UNIVERSIDAD AUTÓNOMA DE MADRID

COMPUTER SCIENCE DEPARTMENT

Biomedical Image Processing
Assignment - 3

Roberto MARABINI RUIZ

Contents

1   Goal

In this assignment we will explore a 3D reconstruction algorithm based on expectation maximization and maximum likelihood (EM-ML) for three dimensional reconstruction of PET data. This is a crucial equation for PET tomography and over the past 30 years, much of the work in statistically based algorithms has been concentrated on finding alternative implementations of this EM-ML algorithm that converge faster or offer improved resolution, contrast and/or signal-to-noise ratio.

2   Introduction

In a PET study, a biochemical metabolite labeled with a positron emitting radioactive material is introduced into the body under study and the radioactive emissions are then counted using a PET scanner. The physics behind this phenomenon is as follows: when a positron is emitted, it "finds" a nearby electron and annihilates with it. The annihilation creates two gamma photons that fly off the point of annihilation, in (nearly) opposite directions along a line with completely random (i.e uniformly distributed in space) orientation. The line defined by the two opposite photons is usually referred to as line of response (LOR). A PET machine consists of an array of detector elements surrounding the body, and the two photons are detected in coincidence by a pair of these detector elements.

Vardi et al. [1] developed an algorithm to estimate the spatial distribution of the radioactive material from a set of measurements acquired by discrete bin detectors. Their mathematical development starts by assuming that positron emissions occur according to a spatial Poisson point process in a certain region $\mathbb{R}^3$ with an unknown intensity function $\lambda(\mathbf{r})$. The measured data set is the total number of coincidences along each LOR. In its list-mode version [2] the 3D reconstruction iterative method has the form.

$$\lambda^{new}(b) = \frac{\lambda^{old}(b)}{p(b,.)} \sum_{n=1}^{N} \frac{p(b,d)}{\displaystyle\sum_{b=1}^{B} \lambda^{old}(b)\, p(b,d)} \tag{1}$$

2

where N is the number of measured events.

The implementation of the EM-ML algorithm requires the estimation of two collections of key terms, namely $p(b,.)$ and $p(b,d)$, which are usually referred to as normalizing term and system matrix respectively. The normalizing term $p(b,.)$ stands for the probability that an emission from within a voxel b is detected by the scanner. The collection of $p(b,d)$ terms that constitute the system matrix is the probability that an event generated in the region defined by the voxel b is detected along the LOR d. In this assignment we will use a Monte-Carlo approach to estimate these two terms

3   Experiment Description

3.1   Description of the Scanner to be simulated

System matrix estimation in PET is a complex and time consuming task. In order to decrease the computer resources needed we will greatly simplify the problem as follows

- we will assume a 2D case. That, is all the system including specimen and detector is contained in a single plane.

- the specimen is discretized by a 4x4 matrix. That is, our sampling rate is such that only 16 voxels are needed to describe the volume.

- the detectors are placed along 2 parallel lines with 4 detectors in each line.

- if a gamma-ray is created within the voxel b we can assume that it has been created at the voxel center.

- we assume that specimen absorption is negligible.

Start the matlab script *pet_student.m*. It will pause after plotting 16 green starts that mark the center of the 16 voxels. Press *enter*, two red lines divided in 4 parts will appear. These are the two detector arrays. If you press *enter* again a set of blue
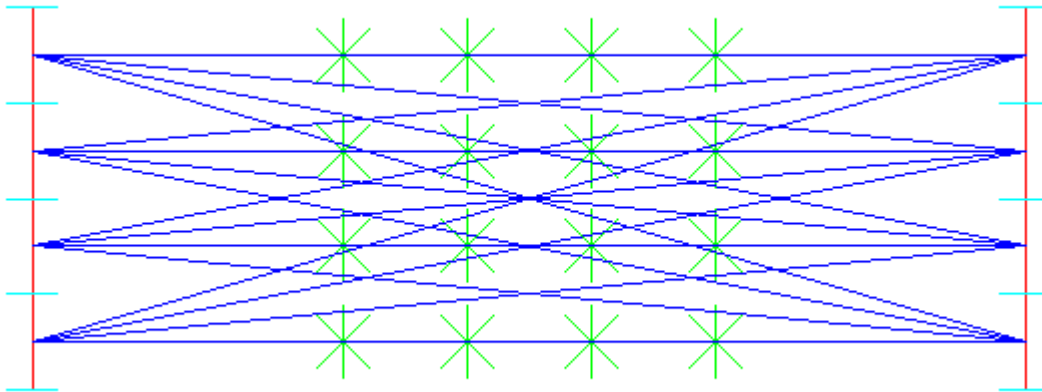
Figure 1: Data acquisition geometry showing: 1) voxel centers (green stars), 2) detectors (red lines) and LORs main axis (dark blue lines)

lines are plotted. These are the LORs. Pairs of gamma rays traveling along these lines will be detected by the system. (In general, LORs are tubes rather than lines but we will represent them in our plot by lines.)

At this point abort the execution of the matlab script.

3.1.1   Conventions

In order to refer to the different voxels and detectors we will index them as follows:

Voxels (4x4 matrix):

13	14	15	16
9	10	11	12
5	6	7	8
1	2	3	4

Detectors

L4	R4
L3	R3
L2	R2
L1	R1

where L refers to the left detector array and R to the right detector array.

Finally, LOR(1) is the LOR defined by detectors L1 and R1, LOR(2) is the one defined by detectors L1 and R2, LOR(5) is the one defined by detectors L2 and R1, ... and LOR(16) is the one defined by detectors L4 and R4

3.2   Monte Carlo Approach

The Monte Carlo method was invented by scientists working on the atomic bomb in the 1940s, who named it for the city in Monaco famed for its casinos and games of chance. Its core idea is to use random samples of parameters or inputs to explore the behavior of a complex system or process.

In our case we will simulate a PET scanner by creating positrons that will produce pairs of gamma photons traveling in random directions.

Now, edit the *pet_student.m* file and:

- comment all the *pause()* statements.

- comment the code that plot the LORs. That is,
  ```
  for  i = 1:numberOfLors
      plot([lors(i,1),lors(i,5)],[lors(i,2)+0.5,lors(i,6)+.5],'b');
  end
  pause()
  ```

- Finally set the variable *numEmissions* equal to 10. That is, we are going to create 10 positrons

Now, you should see ten yellow lines that show the path followed by the ten gamma-rays pairs. Some of them will intersect one detector in both, the left and right detector arrays (3 rays in image 2) and will be detected and the rest intersect only one detector (1 ray in image 2) array or none (6 rays in image 2) and will not be detected.
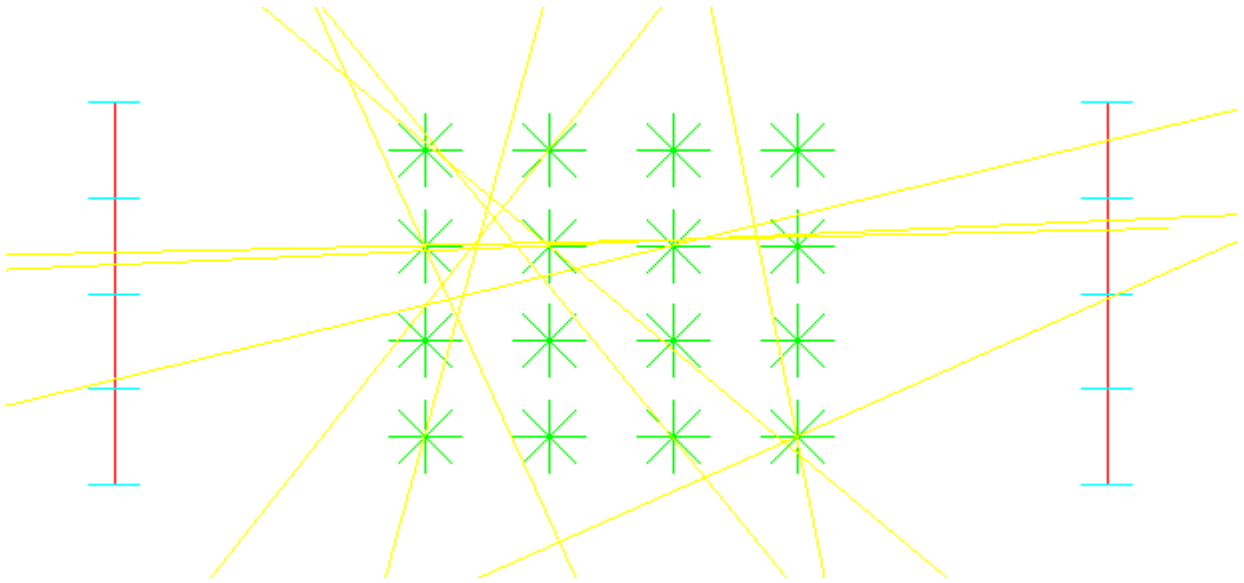
Figure 2: PET scanner showing gamma rays (in yellow)

Figure 3 shows the result of executing the program with *numEmissions* set to 1000. The blue lines represent all the gamma-ray pairs emitted by voxel(1) and that have been detected. The red lines represent all the gamma-ray pairs emitted by voxel (13) and that have been detected.

Questions 1:

- Using figure 3 compute $p(1,1)$ and $p(13,16)$. (First index represents the voxel and second the detector pair)

- Are this numbers different? comment this result

- Repeat figure 3 comparing voxels p(1,1) and p(6,5) (set *numEmissions* to 1000)

- Modify the script so the system matrix is printed and present it in your report as a 2D table. With these values compute $p(b,.)$. (call this script pet_montecarlo.m)

- Execute the code with different values of *numEmissions*. Using symmetry considerations (that is, taken into account which voxels should have similar detection probabilities) select a reasonable value for *numEmissions*. Report the selected *numEmissions* and justify the value.

4   3D Reconstruction

It is time to perform the 3D reconstruction. In the previous section in order to compute the system matrix and the normalization factor we have produce positrons equally distributed in space. Let us modify the matlab script so that all gamma rays are emitted from voxel(5), that is, only voxel(5) contains radiactive material. Run the program until at least one emission have been detected and them, by hand, apply equation 1. Use as initial λ a constant volume, that is, $\lambda(b) = 1$

Questions 2:

- Include in your report the value of the: system matrix and normalization factor, $\sum_{b=1}^{B} \lambda^{old}(b)\, p(b,d)$ and λ^{new}

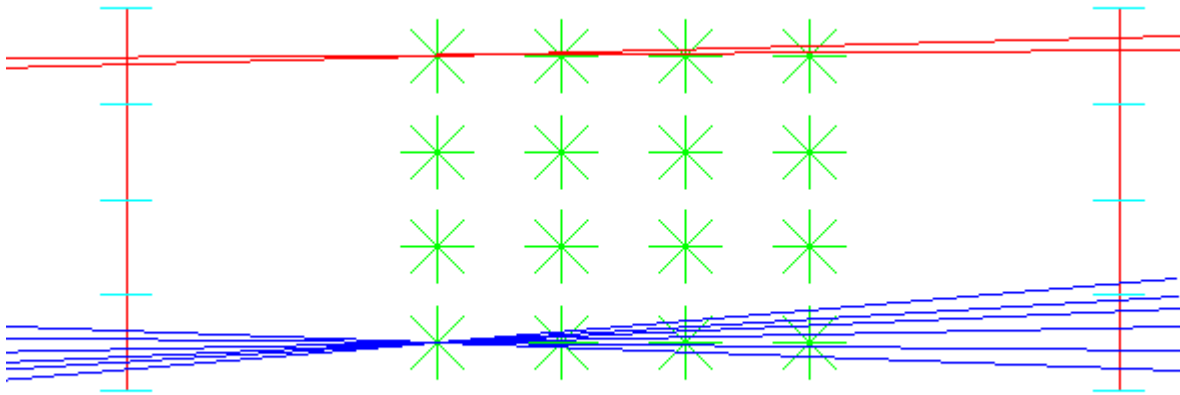- name the script produced for this step pet_3dreconstruction.m.

Figure 3: PET scanner showing the gamma rays detected by the system and emitted either by voxel(0) (blue) or voxel(13) (red)

4.1   Improving the model

4.1.1   Positron free travel

Our model of the PET scanner is very crude (see `http://www.opengatecollaboration.org/` for a full featured PET simulator). Let us make it more accurate. As the radioisotope undergoes positron emission decay, it emits a positron. The emitted positron travels in tissue for a short distance (typically less than 1 mm, but dependent on the isotope), during which time it loses kinetic energy, until it decelerates to a point where it can interact with an electron. Let us model this behavior. Modify the matlab script so that photons are created not in the voxel center but in a neighborhood. This effect can be achieved by adding a small random value to the variables xp1, yp2, xp2 and yp2 that define the lines along each pair of gamma-rays travel. Modify the matlab script as described in this paragraph, use a gaussian random number generator with standard deviation = 0.25 and answer the following questions (CAUTION: do not add the same random number to all gamma rays):

Questions 3:

- Repeat figure 3 comparing voxels 1 and 11 before and after the script modification.

- Include a table with the system matrix before and after the modification and comment on how does it change? The number of LORs that are able to detect a given voxel is higher or smaller than before? How this will affect the reconstruction? Why?

- Modify the script so the system matrix is printed and present it in your report as a 2D table both after and before the script modification. With these values compute $p(b, .)$ both after and before the script modification. (call this script pet_freetravel.m)

4.1.2   False Positives: random coincidences

PET relays on simultaneous or coincident detection of pairs of photons moving in approximately opposite directions. Photons that do not arrive in temporal "pairs" (i.e. within a timing-window of a few nanoseconds) are ignored. Two major sources of noise in PET are scatter (in a detected pair of photons, at least one of them was deflected from its original path by interaction with matter in the field of view, leading to the pair being assigned to an incorrect LOR) and random events (photons originating from two different annihilation events but incorrectly recorded as a coincidence pair because their arrival at their respective detectors occurred within a coincidence timing window). We will explore the effects of the latter source of noise.

So far we have assumed that our detector is so fast and the radioisotope dose so low that random coincidences can be ignored. Let us suppose that this is not the case and that in average two positrons are created in such a way that the detector cannot separate them (this assumption will produce an unrealistic high number of random coincidences). Therefore, we need to modify the program so two gamma rays are generated at the same time. That is, instead of having one pair of gamma rays defined by (x1p,y1p,x2p,y2p), we will have two pairs of gamma rays defined by

(ax1p,ay1p,ax2p,ay2p) and (bx1p,by1p,bx2p,by2p). In order to make the programming easier, we define a "random coincidence" as the case in which the first gamma ray pair intersects one but not both detectors and the second pair intersects one but not both detectors. An extra constrain is that the detector array (either left or right) intersected by the first gamma ray pair should be different from the detector array intersected by the second array

> Questions 4:
>
> - Modify the original matlab script to incorporate the generation of randoms (call this script pet_question3.m).
>
> - Report on the total number of positrons generated, the number of randoms and the number of true detections. (call this script pet_random.m)

4.2 Material to be handed no later than 1 week after the end of this assignment

Using moodle upload a single zip file with

> - A single pdf file containing the answer to all the questions asked in this assignment as well as all the figures, tables and plots requested.
>
> - All matlab scripts mentioned in the question boxes. The scripts should be written in independent text files that can be executed from matlab directly.

NOTE: remember that the system matrix gives probabilities and therefore the sum of all its element cannot be greater than 1.

References

[1] Y. Vardi, L. A. Shepp, and L. Kaufman, "A statistical-model for positron emission tomography," *J. Am. Statistical Association*, vol. 80, no. 389, pp. 8–20, 1985.

[2] A. J. Reader, K. Erlandsson, M. A. Flower, and R. J. Ott, "Fast accurate iterative

reconstruction for low-statistics positron volume imaging," *Physics in Medicine & Biology*, vol. 43, pp. 835–846, 1998.