



BSc thesis APPLIED MATHEMATICS

“Noise reduction in MRI images using partial differential equations”

Lyanne de Haas

Technische Universiteit Delft

Supervisor

Dr.ir. M.B. van Gijzen

Members of the assessment committee

Dr. J.L.A. Dubbeldam

Dr. J.G. Spandaw

X. Shan

August, 2019

Delft

Abstract

MRI machines use superconducting magnets to create an image. However, these magnets are very expensive. It is possible to use weaker magnets in a low-field MRI, but those will result in a lower signal-to-noise ratio, meaning the images will be polluted.

An image can be smoothed by viewing it as the initial condition of a partial differential equation (PDE) and changing it through time integration. The choice for the PDE determines the way the image changes.

This paper compares four PDE's: a second-order equation originally proposed by Perona & Malik, a fourth-order equation as proposed by You & Kaveh, and both aforementioned equations with a fidelity term added to them. Said fidelity term ensures the result does not deviate too far from the original image. All methods use a diffusion coefficient specially designed to preserve edges. These methods are tested on two versions of the Shepp-Logan phantom, one having been corrupted with 'salt-and-pepper' noise, and the other one having been treated with a Gaussian filter, blurring the image.

The salt-and-pepper phantom is improved most by applying the Perona-Malik method with a fidelity term. This method gives a good balance between removing noise and preserving edges and details within the image.

For the blurry phantom the best result is seen using Perona-Malik, where some of the edges become more defined. However, a delicate balance has to be kept between refining the edges and blurring out any lower-contrast detail, and the total effect is limited.

The methods are also tested on images that were created using a prototype of a low-field MRI machine. The noise in these images is mostly the 'salt-and-pepper' type. Though the preferred result is somewhat subjective, the Perona-Malik method with fidelity once again gives the clearest image here.

Contents

1	Introduction	4
1.1	Research questions	5
1.2	Shepp-Logan phantom	5
1.3	Assessment method	6
2	Perona-Malik	8
2.1	Discretisation	8
2.1.1	Spatial discretisation	8
2.1.2	Time integration	10
2.1.3	Finding a stable Δt	10
2.2	Noisy phantom	11
2.3	Blurry phantom	11
3	Fidelity term	13
3.1	Discretisation	13
3.1.1	Picard implementation	13
3.2	Value of λ	14
3.2.1	Noisy phantom	14
3.2.2	Blurry phantom	15
4	Fourth order equation	17
4.1	Discretisation	17
4.1.1	Spatial discretisation	17
4.1.2	Time integration	18
4.1.3	A stable Δt	18
4.2	Noisy phantom	19
4.2.1	The despeckling algorithm	20
4.3	Blurry phantom	21
4.4	Computation time	22
5	Fourth order with fidelity	23
5.1	Discretisation	23
5.2	Noisy phantom	23
5.3	Blurry phantom	24
5.4	Conclusion	24
6	Application to MRI images	25
6.1	Perona-Malik	25
6.2	Fidelity	26
6.3	Fourth Fidelity	28
6.4	Best method	29
7	Conclusion and discussion	30
8	References	32

1 Introduction

Magnetic Resonance Imaging, or MRI, is an incredibly useful tool for doctors to be able to treat patients and analyse their internal organs. The downside is that MRI machines are also incredibly expensive, due to their use of very strong superconducting electromagnets that need to be cooled with liquid helium. This means that though most hospitals in Western countries may have access to these machines, many hospitals in developing countries simply cannot afford to purchase an MRI scanner. For many illnesses, this means the rate of recovery is much lower in developing countries since treatment is harder. One of these illnesses is hydrocephalus. Hydrocephalus is a neurological condition where cerebrospinal fluid (CSF) builds up in the skull, putting pressure on the brain. An estimated 100,000 infants are affected by it every year in Sub-Saharan Africa (Santos et al., 2017). It is usually very treatable, and the most common treatment consists of placing a shunt to drain the CSF. However, usually an MRI scanner is used to determine both the location where the shunt needs to be placed and whether the treatment was successful (Gülbiz Kartal & Algin, 2014).

A group of researchers from TU Delft, Leiden University, Penn State and Mbarara University have set up a project to develop a smaller, inexpensive MRI scanner that can be used in developing countries. One problem this smaller scanner has is that the use of a weaker magnet will inherently result in an image with a lower signal-to-noise ratio.

MR imaging works by aligning the hydrogen atoms along the strong magnetic field created by the electromagnet. A radiofrequency pulse is then fired at the atoms in a specific area, and they absorb some of that energy and then release it. That release of energy can be measured and as different kinds of tissue release different amounts of energy, the kind of tissue in the area can be determined. However, the strength of the magnetic field determines how fast the hydrogen atoms spin and how high the frequency of the released pulse is. This in turn determines the detail in the picture (Freeman, 2003).

The magnetic field in the new MRI scanner is not as strong as that in a traditional MRI scanner. This means there is more noise in the images. There are several mathematical techniques that can be used to improve these images and make them clearer. One such technique will be researched in this paper.

1.1 Research questions

The image can be seen as the initial condition of a partial differential equation (PDE). Time integration will then change the image. If the PDE is chosen the right way, this will make it less noisy. A common choice is the Perona-Malik model (Perona & Malik, 1990). This is a second order equation that works to preserve the edges within an image while smoothing out the areas between edges. In this paper, three methods will be compared to Perona-Malik:

- Perona-Malik with a fidelity term (Weickert, 1998)
- A fourth order equation (You & Kaveh, 2000)
- Fourth order equation with a fidelity term

These methods will each be researched to answer the following questions:

1. For each method, what are good choices for their parameters?
2. Which method performs best on which type of image?
3. What are the results when the methods are applied to images from the MRI prototype?

1.2 Shepp-Logan phantom

To be able to test each method, two degraded forms of the 64 by 64 Shepp-Logan phantom will be used. That way, there will be a model image to compare the result to, so the results can be judged objectively. It will be assumed that all images are 1 by 1 in size.

Figure 1 shows the Shepp-Logan phantom. This will be referred to as the 'model phantom'. This image represents a matrix with values in $[0,1]$.

In figure 2 a matrix with random numbers that are distributed normally with mean $\mu = 0$ and variance $\sigma^2 = 0.01$ has been added to the model phantom. This results in a 'salt-and-pepper' style noise, where each pixel value is different from its neighbour. This will from now on be referred to as the 'noisy phantom'.

In figure 3 a Gaussian filter with standard deviation $\sigma = 0.6$ has been applied to the model phantom. This blurs the edges within the image. This image will be referred to as the 'blurry phantom'. These phantoms will be the initial conditions ϕ^0 for the methods.

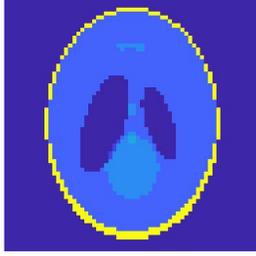


Figure 1: Model phantom

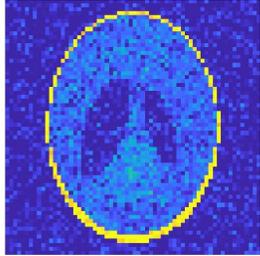


Figure 2: Noisy phantom

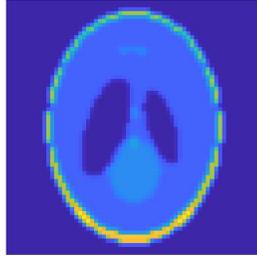


Figure 3: Blurry phantom

The images that are used to test the methods put the fourth-order ones at a disadvantage. The second-order methods tend to a step image while the fourth-order methods tend to a planar one, and our model is a step image. For the purpose of this research, it may be presumed that clearly defined edges between parts of the image are desirable, and therefore the Shepp-Logan phantom is appropriate to test the methods.

1.3 Assessment method

The parameters will be optimized using the standard deviation of the image compared to the model image, defined for every time step n as follows:

$$\sigma_{mod}(n) = \frac{\|\mathbf{w}_{mod} - \mathbf{w}^n\|}{\sqrt{N}} \quad (1)$$

where \mathbf{w}_{mod} is the vector representation of the model phantom, \mathbf{w}^n is the vector representation of the result in time step n , and N is the total number of pixels in the image. This σ_{mod} indicates how far the solution deviates from the model

on average in each time step. This means that $\Delta\sigma_{mod} < 0$ indicates that the image was improved, $\Delta\sigma_{mod} > 0$ indicates that the image became worse, and $\Delta\sigma_{mod} = 0$ indicates that there was no change in quality.

2 Perona-Malik

Perona-Malik diffusion, or anisotropic diffusion, will be the benchmark that the other methods in this paper will be held against. Perona-Malik works to preserve the edges within the image while smoothing the areas between. The equation as proposed by Perona & Malik (1990) is

$$\frac{\partial \phi}{\partial t} = \nabla \cdot (c(x, y, t) \nabla \phi). \quad (2)$$

Here $c(x, y, t)$ is the diffusion coefficient, which influences the rate at which each pixel changes. If $c \equiv 1$ equation 2 becomes the standard heat equation, and will smooth the image equally everywhere. Perona & Malik propose to let c depend on the gradient in the image and choose the function

$$c(\nabla \phi) = e^{-\left(\frac{\|\nabla \phi\|}{K}\right)^2} \quad (3)$$

where K is a fixed constant. If K becomes larger relative to $\|\nabla \phi\|$, c approaches 1 and the image will be smoothed more overall. If K becomes smaller the image will be smoothed less overall.

In equation (3), c becomes small for a large value of $\|\nabla \phi\|$ and vice versa. This means that the function will smooth the image less when there is a large difference between pixels next to each other, so at an edge.

This method does not allow dissipation across the boundary Γ of the image, so the homogeneous Neumann condition

$$c \frac{\partial \phi}{\partial \mathbf{n}} = 0 \quad (4)$$

is in effect on the boundary. A result of this boundary condition is that the sum of all pixels in the image remains the same through every time step.

2.1 Discretisation

2.1.1 Spatial discretisation

In order to be able to apply this method, equation (2) needs to be converted by spatial discretisation to be of the form

$$\frac{\partial \mathbf{w}}{\partial t} = \mathbf{A} \mathbf{w}. \quad (5)$$

where \mathbf{w} is the vector representation of the image.

Equation (2) can be rewritten as

$$\frac{\partial \phi}{\partial t} = \frac{\partial}{\partial x} \left(c \frac{\partial \phi}{\partial x} \right) + \frac{\partial}{\partial y} \left(c \frac{\partial \phi}{\partial y} \right). \quad (6)$$

If we consider the centres of the pixels, equation (6) can be estimated in point (x_i, y_j) by using central differences with the value of $f = c \frac{\partial \phi}{\partial x}$ and $g = c \frac{\partial \phi}{\partial y}$ in

the surrounding points $(x_{i+\frac{1}{2}}, y_j)$, $(x_{i-\frac{1}{2}}, y_j)$, $(x_i, y_{j+\frac{1}{2}})$ and $(x_i, y_{j-\frac{1}{2}})$, the pixel edges. This gives

$$\frac{\partial f_{i,j}}{\partial x} = \frac{f_{i+\frac{1}{2},j} - f_{i-\frac{1}{2},j}}{h} + \mathcal{O}(h^2) \quad (7)$$

$$\frac{\partial g_{i,j}}{\partial y} = \frac{g_{i,j+\frac{1}{2}} - g_{i,j-\frac{1}{2}}}{h} + \mathcal{O}(h^2). \quad (8)$$

Here, h is the width of one pixel, which is $\frac{1}{64}$ for the Shepp-Logan phantoms used for these experiments.

The values of f and g in the necessary points are estimated as follows using central differences:

$$f_{i+\frac{1}{2},j} = c(x_{i+\frac{1}{2}}, y_j) \frac{\phi_{i+1,j} - \phi_{i,j}}{h} + \mathcal{O}(h^2) \quad (9)$$

$$f_{i-\frac{1}{2},j} = c(x_{i-\frac{1}{2}}, y_j) \frac{\phi_{i,j} - \phi_{i-1,j}}{h} + \mathcal{O}(h^2) \quad (10)$$

$$g_{i,j+\frac{1}{2}} = c(x_i, y_{j+\frac{1}{2}}) \frac{\phi_{i,j+1} - \phi_{i,j}}{h} + \mathcal{O}(h^2) \quad (11)$$

$$g_{i,j-\frac{1}{2}} = c(x_i, y_{j-\frac{1}{2}}) \frac{\phi_{i,j} - \phi_{i,j-1}}{h} + \mathcal{O}(h^2) \quad (12)$$

where h is the width of a pixel.

By estimating $\|\nabla\phi\|$ in (x_i, y_j) using central differences

$$\|\nabla\mathbf{w}_{i,j}\| = \sqrt{\left(\frac{\mathbf{w}_{i+1,j} - \mathbf{w}_{i-1,j}}{2h}\right)^2 + \left(\frac{\mathbf{w}_{i,j+1} - \mathbf{w}_{i,j-1}}{2h}\right)^2},$$

a value $c_{i,j}$ for c can be estimated for the center of each pixel. However, we need values for c on the edges of the pixels, which are found by taking the average of c in the neighbouring pixel centers.

$$c_{i+\frac{1}{2},j} = \frac{c_{i,j} + c_{i+1,j}}{2}.$$

The final equation for the spatial discretisation then becomes

$$\begin{aligned} \frac{\partial\phi_{i,j}}{\partial t} = \frac{1}{h} \left\{ c(x_{i+\frac{1}{2}}, y_j) \frac{\phi_{i+1,j} - \phi_{i,j}}{h} - c(x_{i-\frac{1}{2}}, y_j) \frac{\phi_{i,j} - \phi_{i-1,j}}{h} + \right. \\ \left. + c(x_i, y_{j+\frac{1}{2}}) \frac{\phi_{i,j+1} - \phi_{i,j}}{h} - c(x_i, y_{j-\frac{1}{2}}) \frac{\phi_{i,j} - \phi_{i,j-1}}{h} \right\} + \mathcal{O}(h) \end{aligned} \quad (13)$$

Equation (13) contains a virtual point at every boundary, a point that does not exist and we have no value for. For example, to find $\frac{\partial\phi_{1,j}}{\partial t}$, the value of $\phi_{0,j}$

is needed, which doesn't exist. The boundary condition in equation (4) can be used to remove these virtual points. This BC essentially states that a point and its mirror image across the boundary are equal. For the given example, this means

$$\phi_{1,j} - \phi_{0,j} = 0.$$

This means that at the boundaries, the term in (13) that contains the virtual point will equal zero.

We neglect the error and represent the pixel at place (i, j) in the image by $\mathbf{w}(k)$, where $k(i, j) = i + n(j - 1)$. For an image that is nx by ny in size, the matrix \mathbf{A} in equation (5) will then be a square $nx \cdot ny$ by $nx \cdot ny$ matrix, with the $k(i, j)$ -th row representing the value of (13) at $\phi_{i,j}$, and the values of $\frac{1}{h^2}c$ each $\phi_{a,b}$ is multiplied with to find $\phi_{i,j}$ in the $k(a, b)$ -th element of row $k(i, j)$.

2.1.2 Time integration

For the time integration Euler Forward is used, meaning the time integration is represented as follows:

$$\frac{\partial \mathbf{w}^n}{\partial t} = \frac{\mathbf{w}^{n+1} - \mathbf{w}^n}{\Delta t} + \mathcal{O}(\Delta t) \quad (14)$$

where n represents the current time step.

The spatial and time discretisations combined give

$$\mathbf{w}^{n+1} = \mathbf{w}^n + \Delta t(\mathbf{A}\mathbf{w}^n) \quad (15)$$

which can be implemented directly in Matlab. Since \mathbf{A} contains c , which changes for every time step, \mathbf{A} depends on \mathbf{w}^n and needs to be recalculated in every time step.

2.1.3 Finding a stable Δt

If Δt is too small, the method will need a lot of time steps to progress the image. However, Δt can not be too large either, since then the method will not be stable, meaning errors in the image are amplified. Therefore the boundary for Δt past which the method will no longer be stable needs to be found. Equation (15) can be rewritten as

$$\mathbf{w}^{n+1} = (I + \Delta t\mathbf{A})\mathbf{w}^n. \quad (16)$$

$\mathbf{G}(\Delta t\mathbf{A}) = I + \Delta t\mathbf{A}$ is the amplification matrix. From van Kan, Segal, & Vermolen (2014): a sufficient condition for stability is that $\|\mathbf{G}(\Delta t\mathbf{A})\| \leq 1$.

This can be expressed using the eigenvalues λ of \mathbf{G} by $\|\lambda(\mathbf{G}(\Delta t\mathbf{A}))\| \leq 1$.

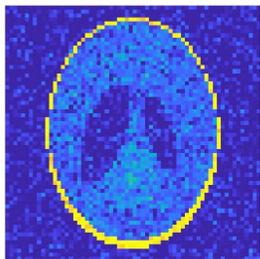
For Euler Forward this gives $|1 + \Delta t\lambda| \leq 1$ and thus $\Delta t \leq \frac{-2}{\lambda_{max}}$.

To estimate the eigenvalues of \mathbf{G} Gershgorin's Circle theorem can be used. Since c always has a value between 0 and 1, the theorem gives us that $-\frac{8}{h^2} \leq \lambda \leq 0$, so $\lambda_{max} \geq -\frac{8}{h^2}$ and $\Delta t \leq \frac{h^2}{4}$. In the case of the Shepp-Logan phantom h is defined as $h = \frac{1}{64}$ and so the method is stable for $\Delta t \leq 6.104 \cdot 10^{-5}$. For the following experiments, Δt is set at $1 \cdot 10^{-5}$

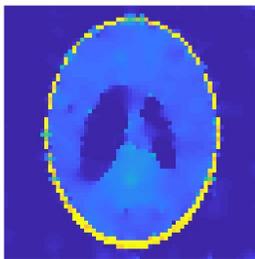
2.2 Noisy phantom

Figure 4 shows the noisy phantom before and after applying Perona-Malik. For the optimal results, K is set at 4. The number of time steps n_{max} is set at 75, since that is when the standard deviation is at its minimum for this choice of K . If n_{max} is higher than that, too much detail in the middle of the image starts to fade.

In figure 4b we see that Perona-Malik has been quite successful at clearing up the image. The two dark ovals in the middle and the lighter patch between them are clearly recognisable. The σ_{mod} of this optimal solution is 0.0521.



(a) Noisy phantom



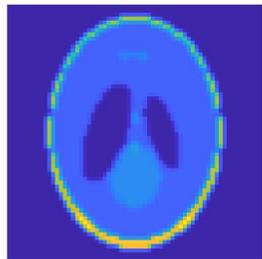
(b) Noisy phantom after Perona-Malik

Figure 4: Applying Perona-Malik to the noisy phantom, $n_{max} = 75$

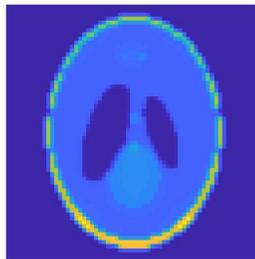
2.3 Blurry phantom

Perona-Malik is now applied to the blurry phantom. Here, the desired result is that the edges, which are unclear, are sharpened.

For most choices of K , σ_{mod} is monotonically increasing, and the image is only blurred further. If $K = 1$, so the overall diffusion is kept low, the image does improve somewhat. σ_{mod} is at its minimum, 0.0786, for $n_{max} = 75$. The result is shown in figure 5b. The improvement is most visible for the dark blue ovals, which are slightly more defined. However, the improvement is limited. If the method is left to run for more time steps, the edges of the dark blue ovals are defined further, but the details with less contrast start to fade out too much. It appears Perona-Malik will be of limited use for the improvement of the blurry phantom.



(a) Blurry phantom



(b) Blurry phantom after Perona-Malik

Figure 5: Applying Perona-Malik to the blurry phantom, $n_{max} = 75$

3 Fidelity term

The first method that will be tested to possibly improve upon Perona-Malik diffusion is the inclusion of a fidelity term.

A fidelity term ensures the image does not stray too far from the original image by creating a relation between the time derivative and the difference between the (partially) filtered image and its original. Weickert (2000) states that the equation for this method can be considered as the steady state of

$$\frac{\partial \phi}{\partial t} = \nabla \cdot (c(|\nabla \phi|^2) \nabla \phi) + \lambda(\phi^0 - \phi) \in \Omega. \quad (17)$$

The definition of c is the same as for Perona-Malik. We also use the same homogeneous Neumann boundary condition as for Perona-Malik:

$$c \frac{\partial \phi}{\partial \mathbf{n}} = 0 \text{ on } \Gamma, \quad (18)$$

The second term of the right hand side of (17) is the fidelity term. λ is a constant that is chosen to determine to what extent the fidelity term influences the outcome of the equation.

3.1 Discretisation

The discretisation of this method is mostly the same as in section 2.1, with \mathbf{A} the same matrix. However, in place of equation 5 we now have

$$\frac{\partial \mathbf{w}}{\partial t} = \mathbf{A} \mathbf{w} + \lambda(\mathbf{w}^0 - \mathbf{w}). \quad (19)$$

3.1.1 Picard implementation

The solution of this method is its steady state. Since that is the case, $\frac{\partial \mathbf{w}}{\partial t} = 0$ can be inserted into equation (19) and the method can be solved using Picard iterations.

$$\begin{aligned} \mathbf{A}(\mathbf{w}^n) \mathbf{w}^{n+1} + \lambda(\mathbf{w}^0 - \mathbf{w}^{n+1}) &= 0 \\ (\mathbf{A}(\mathbf{w}^n) - \lambda \mathbf{I}) \mathbf{w}^{n+1} &= -\lambda \mathbf{w}^0 \\ \mathbf{w}^{n+1} &= (\mathbf{A}(\mathbf{w}^n) - \lambda \mathbf{I})^{-1} (-\lambda \mathbf{w}^0) \end{aligned} \quad (20)$$

Since there is no Δt in this equation, there is no need to go through a certain number of time steps. However, \mathbf{A} depends on \mathbf{w}^n and so the method will have to be run using Picard iterations. These Picard iterations converge to a fixed point, so they will be stopped when the image no longer changes.

3.2 Value of λ

An optimal value for λ needs to be determined to be able to compare the performance of this method to others. However, this optimal value will be different for every image. For the Shepp-Logan phantom it is possible to find an optimal value by minimizing the standard deviation, and this will at least indicate the ballpark in which to choose the value of λ for other images. Of course, the goal is to find a λ which will work for a group of somewhat similar images.

3.2.1 Noisy phantom

First, the method is researched for the noisy phantom. After trying some values, it quickly becomes clear that λ must lie somewhere around 2000-3000. As is visible in figure 6a, when λ is much smaller than that value, most detail is lost from the image. Figure 6b shows that a much larger λ results in too little smoothing, and thus the details are barely any easier to make out than in the original image.



(a) $\lambda = 500$

(b) $\lambda = 6000$

Figure 6: Fidelity method for small and large λ

When the standard deviation for the result is minimized, the best value for λ for the noisy Shepp-Logan phantom turns out to be about 2500. About 25 Picard iterations are needed to reach the stable solution, whereas 75 time steps were used for Perona-Malik. As the image (and therefore the size of the vectors and matrices) gets larger, this reduction in the needed steps will quickly start to make a difference in the computation time. The results can be seen in figure 7.

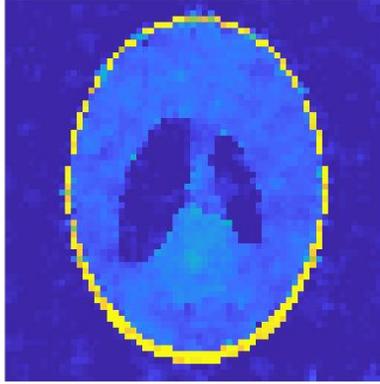


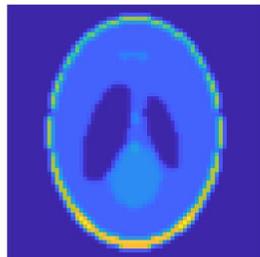
Figure 7: The fidelity method with optimal $\lambda = 2500$

It is clear that the fidelity term results in a less smooth image than Perona-Malik does. However, σ_{mod} is 0.0507 for this method as opposed to 0.0521 for Perona-Malik. The difference is small, but the fidelity term works slightly better to preserve details in the image.

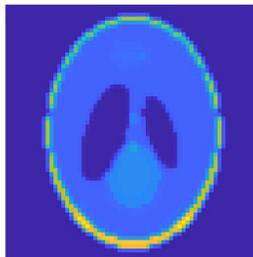
3.2.2 Blurry phantom

For small λ , the edges of the dark ovals are defined but most other detail is lost. Meanwhile for larger λ the image does not lose as much detail, but the edges are not defined as well. Of course, as λ goes toward ∞ , the image is kept closer and closer to the blurry phantom and so the limit of σ_{mod} is the σ_{mod} of the blurry phantom.

The best balance between definition of edges and preservation of detail, and the lowest σ_{mod} , is found at around $\lambda = 1200$. The result is shown in figure 8b. Clearly, not much has changed. The decrease in σ_{mod} is at a scale of $1 \cdot 10^{-5}$. It appears this method will be of limited use for blurry images.



(a) Blurry phantom



(b) Blurry phantom after fidelity, $\lambda = 1200$

Figure 8: Blurry phantom before and after applying fidelity.

4 Fourth order equation

So far this paper has only explored second order PDE's for noise filtering. These methods perform reasonably well to remove the noise and preserve the edges simultaneously, but they have disadvantages as well. The Perona-Malik method minimizes the gradient of the image, and the gradient is at its minimum when the image is (piecewise) constant. This means that the method will ultimately tend to create an image that consists of blocks of the same colour (a step image). This is not always desirable.

You and Kaveh (2000) propose a different method using a fourth order PDE. This will minimize the Laplacian, and therefore deliver a (piecewise) planar image instead of a constant image.

They give the formula for the fourth order method as

$$\frac{\partial \phi}{\partial t} = -\nabla^2 [c(|\nabla^2 \phi|) \nabla^2 \phi] \in \Omega \quad (21)$$

c is now defined by

$$c(|\nabla^2 \phi|) = e^{-\left(\frac{|\nabla^2 \phi|}{\kappa}\right)^2}. \quad (22)$$

Like the Perona-Malik method, this method uses symmetric boundary conditions.

$$\frac{\partial \phi}{\partial \mathbf{n}} = 0 \quad (23)$$

$$c \frac{\partial \nabla^2 \phi}{\partial \mathbf{n}} = 0 \quad (24)$$

4.1 Discretisation

4.1.1 Spatial discretisation

The spatial discretisation for the fourth order method is also given by You and Kaveh (2000). It happens in several stages.

First, the rightmost Laplacian in equation (21) is discretised, adding together the second order central differences in the x- and y-direction. This gives:

$$\nabla^2 \mathbf{w}_{i,j} = \frac{\mathbf{w}_{i+1,j} + \mathbf{w}_{i-1,j} + \mathbf{w}_{i,j+1} + \mathbf{w}_{i,j-1} - 4\mathbf{w}_{i,j}}{h^2} + \mathcal{O}(h^2). \quad (25)$$

At the edges, the symmetry as given by (23) is used to remove the virtual points. For example, at a point $(1, j)$, we have that $\mathbf{w}_{0,j} = \mathbf{w}_{1,j}$. Using this in equation (25) gives

$$\nabla^2 \mathbf{w}_{1,j} = \frac{\mathbf{w}_{2,j} + \mathbf{w}_{1,j+1} + \mathbf{w}_{1,j-1} - 3\mathbf{w}_{1,j}}{h^2} + \mathcal{O}(h^2).$$

We see that for every edge a pixel is at, one term disappears and the factor by which $\mathbf{w}_{i,j}$ is multiplied decreases by one.

Second, the value of the function

$$\mathbf{f}(\nabla^2 \mathbf{w}) = c(|\nabla^2 \mathbf{w}|) \nabla^2 \mathbf{w}$$

is calculated. We define

$$\mathbf{f}_{i,j} = \mathbf{f}(\nabla^2 \mathbf{w}_{i,j}). \quad (26)$$

$c_{i,j}$ for each point is found through

$$c_{i,j} = e^{-\left(\frac{|\nabla^2 \mathbf{w}_{i,j}|}{K}\right)^2}. \quad (27)$$

Last, the Laplacian of f is found with

$$\nabla^2 \mathbf{f}_{i,j} = \frac{\mathbf{f}_{i+1,j} + \mathbf{f}_{i-1,j} + \mathbf{f}_{i,j+1} + \mathbf{f}_{i,j-1} - 4\mathbf{f}_{i,j}}{h^2} + \mathcal{O}(h^2), \quad (28)$$

using the boundary condition (24) to remove the virtual points. The complete method, then, after neglecting the truncation errors, is comprised of a multiplication of \mathbf{w} by three matrices:

$$\frac{\partial \mathbf{w}}{\partial t} = -\mathbf{LCLw}$$

The first and third matrix are the same, \mathbf{L} , representing the Laplacian. This matrix is equal to the one used for Perona-Malik if $c \equiv 1$ in that method. The second matrix \mathbf{C} is the multiplication with c from equation (27).

4.1.2 Time integration

Like for Perona-Malik, the method uses Euler Forward for the time integration. This gives

$$\mathbf{w}^{n+1} = \mathbf{w}^n - \Delta t (\mathbf{LC}(\mathbf{w}^n) \mathbf{Lw}^n). \quad (29)$$

\mathbf{L} is constant, but \mathbf{C} depends on \mathbf{w}^n and needs to be recalculated in every time step.

4.1.3 A stable Δt

To find a Δt for which the method is stable, it can be estimated using the eigenvalues of the matrices in the method. In a way similar to section 2.1.3, but different by a negative sign, we find that $|1 - \Delta t \lambda| \leq 1$ and so $\Delta t \leq \frac{2}{\lambda_{max}}$ and $\lambda_{max}(\mathbf{L}) \geq -\frac{8}{h^2}$. By definition, $0 \leq c \leq 1$, so $\lambda_{max}(\mathbf{C}) \leq 1$. This gives the following.

$$\begin{aligned} |\lambda_{max}(\mathbf{LCL})| &= \|\mathbf{LCL}\|_2 \\ &\leq \|\mathbf{L}\|_2 \|\mathbf{C}\|_2 \|\mathbf{L}\|_2 \\ &\leq \frac{8}{h^2} \cdot 1 \cdot \frac{8}{h^2} \\ &= \frac{64}{h^4}. \end{aligned}$$

This can be inserted into the previous inequality found for Δt to see that for stability $\Delta t \leq 1.863 \cdot 10^{-9}$, which is the square of the maximum Δt for Perona-Malik. In the following experiments, $\Delta t = 1 \cdot 10^{-10}$ is used.

4.2 Noisy phantom

Figure 9 shows what happens when we run the method for a very large n_{max} . We see that both the internal area of the phantom and the surrounding area consist solely of smooth transitions, as we would expect from our equation. The edge is mostly preserved as it was, because our c is very close to 0 in those areas as a result of the large difference between neighbouring pixels.

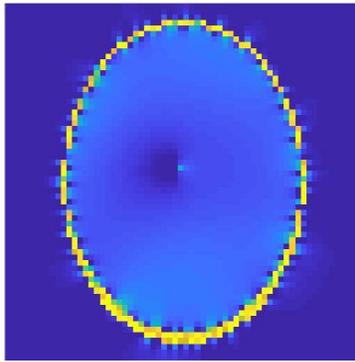


Figure 9: Fourth order solution with $n_{max} = 200000$

Figure 10 shows the result of applying the fourth order method to the noisy phantom with optimal $n_{max} = 1400$ and $K = 1400$. Almost all of the features from the model phantom are recognizable in this image, though sometimes faintly and with the possible exception of the lightblue area at the top of the model phantom. The result is a lot less blotchy than that of the Perona-Malik model. This is because in the Perona-Malik method the execution had to be halted before the diffusion faded out too many details, and thus less smoothing was done on areas that are smooth in the model image. This happens because Perona-Malik ultimately approaches a step image. Using the fourth order method, however, the minimization of the Laplacian instead of the gradient means that details will not fade away as fast, though their edges will fade out. $\sigma_{mod} = 0.0573$ for this result. This is higher than it was for the previous methods. Presumably, the fact that the model image is a step image plays a role in this, since the second order methods will tend to a step image and the fourth order will tend to a planar one.

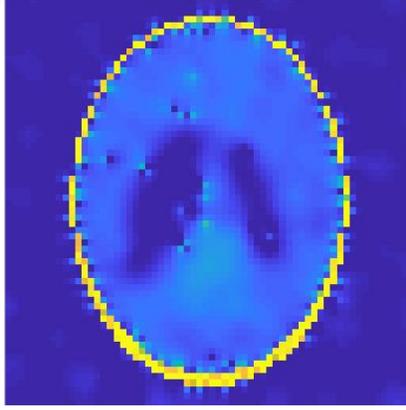


Figure 10: The fourth order method applied to the noisy phantom, $n_{max} = 1400$, $K = 1400$

4.2.1 The despeckling algorithm

We see in figure 10 that there are a few spots around the darker areas in the middle of the phantom where there are so-called speckles. These are pixels whose values are so different from their neighbours in the original image that our c -matrix identifies them as edges. Therefore they are not properly smoothed during the operation of the function. The article by You and Kaveh (2000) proposes a fairly straightforward algorithm to remove these remaining speckles after the main function has finished.

They say the following:

“Denote the mean and variance of the neighbouring pixels around a pixel (i, j) as

$$m = \frac{u_{i,j-1} + u_{i,j+1} + u_{i-1,j} + u_{i+1,j}}{4} \quad (30)$$

and

$$\sigma^2 = \frac{u_{i,j-1}^2 + u_{i,j+1}^2 + u_{i-1,j}^2 + u_{i+1,j}^2}{4} - m^2. \quad (31)$$

Then, this pixel is a speckle if σ is small and $|u_{i,j} - m| \gg \sigma$ ”

This makes sense: if σ is small, that means the surrounding pixels are relatively close together in value and therefore $u_{i,j}$ is probably in an area of the image that should be quite smooth. If the second condition is fulfilled that implies that the difference between $u_{i,j}$ and its neighbours is relatively large. Therefore these two terms make logical conditions for a speckle.

You and Kaveh propose the following choice for the new value of $u_{i,j}$:

$$u_{i,j} = \begin{cases} m, & \text{if } |u_{i,j} - m|^2 > k\sigma; \\ u_{i,j}, & \text{otherwise;} \end{cases} \quad (32)$$

Here k is a constant that can freely be chosen to influence the amount of pixels that will be changed to the mean. Of course, if k is too large none of the pixels will meet the condition in equation (32). However, as is visible in figure 11a, choosing k too low means that pixels in the yellow edge also meet the condition and are removed. After some experimentation a choice of $k = 4$ turns out to give a reasonable balance between removing the speckles and retaining the yellow edge, but there is still a not insignificant amount of loss of the edge. This edge deterioration results in a much higher σ_{mod} , which is now 0.0749.

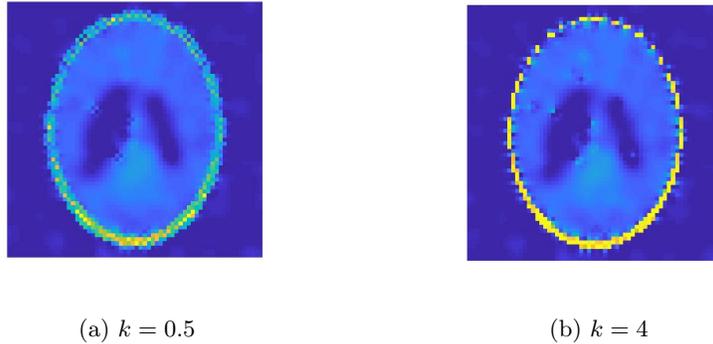


Figure 11: Filtered SL-phantom after despeckling algorithm

Loss of details like this does seem likely to be a problem largely specific to images where there are details a few pixels wide with values that are extremely different from their neighbours. This means that whether this algorithm will add to the result will depend on the shape and resolution of the image being processed. It also means that optimally the resolution of an image needs to be large enough relative to the scale of the results that details like this are at least a few pixels in size.

4.3 Blurry phantom

This method is now tested on the blurry phantom. After testing multiple values of K , we find that for a small enough K the fourth order method can result in some improvement in the image. However, due to the very small Δt , σ_{mod} only decreases by roughly $5 \cdot 10^{-9}$ in 10000 time steps, which means it would take far too long to make any visible changes to the image. This method is not a practical way to improve the blurry phantom.

4.4 Computation time

As the resolution of an image becomes larger, the method becomes very slow computationally. The fourth order method needs a much larger number of time steps to reach a result than Perona-Malik does, since Δt needs to be quadratically smaller for stability. Each time step requires two matrix-vector multiplications, and so the computation time is much higher than for the second-order methods. This is a big disadvantage of the fourth order method.

5 Fourth order with fidelity

As stated in section 4.4, the fourth order method becomes very slow to solve as the images become larger. One way to potentially solve this problem is by combining the fourth order method with the fidelity method. In this chapter that combination will be researched and compared to the separate methods.

A fidelity term is added to the fourth order equation. The equation for this method is

$$\frac{\partial \phi}{\partial t} = -\nabla^2 [c(|\nabla^2 \phi|) \nabla^2 \phi] + \lambda(\phi_0 - \phi) \in \Omega. \quad (33)$$

c and the boundary conditions are all the same as for the Fourth order method.

5.1 Discretisation

The spatial discretisation for the first term on the right hand side of equation (33) is the same as for the Fourth order method. We get

$$\frac{\partial \mathbf{w}}{\partial t} = -\mathbf{LCL}\mathbf{w} + \lambda(\mathbf{w}^0 - \mathbf{w}) \quad (34)$$

As for Perona-Malik with fidelity, the solution being the steady state of (33) means that $\frac{\partial \mathbf{w}}{\partial t} = 0$ can be inserted and the method can be solved with Picard iterations.

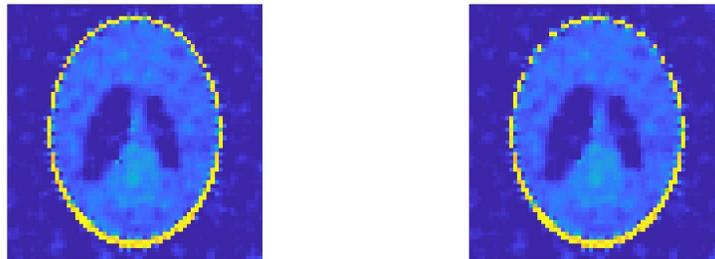
$$\begin{aligned} -\mathbf{LCL}\mathbf{w}^{n+1} + \lambda(\mathbf{w}^0 - \mathbf{w}^{n+1}) &= 0 \\ (-\mathbf{LCL} - \lambda\mathbf{I})\mathbf{w}^{n+1} &= -\lambda\mathbf{w}^0 \\ \mathbf{w}^{n+1} &= (-\mathbf{LCL} - \lambda\mathbf{I})^{-1}(-\lambda\mathbf{w}^0) \end{aligned} \quad (35)$$

Here, only \mathbf{C} depends on \mathbf{w}^n .

5.2 Noisy phantom

Once again values for λ and K need to be found. In this case, a choice of around $\lambda = 45000000$ with about $K = 1750$ delivers the best result. This is roughly the square of the best result for Perona-Malik with a fidelity term. This can be explained by the fact that the vector \mathbf{w} is now multiplied with a matrix twice, and those matrices both have eigenvalues of roughly the same size as the one for Perona-Malik did. Therefore the vector is multiplied by roughly the square of what it was for Perona-Malik, and thus λ needs to be squared to influence the diffusion in the same way. Figure 12a shows the result. Compared to the Fourth order, this figure preserves the details much better. σ_{mod} for this image is 0.0596, which is higher than for the previous methods. Figure 12b shows the result of using the despeckling algorithm on the image. Since the image only contains a few speckles, the despeckling algorithm does more harm than good here, removing a significant portion of the yellow ellipse. λ can be lowered to

smooth out the areas a little more, depending on what the desired use of the images is. Though the result is not quite as smooth as that from the fourth order method, this method is much, much faster computationally. Since the number of Picard iterations needed to converge to a solution (about 50) is much smaller than the number of time steps needed to find a good result for the Fourth order method, this method could be very useful to achieve similar effects to those of the Fourth order method without the long computation times.



(a) Fourth Fidelity method, $\lambda = 45000000$ (b) After despeckling algorithm

Figure 12: Fourth order method with fidelity applied to noisy phantom

5.3 Blurry phantom

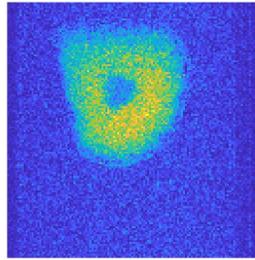
Adding a fidelity term does not improve upon the results from the Fourth order method. Though it is still possible to improve the image, the improvement is so small that it is not visible in the image. Through experimentation, we find that this is the same for all choices for λ . It is not surprising that this method does not help, since the difference from the Fourth order method is that this method keeps the image close to its original, just slowing any change more.

5.4 Conclusion

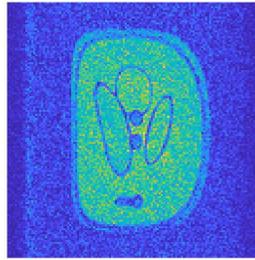
It can be concluded that although this combination of the two methods does not necessarily improve on the results from the fourth order method, it is still very useful. This is because this method improves upon the computation time massively.

6 Application to MRI images

The Perona-Malik, Fidelity and Fourth Fidelity methods will now be applied to results from the MRI prototype. The Fourth order method is already very computationally inefficient on the Shepp-Logan phantom which is half as big as these images; therefore this method will not be considered further since it will not be of use in practice. The image in figure 13a is a scan of an orange slice, and the image in figure 13b is scan of a recreation of the Shepp-Logan phantom. Of course, it is not practical to have to choose new optimal parameters every time a new image is processed. Therefore, the parameters will be optimized on figure 13a and then tested on figure 13b to see whether they also work for a different, yet similar image.



(a) Orange image



(b) Shepp-Logan image

Figure 13: Results from MRI prototype

6.1 Perona-Malik

First, the Perona-Malik method is calibrated to the Orange image. To balance the total diffusion and the edge preservation in the image, K in equation (3) is set to 60. Then n_{max} is set to 10. This gives the image in figure 14. Most of the speckles have disappeared, while the shape of the orange is still very well defined. If n_{max} is made much larger, the details of the orange begin to fade out significantly.

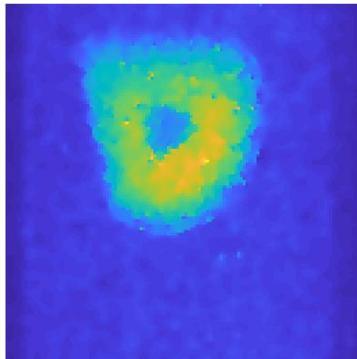


Figure 14: Perona-Malik applied to Orange image

The method is now applied to the SL image using the same parameters. The result is figure 15. Like the Orange image, all edges are clearly defined and all details are distinguishable.

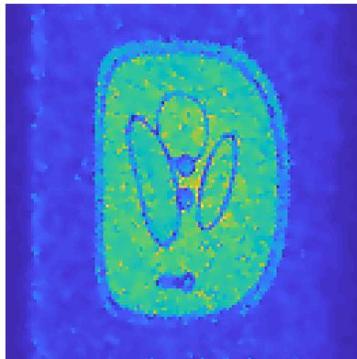


Figure 15: Perona-Malik applied to SL image

6.2 Fidelity

For the fidelity method, K is left at 60. λ is set at 5000. The effect on the Orange image is quite dramatic. In figure 16 nearly all graininess has disappeared from the image. The edges of the orange have been redefined, and the algorithm has done great at preserving the different shades within the orange. This result

looks much better than the result for Perona-Malik, since here more smoothing is possible without fading details.

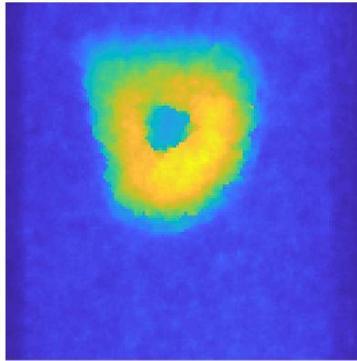


Figure 16: Fidelity method applied to Orange image

When the fidelity method is applied to the SL image as in figure 17 and compared to figure 15, the effect is not quite as dramatic. However, the details and edges in the image are clearly visible, while the areas in between them have been smoothed out. This smoothing is the main difference between figures 17 and 15, and the fidelity method performs better here. The definition of the edges is about the same between the methods.

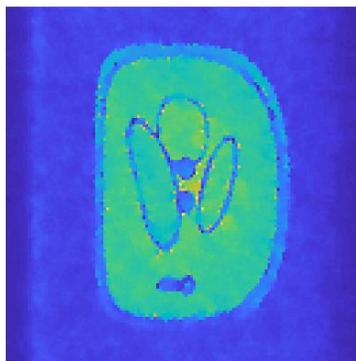


Figure 17: Fidelity method applied to SL image

6.3 Fourth Fidelity

Finally, the Fourth Fidelity method is tested on the images. For a good result, K in equation (3) now needs to be set much higher, at 10000. λ , however, is set to 75. Figure 18 shows the result for the Orange image. This image is not as polished-looking as figure 16. It does preserve more 'detail', but it is not clear whether that is detail that is actually part of the image or whether that results from the noise.

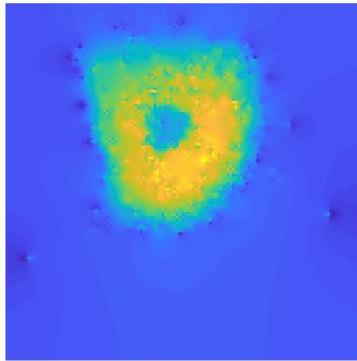


Figure 18: Fourth order method with fidelity applied to Orange image

This method is applied to the SL image in figure 19. When the result is compared to the SL result for the other methods, the interior of the phantom is a lot brighter. This is a result of the despeckling algorithm: since after filtering the high-valued (yellow) pixels outnumber the lower-valued (blue) ones in the interior, the algorithm raises the value for a lot of pixels and therefore brightens the interior.

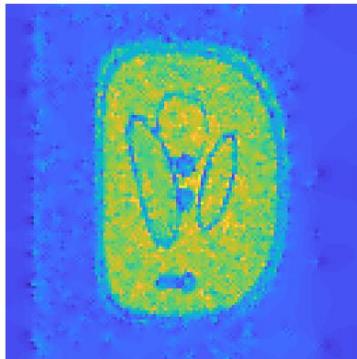


Figure 19: Fourth order method with fidelity applied to SL image

6.4 Best method

Both the fidelity method and the Fourth Fidelity method work well on the problems and improve them more than the Perona-Malik method does. For these images, the fidelity method gives a smoother, more polished-looking result, while still preserving the edges in the images very well. The Fourth Fidelity method smooths edges out a little more. For the purpose for which this smoothing is ultimately needed in this case, to produce clear medical images, clear areas with sharp edges are probably preferable. Therefore, for this purpose the fidelity method is a better fit. The fidelity method would also be a better fit if the images needed to be interpreted by a computer, since a computer is best at noticing sharp transitions.

7 Conclusion and discussion

In this paper four methods of using PDE's for noise reduction were compared: Perona-Malik using a second-order PDE, a method with a fourth-order PDE, and both of these methods with a fidelity term added to them. To compare them, the methods were applied to distorted Shepp-Logan phantoms with optimised parameters. The standard deviation of the resulting image from the model image was then used to judge the results.

For the noisy phantom, the lowest standard deviation, and therefore the best result, was found for the PM-fidelity method. This fidelity method had the best balance between edge- and detail preservation.

Perona-Malik smoothes out the noise, but will quickly deteriorate details with low contrast.

While the fourth-order method may deliver an image with smoother transitions in the interior, this method is very expensive computationally and not practical for use on larger images. Smooth transitions may also be a disadvantage if a computer is used to interpret and recognise parts of the image.

The fourth fidelity method solves the problem of computation time, but still does not perform better than the PM-fidelity method.

Perona-Malik redefines the edges of the high-contrast parts of this phantom. However, the parts with lower contrast start fading fast.

Using the fourth-order method, it was possible to improve the image, but by such a small amount that this is not visible to the naked eye for a practical number of time steps.

When applied to the blurry phantom, the fidelity terms do not improve upon the methods they are added to in terms of edge definition, since they keep the result close to the original and therefore prevent the edges from being defined more.

Out of the methods tested here, Perona-Malik performs best on the blurry phantom.

When the methods were tested on MRI-images the PM-fidelity method delivered the best results, making edges within the image stand out clearly and redefining the interior of the images. The fourth fidelity method also performed well, and the choice for which method is best will depend in large part on the desired effect.

This paper does not investigate the effect of the methods on other types of images. Extending the scope beyond the use for MRI images, this paper could also apply to other noise reduction projects with different images. The Shepp-Logan phantom is a step image with clearly defined zones. Depending on the purpose for which filtering is needed, it may be useful to further research the effect the methods have on other types of images, like planar images or photographs.

Once a method had been chosen, the parameters would need to be calibrated to give a good result. However, presuming images have similar types and amounts of noise every time, calibrating the parameters on one or a few images should enable them to be used for all similar images.

8 References

Freeman, R. (2003). *Magnetic Resonance in Chemistry and Medicine* [Epub]. Oxford University Press. Geraadpleegd van <https://tudelft.on.worldcat.org/oclc/800123639>

Gülbiz Kartal, M., & Algin, O. (2014). Evaluation of hydrocephalus and other cerebrospinal fluid disorders with MRI: An update. *Insights into Imaging*, 5(4), 531–541. <https://doi.org/10.1007/s13244-014-0333-5>

van Kan, J., Segal, A., & Vermolen, F. (2014). *Numerical Methods in Scientific Computing* (2nd edition). Delft, the Netherlands: Delft Academic Press.

Perona, P., & Malik, J. (1990). Scale-space and edge detection using anisotropic diffusion. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 12(7), 629–639. <https://doi.org/10.1109/34.56205>

Santos, M. M., Rubagumya, D. K., Dominic, I., Brighton, A., Colombe, S., O'Donnell, P., . . . Härtl, R. (2017). Infant hydrocephalus in sub-Saharan Africa: the reality on the Tanzanian side of the lake. *Journal of Neurosurgery: Pediatrics*, 20(5), 423–431. <https://doi.org/10.3171/2017.5.PEDS1755>

Weickert, J. (1998). *Anisotropic Diffusion in Image Processing*. Delft: Vieweg+Teubner Verlag.

You, Y., & Kaveh, M. (2000). Fourth-order partial differential equations for noise removal. *IEEE Transactions on Image Processing*, 9(10), 1723–1730. <https://doi.org/10.1109/83.869184>