Image Denoising with MRFs

A maximum-a-posteriori Bayesian image-denoising algorithm is implemented using a Gaussian noise model coupled with a MRF prior that uses a 4- neighbor neighborhood system that has cliques of size no more than 2.

a) RRMSE between noisy and given noiseless image for :-
    1. Phantom image = 0.29858   2. Brain MRI image = 0.14243

b) Optimal values and RRMSE corresponding to the priors are:-

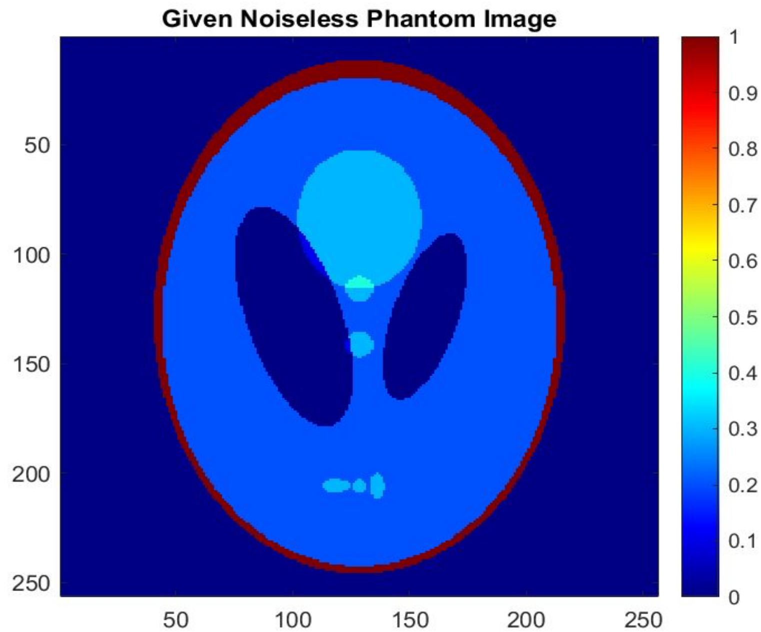
Phantom Image

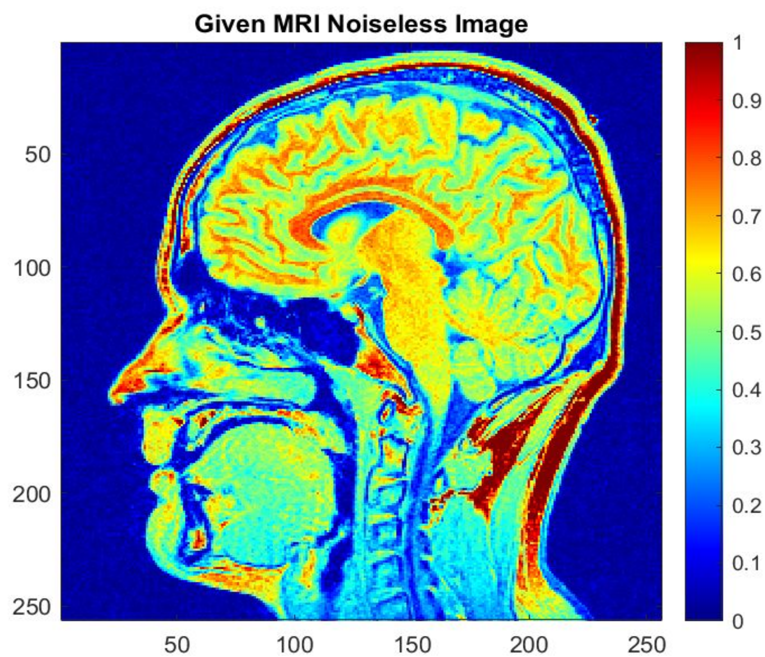
MRF Priors	Alpha (a*)	Gamma (b*)	RRMSE (a*, b*)	RRMSE (1.2a*, b*)	RRMSE (0.8a*, b*)	RRMSE (a*, 1.2b*)	RRMSE (a*,0.8b*)
Quadratic Prior	0.11	NA	0.27648	0.28387	0.27825	NA	NA
Huber Prior	0.113	1	0.22351	0.22863	0.2282	0.22863	NA
Discontinuity Adaptive	0.39	1	0.25428	0.2558	0.25693	0.25629	NA

MRI Image

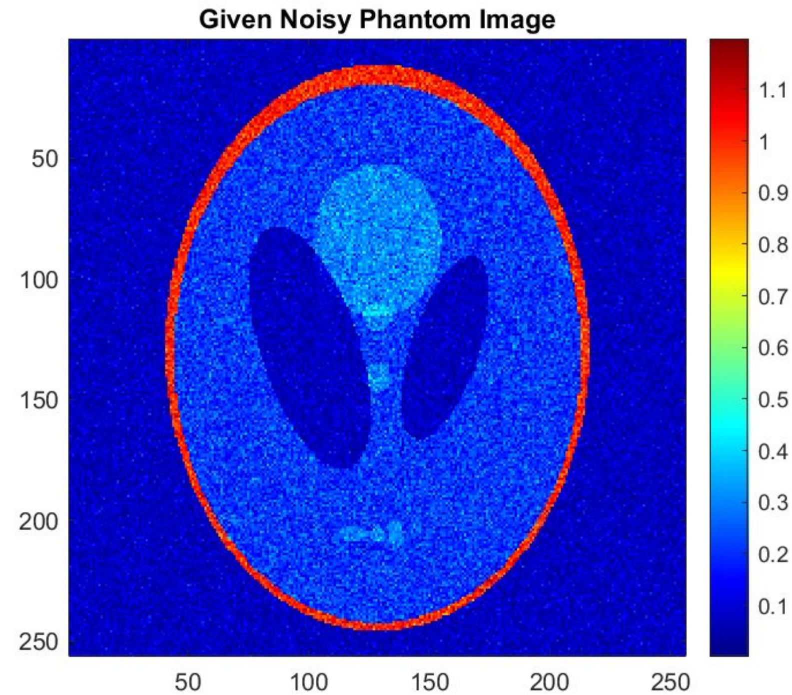
MRF Priors	Alpha (a*)	Gamma (b*)	RRMSE (a*, b*)	RRMSE (1.2a*, b*)	RRMSE (0.8a*, b*)	RRMSE (a*, 1.2b*)	RRMSE (a*,0.8b*)
Quadratic Prior	0.51	NA	0.13905	0.13914	0.13921	NA	NA
Huber Prior	0.17	2	0.118	0.11843	0.11845	0.118	0.118
Discontinuity Adaptive	0.15	70000	0.13533	0.13538	0.13578	0.13533	0.13533

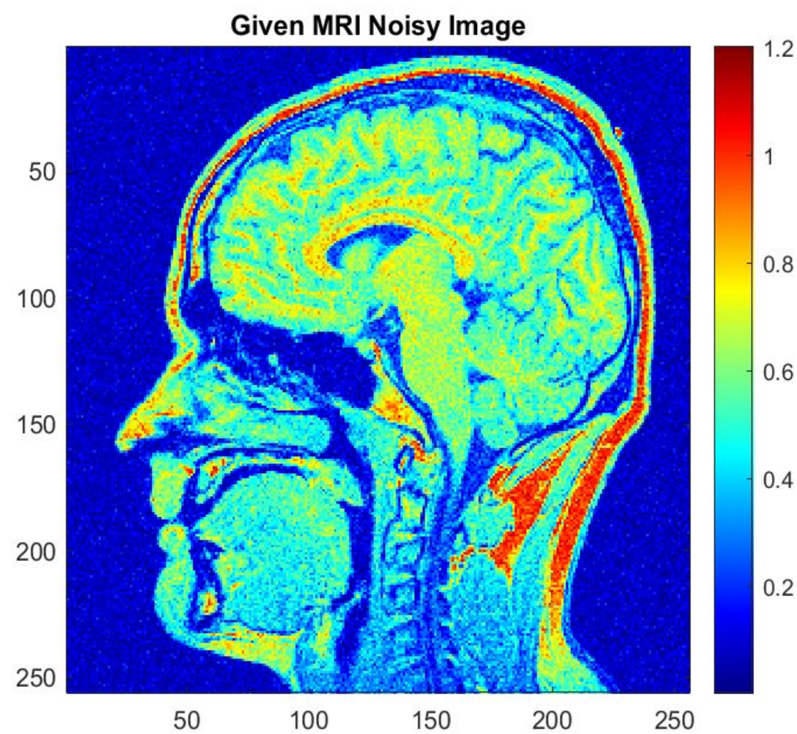
c) i) Given Noiseless Image



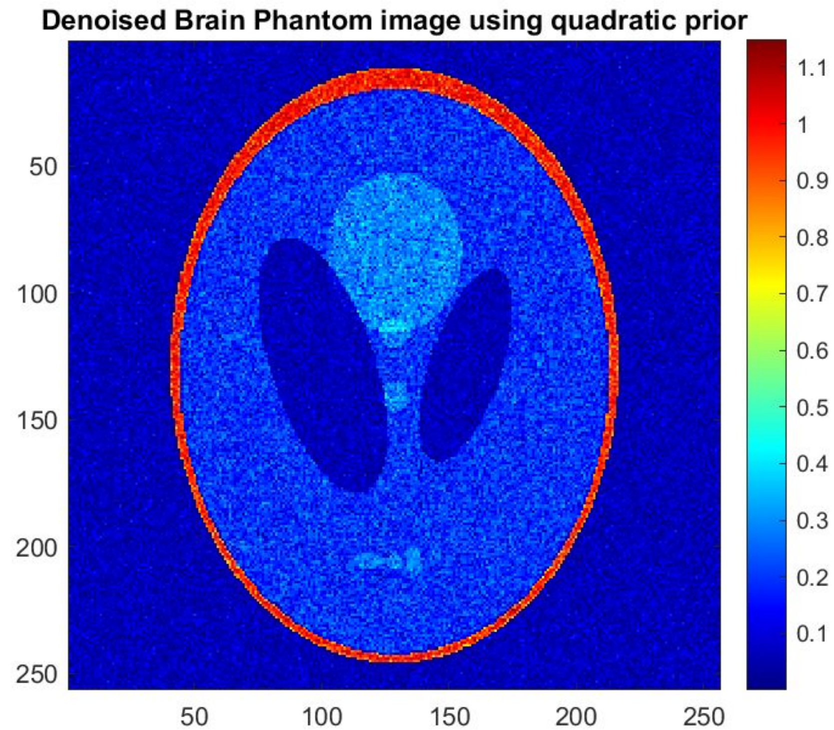


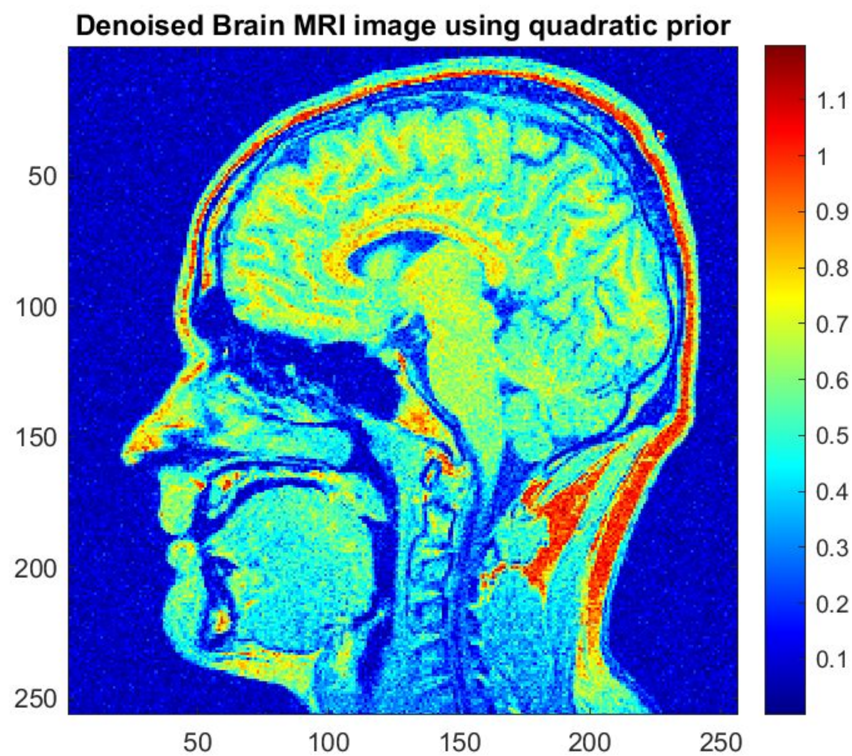
ii) Given Noisy Image



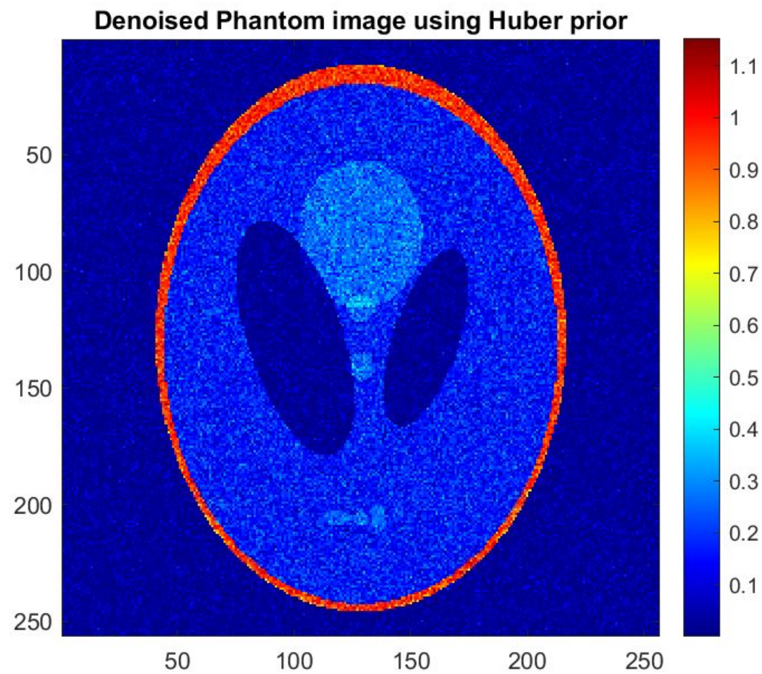


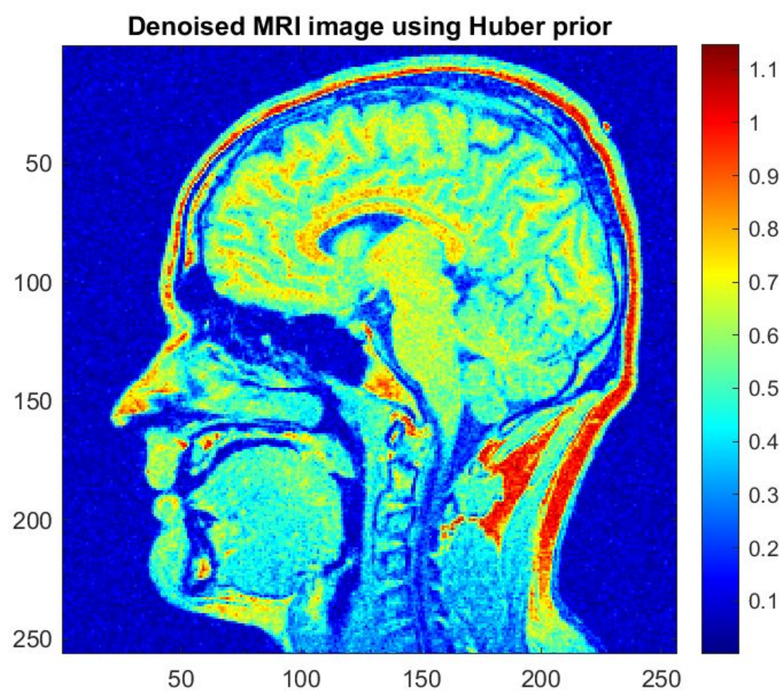
iii) Denoised Image using Quadratic Prior



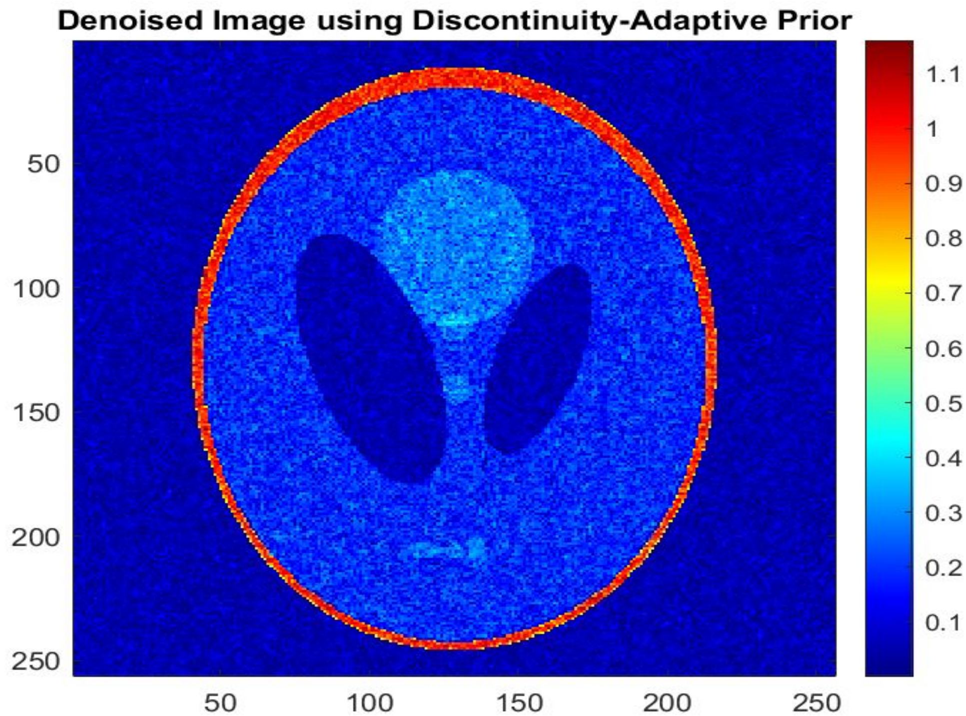


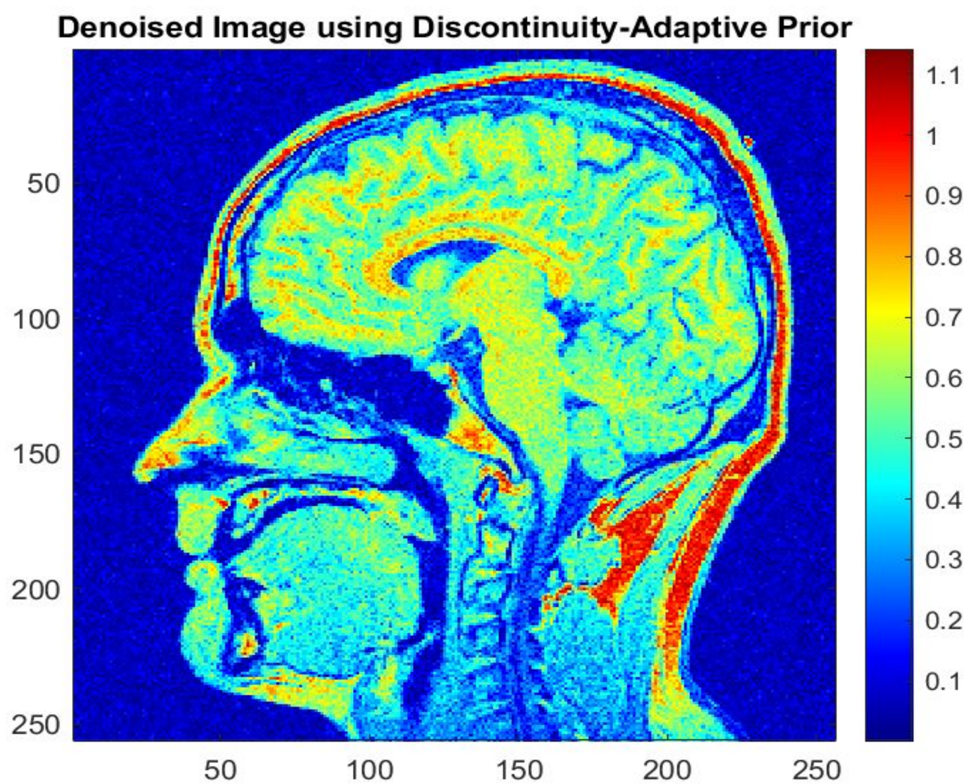
iv) Denoised Image using Huber Prior





v) Denoised Image using Discontinuity-Adaptive Prior





Phantom Image

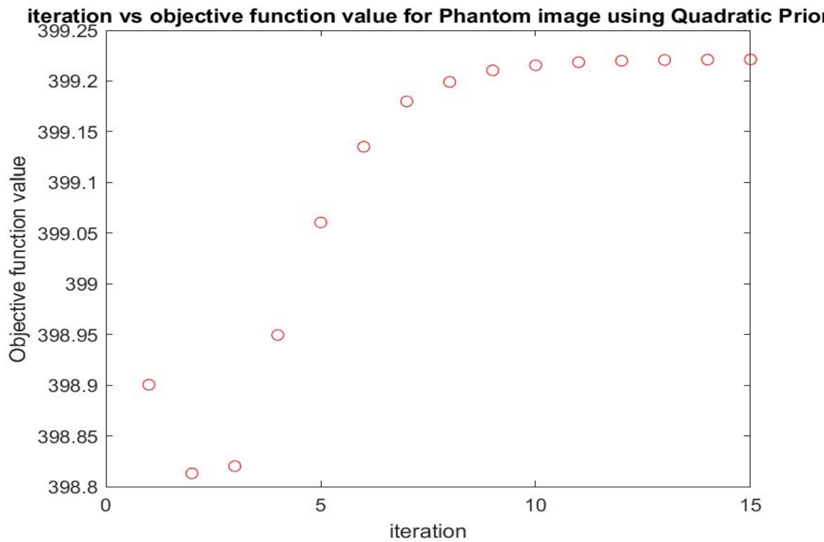
Phantom Image	Name in ‘results’ folder
i) Noiseless Image	Noiseless_phantom
ii) Noisy Image	Noisy_phantom
iii) Denoised Image using Quadratic Prior	DenoisedPhantomQuadratic
iv) Denoised Image using Huber Prior	DenoisedHuberPhantom
v) Denoised Image using Discontinuity Adaptive Prior	Denoised_Adaptive

MRI Image

MRI Image	Name in ‘results’ folder
i) Noiseless Image	Noiseless_MRI
ii) Noisy Image	Noisy_MRI
iii) Denoised Image using Quadratic Prior	DenoisedMRIQuadratic
iv) Denoised Image using Huber Prior	DenoisedHuberMRI
v) Denoised Image using Discontinuity Adaptive Prior	Mri_Denoised_Adaptive

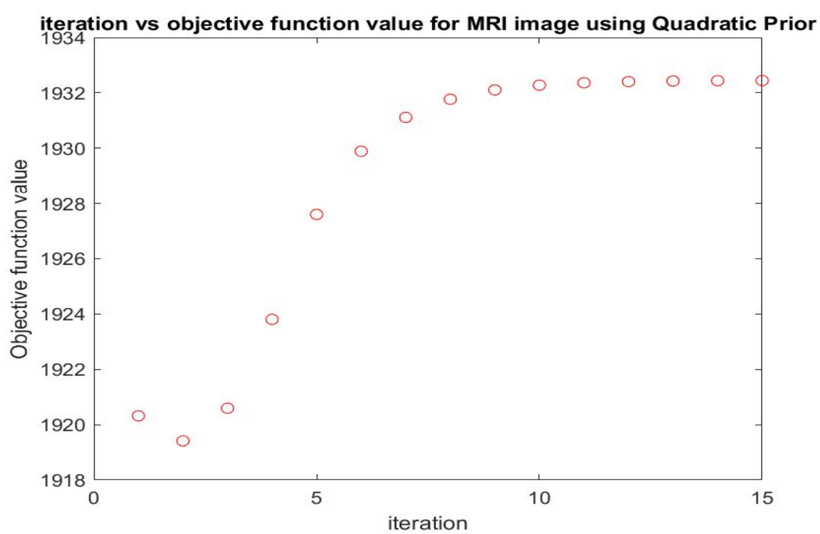
d) i) Objective vs Iteration Using Quadratic Prior

For Phantom Image



The objective function is the negative log posterior so it should keep decreasing to minimize the error. The objective function using Quadratic Prior on Phantom Image first decreases till 3 iterations and then again starts increasing. Hence, the iteration loop in the code is made to run for 3 iterations because the global minima is occurring at 3.

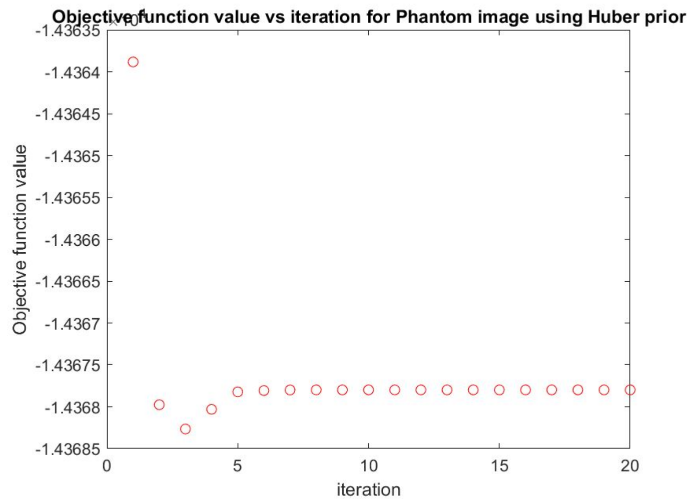
For MRI Image



The objective function is the negative log posterior so it should keep decreasing to minimize the error. The objective function using Quadratic Prior on Brain MRI Image first decreases till 2 iterations and then again starts increasing. Hence, the iteration loop in the code is made to run for 2 iterations because the global minima is occurring at 2.

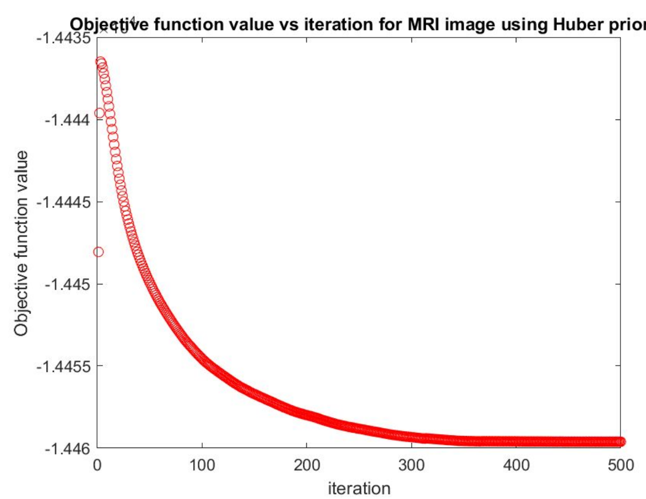
ii) Objective vs Iteration Using Huber Prior

For Phantom Image



The objective function is the negative log posterior so it should keep decreasing to minimize the error. The objective function using Huber Prior on Phantom Image first decreases till 4 iterations and then again starts increasing. Hence, the iteration loop in the code is made to run for 4 iterations because the global minima is occurring at 4.

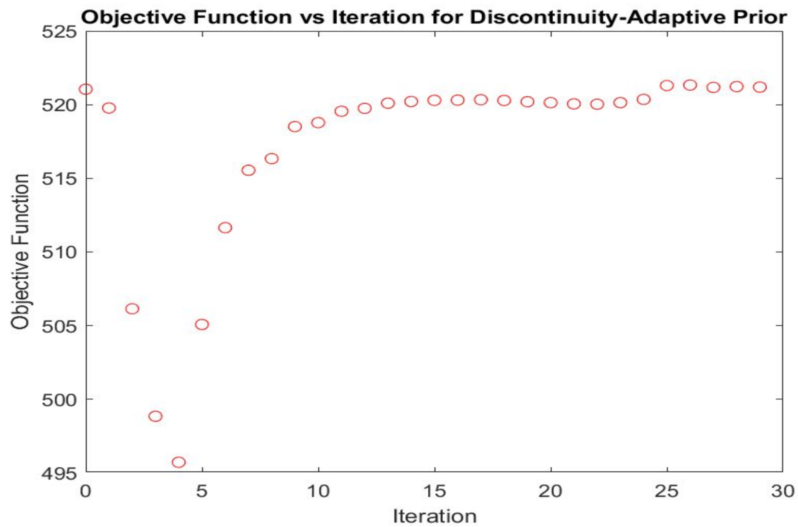
For MRI Image



The objective function is the negative log posterior so it should keep decreasing to minimize the error. The objective function using Huber Prior on MRI image first decreases till 400 iterations and then again starts increasing. Hence, the iteration loop in the code is made to run for 6 iterations because the global minima is occurring at 400.

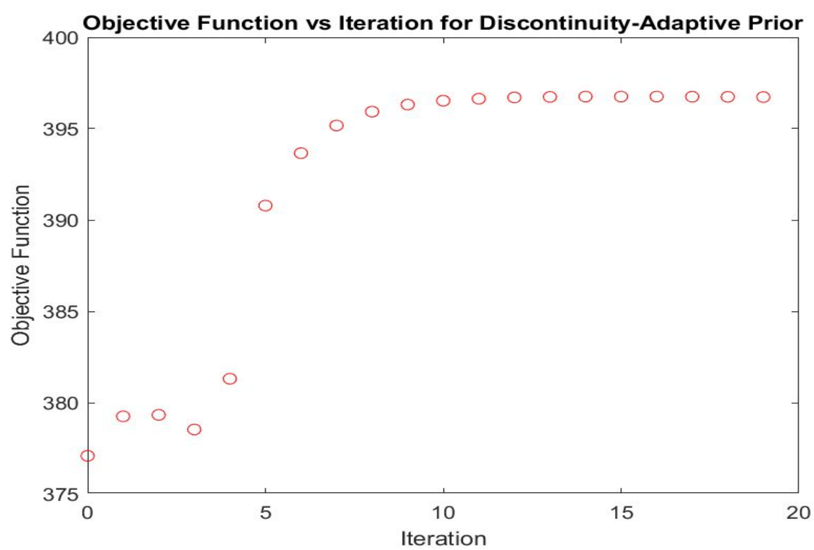
iii) Objective vs Iteration Using Discontinuity-Adaptive Prior

For Phantom Image



The objective function is the negative log posterior so it should keep decreasing to minimize the error. The objective function using Discontinuity-Adaptive Prior on Phantom Image first decreases till 4 iterations and then again starts increasing. Hence, the iteration loop in the code is made to run for 4 iterations because the global minima is occurring at 4.

For MRI Image



The objective function is the negative log posterior so it should keep decreasing to minimize the error. The objective function using Discontinuity-Adaptive Prior on MRI Image first decreases till 4 iterations and then again starts increasing. Hence, the iteration loop in the code is made to run for 4 iterations because the global minima is occurring at 4.

Phantom Image

Objective vs Iteration	Name in ‘results’ folder
i) Quadratic Prior	quadraticObjectivePhantom
ii) Huber Prior	ObjectiveHuberPhantom
iii) Discontinuity- Adaptive	Objective_Adaptive

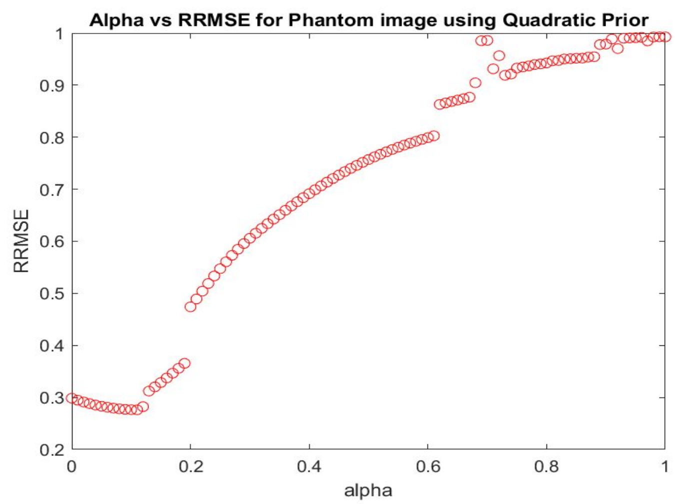
Brain MRI Image

Objective vs Iteration	Name in ‘results’ folder
i) Quadratic Prior	QuadraticObjectiveMRI
ii) Huber Prior	ObjectiveHuberMRI
iii) Discontinuity- Adaptive	Mri_objective_Adaptive

Tuning Parameters

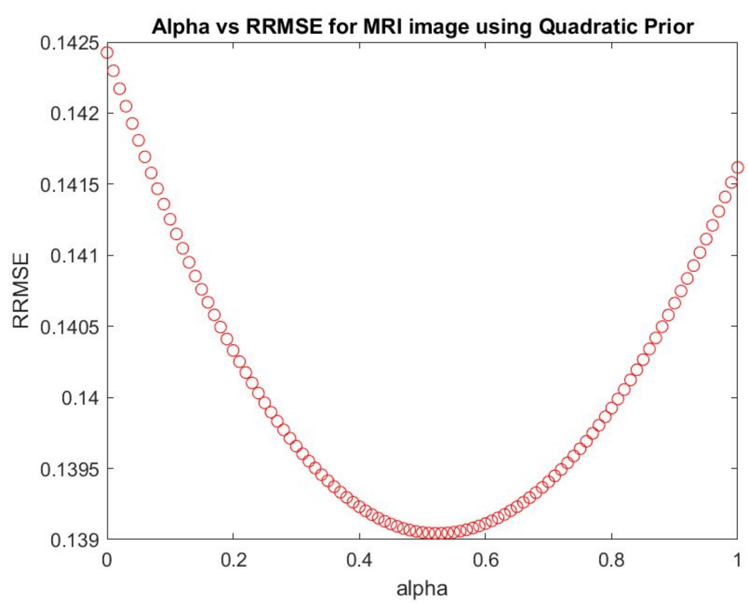
i) Tuning alpha parameter for Quadratic Prior

For Phantom Image



The above graph shows the RRMSE, between the denoised and noiseless phantom image, for different alpha values. At alpha 0.11, the RRMSE is lowest when Quadratic Prior is used. Hence, the value of alpha is kept as 0.11

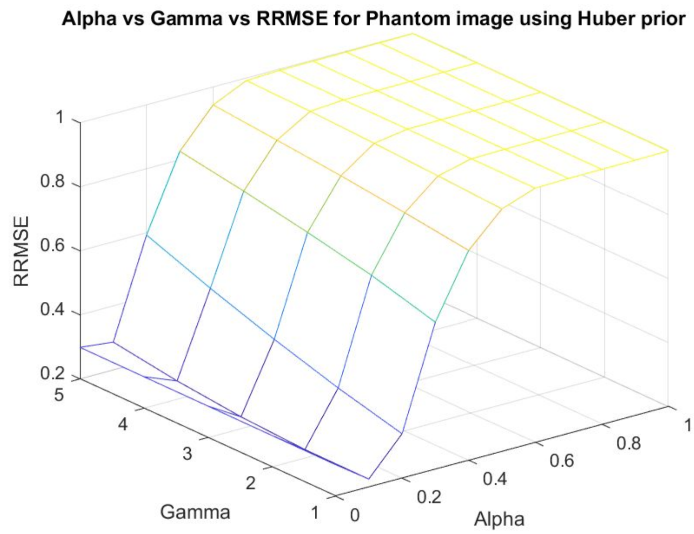
For MRI Image



The above graph shows the RRMSE, between the denoised and noiseless MRI image, for different alpha values. At alpha 0.51, the RRMSE is lowest when Quadratic Prior is used. Hence, the value of alpha is kept as 0.51

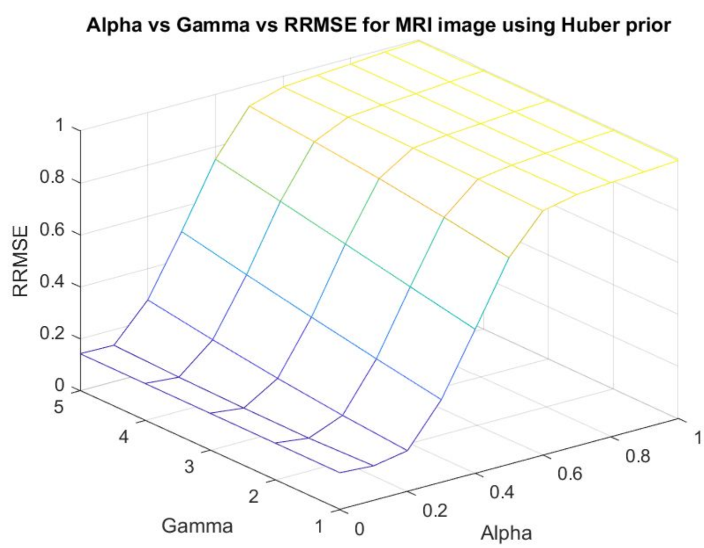
ii) Tuning alpha and gamma parameters for Huber Prior

For Phantom Image



The above graph shows the RRMSE, between the denoised and noiseless phantom image, for different alpha and gamma parameters. At alpha 0.113 and gamma 1, the RRMSE is lowest when Huber Prior is used. Hence, the value of alpha is kept as 0.113 and the value of gamma is kept as 1.

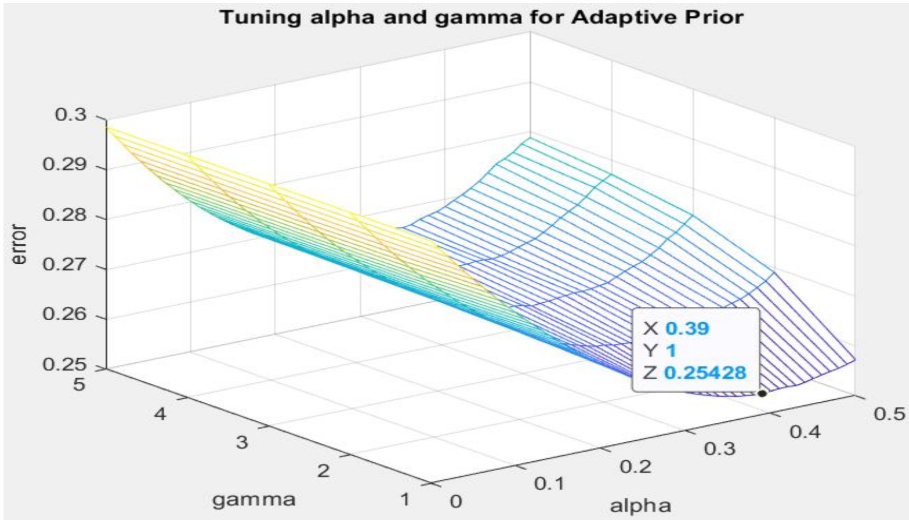
For MRI Image



The above graph shows the RRMSE, between the denoised and noiseless MRI image, for different alpha and gamma parameters. At alpha 0.17 and gamma 2, the RRMSE is lowest when Huber Prior is used. Hence, the value of alpha is kept as 0.17 and the value of gamma is kept as 2.

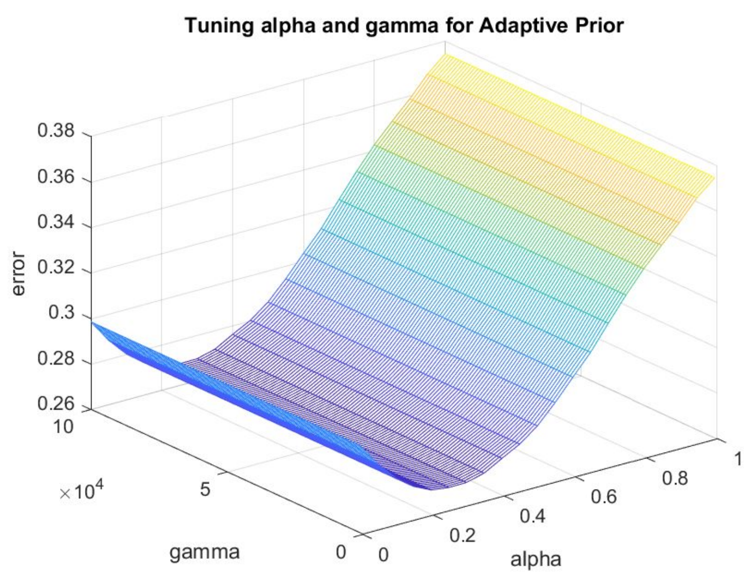
iii) Tuning alpha and gamma parameter for Discontinuity-Adaptive Prior

For Phantom Image



The above graph shows the RRMSE, between the denoised and noiseless phantom image, for different alpha and gamma parameters. At alpha 0.39 and gamma 1, the RRMSE is lowest when Discontinuity-Adaptive Prior is used. Hence, the value of alpha is kept as 0.39 and the value of gamma is kept as 1.

For MRI Image



The above graph shows the RRMSE, between the denoised and noiseless MRI image, for different alpha and gamma parameters. At alpha 0.15 and gamma 70000, the RRMSE is lowest when Discontinuity-Adaptive Prior is used. Hence, the value of alpha is kept as 0.15 and the value of gamma is kept as 70000.

3) a)

- 6 neighbour neighbourhood system is taken for Bayesian image-denoising algorithm (each pixel has 6 neighbors: [left, right, up, down] in the spatial neighbourhood and two neighbours across the three colors channels, i.e., red, green, and blue; the neighborhood wraps around at image boundaries) that has cliques of size no more than 2.

  Ex: Suppose (x,y,B) is a pixel in the Blue channel. Then its neighbours are:
  (x+1,y,B), (x-1,y,B), (x,y+1,B), (x,y-1,B), (x,y,R), (x,y,G).

- Huber MRF prior model is used for statistical dependencies within a spatial neighbourhood and Discontinuity-Adaptive MRF prior is used for dependencies across the three colors channels, i.e., red, green, and blue.

- Huber MRF prior model is used for statistical dependencies within a spatial neighbourhood because the penalty g(u) becomes constant/bounded or increases at a linear rate. Huber prior gives better performance in the spatial neighbourhood as seen in this assignment as well that the RRMSE is lowest when Huber prior is used. ‘u’ is x-y where x is the pixel and y is its neighbour.

$$g(u) = \frac{1}{2}|u|^2 \quad when\ |u| \leq \gamma$$

$$g(u) = \gamma|u| - \frac{\gamma^2}{2} \quad when\ |u| > \gamma$$

$$g`(u) = u \quad when\ |u| \leq \gamma$$

$$g`(u) = \gamma\ sgn(u) \quad when\ |u| > \gamma$$

- Discontinuity-Adaptive MRF prior is used for dependencies across the three colors channels, i.e., red, green, and blue because since the channels colors are different so the difference between intensities across pixels in different channels might be large. So a MRF prior is needed that can allow discontinuities. Hence Discontinuity-Adaptive prior is used. ‘u’ is x-y where x is the pixel and y is its neighbour.

$$g(u) = \gamma|u| - \gamma^2 \log\left(1 + \frac{|u|}{\gamma}\right)$$

$$g`(u) = \frac{\gamma}{2(\gamma + |u|)}$$

- Energy Function:

$$E(x) = \sum_i g\left(y_i - x\right) + \sum_j k\left(y_j - x\right)$$

Where, g(u) is the Huber potential function for spatial neighbourhood and k(u) is the Discontinuity-Adaptive potential function for neighbours across three channels.

b) For RGB Microscopy Image, Poisson distribution can be used as a noise model because in light microscopy the measurements are made by accumulation of photons over a detector. During a certain duration of imaging the average number of photons captured are λ.

Probability of ‘k’ captured photons:

$$P(X = k;\lambda) = \frac{\lambda^k(exp(-\lambda))}{k!}$$

Likelihood function:

$$P(y|x) = \prod_i P\left(y_i| x_i\right) = \prod_{i=1}^{n} \frac{\lambda^{x_i} exp(-\lambda)}{x_i!}$$

$$\max_{x_i} P(y|x) = \sum_{i=1}^{n} \log\left(\frac{\lambda^{x_i} exp(-\lambda)}{x_i!}\right) = \sum_{i=1}^{n} \left[-\lambda - \log\left(x_i!\right) + \log\left(\lambda^{x_i}\right)\right]$$

$$\max_{x_i} P(y|x) = max\left(-n\lambda + \log\lambda \sum_{i=1}^{n} x_i - \sum_{i=1}^{n} \log\left(x_i!\right)\right)$$

c) Objective Function for Bayesian Denoising of RGB Microscopy Image is:

$$\max_{x_i} P(x|y,\theta) = \max_{x_i} P\left(y_i | x_i, \theta\right) P\left(x_i | x_{N_i}, \theta\right)$$

$$= \max_{x_i} \left(\log P\left(y_i | x_i, \theta\right) + \log\left(P\left(x_i | x_{N_i}, \theta\right)\right)\right)$$

$$= \max_{x_i} \left(\log P\left(y_i | x_i, \theta\right) + \sum_{a \in A_i} \left(-V_a\left(x_a\right)\right) + \sum_{b \in B_i} \left(-V_b\left(x_b\right)\right)\right)$$

where,

$V_a\left(x_a\right)$ is the potential function for the spatial neighbourhood pixels which uses Huber prior.

$V_b\left(x_b\right)$ is the potential function for the neighbours across the three colors channels, i.e., red, green, and blue which uses Discontinuity-Adaptive Prior.

Objective function is summation of likelihood function and MRF prior. Here Poisson is the likelihood function and Huber is the MRF prior for spatial neighbourhood pixels and Discontinuity-Adaptive is the MRF prior for the neighbours across the three colors channels.

$$\max_{x_i} -\log(P(x|y,\theta)) = \min_{x_i} \left((1-\beta)\left(n\lambda - \log\lambda \sum_{i=1}^{n} x_i + \sum_{i=1}^{n} \ln\left(x_i\right)\right) + \beta \sum_{a \in A_i} \left(V_a\left(x_a\right)\right) + \beta \sum_{b \in B_i} \left(V_b\left(x_b\right)\right)\right)$$

For gradient-based optimization with parallel updates, derivative at voxel i is:

$$\partial \frac{(-\log(P(x|y,\theta)))}{\partial x_i} = \min_{x_i} \left((1-\beta)\left(n\lambda - n\log\lambda + \sum_{i=1}^{n} \left(\frac{1}{x_i}\right)\right) + \beta \frac{\partial}{\partial x_i} \sum_{a \in A_i} \left(V_a\left(x_a\right)\right) + \beta \frac{\partial}{\partial x_i} \sum_{b \in B_i} \left(V_b\left(x_b\right)\right)\right)$$