COMPTON EFFECT CORRECTION IN SPECT USING PVM

V. BOUCHARD†, L. COLOMBET‡, AND L. DESBAT†

Abstract. A PVM implementation of a Single Photon Emission Computed Tomography (SPECT) reconstruction algorithm with simultaneous compensation for Compton scatter is proposed. The performance of this O(V^3P^3M) 3D reconstruction algorithm is measured on a small heterogeneous workstation network and on a CRAY T3D.

1. Scatter correction in SPECT images. In order to obtain a SPECT image, a radiopharmaceutical with marked particles is injected within the patient body or inhaled. These marked particles fix themselves on the organ to be studied and emit γ-rays which are counted by a γ-camera. As the γ-camera is collimated, each detector counts the γ-rays coming from the straight line perpendicular to the camera plane. Several data are obtained with the camera moving around the patient (cf Fig. 1, left). The aim of SPECT is to provide an estimate of the activity from these data.

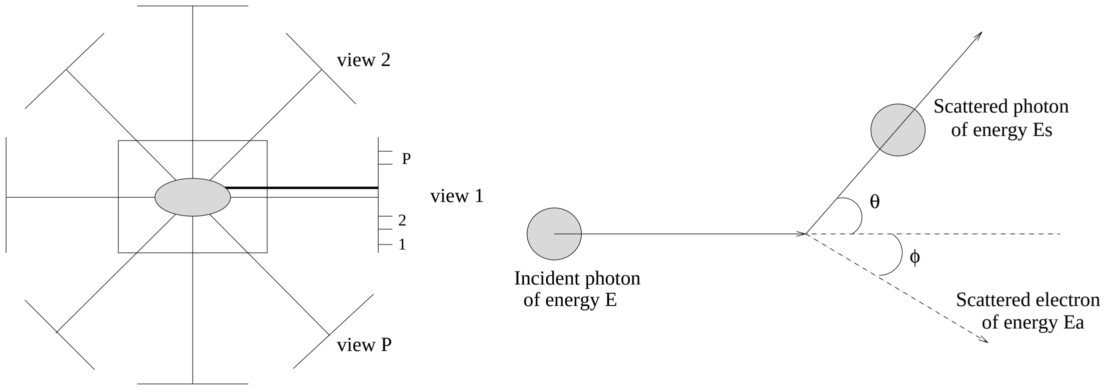


FIG. 1. *Left: the camera moving around the patient; Right: the Compton effect*

The difficulty comes from the fact that, while crossing matter, γ-rays are scattered (cf Fig. 1, right). The main reason is Compton effect: photons interact with electrons, thus can be deviated and loose energy during their propagation.

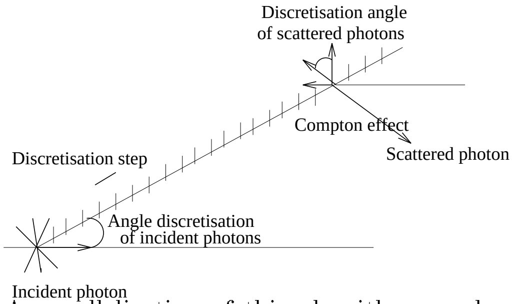
Conservation laws of classical mechanics give four relations between the five parameters described in Fig 1, right. Klein and Nishina (see [4, 3]) have found a fifth relation giving the propability $K(\theta)$ for a photon to be scattered in the angle θ: $K(\theta) = n\, r_0^2\, d\theta\, \frac{1}{1+\alpha(1-\cos\theta)}^2\, \frac{1+\cos^2\theta}{2}\, (1 + \frac{\alpha^2(1-\cos^2\theta)^2}{(1+\cos^2\theta)(1+\alpha(1-\cos\theta))})$, where :$\alpha = \frac{E}{m_0 c^2}$, E is the incident photon energy, c the light celerity, m the electron mass, r_0 the electron radius.

Knowing the electronic density at each point of the patient body - using a scanner image for example - each emitted photon can be followed during its propagation, taking into account Compton scattering. The organ is discretized into V^3 voxels, for which the electronic density is known and the activity is unknown. The γ-camera is made of P^2 pixel detectors. The camera takes P different positions around the patient, yielding P^3 pixel positions in the space. With this model, a probability matrix T can

† TIMC-IMAG, IAB, facullté de Médecine, 38706 La Tronche France
‡ CISI-CENG/SIT, Batiment T, 17 Rue des Martyrs, 38054 Grenoble cedex 9 France

be computed. The entry T_{ij} is the probability for a photon emitted from the voxel j to be counted by the pixel i. Then, we have to solve $\sum_{j=1}^{V^3} T_{ij} a_j = y_i$, $i = 1, \ldots, P^3$, where a_j is the unknown activity in the voxel j, y_i the data read on pixel i. As scattered photons loose energy, we suppose that photons being twice or more times scattered are lost and not counted. So our algorithm for computing T can be described as follows :

The different discretisation steps



for each voxel $O(V^3)$
  for each direction $O(P^2)$
    for each step $O(M)$
      for each camera position $O(P)$
        compute the contribution of this photon to the probability matrix

A parallelisation of this algorithm can be easily proposed because of the data independance. A master slave organisation is implemented. The master distributes to slaves the corresponding computations for the different voxels.

2. Description of the target computers.

2.1. The small heterogeneous DEC Alpha workstation network. A PVM algorithm has been implemented on four DEC Alpha workstations: a DEC 4000-600, 384 Mo, three DEC 3000-500, 128 Mo. These workstations are connected by an ethernet network. The execution time has been measured using the DEC 4000 with one, two or three DEC 3000. the master, executed on the DEC 4000, distributes the computations of each voxel to slaves on the network. The main difference with the PVM implementation on the CRAY T3D is that the DEC 4000 executes also a slave computing task.

2.2. The CRAY T3D. An adapted PVM program has also been executed on the Cray T3D, composed of 128 DEC Alpha (similar processor as the one of the DEC alpha workstations). Each of them has the whole information to compute its own part independtly from the others. Put into pairs sharing a network interface, each pair is directly connected to six other ones, its neighbours, the interconnect network being a 3D torus. The main implementation difference compared to the previous one is that one processor is dedicated to the master task.

3. Speedup and efficiency. The speedup and the efficiency on homogeneous parallel computers like the CRAY T3D are well known (see [1]): $S(N) \stackrel{\text{def}}{=} \frac{T(1)}{T(N)}$, $E(N) = \frac{S(N)}{N}$, where N is the number of processors. In the heterogeneous case, the speedup has to be explicitely expressed as relative to a particular processor called the Reference Processor (RP):

$$S(\mathbf{N}, RP) = \frac{T_{\text{seq}}(RP)}{T(\mathbf{N})},$$

where $\mathbf{N}$ is the network (or processor list). The main problem is then to get a mininal speedup upper bound in order to define the efficiency. Because of the performance differences of the various processors composing the computational ressource, we have to take into account the mapping of the work and we must consider a cost model for the execution time. Let W be the total amount of work, let W^j be the amount of work of type j ($W = \sum_{j=1}^{J} W^j$), let W_i be the work executed on the processor i ($W = \sum_{i=1}^{I} W_i$), the mapping of the algorithm is described by the quantities $w_{i,j}$ being the amount of work j executed on the processor i. We denote also $\mathbf{c}^i = (c_{i,1}, \ldots, c_{i,j}, \ldots, c_{i,J})$ the cost vector of processor i, i.e., the execution time of the work $W_i = \sum_{j=1}^{J} w_{i,j}$ is $T_i = \sum_{j=1}^{J} c_{i,j} w_{i,j}$. It can be proved (see [2]) that the following simplex, giving a parallel execution time lower bound, is always feasible:

$$T_{//} \stackrel{\text{def}}{=} \min_{w_{i,j}} \sum_{j=1}^{J} c_{N,j} w_{N,j}$$

$$(1) \qquad \begin{cases} w_{i,j} \geq 0 \\ \displaystyle\sum_{j=1}^{J} c_{i,j} w_{i,j} = \sum_{j=1}^{J} c_{N,j} w_{N,j}, \ \forall i = 1, \ldots, N-1 \\ \displaystyle\sum_{i=1}^{N} w_{i,j} = W^j, \ \forall j = 1, \ldots, J \end{cases}$$

We define efficiency as

$$E(\mathbf{N}) = \frac{T_{//}}{T(\mathbf{N})}.$$

The linear problem (1) is totally degenerated (i.e., all feasible solutions are optimal) iff the vector costs $\mathbf{c}^i$ are two by two linear dependant. In this case, the network is said "weakly heterogeneous". If we choose a RP having a vector cost $\mathbf{c} = (c_1, c_2, \ldots, c_J)$ such as $c_{i,j} = c_j/\lambda_i$ then the efficiency can be written

$$(2) \qquad E(\mathbf{N}) = \frac{S(\mathbf{N}, \mathrm{RP})}{\displaystyle\sum_{i=1}^{N} \lambda_i}$$

yielding a simple generalization of the classical Efficiency definition ($\lambda_i = 1$).

4. Experimental results. The first results show the comparison of the sequential times on the DEC 4000 and DEC 3000 (2 first lines of the following table), PVM execution times and efficiencies on increasing size network, for computing the probability matrix for a space discretization varying from $V^3 = 4^3$ to $V^3 = 7^3$. The DEC 4000 is taken as the RP for computing the efficiency. The network is suposed to be weakly heterogeneous. In this case $\lambda_{DEC4000} = 1$ and $\lambda_{DEC3000} = \frac{T_{DEC4000}}{T_{DEC3000}}$. From our experiments, we take for $\lambda_{DEC3000}$ the mean value of line 1 divided by line 2: $\lambda_{DEC3000} = 0.94$ (note that the standard deviation is only $\sigma_n = 0.03$ for the value of $\lambda_{DEC3000}$, justifying our hypothesis). We derive the efficiency using the definition (2), with $\sum_i \lambda_i$ being 1.94, 2.88, 3.82 for $N = 2$,i.e., 1 DEC 4000, 1 DEC 3000, resp. $N = 3$,i.e., 1 DEC 4000, 2 DEC 3000, resp. $N = 4$,i.e., 1 DEC 4000, 3 DEC 3000.

N\V	4	5	6	7
DEC 4000	84.5	222.9	494.6	935.9
DEC 3000	86.8	234.8	526.7	1051.2
2	55.25	141.18	304.44	524.24
E	.79	.81	.84	.92
3	39.51	101.42	211.52	371.66
E	.74	.76	.81	.87
4	29.55	78.72	169.59	289.50
E	.75	.74	.76	.85

This generalized efficiency has globally a standard behavior: it increases with the problem size and decreases with the number of workstations.

The next results show the computing times using the Cray T3D, generating the same matrix, for a small and a bigger configuration.

N\V	4	5	6
8	10.04	25.48	55.22
16	5.38	12.87	27.32
32	4.88	8.82	16.98
64	7.23	10.00	17.24

N\V	10	11	12
32	546.24	729.82	966.06
64	348.84	457.95	610.38
128	335.59	426.35	575.76

We did not measure the sequential time for 1 Processor Element on the CRAY T3D, but we can already detect some problems. The problem size has to be adapted to the dimension of the computational ressource (or inversely). We can also suspect that our too simple master slave organisation with a centralized distribution of the computation, is no more efficient when the number of processors increases. This is particularly obvious in the second table: even for $V = 12$, i.e. for a large amount of computations, the difference between $N = 64$ and $N = 128$ is very small. The computations repartition has to be adapted. This is our next aim for the CRAY T3D.

REFERENCES

[1] G.M. Amdahl. Validity of the single-processor approach to achieving large scal computing capabilities. In D.J. Evans, G.R. Joubert, and H. Liddell, editors, *IAFIPS Conference Proceedings*, volume 30, pages 483–485. AFIPS press, 1967.

[2] L. Colombet and L. Desbat. Speedup and efficiency of large size applications on heterogeneous networks. Technical report, UJF-CNRS-CHUG, TIMC IMAG, IAB, faculté de Médecine, F-38706 La Tronche, 1994. accepted for publication in the ICAE Journal, Special Issue on High Performance Distributed Computing.

[3] R.D. Evans. *Encyclopedia of physics (Compton effect)*. S. Flugge, 1985.

[4] M. Tubiana, J. Dutreix, A. Dutreix, and P. Jockey. *Bases physiques de la radiothérapie et de la radiobiologie*. Masson et Cie., 1963.