SPEEDING UP IMAGE RECONSTRUCTION METHODS IN CODED MASK γ CAMERAS USING NEURAL NETWORKS: APPLICATION TO THE EM ALGORITHM

FERNANDO J. BALLESTEROS, ENRIQUE M. MURO* and BARTOLO LUQUE
Centro de Astrobiología, Instituto Nacional de Técnica Aeroespacial, Ctra. de Ajalvir km 4, Torrejón de Ardoz, 28850 Madrid, Spain
(author for correspondence, e-mail: e-mail: emuro@bossa-nova.cnb.uam.es)*

(Received 20 March 2000; accepted in revised form 12 March 2001)

Abstract. When using γ-ray coded-mask cameras, one does not get a direct image as in classical optical cameras but the correlation of the mask response with the source. Therefore the data must be mathematically treated in order to reconstruct the original sky sources. Generally this reconstruction is based on linear methods, such as correlating the detector plane with a reconstruction array, or non-linear ones such as iterative or maximization methods (i.e. the EM algorithm). The latter have a better performance but they increase the computational complexity by taking a lot of time to reconstruct an image. Here we present a method for speeding up such kind of algorithms by making use of a neural network with a back-propagation learning rule.

Keywords: coded-masks, EM algorithm, neural networks, γ-ray cameras

1. Introduction

The possibility of imaging in γ-rays is very useful in different research fields besides γ-ray astronomy: for instance, in nuclear medicine it is used to know the distribution of the emitter isotopes previously introduced inside the patient body or in nuclear fusion since it is interesting to know the plasma evolution inside the reactor (Pedersen and Granetz, 1984). In any case, we need to obtain an image of the emitting sources, but this is a difficult task. The classical cameras based on lenses or mirrors are here inappropriate due to the high energy of this radiation (Skinner, 1998) passing through the glass without suffering any significant deviation.

A coded mask camera, commonly used for imaging γ sources, is an opaque body with a certain transparent pattern of holes interposed between the detector plane and the source. This pattern codes the received signal. In fact, what we get is the correlation of the sources with the mask pattern. Therefore we do not get a direct image of the source (as it happens when using lenses, mirrors or pinhole cameras) but an image to be mathematically processed.

The straightforward reconstruction methods in these cameras (Fenimore et al., 1981) are based on correlating the detector plane with a reconstruction array (a modification of the mask array). These methods are very popular because they are easy and quick. Usually they are preferred when the coded mask camera is rather

Experimental Astronomy **11:** 207–222, 2001.

complex and it has many components. Unfortunately they have some problems in handling data in difficult situations, for instance when the camera moves.

In order to improve the reconstructed image, more complex methods must be used: TK-correlation (Hammersley, 1986), Iterative Removal of Sources (IROS) (Hammersley et al., 1992), Maximum Entropy methods (MEM) (Ponman, 1984) or Maximum Likelihood methods as the EM algorithm (Ballesteros et al., 1998). These methods have better angular resolution and performance, but even in the simplest cases, the reconstruction is very much slower than using correlation methods.

In this article we present a proposal to speed up these reconstruction methods using a well known neural network (NN), a Multilayer Perceptron which uses a Back-propagation (Rumelhart et al., 1986; Werbos, 1974; LeCun, 1985) learning rule with momentum (Haykin, 1998). For our purpose we have used a public NN package⋆: the Stuttgart Neural Network Simulator SNNS v. 4.1 (Zell et al., 1991).

2. Framework

A coded mask camera is composed by a discrete detector plane and an opaque plate (the mask) consisting of a pattern of holes which receives direct inputs from the sky (see Figure 1).

The detection process can be described by the following equation:

$$D_{kl} = \sum_{ij} O_{ij} \Phi_{kl}^{ij} , \tag{1}$$

where the indices kl label the detector plane units and ij the sky pixels. Φ is the response array which has to include all the effects of the detection process: different efficiencies for each detector unit, passive elements such as honeycombs, collimators, etc. It indicates how many photons coming from the sky pixel ij contributes to the signal of the detector unit kl.

The goal is to obtain the O_{ij} sky intensity parameters (photons per area unit per time unit), having the detected data D_{kl} (counts per detector unit per time unit), that is, inverting Equation (1). Several of these methods with different performance are mentioned in the Introduction.

Unfortunately those which give better image resolution and with better performance are slower than linear methods, as correlation. In fact, for very complex telescopes (with many pixels in the detector plane and cells in the mask) the reconstruction using this kind of method could result impracticable. For instance, using a Pentium II with 400 MHz (this will be the computer assumed to be used in the whole paper), and in the case of the telescope IBIS on board INTEGRAL (Reglero et al., 1996; Ballesteros et al., 1998) composed of 128 × 128 detector units and

⋆ http://www.informatik.uni-stuttgart.de/ipvr/bv/projekte/snns/snns.html.

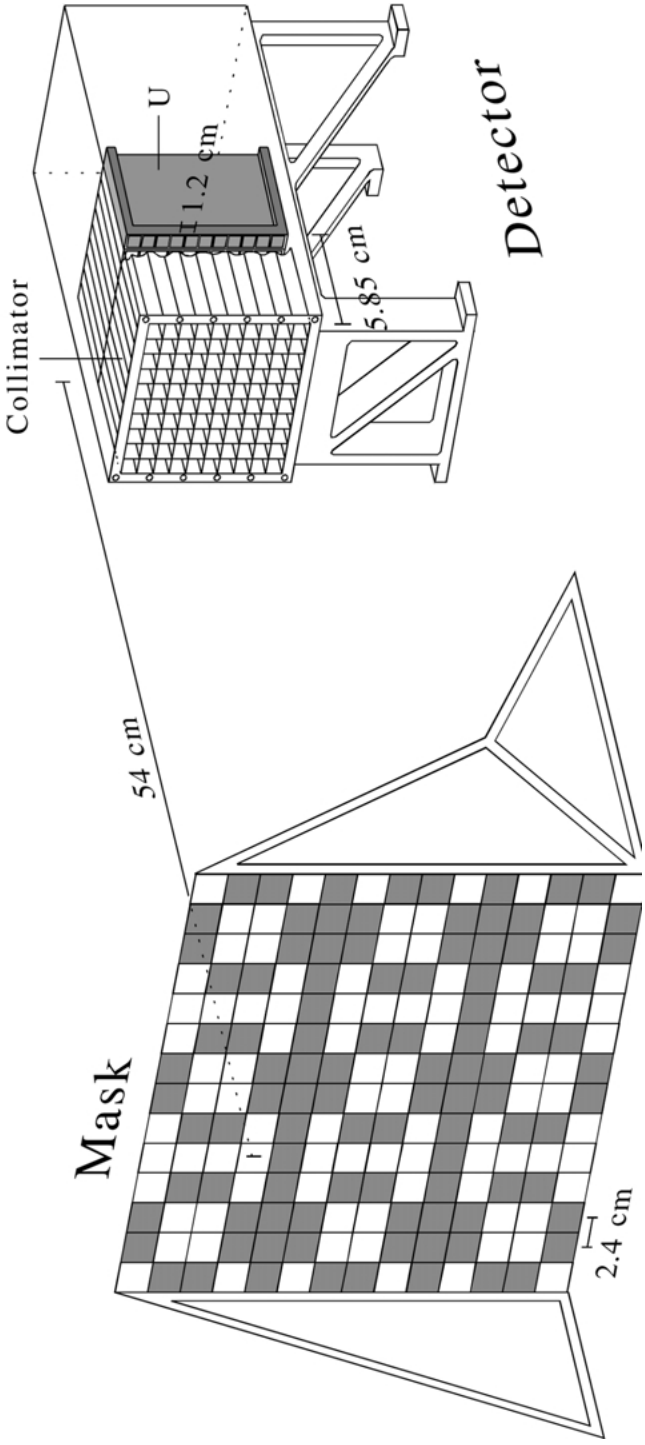


Figure 1. The LEGRI camera of the University of Valencia on board Minisat 01 (Reglero et al., 1996). The mask is a 5×5 MURA pattern (Gottesman and Fenimore, 1989) placed in a 2.8×2.8 mosaic and the detector plane is composed by 10×10 detector units placed in 10 elements rows labeled U in the figure.

a typical sky of 580 × 580 pixels, a reconstruction using the EM algorithm takes about 65 days. Hence, a linear reconstruction method such as correlation has to be used (which only takes 25 min).

In the opposite case, using very simple cameras with few detector units and mask cells, the reconstruction with correlation methods gives few pixels of great size, that is, a poor angular resolution and bad imaging capabilities. Some examples of such simple cameras are: the telescope SPI, also on board INTEGRAL (Lavigne et al., 1998) (19 pixels in a hexagonal detector plane, and a typical hexagonal sky of about 61 pixels), LEGRI (Figure 1) (10 × 10 pixels in the detector plane and a typical sky of 33×33 pixels) or in a non-astronomical context, cameras monitoring tokamaks (Pedersen and Granetz, 1984), nuclear power stations, and the coded-mask cameras used in nuclear medicine (Ohyama et al., 1983).

Therefore one must improve the image using more sophisticated algorithms like the Maximum Entropy Method or the EM algorithm, although the reconstruction time increases (an image reconstruction of the LEGRI data with the EM algorithm takes about 2.5 min whereas with correlation takes milliseconds). Our goal is to speed up these reconstruction processes in the case of simple telescopes in order to make it as fast as linear methods but without loosing their good performance.

In this work we will deal with the EM algorithm introduced in the field of γ-ray astronomy in a previous article (Ballesteros et al., 1998). It is an iterative algorithm of maximization that computes maximum likelihood estimators from incomplete and noisy data. This method has great advantages respect to linear methods, being comparable to MEM. It is a flexible method with very good performance and easy-to-implement. Unfortunately, it is slower than linear methods and depending on the camera complexity, reconstruction of an image could take a long time.

The iterative method is described by:

$$\tilde{O}_{ij}^{n+1} = \tilde{O}_{ij}^{n} \frac{\sum_{kl} \Phi_{kl}^{ij} \left(\frac{D_{kl}}{\tilde{D}_{kl}^{n}} \right)}{\sum_{kl} \Phi_{kl}^{ij}}, \tag{2}$$

where $\tilde{O}_{ij}^{n}$ is the estimation of the ij pixel intensity at iteration n, D_{kl} stands for the detected data, $\tilde{D}_{kl}^{n}$ is the current estimation of the detected data (estimated using Equation (1) and $\tilde{O}_{ij}^{n}$). We use for $n = 0$ a flat image filled with 1's, i.e. $\tilde{O}_{ij}^{0} = 1$ for all ij. The convergence criterion of EM is described in Ballesteros et al. (1998).

In order to speed up the reconstruction we implement a NN which will carry out the iterative EM algorithm in real time after the training process. The implemented NN is a Multilayer Perceptron which uses a Backpropagation learning rule with momentum. The NN uses a three layer feedforward architecture: an input layer with $M \times M$ neurons (which stands for the detector plane), a hidden layer which must be initializated with a certain amount of neurons and an output layer composed of $L \times L$ neurons (which stands for the reconstructed sky).

3. Neural network: learning and results

To train the network we generate a set of N sky images ($\{[O_{ij}]^{(s)}\}$ with $s = 1, \ldots, N$) just by placing a random number of γ sources in random positions inside the field of view in a $L \times L$ lattice for each sky image.

All the generated sky sources have an intensity of 1 photon per area unit per time unit. The length of the simulated observation is assumed to be equal to a time unit. The area of each detector unit is assumed to be equal to an area unit.

The number of sources varies randomly following a uniform distribution between 1 and a maximum value. In order not to saturate too much the initial sky images, the maximum value has been assumed to be the 80% of the number of pixels in the $L \times L$ lattice.

An example of $L \times L = 9 \times 9$ cells and 17 points randomly placed, using a uniform distribution, is shown in Figure 2a.

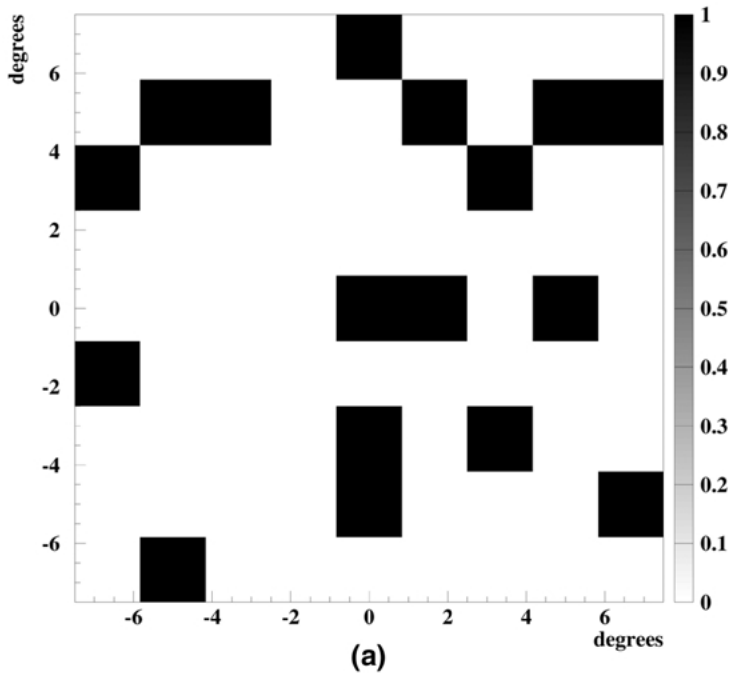
We compute the detected data $[D_{kl}]$, a $M \times M$ lattice (see Figure 2b), using Equation (1). Each instrument has its own Φ response function which depends on different effects: real efficiency of the detector units, transparency of the mask, passive elements, strongbacks,... as was explained above. In any case, all of them are linear systems (see Equation 1) and the differences among them come from the different possible values of the response array Φ and its size. Given that these systems are analogous, in order to test the method we have simulated a realistic Φ response, using two cases:

(a) A 5×5 detector plane plus a 13×13 coded mask. This configuration produces a reconstructed sky with 9×9 pixels. The pattern of the mask has been a 5×5 MURA pattern (Gottesman et al., 1989) placed in a mosaic. (b) A 7×7 detector plane plus a 19×19 coded mask (giving a reconstructed sky with 13×13 pixels) with a 7×7 MURA pattern placed also in a mosaic. This second case has been less used due to be slower. In both cases each detector unit has a different efficiency, randomly chosen in the range (0.5,1) (as said before, for simplicity each detector unit has an area equal to 1 area unit). Similarly the mask transparency has been randomly chosen in the range (0.8,1) for the transparent elements and in the range (0,0.2) for the opaque elements.

We provide these detected data $[D_{kl}]$ to the EM algorithm getting a set of reconstructed sky images $[\tilde{O}_{ij}]$ (see Figure 2c).

The NN uses as input the detected data set $[D_{kl}]$ and as desired output the EM algorithm reconstructions $[\tilde{O}_{ij}]$. Therefore, the NN learns the EM algorithm reconstruction.

Why do we not train the NN by using directly the detected data together with the original sky images, instead of using the reconstructions from the EM algorithm? Because the mathematical mapping from $[O_{ij}]$ to $[D_{kl}]$ (see Equation 1) is not bijective but surjective: that is, there are many different skies that will produce the same detector response. If we try to train the NN with real images, the NN never



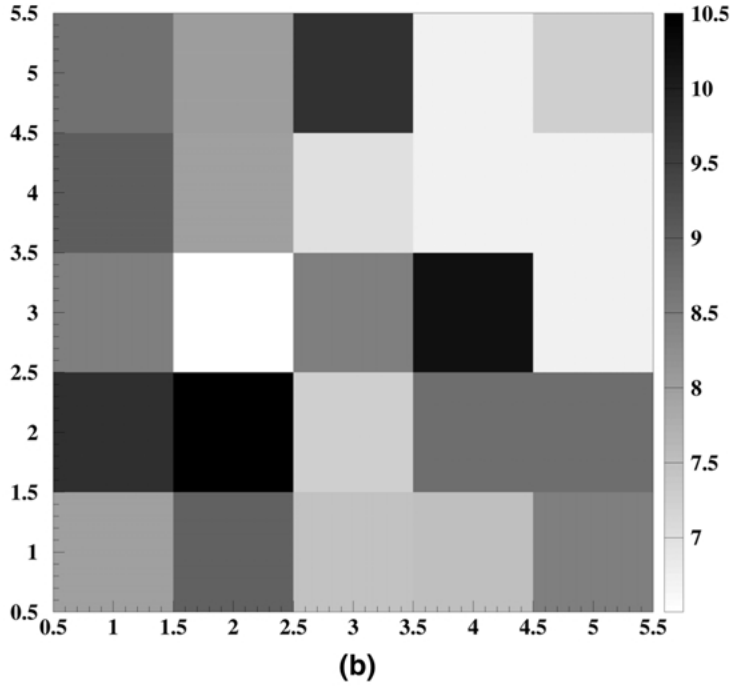


Figure 2a-b. (a) Example of a random sky $[O_{ij}]$ for the learning process. Each black pixel represents a γ-ray point source with intensity equal to one photon per area unit per time unit. (b) Signal detected $[D_{kl}]$ in the detector plane.

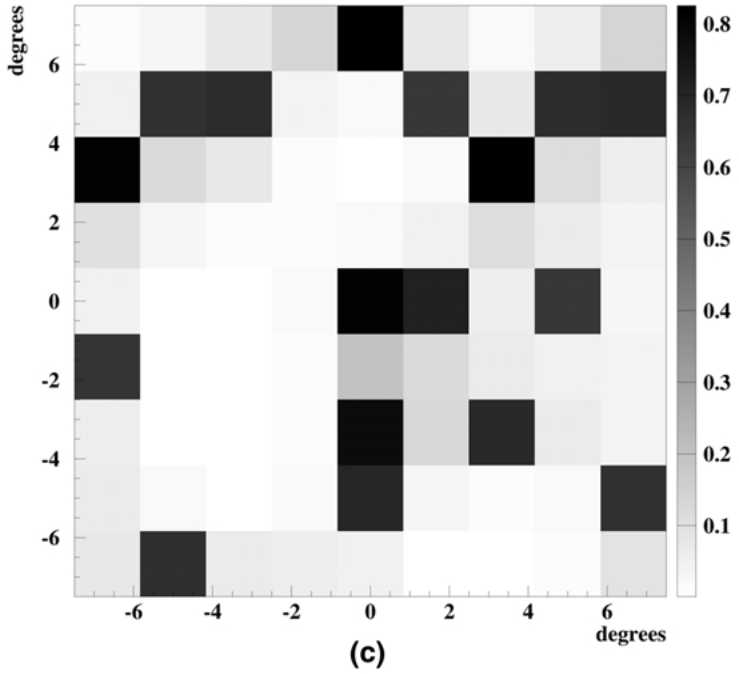


Figure 2c. (c) Image reconstructed $[\tilde{O}_{ij}]$ by the EM algorithm. The bar at the right shows the meaning of the grey scale.

will learn, because different sky configurations producing identical detector planes obstruct the convergence of the learning process. Therefore, we provide a criterion in order to make the mapping bijective. This criterion is given by a previously existent reconstruction method (the EM algorithm). That is, the NN cannot be used as a direct reconstruction method in this context, but it can simulate others.

Obviously this method cannot be applied to emulate the EM algorithm in the case of very complex telescopes as IBIS, since generating each image of the set takes 65 days, but the method could also be used to speed up reconstructions by means of correlation (where each image takes 25 min).

Here we have used for the learning process sky images of 9×9 pixels and a detector plane composed of 5×5 elements. The architecture of the NN is composed of three layers: an input layer of 5×5 neurons, a hidden layer of 9×9 neurons and an output layer with 9×9 neurons. The NN has been trained with a learning set of $N = 5000$ samples (total pairs of detected data $[D_{kl}]$-desired output $[\tilde{O}_{ij}]$). It uses the following parameters: learning parameter equal to 0.001 and momentum parameter equal to 0.01.

We evaluate at each learning momentum the behavior of the NN by the relative error RE defined (Zell et al., 1991) as:

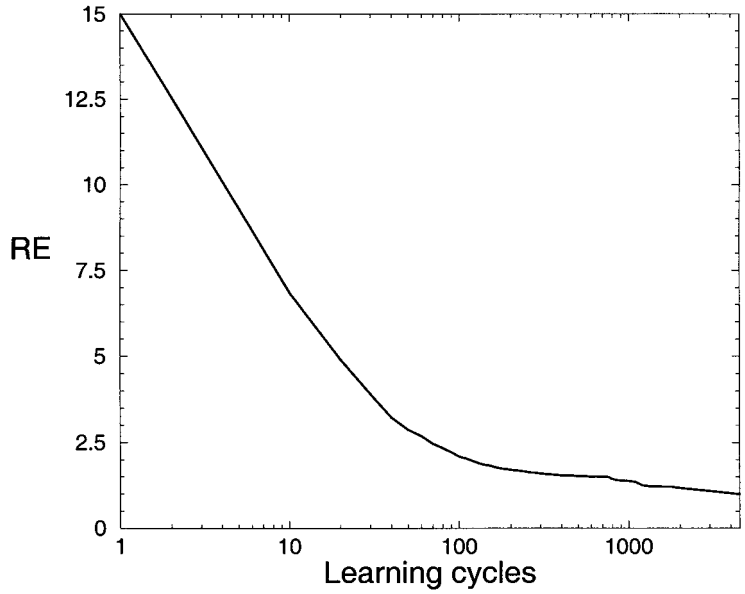


Figure 3. Evolution of the mean relative error RE (see Equation (3)) during the learning of the NN. Each iteration represents a learning step where all the samples of the learning set are presented to the NN. The process stops when $RE = 1$.

$$RE = \frac{1}{N} \sum_{s=1}^{N} \sum_{ij} \left(\tilde{O}_{ij}^{(s)} - \Theta_{ij}^{(s)} \right)^2, \tag{3}$$

where N is the number of samples in the learning set (5000), $[\tilde{O}_{ij}^{(s)}]$ is the s-th sample of the desired output set and $[\Theta_{ij}^{(s)}]$ is the s-th neural network estimation.

The learning process (see Figure 3) is stopped when the RE for the learning data is under a certain threshold. In this case $RE = 1$. For this purpose 4500 learning cycles, about 12 hours, are needed.

We use the error definition of Equation (3) as the stopping criterion because it is the one displayed by the SNNS package (together with the 'absolute' error, which is equal to Equation (3) but not divided by N). The name 'relative' comes from the fact that it is averaged by the number of elements of the training set. But it is not actually a relative error. Thus, in order to compare the difference between EM and NN images and to test the goodness of the reconstruction we are going to define a *really* relative error R which can be written as a function of RE:

$$R = \sqrt{\frac{\sum_{s=1}^{N} \sum_{ij} \left(\tilde{O}_{ij}^{(s)} - \Theta_{ij}^{(s)} \right)^2}{\sum_{s=1}^{N} \sum_{ij} \left(\tilde{O}_{ij}^{(s)} \right)^2}} = \sqrt{\frac{N \times RE}{\sum_{s=1}^{N} \sum_{ij} \left(\tilde{O}_{ij}^{(s)} \right)^2}}, \tag{4}$$

In this case, $RE = 1$ gives $R = 0.038$, that is, the final relative error at the end of the learning process is 3.8%.

Though the EM algorithm is not a linear algorithm, its final response is rather linear, that is, the reconstruction of a sky source with intensity $2I$ is just the double of the reconstruction of a sky source in the same position with intensity I almost independently of the other sky sources (Ballesteros, 1992). Moreover, it is quite insensitive to noise in the detector plane. Given this linearity in the response, training the NN with point sources with an intensity of 1 and without noise is enough to teach the NN.

In Figure 4 we show an example of a more realistic case, not used in the training set. In Figure 4a we have as sky input for our simulator 4 sky point sources with intensities 6.4, 8.8, 15.3 and 20.2 photons per area unit per time unit, that is a mean source intensity of about 12. There is also a Poissonian random background with a mean value of 3 counts per detector unit per time unit. Considering the square root of the background (an estimation of the statistical fluctuation of the background) measured in the whole detector plane as the noise ($5 \times 5 = 25$ detector units) as the noise, considering the efficiency of the detector plane equal to 0.75 (for each detector unit, the efficiency was generated randomly between 0.5 and 1.0), and considering that only a half of the signal from the sky sources passes through the mask, the signal-to-noise ratio is given by

$$\frac{S}{N} = \frac{0.5 \times 0.75 \times 25 \times \sum skysources}{\sqrt{background \times 25}} = \frac{473.3}{8.7} = 54.4 \, . \tag{5}$$

The direct reconstruction with the EM algorithm (Figure 4b) gives a typical spreading of the reconstructed signal in the cases where the source is not centered at a pixel of the reconstructed signal. The spreading of the sources is as follows: 15.3 and 20.2 over four pixels, 6.4 over two pixels and 8.8 over one pixel. The reason of this distribution is that the simulator (and the real detector too), is unable to properly resolve a source in a non-centered position: the detected response is equivalent to have two (or four) well centered weaker sources.

As can be seen, the reconstruction produced by the NN (Figure 4c) from the detector data (in this case the data do not belong to the training set) is very similar to the one produce by the EM algorithm. In fact the relative error between them is just $R = 0.047$ (in this case, Equation (4) has been applied with $N = 1$, as the set has only one case).

In order to test the goodness of the NN reconstructions, we have carried out a set of one hundred simulations, similar to the last one. Each one has a number of sources randomly chosen between 1 and 20 in random positions and with values for their intensities chosen randomly in the range 1 to 100. Besides, for each case a random noise with mean values ranging between 0 and 50 counts has been added to the detector units.

From these simulated data we reconstruct the sky using the trained NN and the EM algorithm, calculating afterwards their relative error between them, R, and

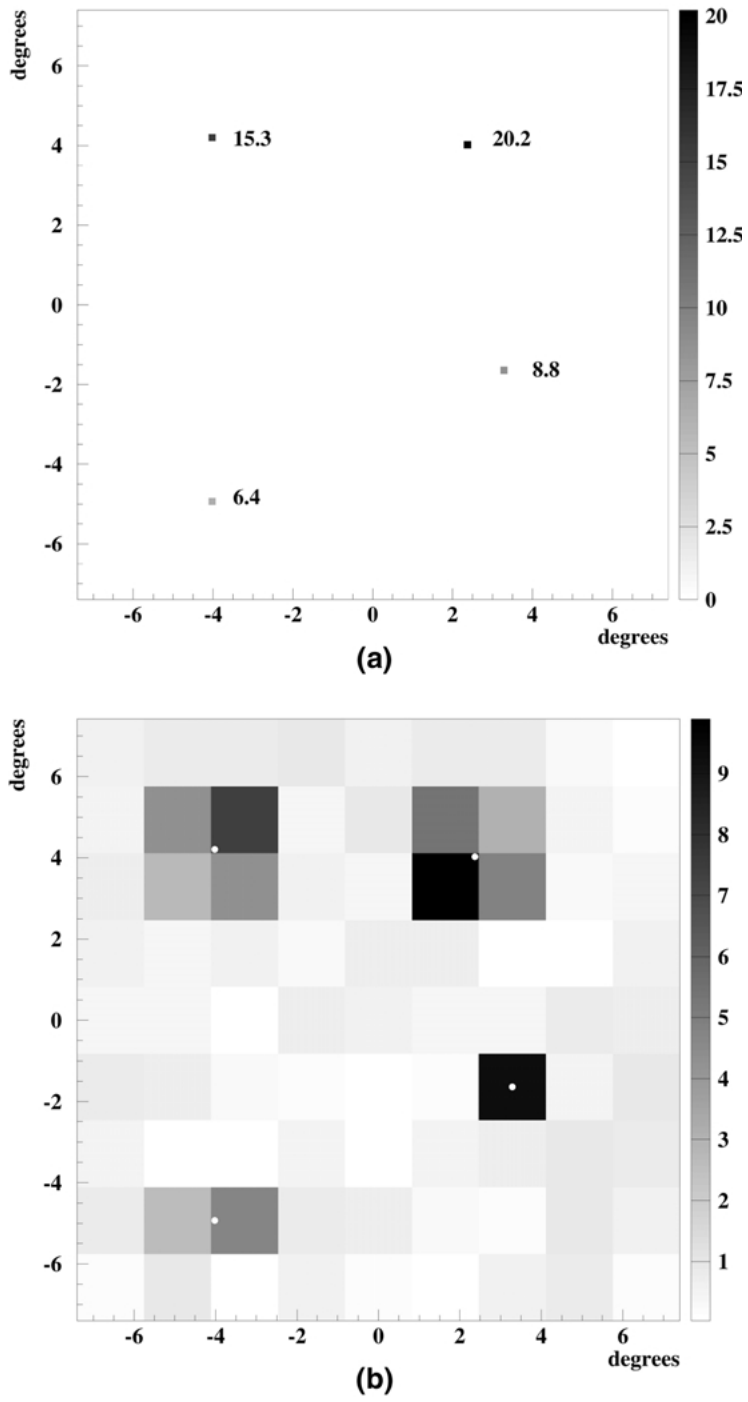


Figure 4a-b. (a) Example of sky to test the neural network: four sky point-sources with different intensities used as input sky in the simulator. Besides a random noise with mean 3 counts per pixel in the detector plane is added. (b) Reconstruction $[\tilde{O}_{ij}]$ using the EM algorithm from the signal detected with noise.

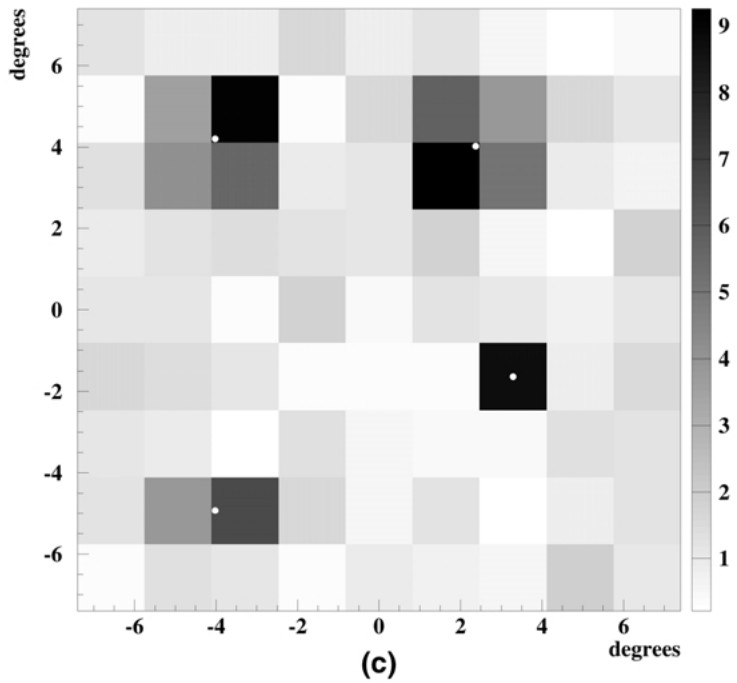


Figure 4c. (c) Reconstruction $[\Theta_{ij}]$ using the trained NN. The bar at the right shows the meaning of the grey scale.

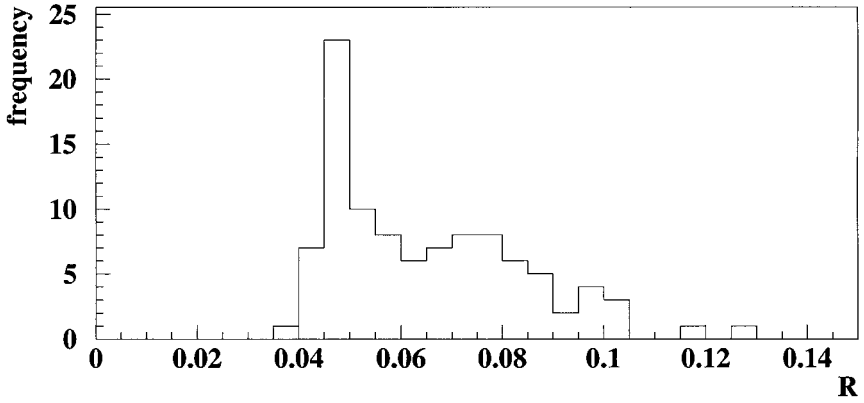


Figure 5. Histogram of the relative error, R, between reconstructions with the trained NN and the EM algorithm, obtained for a set of 100 image reconstructions, which do not belong to the training set.

considering separately each case (that is, using $N = 1$ in Equation (4)). For greater clarity the resulting relative error of these 100 simulations (not belonging to the training set) is presented in the form of an histogram (Figure 5). As can be seen, the relative error, R, produced by the neural network in the reconstruction ranges mainly between 4 and 9%. Considering as a single set with $N = 100$ all the simulations, the relative error of the whole set is $R = 0.068$ (6.8%). Therefore it can be concluded that the trained NN simulates quite well the original reconstruction method (the EM algorithm) and it seems that, due to the linearity in the response of the EM algorithm, it is enough for our purposes to train the NN using sources of intensity 1 without noise. Notice also that once the NN has learned it is able to carry out a reconstruction in real time ($\sim$1 ms).

In order to show that this good response is not limited to the case of point sources but also to well-defined extended sources, we are going to work with the other camera previously defined, as it allows a greater number of pixels in the reconstructed image and therefore better image capabilities. The associated network has a 7×7 detector plane and 13×13 reconstructed images, with 13×13 cells in the hidden layer and also a set of $N = 5000$ samples. The training takes about 3 days with the same kind of learning curve depicted in Figure 3. It uses a learning parameter of 0.001 and a momentum of 0.035.

The sky source ($[O_{ij}]$) used in this case is a cross configuration, as can be seen in Figure 6a. In Figure 6b can be seen the signal detected in the simulator, ($[D_{ij}]$). If we reconstruct the sky from this detected data using the EM algorithm, we obtain Figure 6c, which shows a typical blur in the center due to the closeness of the sources. If we reconstruct the sky from the same data $[D_{ij}]$ using our NN, we obtain the $[\Theta_{ij}]$ shown in Figure 6d. The relative error in this case is a bit higher, $R = 0.088$, although the error is still under 10%.

4. Discussion

Our analysis shows that it is possible to speed up a reconstruction method, such as the EM algorithm, by using a classical NN. Its main shortcoming is that it takes a long time for learning. However, once it has learned, the reconstruction of the image is almost instantaneous ($\sim$1 ms).

Here we have to notice that the learning of the NN can be performed before the camera is operative. In telescopes on board satellites (like SPI or LEGRI), there is a long time interval between the instant when all the instruments are fully developed (but still not inside the satellite) and the moment when they are fully operational, already in orbit and integrated to the satellite. This allows us to train the net using the real response of the telescope before the first images are obtained.

A real-time image reconstruction is useful not only when one has to deal with many sets of data (as those that can be produced in the lifetime of a mission, between two to five years) but also when a quick response is needed. An example

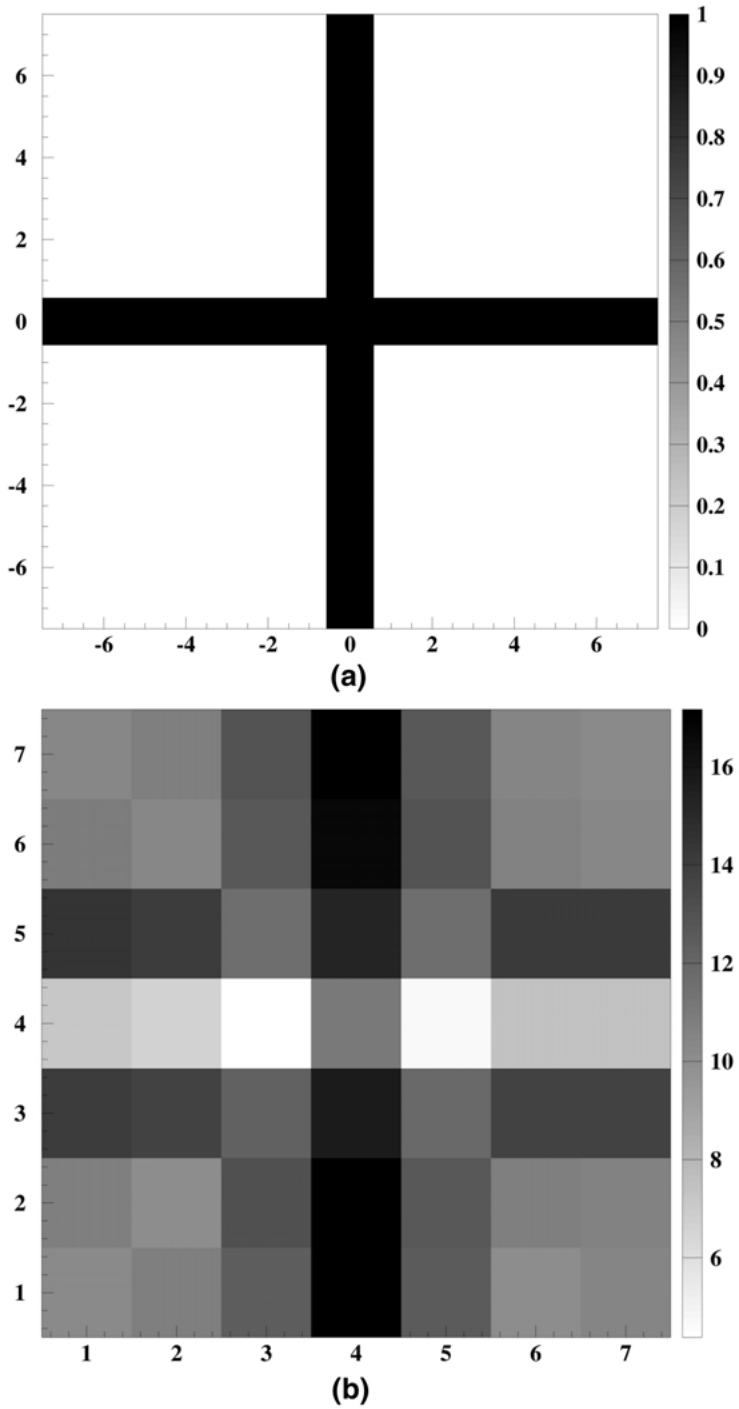


Figure 6a-b. (a) 13×13 cross-like source [O_{ij}] used for test the NN. (b) Signal detected [D_{kl}] in the detector plane.

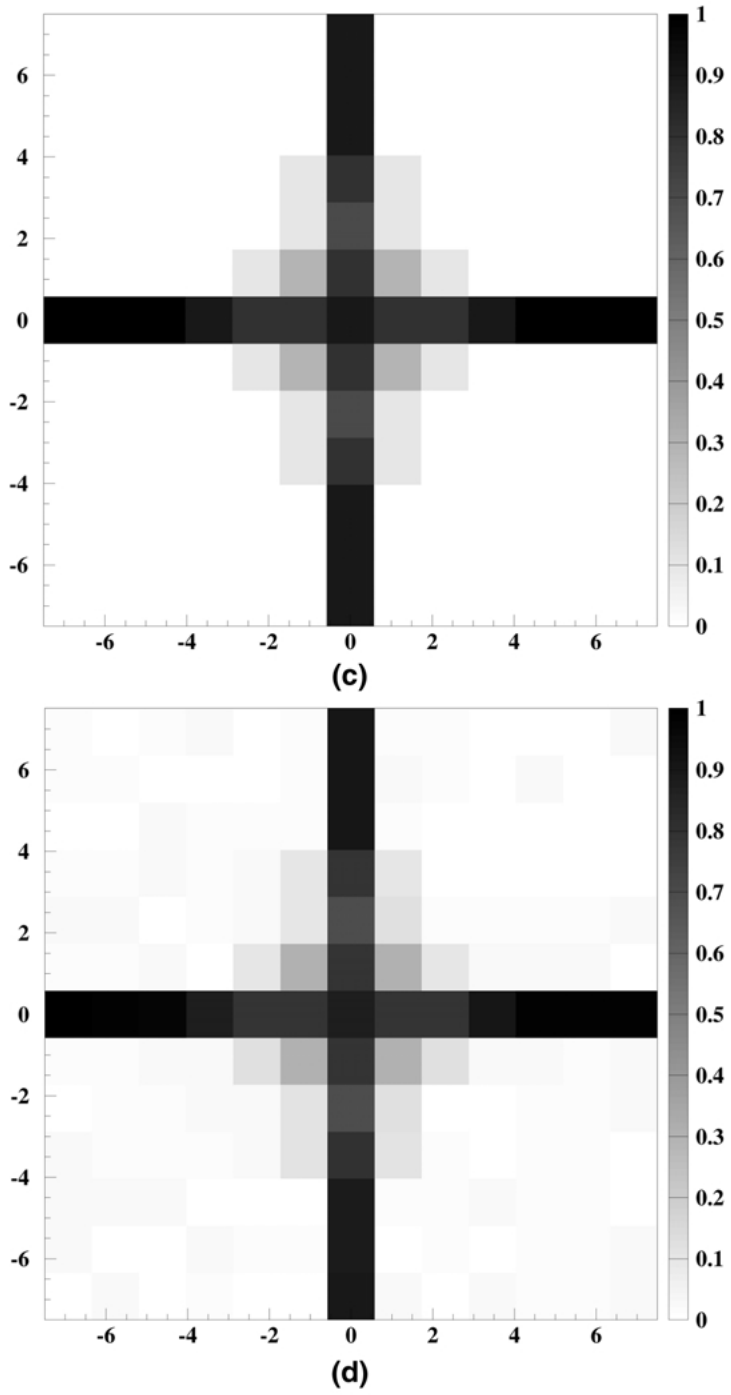


Figure 6c-d. (c) Reconstruction $[\tilde{O}_{ij}]$ using the EM algorithm from the signal detected. (d) Reconstruction $[\Theta_{ij}]$ using the trained NN. The bar at the right shows the meaning of the grey scale.

is the quick-look procedure (Roberts et al., 1996). The quick-look is a standard procedure in space-borne telescopes where a quick look of the data is done periodically, obtaining rough, fast image reconstructions. Its aim is to detect the location of any unexpected (and interesting) sky source (in this case gamma or X sources) inside the field of view of the camera (i.e. a gamma bursts) in the shortest time, in order to change as soon as possible the pointing. Another example where a quick response is needed is the monitoring of nuclear power stations or tokamaks (Ress et al., 1985) to control possible radiation or plasma leaks.

An obvious extension of our proposal is to implement this type of NN in a hardware chip installed in the proper instrument of measure, so real time images can be obtained. It would be also interesting in nuclear medicine and nuclear physics fields.

We have to consider that the NN we have used is a public domain NN package, which can simulate several types of NN's and therefore it is not optimized in time for any concrete neural network. A fully dedicated NN, specifically programmed for this problem will substantially increase the speed of learning process.

Acknowledgments

The authors thank Luisa M. Lara and R. García-Pelayo for assistance and our anonymous referee for the comments which provided useful perspectives on our results. This work has been supported by the Centro de Astrobiología.

References

Ballesteros, F. J., et al.: 1998, The EM imaging reconstruction method in γ-ray astronomy, *Nucl. Inst. and Meth.* **B 145**.

Ballesteros, F. J.: 1992, Development of imaging techniques in gamma-ray astronomy using coded mask systems. Application to the Telescope LEGRI, Ph.D. Thesis, University of Valencia, ISBN 84-370-3645-3.

Fenimore, E. E. and Cannon, T. M.: 1981, Uniformly redundant arrays: digital reconstruction methods, *Appl. Opt.* **20**, 1858.

Gottesman, S. R. and Fenimore, E. E.: 1989, New family of binary arrays for coded aperture imaging, *Appl. Opt.* **28**, 4344.

Hammersley, A. P.: 1986, The reconstruction of coded mask under conditions realistic to X-ray astronomy observations, Ph.D. Thesis, University of Birminghan.

Hammersley, A. P., et al.: 1992, Reconstruction of images from a coded-aperture box camera, *Nucl. Inst. and Meth.* **A 311**, 585.

Haykin, S.: 1998, *Neural Networks: A Comprehensive Foundation*, Mac-Millan.

Lavigne, J. M., Jean, P., Kandel, B., Borrel, V., Roques, J. P., Lichti, G., Schönfelder, V., Diehl, R., Georgii, R., Kirchner, T., Durouchoux, P., Cordier, B., Diallo, N., Sánchez, F., Payne, B., Leleux, P., Caraveo, P., Teegarden, B., Matteson, J., Slassi-Sennou, S., Skinner, G. K. and Connell, P. H.: 1998, The INTEGRAL experiment, *Nucl. Physics (Suppl.)* **B 60**, 69.

LeCun, Y.: 1985, Une procedure d'apprentissage pour reseau a seuil assymetrique, *Proc. Cognitiva* **85**, 599.

Ohyama, N., Honda, T., Tsujiuchi, J., Matumoto, T., Iinuma, T. A. and Ishimatsu, K.: 1983, Advanced coded-aperture imaging-system for nuclear-medicine, *Applied Optics* **22**, 3555.

Pedersen, T. Sunn and Granetz, R. S.: 1984, Edge X-ray imaging measurements of plasma edge in Alcator, *Review of Scientific Instruments* **70** (No. 1 January), 586.

Ponman, T. J.: 1984, 'Maximum Entropy Methods', *Nucl. Inst. and Meth.* **72**, 221.

Reglero, V., et al.: 1996, in *Proceedings of the Second INTEGRAL International Workshop 'The Transparent Universe'*, 16–20 September 1996, St. Malo, France, p. 343.

Ress, D., Bell, P. M. and Bradley, D. K.: 1993, A time-resolved X-ray ring coded-aperture microscope for inertial confinement fusion aplications, *Re. Sci. Instrum.* **64**, 1404.

Roberts, A., Ballesteros, F. et al.: 1998, in *International Symposium 'Small Satellite Systems and Services'*, 14–18 September 1998, Antibes – Juan Les Pins, France.

Rumelhart, D. E., Hinton, D. E. and Williams, R. J.: 1986, *Parallel Distributed Processing: Exploration in the Microstructure of Cognition*, Cambridge, MA, MIT Press.

Skinner, G. K.: 1988, *X-Ray Imaging with Coded Mask*, Scientific American, August.

Werbos, P. J.: 1974, *Beyond Regression: New Tools for Prediction and Analysis in the Behavioral Sciences*, Harvard University, Cambridge, MA.

Zell, A., Mache, N., Sommer, T. and Korb, T. G.: 1991, The SNNS neural network simulator, in *GWAI-91, 15. Fachtagung für künstliche Intelligenz*, Informatik-Fachberichte 285, pp. 254–263, Springer-Verlag,