Computational Data Acquisition in Positron Emission Tomography

J. J. JOHANSSON[a] AND P. NUMMINEN

[a]Turku PET Centre
Contact: J. J. Johansson
jarjoh@utu.fi

Abstract

Iterative methods for computing positron emission tomography (PET) reconstruction, estimate the image of the scanned object by comparing the data acquired with a PET scanner with the data generated by a computational method. In our work, we define the computational data acquisition by simple geometrical rules originating from the arrangement of the detectors in a PET scanner. Our aim is to define and implement efficient algorithm performing accurate computational data acquisition for PET scanners having arbitrarily many detectors.

Pairs of detectors in a PET scanner are connected to form coincidence lines, in which the coincident positron emissions are detected during a scanning period. In computational data acquisition the positron emitting object is replaced by a discrete image presenting a map of the spatial distribution of radioactivity concentrations. The data acquisition is then simulated by computing a weighted sum over the image elements in all the lines corresponding the coincidence lines in a PET scanner. The weighting coefficient describing the contribution of an image element to a coincidence line is computed for each image element - coincidence line pair.

As the number of coincidence lines in PET scanners grows at the rate that outpaces the development of fast computers, the algorithm performing the computational data acquisition must be implemented to run very fast. In our work we have fully exploited the symmetries of the set of coincidence lines to obtain a very fast algorithm which is accurate and requires no interpolation. By using the symmetries of the set of coincidence lines we also obtain a space-saving method for storing a matrix of weighting coefficients.

Our algorithm is utilised to simulate the 2D data acquisition in PET, but it is a good starting point for the 3D case, which has not been thoroughly studied yet.

1 Purpose

1.1 Introduction

In all methods for computing positron emission tomography (PET) image reconstruction, we utilise the known connection between spatial domain and the acquisition domain. Mathematically, in continuos space, the connection is defined by Radon transform;

$$g(s,\theta) \triangleq \mathcal{R}f = \iint\limits_{\mathbf{R}\times\mathbf{R}} f(x,y)\delta(x\cos\theta + y\sin\theta - s)\,dx\,dy. \tag{1}$$

Where for s and θ; $-\infty < s < \infty$, $0 \leq \theta < \pi$, or with the associated back-projection transform, which is defined as

$$b(x,y) \triangleq \mathcal{B}g = \int\limits_0^\pi g(x\cos\theta + y\sin\theta, \theta)\,d\theta. \tag{2}$$

In (1), $g(s,\theta)$ on the lefthand side represents the information acquired from the function $f(x,y)$ by calculating an integral along a line defined by distance s and angle θ. A PET scanner detects coincident events of annihilation so that the data gathered during a scanning session represents actually line integrals of the radioactivity concentrations inside the scanned object. Thus it is reasonable to model the real data acquisition with the Radon transform.

To reconstruct the scanned object from the acquired data, line integrals, we use computer programs which implement the Radon transfrom and the back projection in some way. In computer implementation the scanned object is presented as a discrete image volume composed of certain number of volumetric boxes, voxels. A discrete Radon transform along a line defined by pair (s,θ) is a weighted sum over the voxels. Each weighting factor describes the contribution of a voxel to a coincidence line. Weighting factor may include physical aspects like attenuation, photon range and non-isotropic PSF, but in this work we will only consider different methods to implement the discrete Radon transform.

1.2 Aim

Motivation to our work is lack of well documented and tested, efficient and easy to use C library functions implementing different discretisation methods for computational data acquisition. Our aim is to define, implement and evaluate a C library containing functions for Radon transform and back-projection and for storing the sparse projection matrix containing the weighting coefficients.