Answers to the exam in
Medical image analysis - TBMI02
2010-12-22

The answers presented below may be more exhaustive compared to what is normally required at the exam.

Part 1

1. a) There are several reasons why the correct solution is not given in one iteration.
   The assumptions that we make for the algorithm are not valid everywhere in the image. Due to the aperture problem only a few pixels (corners, edges, lines) contribute to the solution. Flat areas do not contribute at all. This also leads to the problem that there are more than one solution in many pixels.
   The solution that is given by one iteration is the best in an average sense (since we average the equation system over all the pixels), but it might not be locally optimal for all pixels.
   b) One way is to do the registration in several scales. Mats can lowpass filter and downsample the images a number of octaves and for example start the registration at the resolution of 512 x 512 pixels to find an approximate solution. The registration then continues on the resolution 1024 x 1024 pixels to improve the solution and finally a few iterations are made on the original resolution. To run the algorithm on 512 x 512 pixels is 16 times faster than on 2048 x 2048 pixels. By using several scales, the algorithm is also able to handle larger translations and rotations than if only one scale is used.

2. a) The Hermitian property in 2D states that \( f(x, y) = f(-x, -y)^* \), i.e. that the function value at \((x, y)\) is the same as the value at \((-x, -y)\) but with a complex conjugate. This is for example true for the Fourier transform of a real valued signal (but not for a complex valued signal). This means that it is sufficient to sample half of k-space, since the other half can be calculated by using the Hermitian property.
   b) One advantage is that the low frequencies (which contain most of the energy in natural images) are sampled more than once. Due to this the signal to noise ratio (SNR) increases (we can average out noise).
   One disadvantage is that the sampling time is increased.

3. \[
\min \epsilon^2 = \sum_{x=0}^{2} \left[ w(x) (\alpha + \beta x - 3x) \right]^2 \quad w(x) = [1, 1, 3]
\]
   \[
\epsilon^2 = 1 (\alpha - 1)^2 + 1 (\alpha + \beta - 1)^2 + 9 (\alpha + 2 \beta - 9)^2
\]
   Compute \( \frac{d\epsilon^2}{d\alpha} = 0 \) and \( \frac{d\epsilon^2}{d\beta} = 0 \) and solve \( \alpha = 0.22 \) och \( \beta = 4.35 \).
   weighted error = \( \sqrt{\frac{\sum_{x=0}^{2} w(x) (\alpha + \beta x - 3x)^2}{\sum_{x=0}^{2} w(x)^2}} \) = 0.53
4. a) In the middle image there is spatial aliasing (note that the head is bigger than in the other images) due to the fact that the distance between the samples in k-space is too large. The solution is to sample more dense in k-space.

b) The right image is blurry since we do not sample the high frequencies. The solution is to sample further out in k-space.

5. **watersheds** correspond to edges of regions, i.e. where we have high gradients in the image **catchment basins** correspond to regions with low gradients, i.e. no edges, these regions are surrounded by the edges

The water will be divided by the watersheds and fall into the catchment basins and form different regions.

6. a) The local phase is a better representation since it is invariant to the intensity in the image and it can thereby handle registration of images with different intensity. The local phase is better suited for the assumptions that are necessary in the optical flow algorithm.

b) The local phase from quadrature filters describe a signal only for a certain direction. Therefore it is necessary to use a certainty that tells us which filters to trust in each point. It is also important to only use the phase if the magnitude of the filter response is high, otherwise the phase is more or less noise.

Since the phase is periodic with \(2\pi\), and the fact that the uncertainty increases with the phase difference, it is also important to give a low weight to big phase differences (a big phase difference indicates that the filters have picked up different things in the reference image and in the altered image).

7. To facilitate the filter optimization we optimize the different features of the filter in different domains. To maintain a good spatial locality and limit the effective spatial size of the filter a spatial ideal function and a spatial weight function are used.

The spatial ideal filter is the most compact filter we know, an impulse function which is zero everywhere except at the origin. The spatial weight function is usually a quadratic function, \(x^2\), which is zero at the origin and prevent the use of large filter coefficients close to the border of the filter.
8. a) The Fourier domain is periodic with a period of $2\pi$. For $N=3$

$$u = [-\frac{2\pi}{3}, 0, \frac{2\pi}{3}] (n \cdot 2\pi)$$

gives a uniform sampling that is symmetric and includes the origin.

b) 

$$u = \sum_{k=-\frac{N-1}{2}}^{\frac{N-1}{2}} k \frac{2\pi}{N}$$

9. The noise is amplified in an orientation that is perpendicular to the local structure. The adaptive filtering is performed in the 'opposite direction'. This may be caused by an accidental confusion of $e_1e_1^T$ and $e_2e_2^T$ in the construction of the control tensor.

10. If the Fourier transform of $f(x)$ is given by $F(u)$ and $x$ and $u$ are one-dimensional variables the Fourier transform of $f(\alpha x)$ is given by:

$$|\alpha|^{-1} F(\alpha^{-1} u).$$

11. True False

1. The $\gamma_2$-image is low in areas that just contain noise. x

2. $\mu_2$ is a mapping of $\lambda_1/\lambda_2$. x

3. $\|C\| = 0$ implies that only the lowpass filter is used. x

4. If the SNR in the image is increased the $\sigma$ parameter in the $m$-function should be decreased. x

12. Filter 1 is optimized with frequency weight 1 and spatial weight 2.

Filter 2 is optimized with frequency weight 1 and spatial weight 1.

Filter 3 is optimized with frequency weight 2 and spatial weight 1.

Filter 4 is optimized with frequency weight 2 and spatial weight 2.
13. a) $T_1$ is related to the relaxation of the spin vector in the $z$-direction, i.e. how long it takes for the spin vector to get back to $z$ after the excitation. $T_1$ is related to the repetition time, we have to wait a certain time before we can excite the protons again, otherwise there is no $z$-component to flip down and we do not get any signal.

$T_2$ is related to the dephasing of the spin vectors in the $x$-$y$-plane, how long it takes for the spins to get out of phase. $T_2$ is related to the echo time, we have to wait a certain time to form an echo where the spins are in phase, to get an as good signal as possible.

b) Use two facts, phase = integral of frequency and $\omega = \gamma B$.

$B = g_x(t)\hat{x} + g_y(t)\hat{y} + g_z(t)\hat{z}$

The Fourier transform of $m(x)$ can thereby be written as

$$F\{m(x)\} = \int_{-\infty}^{\infty} m(x)e^{-i\varphi(x,t)}dx = \int_{-\infty}^{\infty} m(x)e^{-i\int_0^t (g_x(\tau) x + g_y(\tau) y + g_z(\tau) z) d\tau}dx$$

If we compare the exponents of the more common way of writing the multidimensional Fourier transform

$$F\{m(x)\} = \int_{-\infty}^{\infty} m(x)e^{-i2\pi u^T x}dx$$

where

$$u = (k_x, k_y, k_z)^T$$

we get that the positions of the sample in k-space at timepoint $t$, $k_x(t)$, $k_y(t)$ and $k_z(t)$, are given by

$$k_x(t) = \frac{\gamma}{2\pi} \int_0^t g_x(\tau) d\tau, \quad k_y(t) = \frac{\gamma}{2\pi} \int_0^t g_y(\tau) d\tau, \quad k_z(t) = \frac{\gamma}{2\pi} \int_0^t g_z(\tau) d\tau$$

where $g_x$, $g_y$ and $g_z$ are the gradient fields in the $x$-, $y$-, and $z$-direction respectively.

14. Two quadrature filters, of the type used in the mini-project, has a zero scalar product. The filters have the same radial function.

a) How are the filters related to each other? (2p)

b) Show that both filters will always have the same output magnitude when convolved with a real signal. (2p)

**ANSWER**

a) Let the filters be directed in opposite directions (i.e. one in direction $\mathbf{n}$ and the other in direction $-\mathbf{n}$). It is then easy to see that the scalar product will be zero since one of the filter will be zero (and hence also their product) everywhere in the Fourier domain, i.e:

$$< Q(u), Q(-u) > = \int_u Q(u) Q(-u) du = \int_u 0 du = 0$$

b) The two filters in a) can in the spatial domain be written $q(x)$ and $q(x)^*$. Since the signal is real the contribution from an image point for one filter will be the complex conjugate of the contribution from that point to the other filter. Thus, the filter responses (which is just a sum from all contributing image points) will also be the complex conjugate of each other, i.e. will have the same magnitude.
15. Below is an example of what the process can look like.

- Define internal forces, i.e. the forces that determine how the snake moves. These forces are defined such that the arc length of the curve is minimized. This can for example be done using the neighbouring nodes to calculate the normal direction of each node and take a step in this direction.
- Define the external forces, i.e. the forces that are calculated from the image and prevents the snake from moving too far. The gradients of the image can be used to localize edges that should catch the snake. The external forces are pointed in the opposite direction of the internal forces.
- Calculate internal and external forces for each node.
- Get new positions of the nodes by adding the internal and external force to the position of the node \( F_{\text{new}} = F + F_{\text{int}} + F_{\text{ext}} \).
- Iterate the algorithm until the snake has segmented the object.

Problem 1

The problem is that the internal forces have been defined to minimize the arc length of the contour. This will lead to that the curve will get stuck in the convex hull of the object, it does not want to grow in along the contour of the object since this will increase the arc length. This is solved by always pointing the internal force towards the object (using a cross product).

Problem 2

The loop that has been created is caused by the fact that the nodes are moved closer and closer to each other and finally crosses each other. The curve is turned inside out and grows in the wrong direction. To prevent this some sort of cluster elimination should be used. For example rearrange the nodes in each iteration such that they always are equally distributed, or resample the curve and add or remove nodes where it is necessary.

16. We want to produce a 2D tensor that has only one positive non-zero eigenvalue if the signal is simple. We remember that the basic idea given by:

\[
T = \sum_k q_k (\hat{n}_k \hat{n}_k^T - \alpha I)
\]

where \(\hat{n}_k\) are the filter orienting unit vectors and \(q_k\) are the magnitudes of the corresponding quadrature filter outputs given by:

\[
q_k = A (\hat{n}_k^T \hat{x})^2
\]

for a simple signal with orientation \(\hat{x}\) and amplitude \(A\). The filter orienting vectors are given by:

\[
\hat{n}_k = \begin{pmatrix} \cos(k \pi/4) \\ \sin(k \pi/4) \end{pmatrix} \quad k = 0, 1, 2, 3
\]

a) What is the determinant of \(T\) for \(\alpha = 0\)? (1p)

b) Which value should \(\alpha\) have for \(T\) to have only one non-zero eigenvalue? (3p)

**HINT:** We remember that the tensor eigenvalues are rotation invariant for simple signals, i.e. we can choose an ‘easy’ \(\hat{x}\). For example \(\hat{x} = (1, 0)^T\).
**ANSWER:**

Using the hint and letting $\hat{x} = (1, 0)^T$ we get:

$$q_k = A \left( \hat{n}_k^T \begin{bmatrix} 1 \\ 0 \end{bmatrix} \right)^2 = A \left( \cos(k \pi/4), 0 \right) \begin{bmatrix} 1 \\ 0 \end{bmatrix}^2 = A \cos(k \pi/4)^2$$

This yields: $q_1 = A$, $q_2 = 0.5A$, $q_3 = 0$ and $q_4 = 0.5A$.

We also need the corresponding tensors $\hat{n}_k \hat{n}_k^T$ for each $k$.

$$\hat{n}_1 \hat{n}_1^T = \begin{bmatrix} 1 & 0 \\ 0 & 0 \end{bmatrix}; \quad \hat{n}_2 \hat{n}_2^T = \begin{bmatrix} 0.5 & 0.5 \\ 0.5 & 0.5 \end{bmatrix}; \quad \hat{n}_3 \hat{n}_3^T = \begin{bmatrix} 0 & 0 \\ 0 & 1 \end{bmatrix}; \quad \hat{n}_4 \hat{n}_4^T = \begin{bmatrix} 0.5 & -0.5 \\ -0.5 & 0.5 \end{bmatrix}$$

Calculating $T$ with $\alpha = 0$ we get:

$$T = A \left[ \begin{array}{cc} 1 & 0 \\ 0 & 0 \end{array} \right] + 0.5A \left[ \begin{array}{cc} 0.5 & 0.5 \\ 0.5 & 0.5 \end{array} \right] + 0 \left[ \begin{array}{cc} 0 & 0 \\ 0 & 1 \end{array} \right] + 0.5A \left[ \begin{array}{cc} 0.5 & -0.5 \\ -0.5 & 0.5 \end{array} \right]$$

i.e we get: $T = A \left[ \begin{array}{cc} 1.5 & 0 \\ 0 & 0.5 \end{array} \right]$

The answer to a) is then directly seen as the determinant of $T$ is $1.5A \times 0.5A = 0.75A^2$.

Whith $\alpha > 0$ we get the tensor to subract as: $\sum_k q_k \alpha I = 2A \left[ \begin{array}{cc} \alpha & 0 \\ 0 & \alpha \end{array} \right]$ and $T$ will be given by:

$$T = A \left[ \begin{array}{cc} 1.5 - 2\alpha & 0 \\ 0 & 0.5 - 2\alpha \end{array} \right]$$

It is now easy to see that the anwer to b) is that we get a zero determinant if we set $\alpha = 0.25$.

($\alpha = 0.75$ will also give a zero determinant but $T$ will hen be negative definite, and we don't want that.)