Matlab Work 1 - Image Deconvolution: Wiener-Hunt method

Sebestyén Németh, Kush Gupta, Bendegúz H. Zováthi, Levente Pető
IPCV

November 27, 2022

1 Exercise 1.

$$J_{PLS}(\mathbf{x}) = ||\mathbf{y} - \mathbf{H}\mathbf{x}||^2 + \mu||\mathbf{D}\mathbf{x}||^2 = (\mathbf{y} - \mathbf{H}\mathbf{x})^t(\mathbf{y} - \mathbf{H}\mathbf{x}) + \mu\mathbf{x}^t\mathbf{D}^t\mathbf{D}\mathbf{x} \tag{1}$$

$$\operatorname*{argmin}_x J_{PLS}(\mathbf{x}) = ? \tag{2}$$

$$\nabla J_{PLS}(\hat{\mathbf{x}}) = 0 \tag{3}$$

$$\nabla J_{PLS}(\mathbf{x}) = -2\mathbf{H}^t(\mathbf{y} - \mathbf{H}\mathbf{x}) + 2\mu\mathbf{D}^t\mathbf{D}\mathbf{x} \tag{4}$$

$$\mathbf{H}^t\mathbf{H}\hat{\mathbf{x}} - \mathbf{H}^t\mathbf{y} + \mu\mathbf{D}^t\mathbf{D}\hat{\mathbf{x}} = 0 \tag{5}$$

$$(\mathbf{H}^t\mathbf{H} - \mu\mathbf{D}^t\mathbf{D})\hat{\mathbf{x}} = \mathbf{H}^t\mathbf{y} \tag{6}$$

$$\hat{\mathbf{x}} = (\mathbf{H}^t\mathbf{H} - \mu\mathbf{D}^t\mathbf{D})^{-1}\mathbf{H}^t\mathbf{y} \tag{7}$$

If $\mu = 0$, we get back the basic Wiener-filter formula:

$$\hat{\mathbf{x}} = \mathbf{H}^{-1}\mathbf{y} \tag{8}$$

2 Exercise 2.

$$\hat{H} = \mathbf{F}^\dagger\Lambda_h\mathbf{F} \qquad \hat{D} = \mathbf{F}^\dagger\Lambda_d\mathbf{F} \tag{9}$$

$$\mathbf{F}^\dagger\mathbf{F} = \mathbf{I} \qquad \mathbf{F}\mathbf{y} = \mathring{\mathbf{y}} \qquad \mathbf{F}\hat{\mathbf{x}} = \mathring{\hat{\mathbf{x}}} \tag{10}$$

$$(\mathbf{F}^\dagger\Lambda_h^\dagger\mathbf{F}\mathbf{F}^\dagger\Lambda_h\mathbf{F} - \mu\mathbf{F}^\dagger\Lambda_d^\dagger\mathbf{F}\mathbf{F}^\dagger\Lambda_d\mathbf{F})\hat{\mathbf{x}} = \mathbf{F}^\dagger\Lambda_h^\dagger\mathbf{F}\mathbf{y} \tag{11}$$

$$\mathbf{F}^\dagger(\Lambda_h^\dagger\Lambda_h) - \mu\Lambda_d^\dagger\Lambda_d)\mathbf{F}\hat{\mathbf{x}} = \mathbf{F}^\dagger\Lambda_h^\dagger\mathbf{F}\mathbf{y} \tag{12}$$

$$(\Lambda_h^\dagger\Lambda_h) - \mu\Lambda_d^\dagger\Lambda_d)\mathring{\hat{\mathbf{x}}} = \Lambda_h^\dagger\mathring{\mathbf{y}} \tag{13}$$

$$\mathring{\hat{\mathbf{x}}} = (\Lambda_h^\dagger\Lambda_h) - \mu\Lambda_d^\dagger\Lambda_d)^{-1}\Lambda_h^\dagger\mathring{\mathbf{y}} \tag{14}$$

If $\mu = 0$, we get back the basic Wiener-filter formula:

$$\mathring{\hat{\mathbf{x}}} = \Lambda_h^{-1}\mathring{\mathbf{y}} \tag{15}$$

3 Exercise 3.

By observation (Figure 1), we can see that Data 1 has less blurry observation whereas the Data 2 observations are more blurry. Since blurring removes the sharp edges, in the frequency domain we can observe that for Data 1 highest frequency (at the center) and some lower frequencies are present. For Data 2 high frequencies are attenuated, hence the edges are blurred.

As the highest (zeroth) frequency is at the center we displayed the images on a scale of -0.5 and 0.5 in the frequency domain (on both axes) at Figure 2, 3.

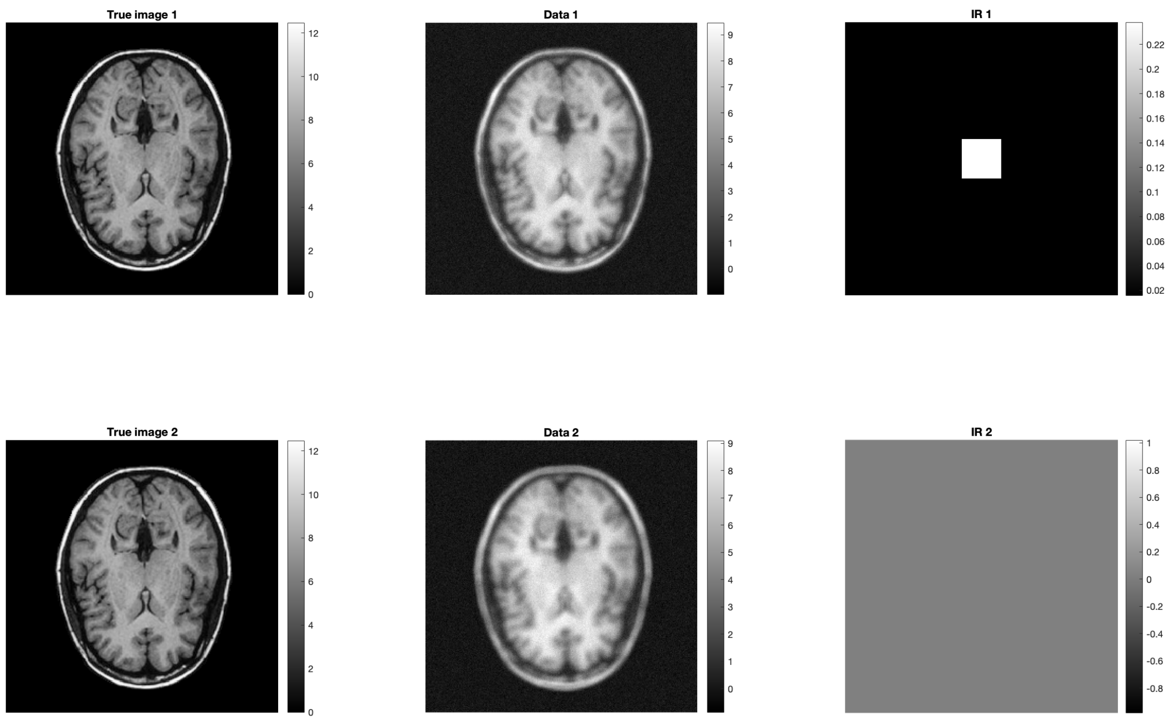


Figure 1: Original and observed images and the impulse response of the filter

4 Exercise 4.

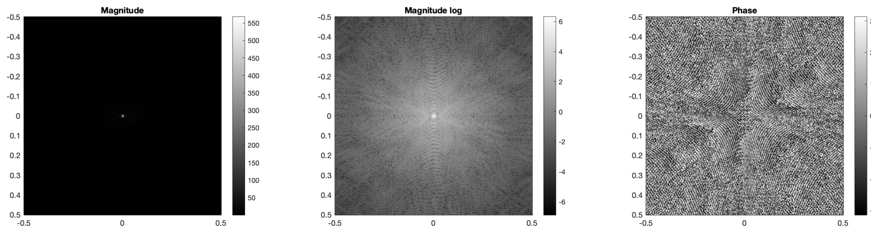
The two impulse responses and the transfer functions for the filters 1 and 2 are displayed in Figure 4. Both are low-pass filters. The difference between filter 1 and 2 is, when using the filter 2 for inversion it will explode the frequencies at magnitude zero as seen in the impulse response for filter 2. However, in filter 1 the magnitude for the frequency is not zero.

5 Exercise 5.

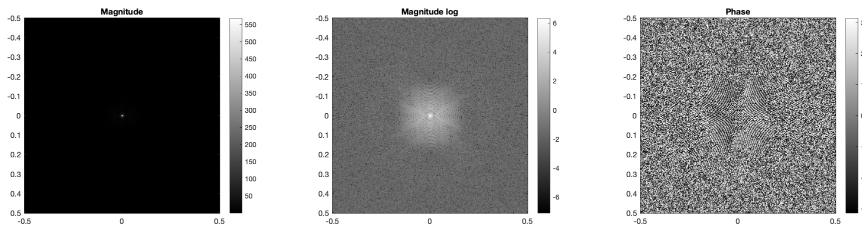
You can find the implementation in the submitted MATLAB script.

6 Exercise 6.

We used Data 1 for this exercise. In this case, when the $\mu = 0$ we are eliminating the penalizing term which preserves the edges. In the spatial domain, for $\mu = 0$ (Figure 5a) we can notice noisy observation and there are ringing effects in the observed image. In the frequency domain, (Figure 5b) for $\mu = 0$ when we compared the resulting image with respect to the true image we can see that the higher frequencies are attenuated which results in the smoothing of the observed image (minute details are not preserved).

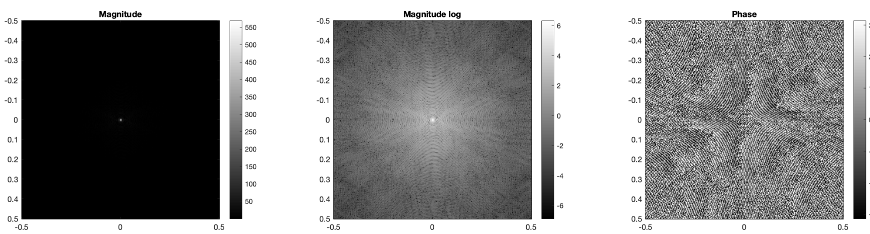


(a) True image in frequency domain

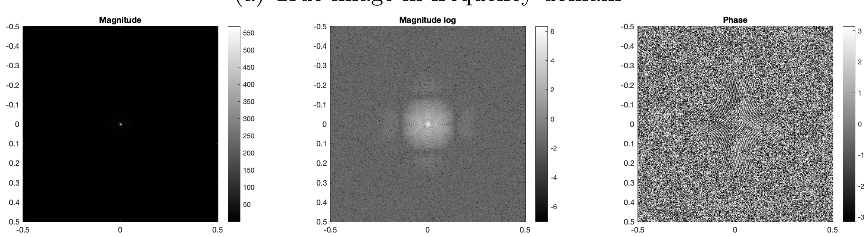


(b) Blurred image in frequency domain

Figure 2: Data 1



(a) True image in frequency domain



(b) Blurred image in frequency domain

Figure 3: Data 2

7 Exercise 7.

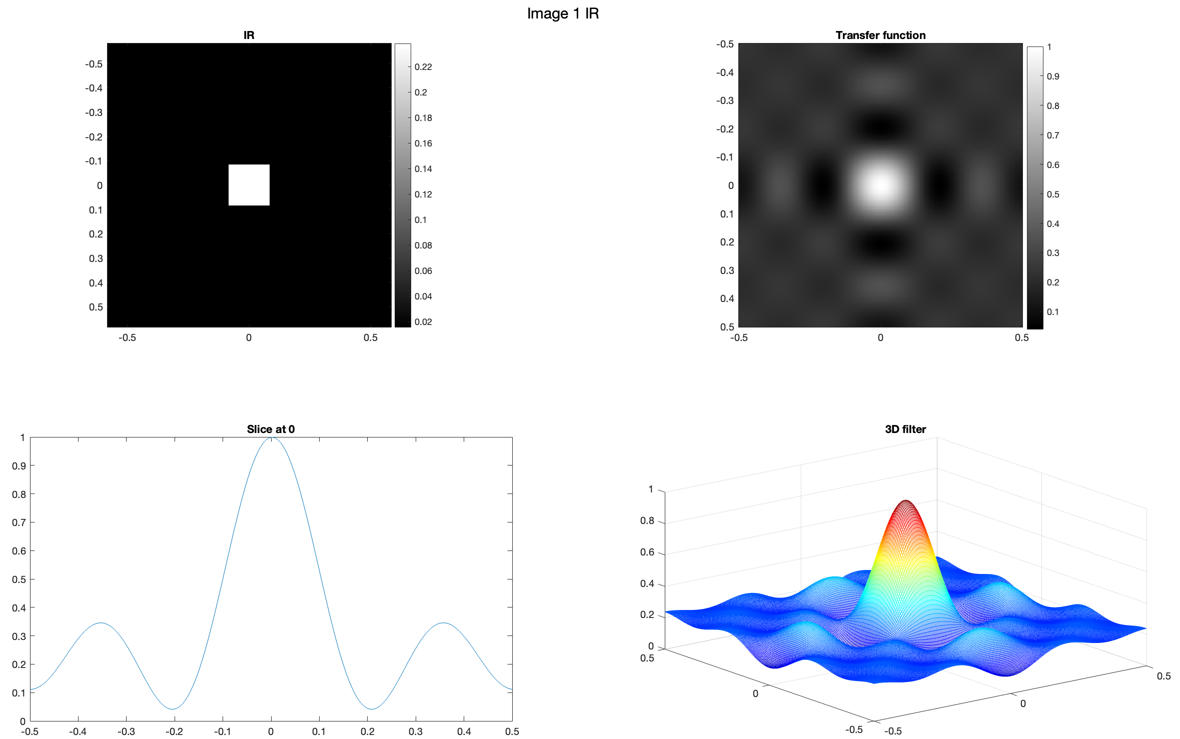
For Data 1:
In the spatial domain, (Figure 6a) we can notice that as we increase μ the penalty increases resulting in smoother images (until $\mu = 0.1$), however as we further increase μ from $\mu = 1$ on wards we can notice that the image gets blurry and for $\mu = 10$ the resulting image is too blurry due explosion of low frequencies (?). In the frequency domain, (Figure 6b), when the μ is low (i.e 0.01 and 0.1) some high frequencies are attenuated and as we further increase μ (i.e from $\mu = 1$ on wards) a lot of high frequencies are attenuated resulting in a blurry image.
By trial and error, the appropriate value of μ for data1 is $\mu = 0.01$.

For Data 2:
In the spatial domain, (Figure 7a) we noticed that the penalty increases as we increase μ resulting smoother images (until $\mu = 0.1$), however as we further increase μ from $\mu = 1$ on wards we can notice that the image gets blurry and for $\mu = 10$ the resulting image is too blurry due explosion of low frequencies (?).

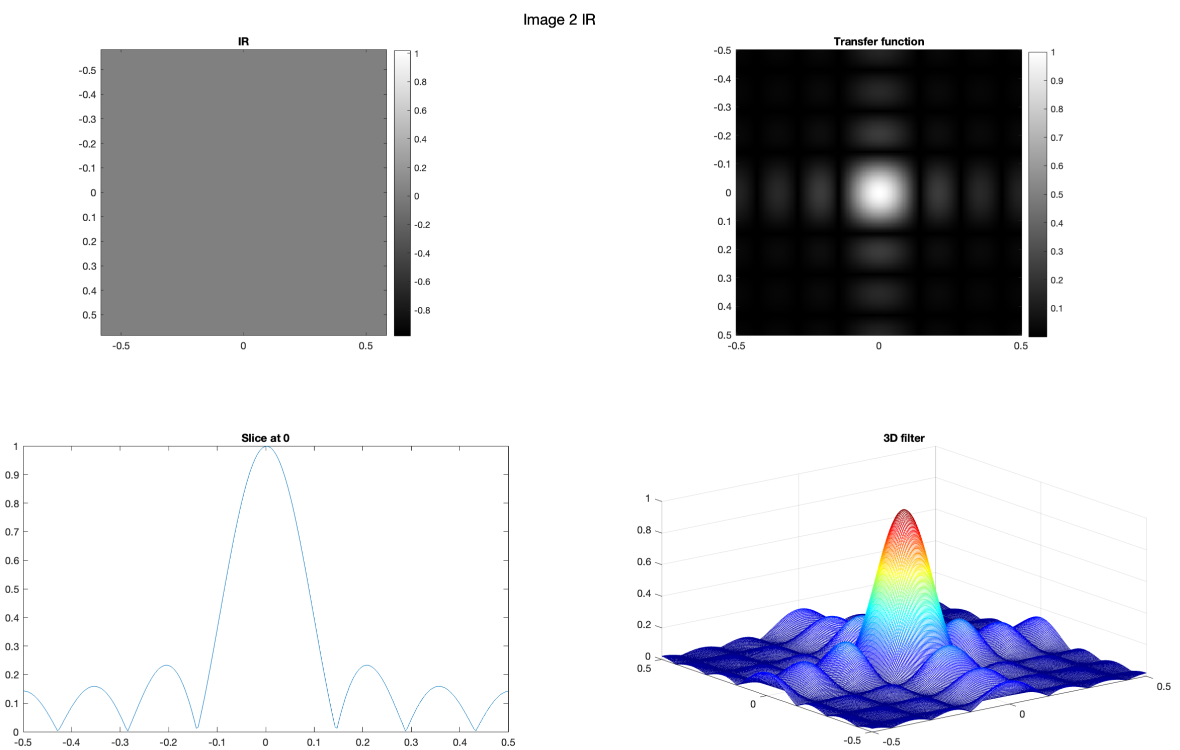
In the frequency domain, (Figure 7b), when the μ is low (i.e 0.01 and 0.1) some high frequencies are attenuated and as we further increase μ (i.e from $\mu = 1$ on wards) a lot of high frequencies are attenuated resulting in blurry image. Also, we can clearly notice the ringing effect and the artifacts in (Figure 7b).
By trial and error, the appropriate value of μ for data2 is $\mu = 0.01$.

8 Exercise 8.

In the figure,8 we can observe the different distance metrics from left to right: D1, D2, Dinf and their corresponding μ values (red line on each graph) which minimize each distance function. For Data 2, we choose $\mu = 0.01$ empirically, which gives a good resulting image. From the calculated distance metric for data2, 8b we can observe that for distance metrics D2 and Dinf the minimum μ value is 0.01 and for D1 it's 0.1. Similarly, for Data 1, we choose $\mu = 0.01$ empirically, which gives a good resulting image, and from the calculated distance metric for data1, 8a we can observe that for distance metrics D2 and Dinf the minimum μ value is 0.01 and for D1 it's 0.1.

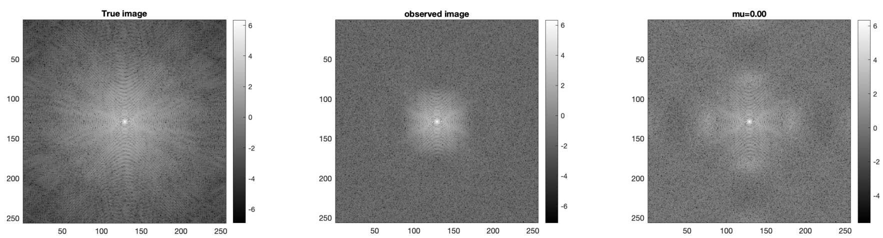


(a) Impulse response for filter 1.

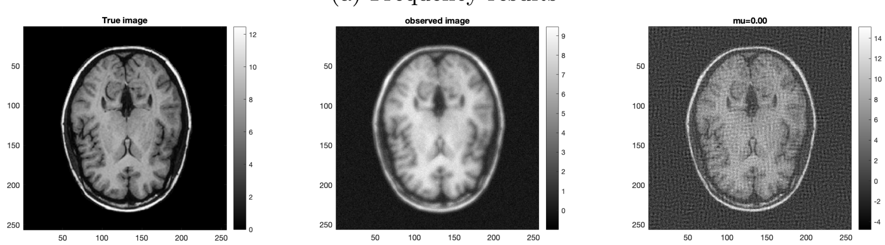


(b) Impulse response for filter 2.

Figure 4: Impulse responses and transfer functions for filter 1 and 2

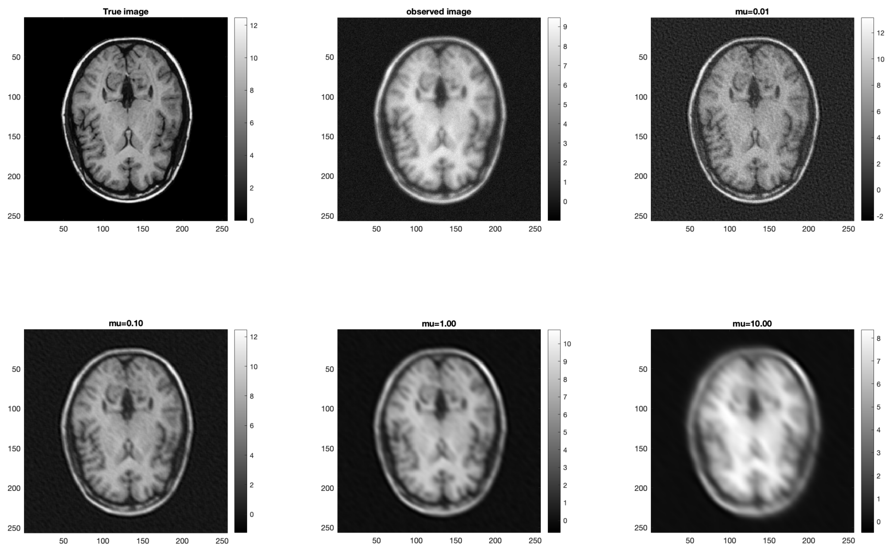


(a) Frequency results

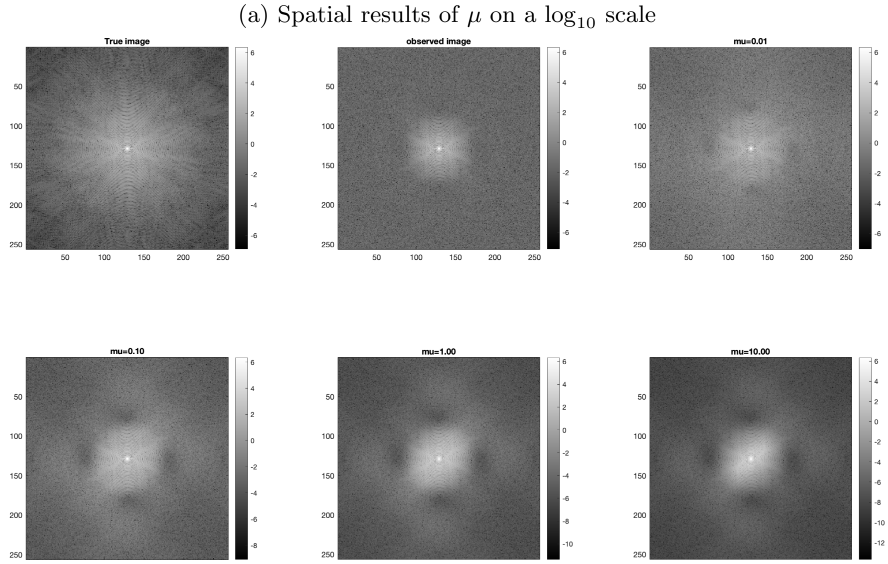


(b) Spatial results

Figure 5: Frequency and spatial results where $\mu = 0$

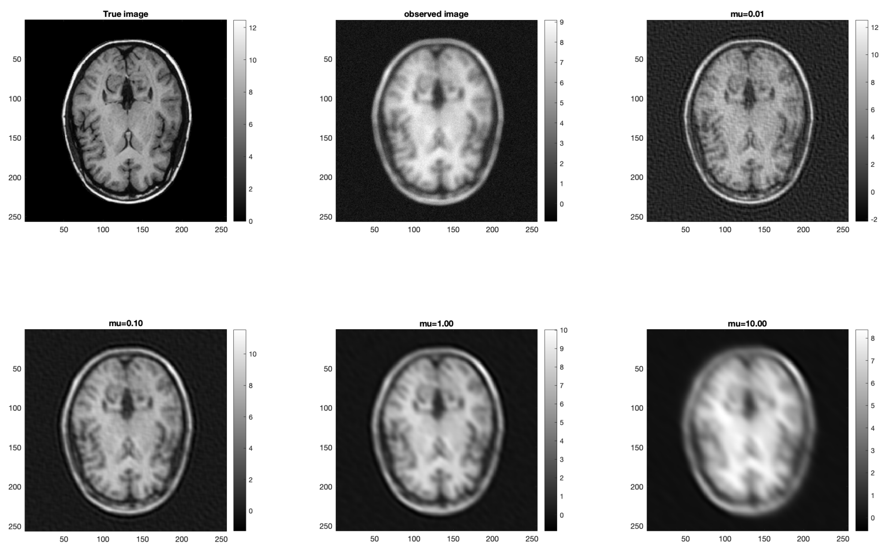


(a) Spatial results of μ on a $\log_{10}$ scale

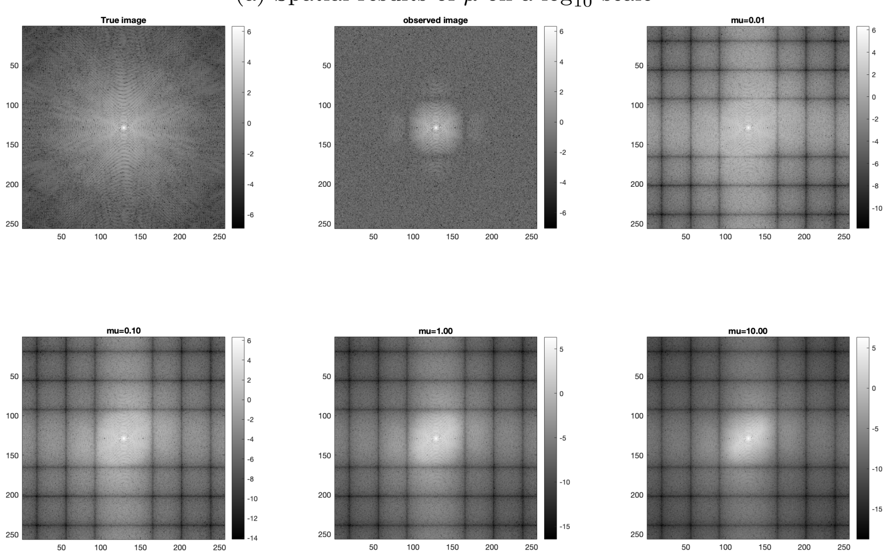


(b) Frequency results of μ on a $\log_{10}$ scale

Figure 6: Results for Data 1

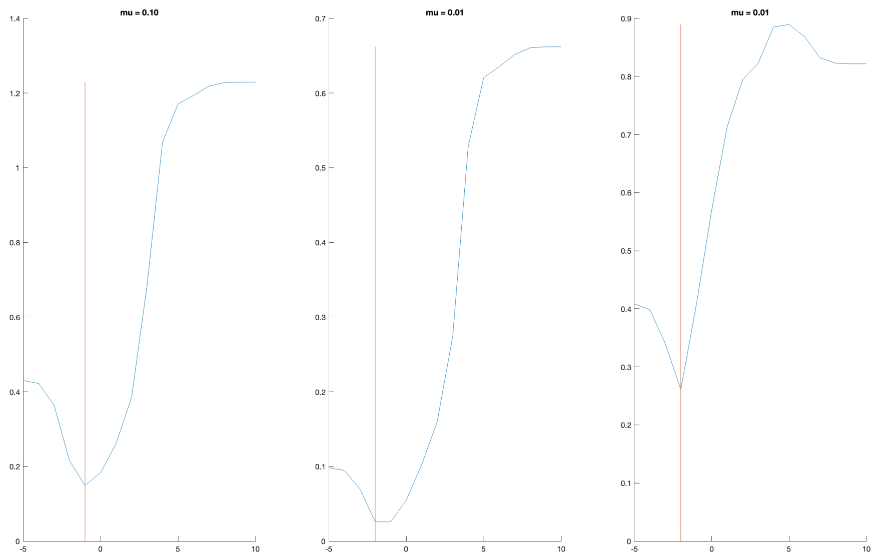


(a) Spatial results of μ on a $\log_{10}$ scale

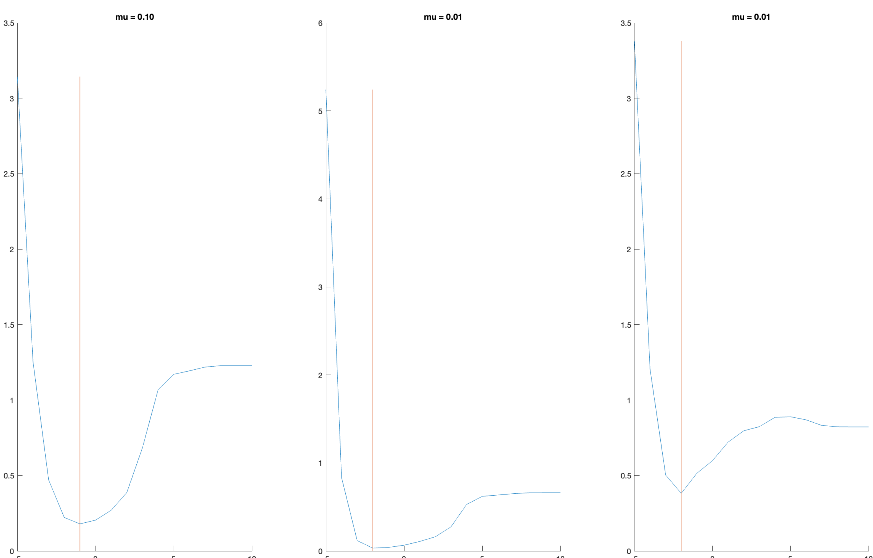


(b) Frequency results of μ on a $\log_{10}$ scale

Figure 7: Results for Data 2



(a) Different distance metrics with their corresponding minimum μ values in each case for data1



(b) Different distance metrics with their corresponding minimum μ values in each case for data2

Figure 8: Distances metrics (L to R: D1, D2, Dinf) for Data1 and Data2