Proc. of the International Meeting. on Fully 3D Image Reconstruction in Radiology and Nuclear Medicine, 1999

On various approximations for the projectors in iterative reconstruction algorithms for 3D-PET

K. Thielemans, MRC CU, Hammersmith Hospital, UK
M.W. Jacobson, The Optimization Lab, Technion Univ., Israel
D. Belluzzo, HSR, Milano, Italy

1. Introduction

An essential ingredient of all probabilistic reconstruction methods is the matrix P_{bv} giving the conditional probability that an event in voxel v is detected in a detector pair (indexed by a single index b). We discuss in this paper the relation between various approximations used for this matrix. We also introduce a new piece-wise linear interpolation method, which approximates a Tube of Response calculation, but is much more efficient in CPU time. We will ignore scatter and random coincidences.

The image is discretised in some basis functions f_v (usually voxels). We will loosely talk about an event in voxel v when we mean that it is generated in $\hat{r}$ with probability $f_v(\hat{r})$. The above mentioned conditional probability can be written as

$$P_{bv} = \int d\bar{l}\, \mu(\bar{l})\varepsilon_b(\bar{l})\, \wp f_v(\bar{l}), \qquad (1)$$

where $\bar{l}$ is a coordinate in projection space, $\varepsilon_b(\bar{l})$ is the probability that the detector pair ij detects the (511 keV) photons coming along the LOR corresponding to $\bar{l}$, $\mu(\bar{l})$ is an integration measure for projection space and $\wp$ is the continuous forward projection operator (i.e. it computes line integrals). The detector-pair sensitivities $\varepsilon_b(\bar{l})$ will be concentrated around the LOR $\bar{l}_b$ between the centres of the two detectors.

There are two practical difficulties: the functions $\varepsilon_b(\bar{l})$ are not known, and when realistic approximations for the sensitivities $\varepsilon_b(\bar{l})$ are used, the computation of the matrix P_{bv} is very CPU intensive. Various models for the sensitivities are used in the literature (see Fig. 1).

- Solid angle model, see e.g. [Ter97]
  In this model, the sensitivity $\varepsilon_b(\bar{l})$ of a detector-pair is 1 when the LOR intersects the two detectors, and 0 otherwise. The integration over the voxel is generally done by choosing some points in the voxel and averaging the different solid angles.
- Volume of intersection of the Tube of Response (TOR) and the voxel, see e.g. [Oll97]
  This is equivalent to assuming that the sensitivity $\varepsilon_b(\bar{l})$ of a detector-pair is 1 when the LOR intersects the two detectors *and* it is parallel to the 'centre' LOR $\bar{l}_b$ connecting the centre of the two detectors, and 0 otherwise.
- Length of intersection of the 'centre' Line of Response (LOR) with the voxel.
- Interpolation
  The centre of every voxel is projected onto the projection plane, and interpolation between the four nearest projection elements determines their respective weights. This can be seen as a discretisation of the two angular integrals in the continuous backprojection. Different interpolation methods are being used (see e.g. [Wal97]), most common are bilinear and Gaussian interpolation.

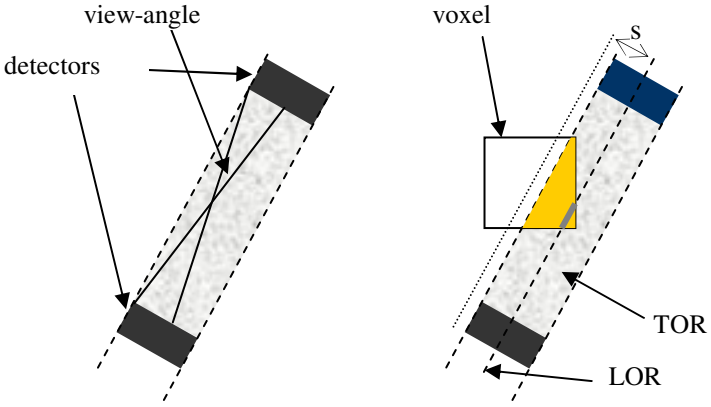

Figure 1: *Illustration (in the 2D case) of three ways to approximate the probability of detection of an event in a voxel. Left: Solid angle (for one point). Right: The intersection length of the central LOR between the two detectors with the voxel, and the intersection area of the TOR with the voxel.*

2. Approximations for the projectors valid for smooth data

In many iterative algorithms the forward projection of an image, and backprojection of projection data are the only manipulations that need the matrix P_{bv}. Under certain smoothness conditions these operations can be approximated by much more efficient calculations.

2.1 Forward projection

Forward projection of an image i_v is given by

$$\sum_v P_{bv} i_v = \int d\bar{l}\, \mu(\bar{l})\varepsilon_b(\bar{l})\wp\left(\sum_v i_v f_v\right)(\bar{l}), \qquad (2)$$

where we used the definition of P_{bv} and interchanged the order of the integration and the summation. Assume now that the function $\sum_v i_v f_v(\hat{r})$ varies smoothly over a scale set by the detector sizes. We see that the integrand consists of two factors, $\mu(\bar{l})\varepsilon_b(\bar{l})$ which is non-zero only in the

neighbourhood of the 'centre' LOR $\bar{l}_b$, and the image dependent factor which is (nearly) constant where the first factor is non-zero. So, the integral can be approximated by bringing out the constant factor:

$$\sum_v P_{bv} i_v \approx \wp\left(\sum_v i_v f_v\right)(\bar{l}_b)\int d\bar{l}\,\mu(\bar{l})\varepsilon_b(\bar{l})\,. \qquad (3)$$

This means we can compute the forward projection by using line integrals of the image for all LORs $\bar{l}_b$, and multiplying the result with the normalisation factors $n_b = \int d\bar{l}\,\mu(\bar{l})\varepsilon_b(\bar{l})$. These factors can be *measured* by scanning a uniform source, and their determination is a standard procedure in PET. We can conclude that this approximation amounts to first using the LOR model, and then adjusting the result by the measured normalisation factors.

2.2 Backprojection

The backprojection of projection data p_b is given by

$$\begin{aligned}\sum_b P_{bv} p_b &= \int d\bar{l}\,\mu(\bar{l})\left(\sum_b p_b\varepsilon_b(\bar{l})\right)\wp f_v(\bar{l}) \\ &= \int d\hat{r}\, f_v(\hat{r})\,\wp^+\left(\sum_b p_b\varepsilon_b\right)(\hat{r})\end{aligned} \qquad (4)$$

where in the second equation we have used the adjoint operator $\wp^+$ of the continuous projection $\wp$, which is well-known to be the (continuous) backprojection operator [Natt86] . Note the similarity of the last formula with the forward projection formula if we interchange f_v and ε_b, i_v and p_b, continuous forward projection $\wp$ and backprojection $\wp^+$. We can now do the analogous approximation as in the case of the forward projection if $\wp^+\left(\sum_b p_b\varepsilon_b\right)(\hat{r})$ varies much more slowly than the basis function $f_v(\hat{r})$ (e.g. when the p_b vary slowly with b):

$$\sum_b P_{bv} p_b \approx \wp^+\left(\sum_b p_b\varepsilon_b\right)(\hat{r}_v)\int d\hat{r}\, f_v(\hat{r}) \qquad (5)$$

where $\hat{r}_v$ is the point in the middle of the 'voxel'. That is, for 'smooth' data, the discrete backprojection can be computed by sampling the continuous backprojection, times a factor which measures the volume of the voxel.

Unfortunately, even with this approximation we need to sample the continuous backprojection $\wp^+\left(\sum_b p_b\varepsilon_b\right)$.

However, the only thing we can easily measure about the sensitivities ε_b are their integrals (the normalisation factors n_b). We still need to have a 'model' for the dependency of the sensitivities $\varepsilon_b(\bar{l})$ on the LOR[1]. This should then be scaled appropriately such that the integral corresponds to the measured normalisation factors.

Of course, using this approximation simplifies the calculations. For example, if the solid angle model is used, we need to compute the solid angle only for 1 point per voxel. In the TOR model, this backprojection amounts to using a nearest neighbour interpolation.

3. Correspondence between interpolation and the TOR model

We can view the action of an interpolating backprojector as the convolution of the data with an interpolating kernel. In 3D PET the interpolation kernel is two-dimensional (with coordinates *bin* and *ring*). The contribution to a voxel of 1 data element *s,r* of a projection plane is the shifted kernel $K(s-s_v, r-r_v)$, where s_v, r_v are the coordinates of the projection of the centre of the voxel on the projection plane. In this section, we briefly show that for an appropriate choice of the kernel, this approximates the volume of intersection of the TOR with the voxel.

Fig.2 shows the area of intersection in the two-dimensional case, for different relative locations of the detectors and the voxel. The graph on the left shows that a triangle shape (i.e. linear interpolation) is a reasonable approximation for the area of intersection for *any* angle, but this is *only* in the case where voxel sizes are equal to the detector sizes.

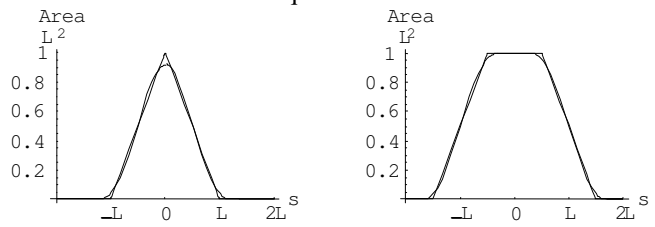

Figure 2: *Plots of the area of intersection area of the TOR and a pixel of size LxL. See Fig. 1 for the definition of* s. *Each plot contains curves for 2 different angles of the TOR: 0 and π/4. Left: detector size = L. Right: detector size = 2*L.*

In the case that voxel size is not equal to the detector size, we propose to use the new (piece-wise linear) interpolation kernel shown in Fig. 3. Obviously, the piece-wise linear interpolation corresponds much better to the probability than linear interpolation. Moreover, it will improve resolution, see section 5.2.

[1] In the forward projection case, the analogous step was no problem as we know the basis functions f_v.

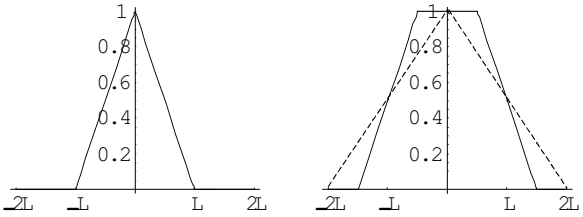

Figure 3: *Interpolating kernels: linear interpolation (dashed), proposed piece-wise linear interpolation. Left: detector size =L (the two curves are identical). Right: detector size = 2*L.*

4. Implementation of the piece-wise bilinear interpolation backprojector

For the 2D case, a fast incremental version of linear interpolation was described in [Cho90], [He93]. The algorithm works by selecting a 'beam' of voxels that get contributions of 2 projection elements. These contributions are found incrementally by adding to the contribution of the previous voxel in the beam. For efficiency reasons, voxel size in the *x* and *y* direction is taken equal to the size of an (arc-corrected) bin. This algorithm was extended to the 3D case in [Egg96, 98] for linear interpolation. There it was suggested to take the z-voxel size equal to half the ring spacing of the scanner[2] to improve resolution in the z-direction. However, as the previous section shows, in this case the piece-wise linear interpolation should be used (in z-direction). We discuss here the generalisation of the partially incremental algorithm of [Egg96, 98] to the piece-wise linear case for this choice of voxel size.

If we scale coordinates such that the voxel sizes are 1, the update value V_A of a voxel A is obtained as follows:

$$V_A = (1-ds_A)\,(3/2-dz_A)\,V_1 + ds_A\,(3/2-dz_A)\,V_2 + (1-ds_A)$$

$$(dz_A-1/2)\,V_3 + ds_A\,(dz_A-1/2)V_4, \text{ if } 1/2 < dz_A < 3/2 \qquad (6)$$

$$V_A = (1-ds_A)V_1 + ds_A\,V_2, \text{ if } dz_A <= 1/2 \qquad (7)$$

$$V_A = (1-ds_A)V_3 + ds_A\,V_4, \text{ if } dz_A >= 3/2 \qquad (8)$$

where $V_1 \ldots V_4$ are the values of the projection data on the four corners of the beam. The two last equations can easily be incrementalised as in [Cho90]. We concentrate here on Eq. 6, and in the following we assume that $1/2 < dz_A < 3/2$. We can write Eq. 6 in terms of the same beam-constants as defined in [Egg98] (they are linear combinations of the $V_1 \ldots V_4$):

$$V_A = V_1 - K_2/2 + ds_A\,(K_1 - K_3/2) + dz_A\,K_2 + ds_A\,dz_A\,K_3$$

For a voxel B adjacent to A and whose centre lies inside the same beam, we find:

$$V_B = V_{B,incr} + dz_B\,ds_B\,K_3,$$

with

$$V_{B,incr} = V_{A,incr} + (ds_B - ds_A)\,(K_1 - K_3/2) + (dz_B - dz_A)\,K_2.$$

The implementation follows the same lines as for linear interpolation [Egg96,98]. The main change is that incremental values for Eqs. 7,8 have to be stored, and that at each update of a voxel a test has to be made on the value of dz_A. Note that this implementation can be accelerated by the usage of symmetries as in [Egg96,98].

Interpolating backprojection can be seen as a discrete implementation of the backprojection operation (replacing the integrals by summation). In the continuous backprojection a geometric factor has to be included (depending on the ring difference and the *s* coordinate) [Def90]. This means that the beam values $V_1 \ldots V_4$ are in fact the projection data multiplied by the average value of this geometric factor for that beam (this was ignored in [Egg96,98]).

We have implemented this algorithm as part of an object-oriented library for iterative reconstructions in 3D PET [Lab98]. It turns out that the increase in CPU time for the incremental versions of piece-wise bilinear over ordinary bilinear interpolation is only a few percent. This can be explained by the fact that in half of the cases for dz_A , Eq. 6 is not used, but a fully incremental version Eqs. 7,8.

5. Results on the piece-wise linear interpolation

5.1 Sensitivity image

An important ingredient of many iterative algorithms is the sensitivity image $S = \sum_b P_{bv}$. This computes the probability that an event in a voxel is detected at all. *S* can be computed as the backprojection of data p_b set all equal to 1 (see Eq. 4). This means that the smoothness assumption mentioned in section 2.2 is optimally satisfied, and hence for a given model for the detector-pair sensitivities ε_b , the approximation Eq. 5 will give (nearly) the same result as the exact result Eq. 4.

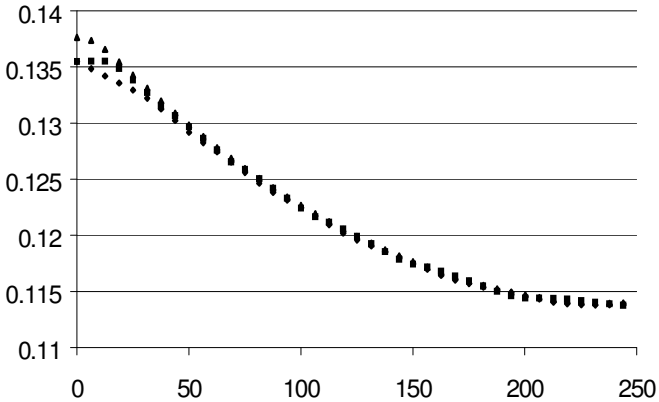

Figure 4*: Profiles of the sensitivity image (along a horizontal line through the axis of the scanner, and in the*

[2] This corresponds to the usual practice in 2D PET to have 'direct' and 'indirect' planes.

middle transaxial plane) for the 953b. Triangles: solid angle probabilities. Squares: piece-wise linear interpolation. Diamonds: linear interpolation.

In Fig. 4 we compare the sensitivity images as computed by various backprojectors for the Ecat 953b scanner (ignoring effects of detector normalisation). The slight mismatch between the solid angle and the piece-wise bilinear interpolation can be explained by remembering that the interpolation approximates the backprojection integral by summing over the ring differences and views. This only works well when the number of rings and the number of views are sufficiently large. For a 16-ring scanner this discretisation gives deviations in the middle of the scanner. For the CTI Exact 3D, a 48-ring scanner, the sensitivity image obtained by the piece-wise linear interpolation and a solid angle computation are indistinguishable (plots not shown).

5.2 Resolution in z-direction

In the case where detector size is larger than the voxel size, the second plot in Fig. 3 shows that the kernel of the piece-wise linear interpolation has a smaller spread than in linear interpolation. This means that backprojected images will have a higher resolution. To test this, we scanned point sources on a GE Advance scanner. We reconstructed the data with our own implementation (using the library [Lab98]) of the PROMIS algorithm [Kin98], without using axial compression of the projection data.

Figure 5 shows the improvement in the FWHM in axial direction by showing the ratio between the result using piece-wise linear interpolation and the result by using linear interpolation.

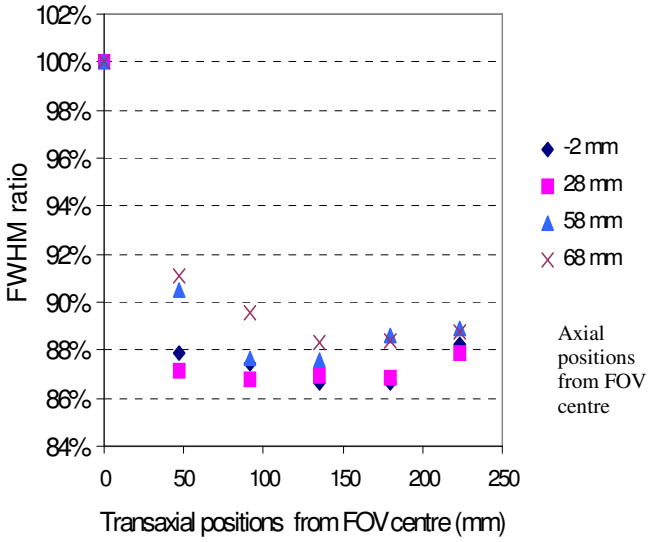

Figure 5: *ratio of FWHM in axial direction (see text)*

6. Discussion

For present day 3D-PET scanners, it is computationally expensive to use realistic models for the detector-pair sensitivities. It has been argued in various places (see e.g. [Zen90], [Egg96]) that an LOR model can be used for the forward projection, but interpolation in the backprojection. We showed in this paper that this is because the LOR model (together with normalisation) is a good approximation of the (scatter-free) measurement process when the image is smooth. As backprojection essentially works on 1 voxel, this smoothness assumption is not satisfied.

Instead, backprojection can be performed by making a solid angle computation for 1 point per voxel, or by interpolation. The new piece-wise bilinear algorithm provides an efficient alternative for the TOR model. When voxel sizes are equal to detector sizes, the piece-wise bilinear interpolation reduces to ordinary linear interpolation.

All models for the detector efficiencies discussed here ignore scatter in the detectors, resulting in a 'spreading out' of the efficiencies. This can be handled in a factorised matrix approach, where this effect is handled by a convolution of the projection data, see e.g. [Qi98]. The models discussed in the first section form then the geometric projection matrix.

Acknowledgements

This work was funded by the European Esprit LTR project PARAPET (EP23493). The authors thank the partners within the project for many discussions and suggestions. KT also thanks Darren Hogg for suggestions on the manuscript.

REFERENCES

Cho Z H, Chen C M and Lee S Y **(1990)** IEEE Trans. Med. Imaging, 9-2, p. 207-217.

Defrise, D. Townsend, A. Geissbuhler, **(1990)** Phys. Med. Biol. (1990), Vol. 35, No 10, 1361-1372.

Egger M L **(1996)** PhD Thesis, University of Lausanne

Egger ML, Joseph C, and Morel C. **(1998)** Phys. Med. Biol.43 p. 3009-3024.

Labbé C, Thielemans K, Zaidi H, Morel C **(1998)**, submitted to 3D99.

He Y J, Cai A and Sun J-A **(1993)** IEEE Trans. Med. Imaging, 12-3, p. 555-559.

Natterer F, **(1986)** *"The mathematics of computerized tomography"*, Wiley, New York.

Qi J, Leahy RM, Cherry SR, Chatziioannou A, Farquhar TH **(1998)**, PMB Vol. 43, No. 4, p.1001.

Terstegge A, Weber S, Herzog H, Müller-Gärtner, HW, Halling H **(1997)**, Proc. IEEE Nucl. Sc. Symp. and Med. Im. Conf., p.1603-1607.

Ollinger JM, Goggin AS **(1997)**, Proc. IEEE Nucl. Sc. Symp. and Med. Im. Conf., p. 1594-1598.

Siddon R L **(1985)** Medical Physics, 12-2, p. 252-255.

Wallis JW, Miller TR **(1997)** IEEE Trans. Med. Im. Vol. 16, No. 1, p. 118-123.

Zeng GL, Gullberg GT **(1990)** IEEE Trans. Nucl. Sc. Vol. 37, No. 2, p 759-767.