Exercise: Graph Cuts based Phase Unwrapping of MRI Images.

Pankaj Daga

August 12, 2015

1 Introduction

Not all clinical MR images are created equal.

The Fourier transform is used to transform the measured k-space data in MR imaging to image space resulting in an image which is of *complex* type. The reader is, perhaps, more used to the **magnitude** MR images calculated after the Fourier transform as:

$$M = \sqrt{\mathcal{R}^2 + \mathcal{I}^2}$$

where $\mathcal{R}$ and $\mathcal{I}$ are the real and imaginary components of the measured signal. However the MRI **phase** image, which is reconstructed from the real and the imaginary signal by calculating the arctangent of their ratio, is of use in many clinical and image analysis tasks. This includes, but is not limited to, correction of susceptibility artifacts, MRI based flow imaging, susceptibility weighted imaging etc. However, the phase images are uniquely defined only in the range of $(-\pi, \pi]$ and the phase images need to be *unwrapped* at each voxel by an unknown integer multiple of 2π to obtain the true phase as in equation (1).

$$\phi_t(i) = \phi_w(i) + 2\pi k_i \tag{1}$$

where $\phi_t(i)$ is the true phase at a given voxel i, $\phi_w(i)$is the wrapped phase and k_i is the unknown integer multiple of 2π that needs to be estimated. In the absence of noise provided that the underlying field is spatially continuous, the only discontinuities that can occur in the measured phase image is due to wrapping itself. In that specific case, phase unwrapping is relatively easy to address. To unwrap, the phase difference between adjacent samples is calculated and if it is greater than π, phase wrapping has occured. In the absence of noise, the measured phase image can be correctly unwrapped provided that there are no discontinuities between adjacent voxels in the true phase image that are greater than π. While this algorithm is simple to implement, it can fail in areas with low signal to noise and these errors can propagate through the overall unwrapping process creating unwrapping failure over a large area.

In this workshop, we will tackle this phase unwrapping problem using a global optimization scheme using Graph Cuts. In particular, you will be implementing part of the work presented in [Daga and et al.(2014)]. The phase is modeled as a Markov Random Field (MRF) where the true phase ϕ_t and the

wrapped phase ϕ_w are treated as random variables. The aim is to find the discrete label configuration k that gives the maximum a posteriori (MAP) estimate of the phase wraps as shown in equation (2). MRF is an intuitive choice for this problem as an individual voxel does not provide any information to perform the phase unwrapping and there is a need to specify spatial constraint and relationships among neighboring voxels, which can be done conveniently through an MRF.

$$\phi_t = \max_k \underbrace{P(\phi_w|\phi_t)}_{\textbf{Likelihood}} \underbrace{P(\phi_t)}_{\textbf{Prior}} \tag{2}$$

Modeling the likelihood term as a delta function and owing to the MRF-Gibbs equivalence, the phase unwrapping problem is to find the MRF labeling or configuration that minimizes the energy $E(k|\phi_w)$:

$$E(k|\phi_w) = \arg\min_k \sum_{i \in I} \sum_{\Omega} V(\Delta\phi_t^i) \tag{3}$$

where I are the image voxels, Ω is the set of neighbours for a given voxel at location i. $V(\Delta\phi_t^i)$ is the potential function defined on the difference potential between a voxel i and its neighbors in Ω. The unknown integer wraps are denoted by k.

2 Minimization via Graph Cuts

Now that we have formulated the problem, we will look at how we can minimize it via Graph Cuts. Before continuing, recall from the lecture that an energy function with the form of equation (4), where E^i is the unary energy term and E^{ij} is the pair-wise energy term, can be represented by a graph as long as each pair-wise term E^{ij} satisfies the inequality in equation (5). The proposed energy function of equation (3) has the structure of equation (4) with null unary data term.

$$E(x_1, x_2, ...x_u) = \sum_{i=1}^{u} E^i(x_i) + \sum_{i=1; i<j}^{u} E^{ij}(x_i, x_j) \tag{4}$$

$$E^{ij}(0,0) + E^{ij}(1,1) \leq E^{ij}(0,1) + E^{ij}(1,0) \tag{5}$$

Keeping this is mind, we will create an iterative optimization scheme for this problem. If the pairwise energy term V is convex and if the minima of $E(k|\phi_w)$ is not reached, a binary image $\delta \in (0,1)$ exists such that $E(k+\delta|\phi_w) < E(k|\phi_w)$. For brevity let us consider the problem in one dimension and assume a two neighbourhood MRF system i.e. we only consider a single pair of neighbours. This can be easily extended to multiple dimensions by simply adding the terms corresponding to the neighbours in the other spatial dimensions. Let $k_{t+1}^i = k_t^i + \delta^i$ be the wrap count at time $t+1$ at voxel i. Then, we have equation (6) where $\Delta\phi_t$ is the difference in the true phase between the MRF neighbours.

$$\Delta\phi_t = 2\pi(k_{t+1}^i - k_{t+1}^{i-1}) + (\phi_w^i - \phi_w^{i-1}) \tag{6}$$

After algebraic manipulation of equation (6), we can rewrite the energy function as equation (7).

$$E(k_t + \delta|\phi_w) = \arg\min_n \sum_{i \in I} \sum_{s \in \Omega} V(2\pi(\delta^i - \delta^{i-1}) + 2\pi(k_t^i - k_t^{i-1}) + (\phi_w^i - \phi_w^{i-1})) \quad (7)$$

Now considering the terms in equation (5), we have:

$$\begin{aligned} E(0,0) &= V(t) \\ E(1,1) &= V(t) \\ E(1,0) &= V(2\pi + t) \\ E(0,1) &= V(-2\pi + t) \end{aligned}$$

where

$$t = 2\pi(k_t^i - k_t^{i-1}) + (\phi_w^i - \phi_w^{i-1})$$

As V is convex, $E^{ij}(0,0) + E^{ij}(1,1) \leq E^{ij}(0,1) + E^{ij}(1,0)$ or $V(2\pi + t) + V(-2\pi + t) \geq 2 \times V(t)$. Hence, the proposed energy term can be represented by a graph.

3 Graph Construction

We will use the construction from [Kolmogorov and Zabin(2004)] to construct the graph. Have a look at the paper on how to construct this energy function efficiently.

4 Setting up the code environment

Now that we have ensured that we have the theoretical understanding, it is time to put it to practice! Clone the following git repository, which will give you the basic utilities to get started. You can clone the git repository with the following command: git clone https://github.com/dagap/ucl-graph-workshop.git

Go to the cloned project and in the phase-unwrapping/code directory, you can find the basic project stub. You will find the image data that you will need for the project in the phase-unwrapping/data directory. In addition, you have the papers referred to in this exercise as well.

The data consists of a raw phase image that we will unwrap using the algorithm described before. Additionally, there is also an image mask, which suppresses the non-brain portion of the image. Before you proceed, read the images into your favorite medical image viewer and familiarize yourself with it. Observe, how the integer phase wraps manifest themselves in the phase image intensities.

The code consists of two files. The image.py consists of a utility class that will help with the loading of the nifti image files. The unwrap.py is the basic project file that you will develop during the course of this exercise. This file comments all the different steps you will need to implement to arrive at a successful solution.

Let's have some fun with Graph Cuts!

References

[Daga and et al.(2014)] P. Daga and et al. Susceptibility artefact correction using dynamic graph cuts: Application to neurosurgery. *Medical Image Analysis*, 2014.

[Kolmogorov and Zabin(2004)] V. Kolmogorov and R. Zabin. What energy functions can be minimized via graph cuts? *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2004.