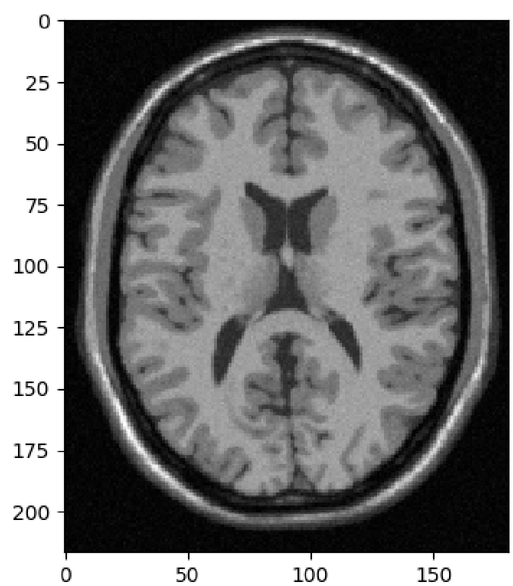

Figure 2: Noisy Image

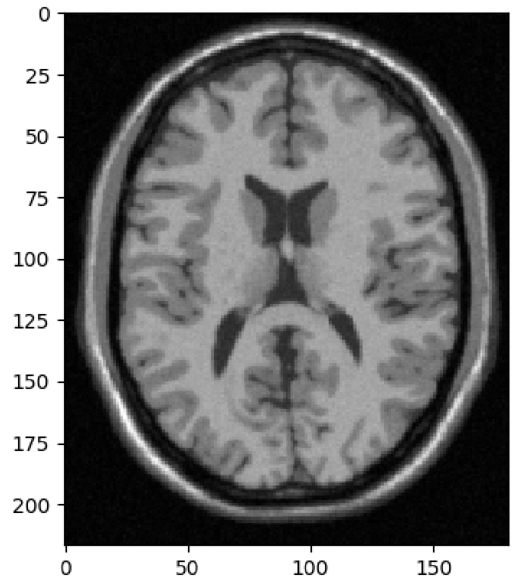

Figure 3: Image Denoised Using Quadratic Prior

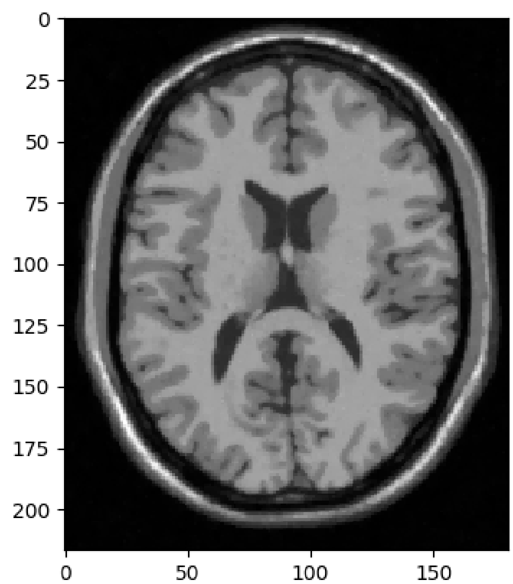

Figure 4: Image Denoised Using Discontinuity Adaptive Huber Prior

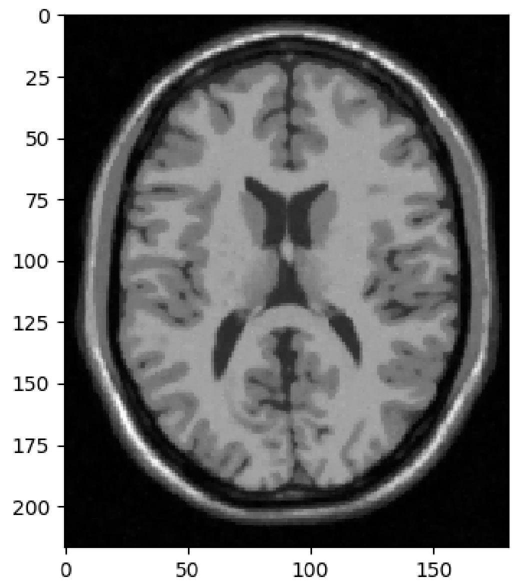

Figure 5: Image Denoised Using Discontinuity Adaptive Prior

3.3.2 Medium Noise Image

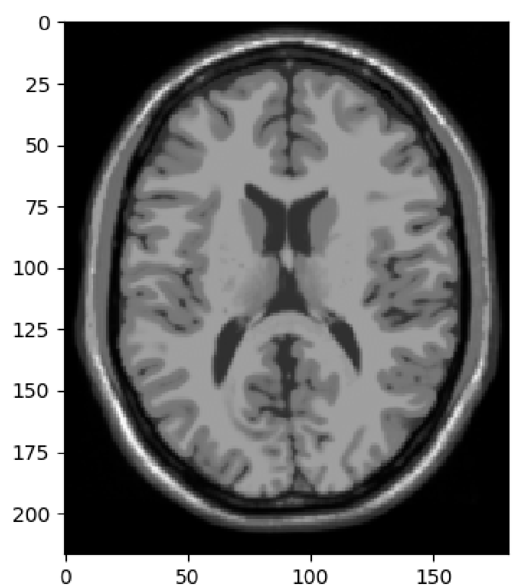

Figure 6: Ground Truth Noiseless Image

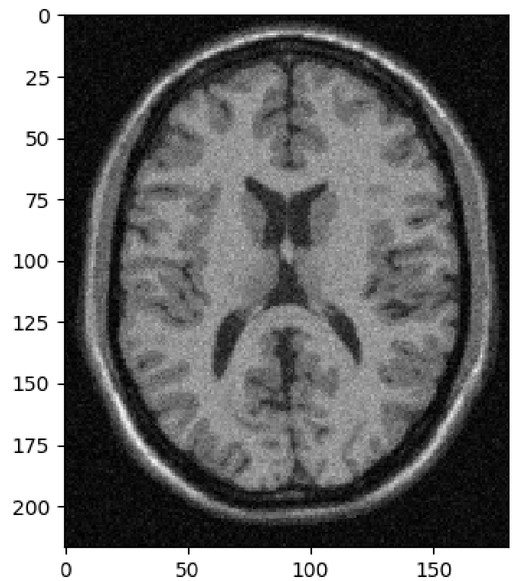

Figure 7: Noisy Image

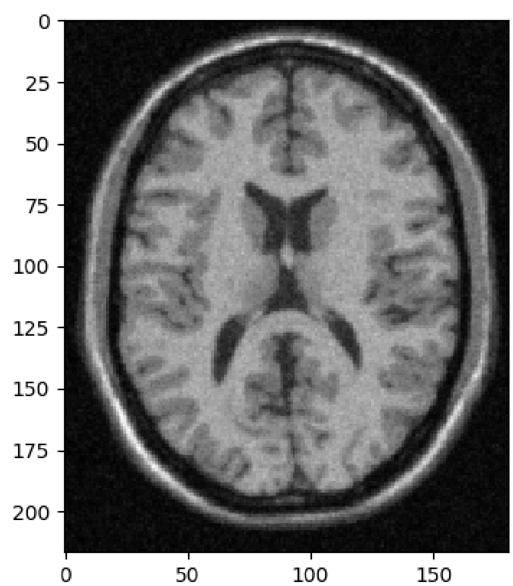

Figure 8: Image Denoised Using Quadratic Prior

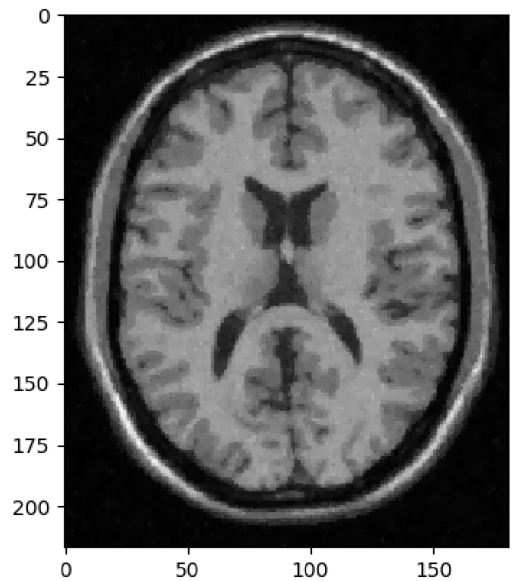

Figure 9: Image Denoised Using Discontinuity Adaptive Huber Prior

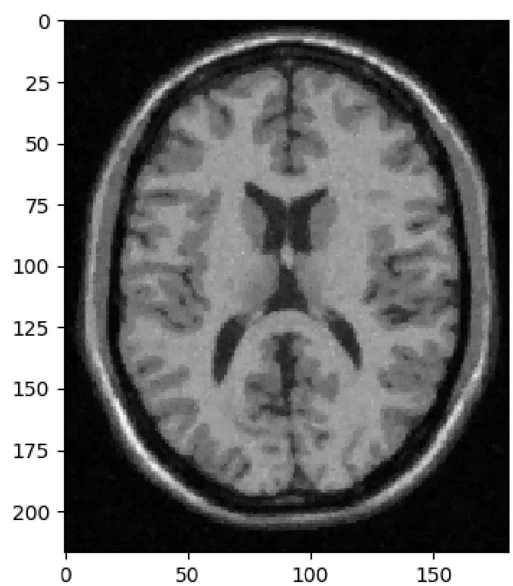

Figure 10: Image Denoised Using Discontinuity Adaptive Prior

3.3.3 High Noise Image

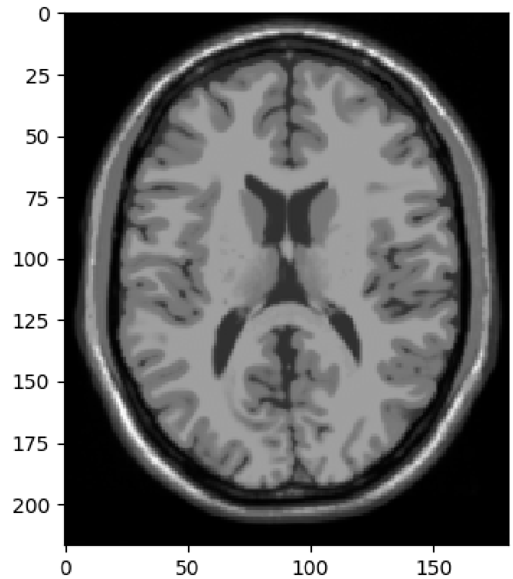

Figure 11: Ground Truth Noiseless Image

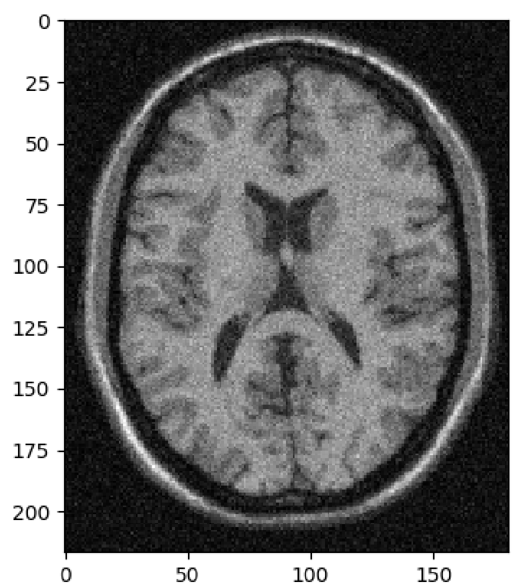

Figure 12: Noisy Image

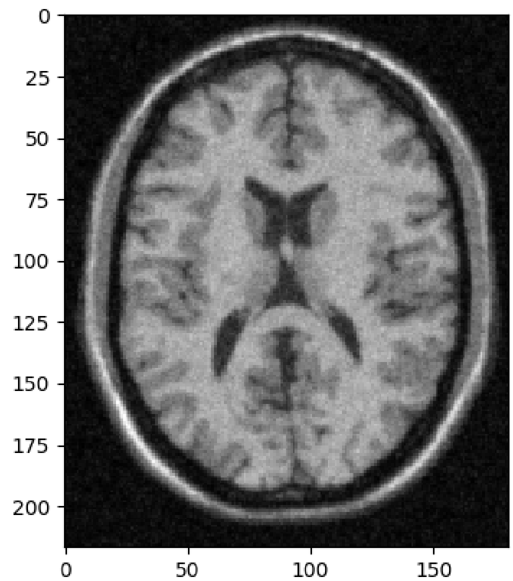

Figure 13: Image Denoised Using Quadratic Prior

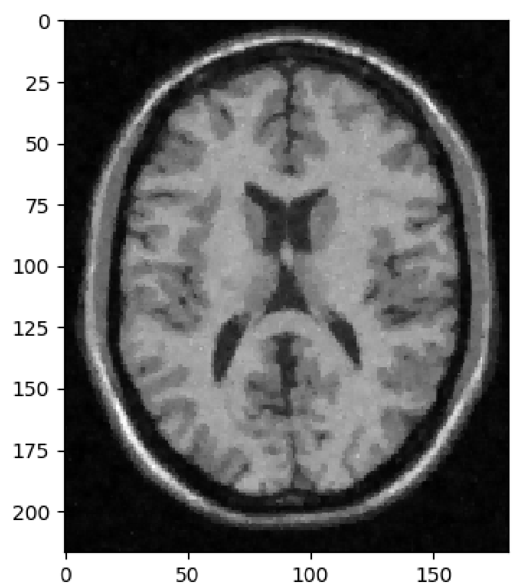

Figure 14: Image Denoised Using Discontinuity Adaptive Huber Prior

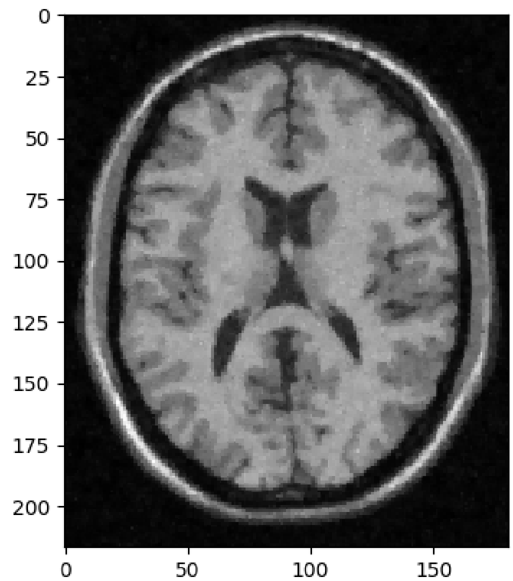

Figure 15: Image Denoised Using Discontinuity Adaptive Prior

3.4 Part D: Objective Function Plots

3.4.1 Low Noise Image

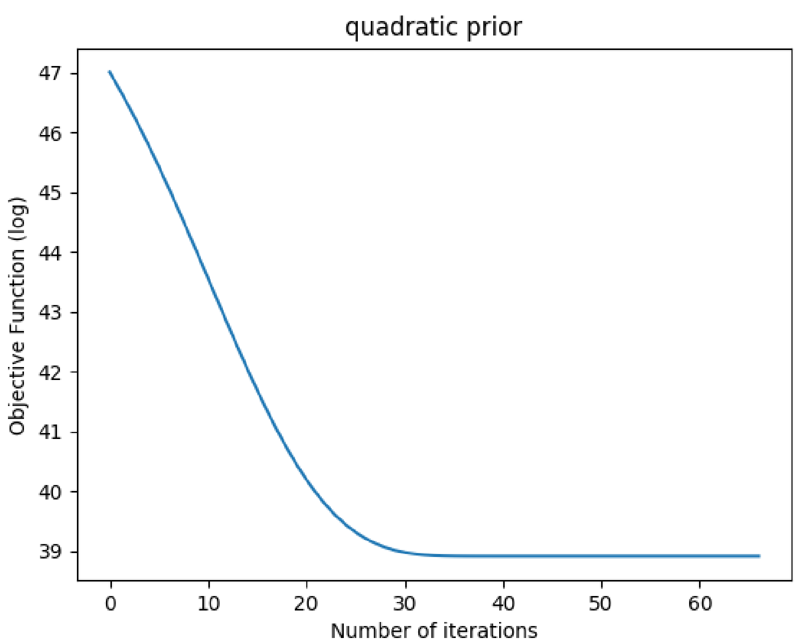

Figure 16: Objective function variation: Low Noise Image, Quadratic Prior

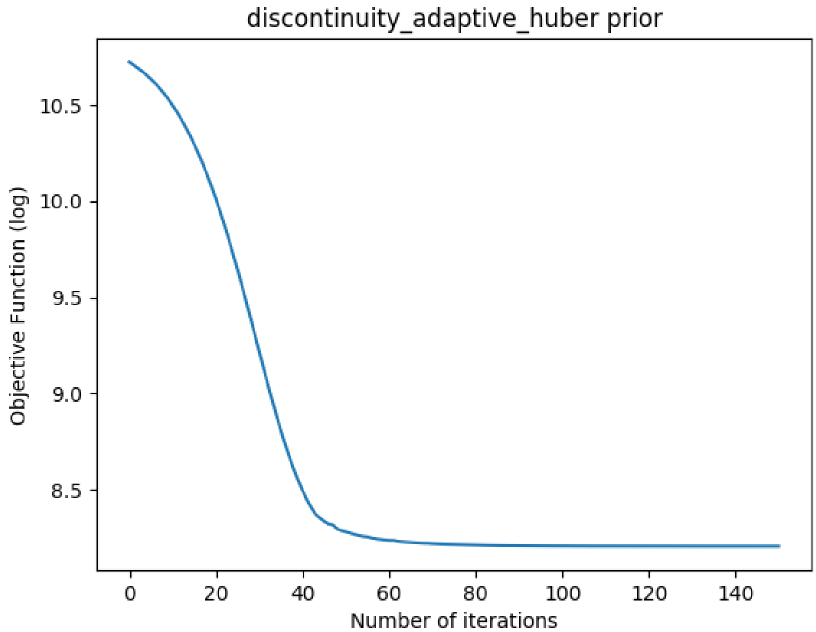

Figure 17: Objective function variation: Low Noise Image, Discontinuity Adaptive Huber Prior

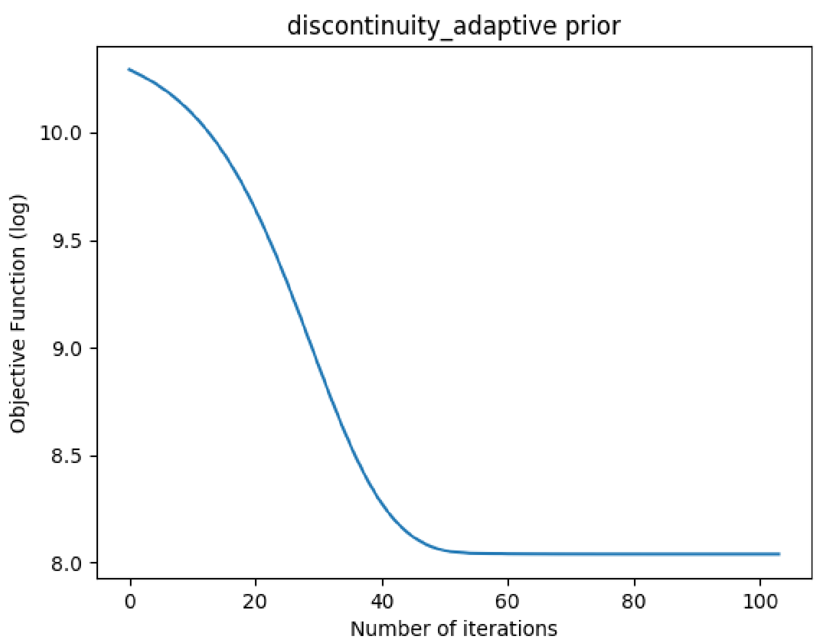

Figure 18: Objective function variation: Low Noise Image, Discontinuity Adaptive Huber Prior

3.4.2 Medium Noise Image

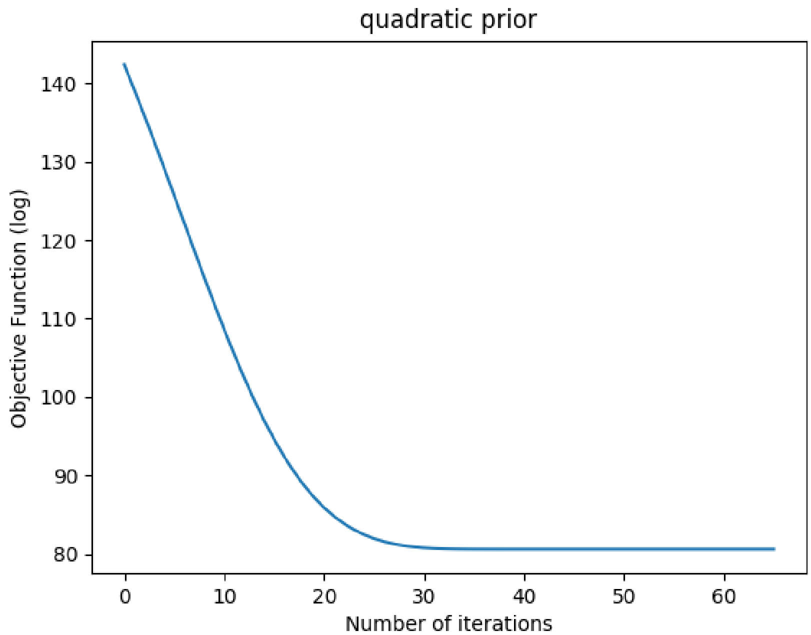

Figure 19: Objective function variation: Medium Noise Image, Quadratic Prior

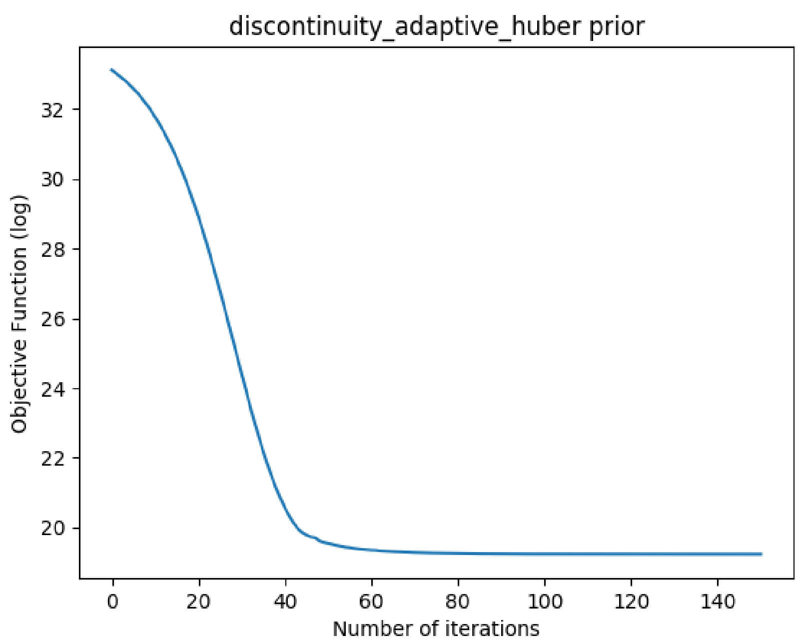

Figure 20: Objective function variation: Medium Noise Image, Discontinuity Adaptive Huber Prior

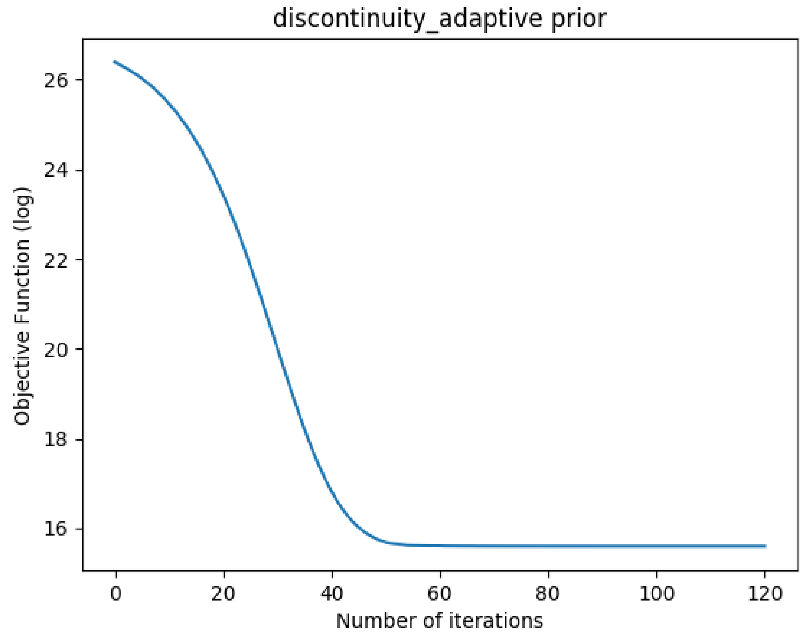

Figure 21: Objective function variation: Medium Noise Image, Discontinuity Adaptive Huber Prior

3.4.3 High Noise Image

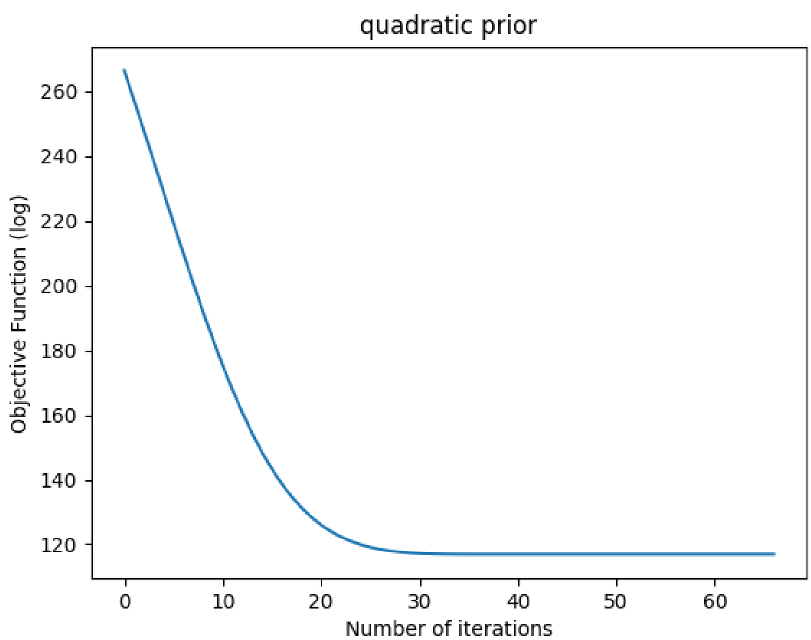

Figure 22: Objective function variation: High Noise Image, Quadratic Prior

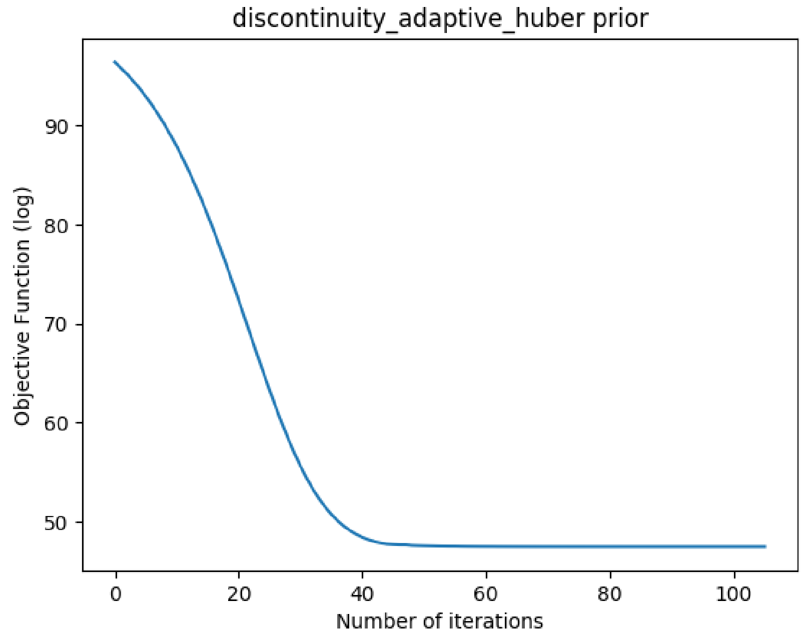

Figure 23: Objective function variation: High Noise Image, Discontinuity Adaptive Huber Prior

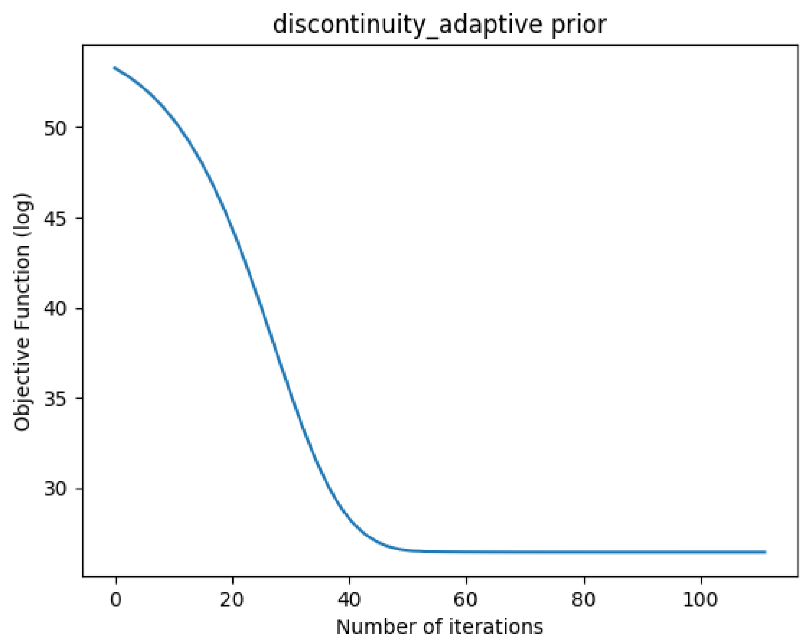

Figure 24: Objective function variation: High Noise Image, Discontinuity Adaptive Huber Prior

4 Analysis And Results: Coloured Digital Pathology Images

4.1 Proposed Strategy

We initially tried to do channel-wise denoising in the RGB space but the results were not that good which we believe is due to us trying to independently optimize different channels which have high correlation between them.

So we converted our image to HSV color-space. Although the three channels are not strictly independent in HSV, there is only one channel that controls the color. We independently find the optimal parameter values for each of channel of the image and finally combine the three denoised channels to get our final image.

4.2 Original RRMSE Values

For the complete image, Original RRMSE: 0.28595

4.2.1 H channel

Original RRMSE value for H channel: 0.28595

4.2.2 S channel

Original RRMSE value for S channel: 0.44423

4.2.3 V channel

Original RRMSE value for V channel: 0.17661

4.3 RRMSE Values for different parameter values

RRMSE for Quadratic prior denoised Image = 0.094847 RRMSE for Discontinuity Adaptive Huber prior denoised Image = 0.22235 RRMSE for Discontinuity Adaptive prior denoised Image = 0.09598

4.3.1 H channel

Quadratic Prior: Parameters: α

1. $\alpha^* = 0.9919$
2. RRMSE$(\alpha^*) = 0.09485$
3. RRMSE$(1.2\alpha^*) = 0.12077$
4. RRMSE$(0.8\alpha^*) = 0.1043$

Discontinuity Adaptive Huber Prior: Parameters: α, γ

1. $\alpha^* = 1, \gamma^* = 1$
2. RRMSE$(\alpha^*, \gamma^*) = 0.09714$
3. RRMSE$(1.2\alpha^*, \gamma^*) = 0.22235$
4. RRMSE$(0.8\alpha^*, \gamma^*) = 0.27431$
5. RRMSE$(\alpha^*, 1.2\gamma^*) = 0.09651$
6. RRMSE$(\alpha^*, 0.8\gamma^*) = 0.09776$

Discontinuity Adaptive Prior: Parameters: α, γ

1. $\alpha^* = 1, \gamma^* = 1$

2. RRMSE(α^*, γ^*) = 0.09599

3. RRMSE($1.2\alpha^*, \gamma^*$) = 0.22274

4. RRMSE($0.8\alpha^*, \gamma^*$) = 0.27446

5. RRMSE($\alpha^*, 1.2\gamma^*$) = 0.09615

6. RRMSE($\alpha^*, 0.8\gamma^*$) = 0.09647

4.3.2 S channel

Quadratic Prior: Parameters: α

1. $\alpha^* = 0.75275$

2. RRMSE(α^*) = 0.20901

3. RRMSE($1.2\alpha^*$) = 0.21847

4. RRMSE($0.8\alpha^*$) = 0.21425

Discontinuity Adaptive Huber Prior: Parameters: α, γ

1. $\alpha^* = 0.8974, \gamma^* = 0.02662$

2. RRMSE(α^*, γ^*) = 0.20841

3. RRMSE($1.2\alpha^*, \gamma^*$) = 0.21497

4. RRMSE($0.8\alpha^*, \gamma^*$) = 0.29182

5. RRMSE($\alpha^*, 1.2\gamma^*$) = 0.2088

6. RRMSE($\alpha^*, 0.8\gamma^*$) = 0.2098

Discontinuity Adaptive Prior: Parameters: α, γ

1. $\alpha^* = 0.93878, \gamma^* = 0.02139$

2. RRMSE(α^*, γ^*) = 0.20846

3. RRMSE($1.2\alpha^*, \gamma^*$) = 0.22976

4. RRMSE($0.8\alpha^*, \gamma^*$) = 0.31367

5. RRMSE($\alpha^*, 1.2\gamma^*$) = 0.20862

6. RRMSE($\alpha^*, 0.8\gamma^*$) = 0.20929

4.3.3 V channel

Quadratic Prior: Parameters: α

1. $\alpha^* = 0.7137$
2. $\text{RRMSE}(\alpha^*) = 0.09516$
3. $\text{RRMSE}(1.2\alpha^*) = 0.09796$
4. $\text{RRMSE}(0.8\alpha^*) = 0.09689$

Discontinuity Adaptive Huber Prior: Parameters: α, γ

1. $\alpha^* = 0.8718, \gamma^* = 0.02662$
2. $\text{RRMSE}(\alpha^*, \gamma^*) = 0.09341$
3. $\text{RRMSE}(1.2\alpha^*, \gamma^*) = 0.09463$
4. $\text{RRMSE}(0.8\alpha^*, \gamma^*) = 0.11408$
5. $\text{RRMSE}(\alpha^*, 1.2\gamma^*) = 0.0948$
6. $\text{RRMSE}(\alpha^*, 0.8\gamma^*) = 0.09495$

Discontinuity Adaptive Prior: Parameters: α, γ

1. $\alpha^* = 0.9184, \gamma^* = 0.02138$
2. $\text{RRMSE}(\alpha^*, \gamma^*) = 0.09473$
3. $\text{RRMSE}(1.2\alpha^*, \gamma^*) = 0.09693$
4. $\text{RRMSE}(0.8\alpha^*, \gamma^*) = 0.1233$
5. $\text{RRMSE}(\alpha^*, 1.2\gamma^*) = 0.09468$
6. $\text{RRMSE}(\alpha^*, 0.8\gamma^*) = 0.0952$

4.4 Noiseless, Noisy and Denoised Images

Note: The images were converted to HSV format for denoising and were saved as is and that is why they look different from the original images. The denoising is much more evident in HSV format and the denoised image looks to have slighly different colors when converted back to RGB space, so that is why the images have not been converted back to RGB space for saving and have been saved as is in the HSV space.

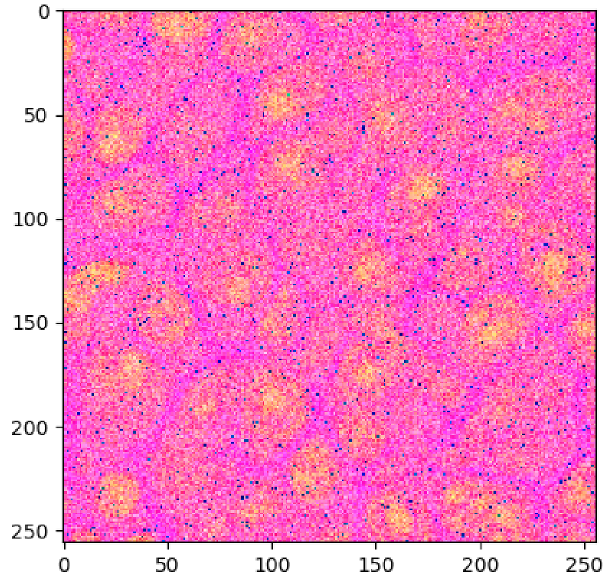

Figure 25: Original Noisy Image.

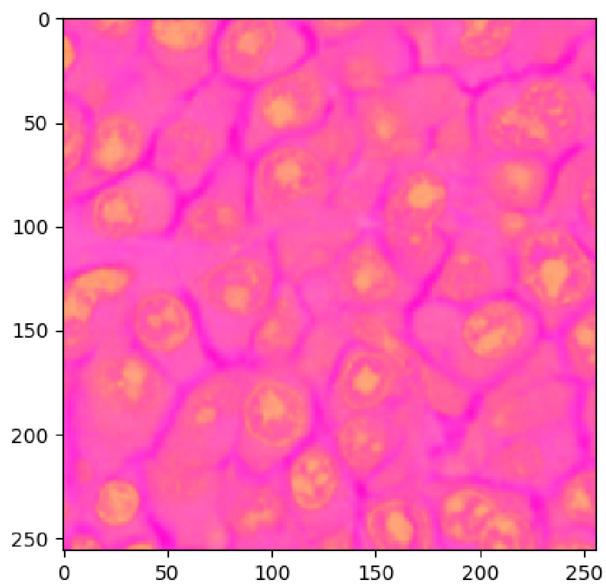

Figure 26: Original Ground truth noiseless Image.

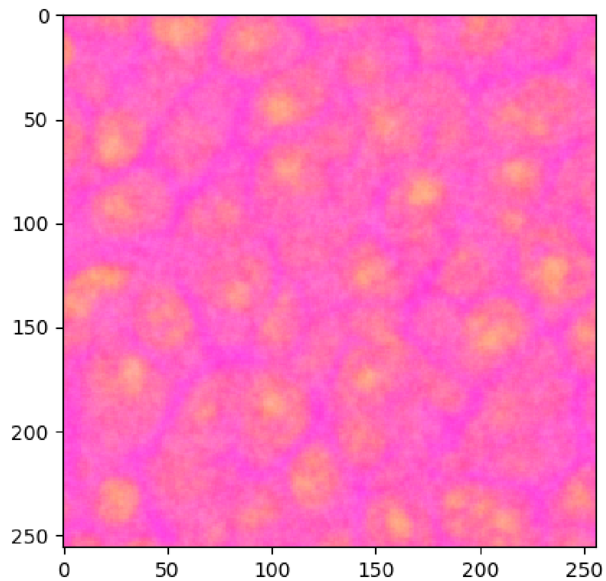

Figure 27: Quadratic Prior Denoised Image

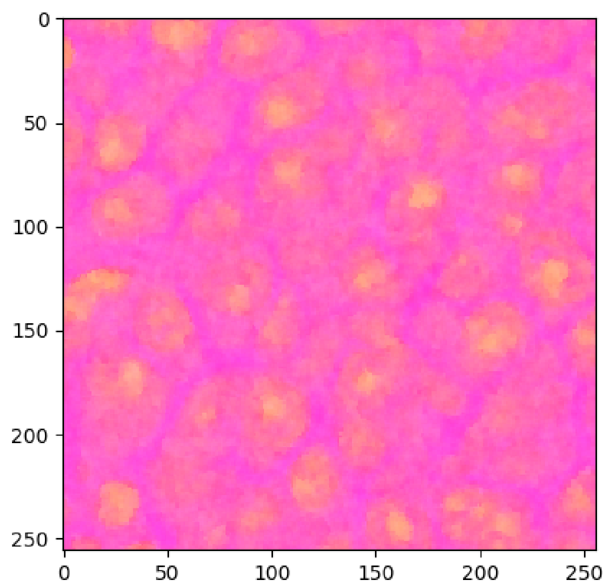

Figure 28: Discontinuity Adaptive Huber Prior Denoised Image.

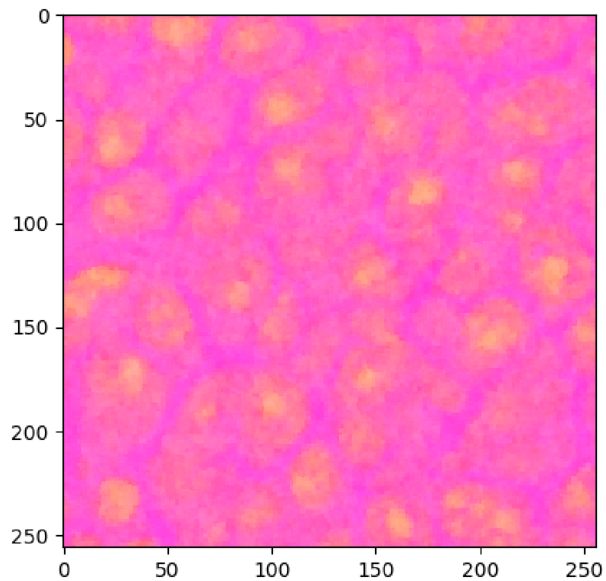

Figure 29: Discontinuity Adaptive Prior Denoised Image.

4.4.1 H channel

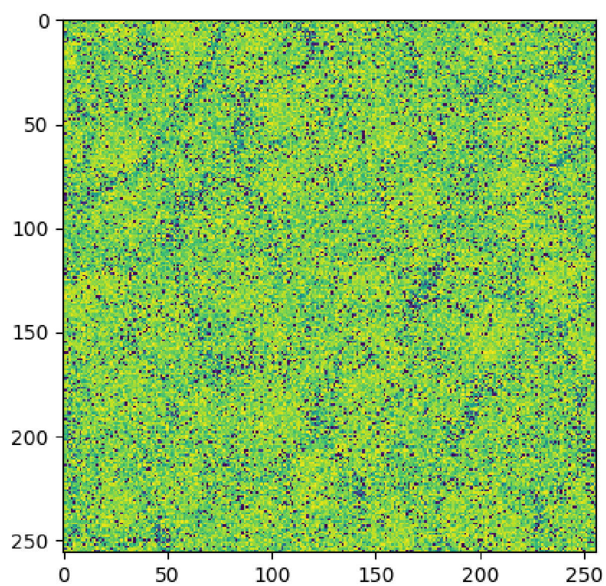

Figure 30: Noisy H channel

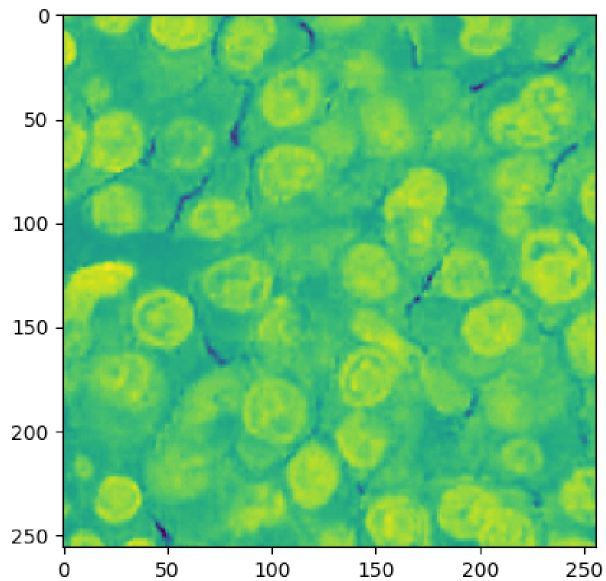

Figure 31: Noiseless H channel

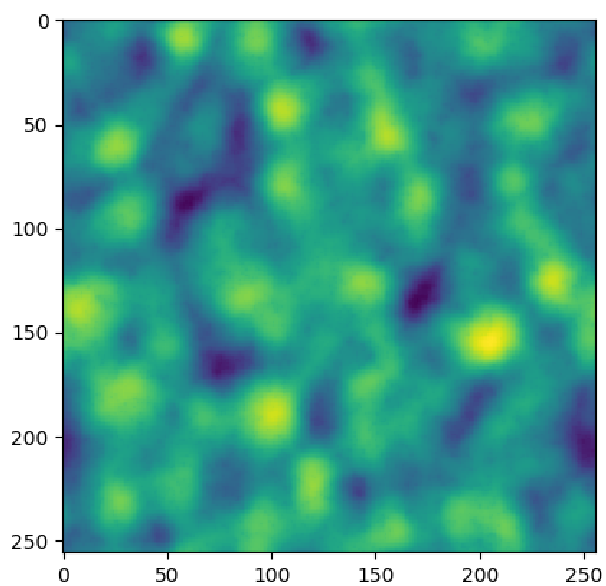

Figure 32: Quadratic Denoised H channel

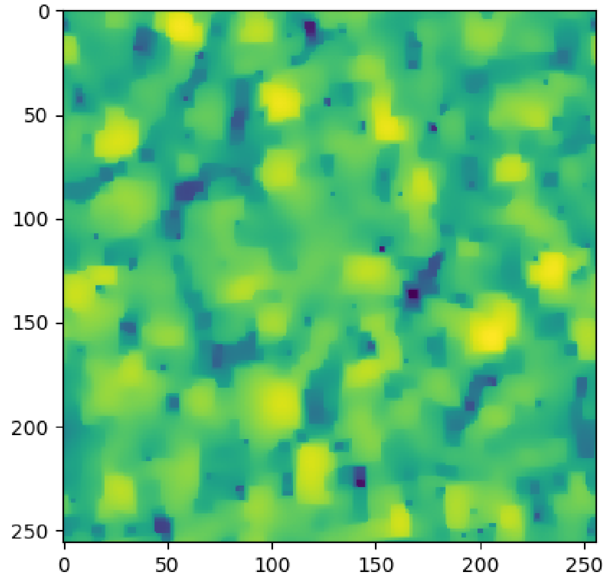

Figure 33: Discontinuity Adaptive Huber Denoised H channel

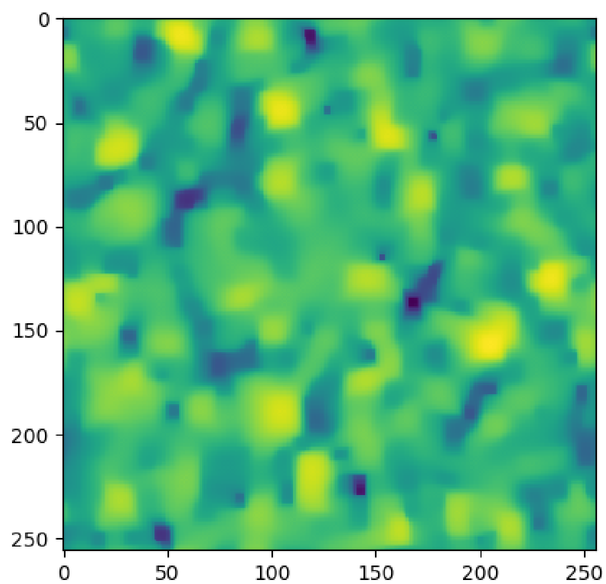

Figure 34: Discontinuity Adaptive H channel

4.4.2 S channel

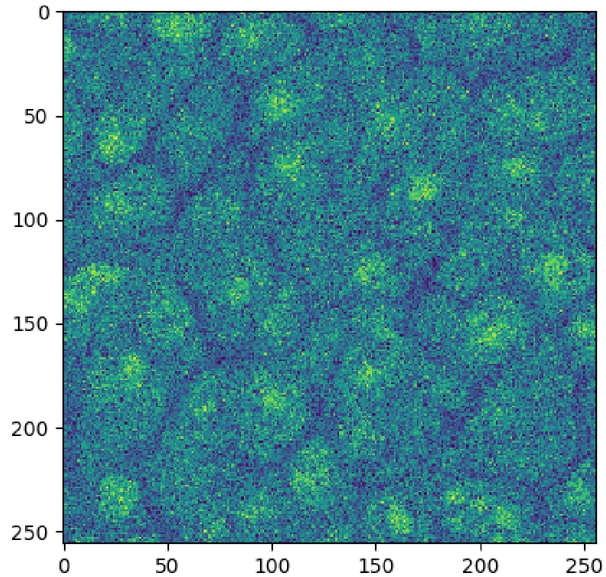

Figure 35: Noisy S channel

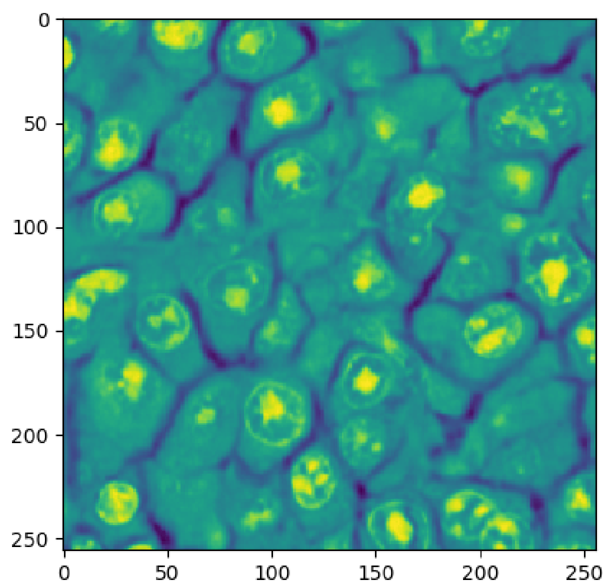

Figure 36: Noiseless S channel

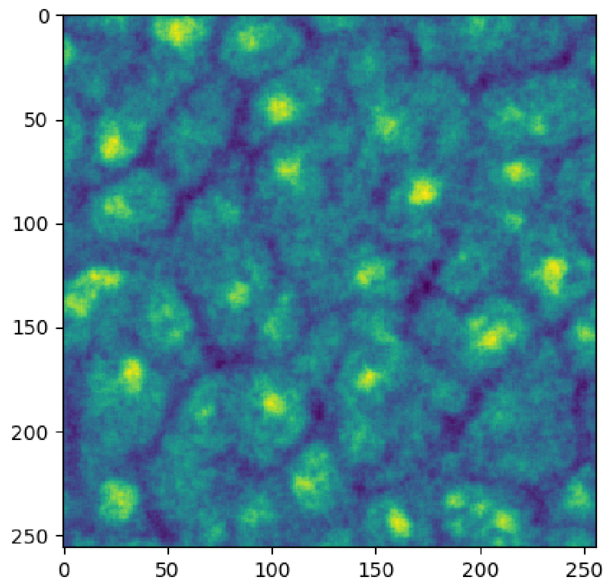

Figure 37: Quadratic Denoised S channel

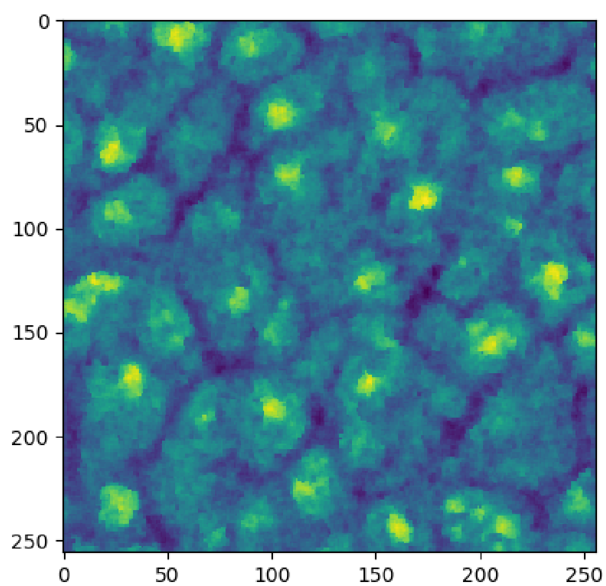

Figure 38: Discontinuity Adaptive Huber Denoised S channel

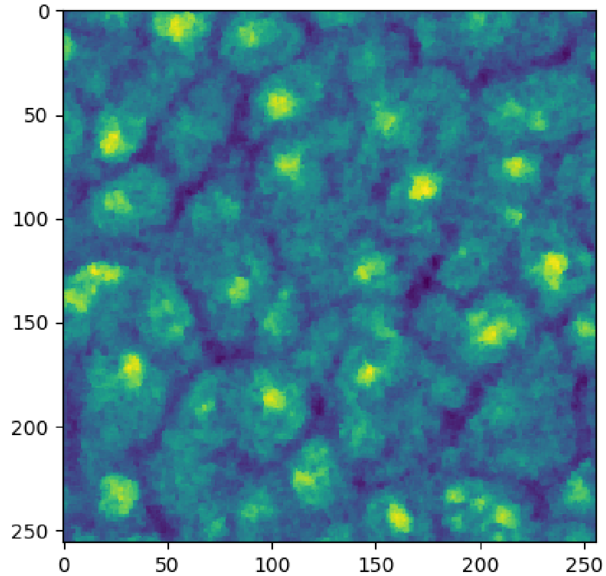

Figure 39: Discontinuity Adaptive S channel

4.4.3 V channel

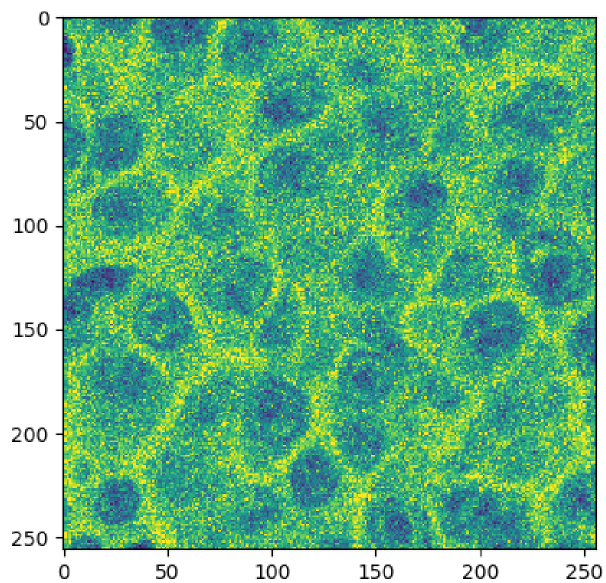

Figure 40: Noisy V channel

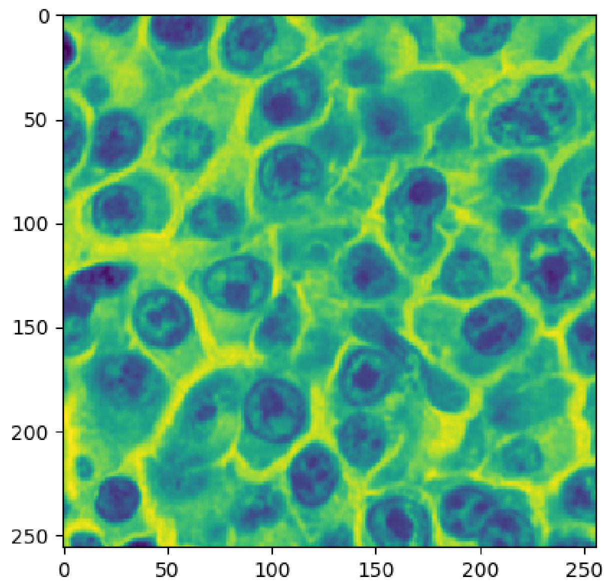

Figure 41: Noiseless V channel

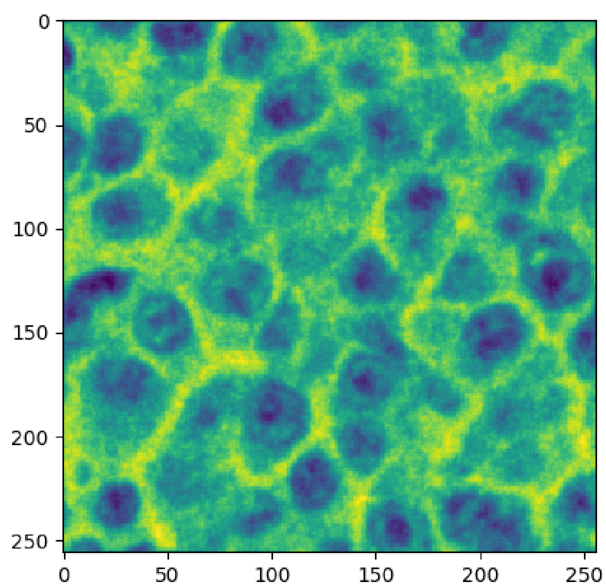

Figure 42: Quadratic Denoised V channel

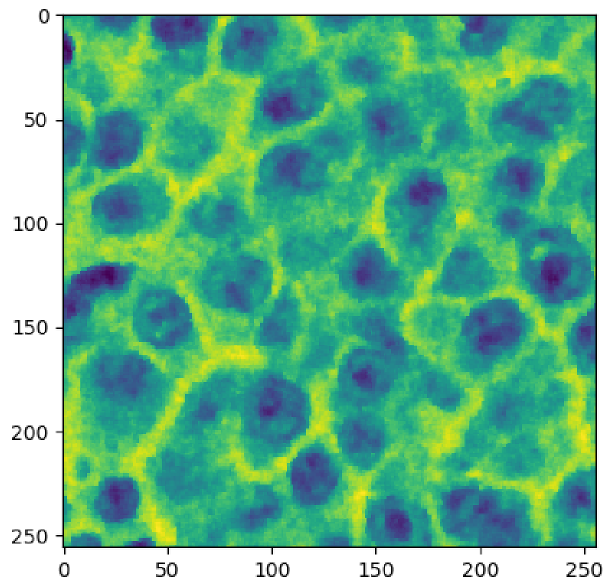

Figure 43: Discontinuity Adaptive Huber Denoised V channel

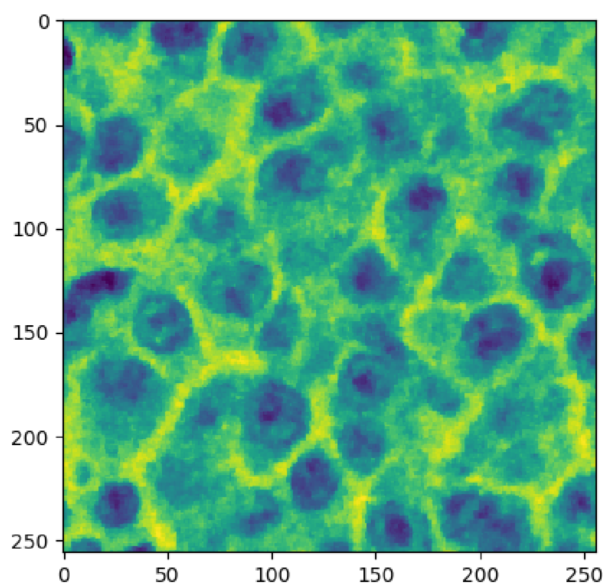

Figure 44: Discontinuity Adaptive V channel

4.5 Objective Function Plots

4.5.1 H channel

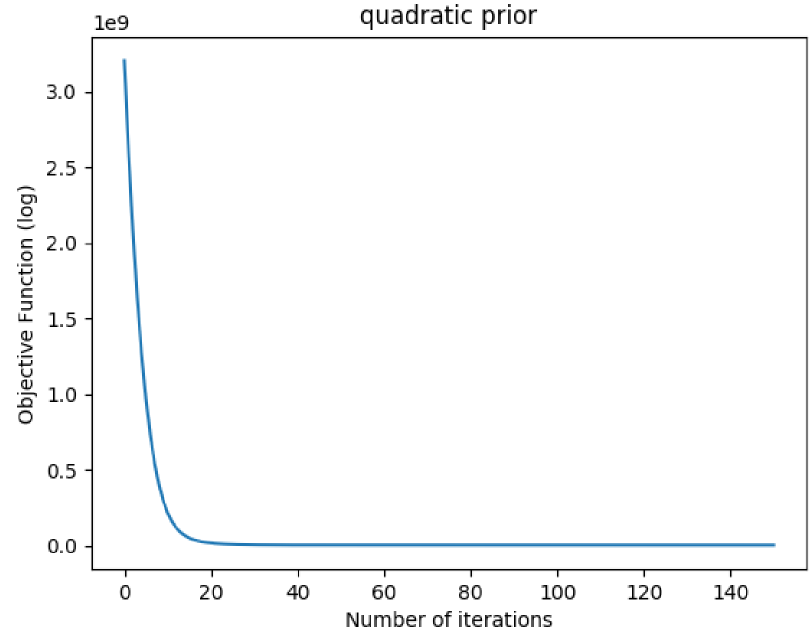

Figure 45: Objective Function Variation: H channel, Quadratic Prior

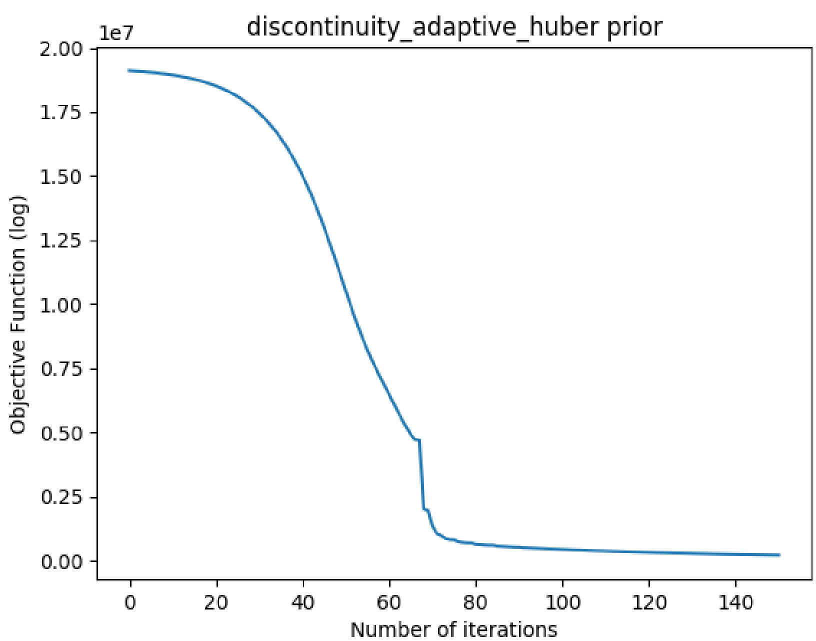

Figure 46: Objective Function Variation: H channel, Discontinuity Adaptive Huber Prior

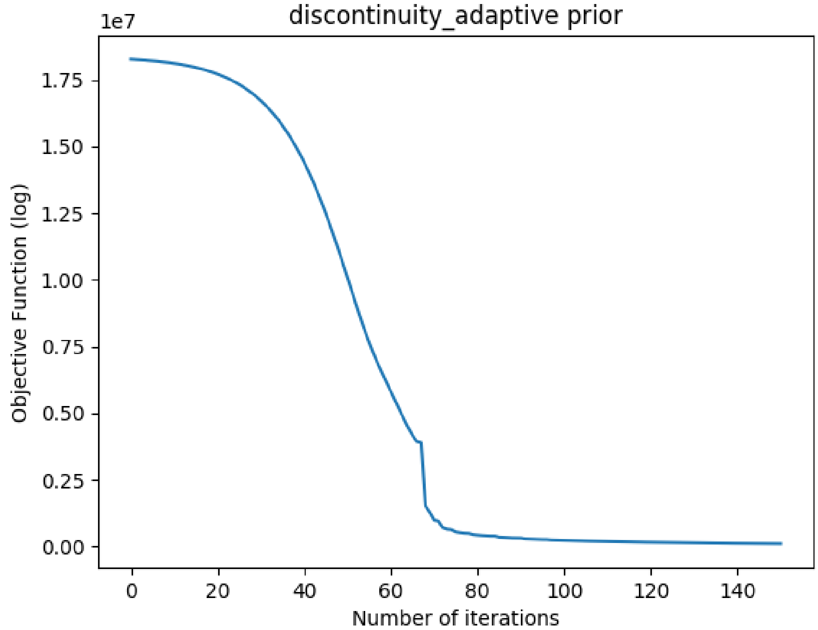

Figure 47: Objective Function Variation: H channel, Discontinuity Adaptive Prior

4.5.2 S channel

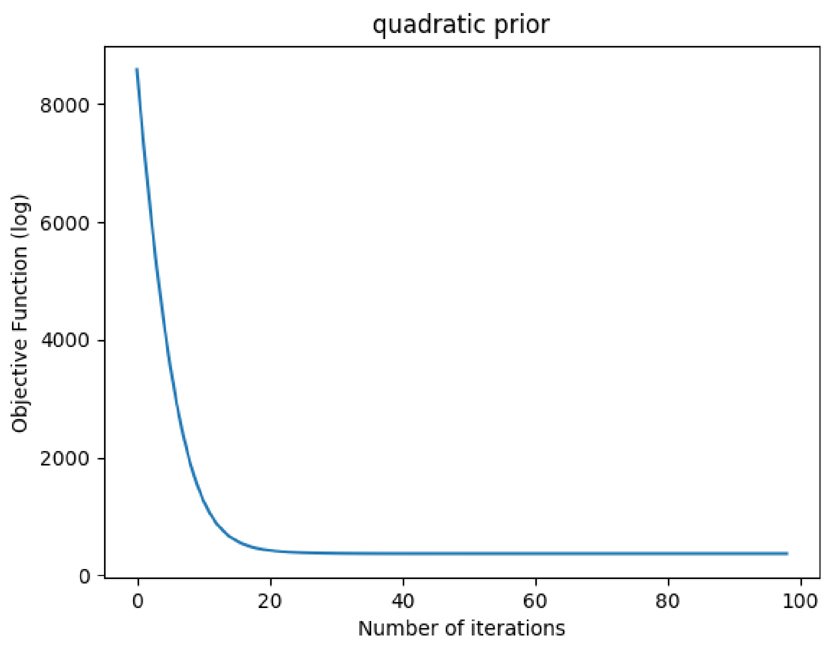

Figure 48: Objective Function Variation: S channel, Quadratic Prior

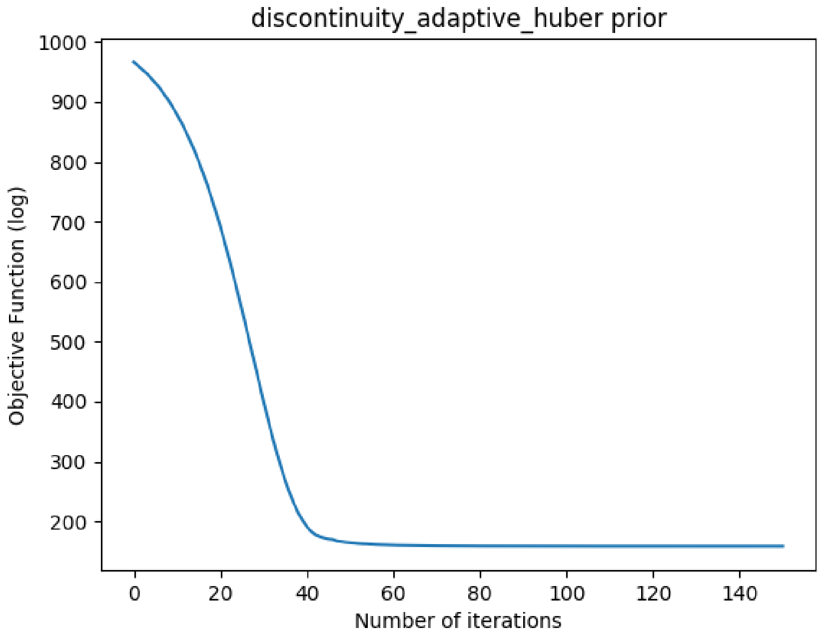

Figure 49: Objective Function Variation: S channel, Discontinuity Adaptive Huber Prior

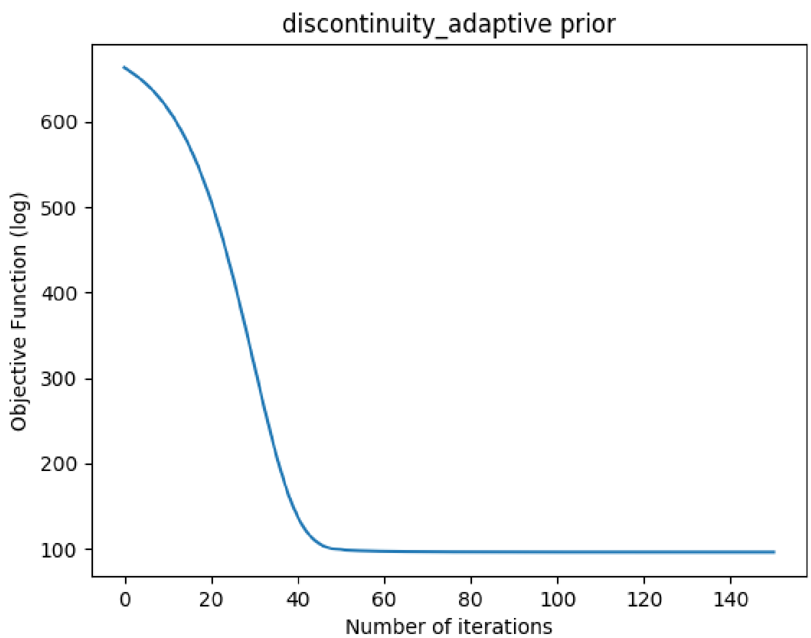

Figure 50: Objective Function Variation: S channel, Discontinuity Adaptive Prior

4.5.3 V channel

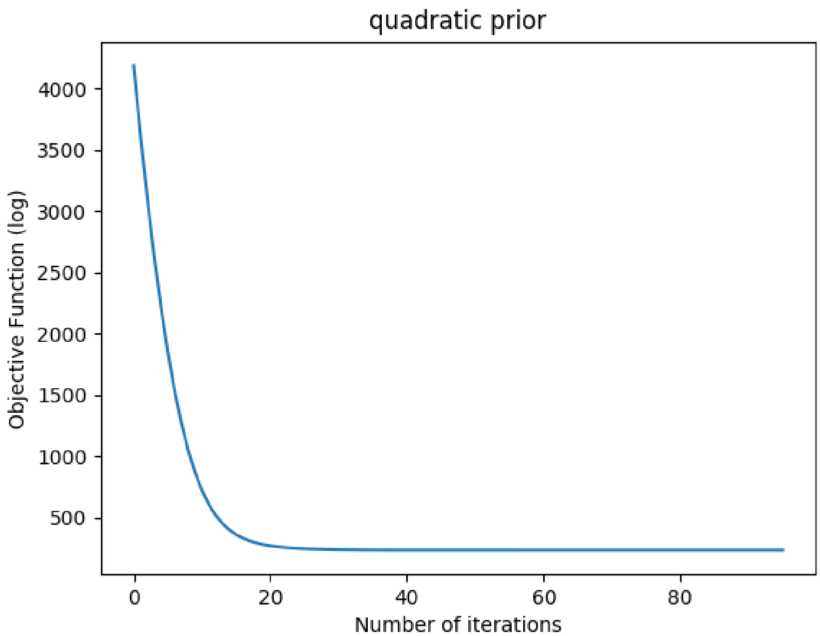

Figure 51: Objective Function Variation: V channel, Quadratic Prior

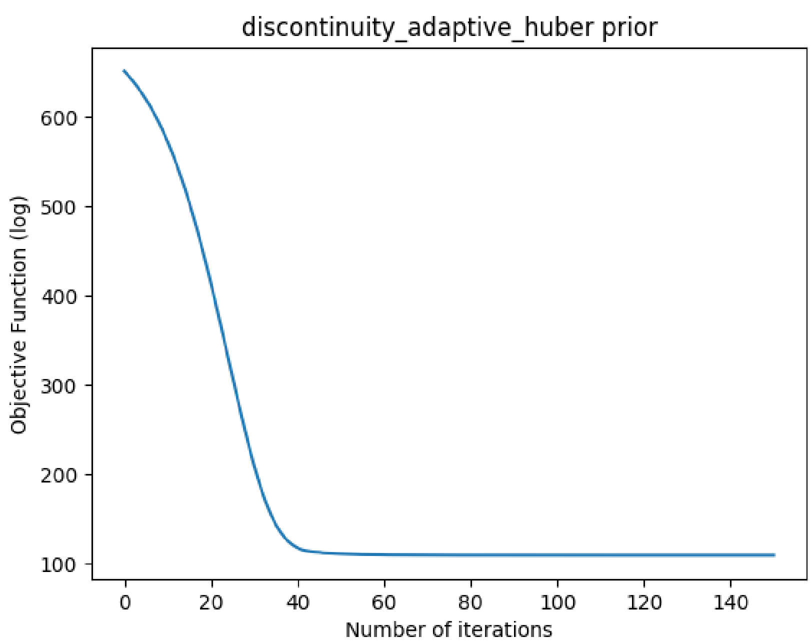

Figure 52: Objective Function Variation: V channel, Discontinuity Adaptive Huber Prior

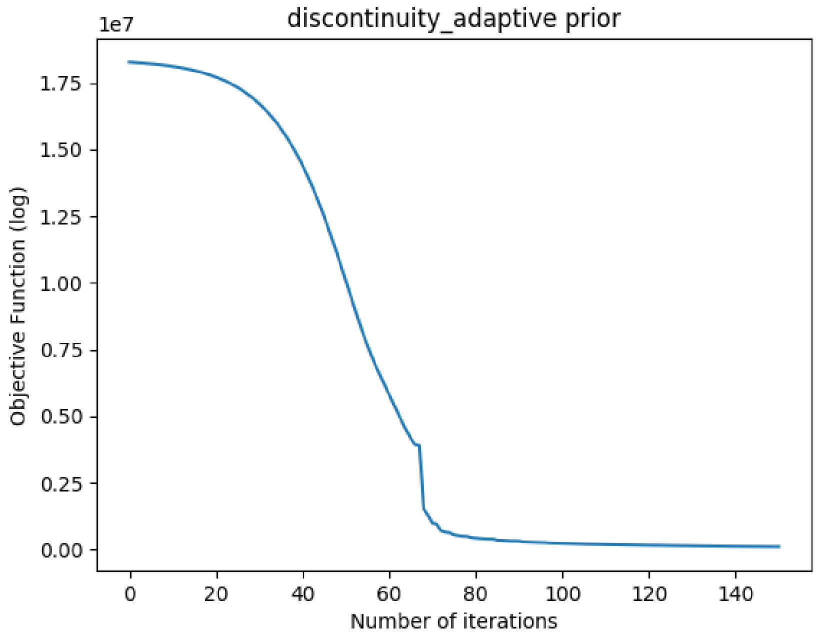

Figure 53: Objective Function Variation: V channel, Discontinuity Adaptive Prior