Three-dimensional reconstruction of the electron-positron momentum density via supervised artificial neural network

1998 J. Phys.: Condens. Matter 10 10517

(http://iopscience.iop.org/0953-8984/10/46/018)

J. Phys.: Condens. Matter **10** (1998) 10517–10528. Printed in the UK PII: S0953-8984(98)97466-5

Three-dimensional reconstruction of the electron–positron momentum density via supervised artificial neural network

Maurizio Biasini

ENEA (Istituto Nazionale di Fisica della Materia), via don Fiammelli 2, 40128 Bologna, Italy

Received 30 April 1998

Abstract. Preliminary results of an artificial neural network approach to 3D image reconstruction are reported. The set of $(n-1)$-dimensional projections of the n-dimensional image and the pixels of the image itself constitute the input and output layers of the system, respectively. The layers are directly interconnected via a network of synapses, each associated with a specific weight. The network is trained using a supervised scheme: the projections of a set of point patterns are fed into the network and the reconstructed images are compared to the known patterns. The weights, organized as a matrix, are iteratively updated to minimize the mean square error between the ideal and reconstructed patterns, until convergence is reached. Once the training process is completed, the network can generalize and reconstruct any 3D density, one plane at a time, from the one-dimensional slices of the 2D experimental data. In this work, the network is trained to simultaneously deconvolute the ideal data from the experimental resolution. The symmetry of the momentum density is inserted in the learning procedure to reduce the memory requirements of the algorithm. The reconstructed data are compared with those obtained from the standard filtered-back-projection method. The numerical simulations considered show a reduction in mean square error of up to a factor of five compared to the back-projection results.

1. Introduction

The two-dimensional angular correlation of annihilation technique (2D-ACAR), thanks to its direct access to the electron momentum wave function amplitude, is a unique tool to investigate the electronic structure of intermetallic compounds [1]. It is well known that the data analysis of the large acquisition matrices produced in the 2D-ACAR experiments is a very delicate and time consuming task. Probably the most challenging numerical problem is the reconstruction of the full 3D momentum density $\rho(\boldsymbol{p})$ or $\boldsymbol{k}$-space density $\rho(\boldsymbol{k})$ from the 2D-ACAR projections. As the most useful information which can be obtained from the knowledge of $\rho(\boldsymbol{k})$ consists in the Fermi surface (FS), which is revealed by the discontinuities of $\rho(\boldsymbol{k})$ (usually denoted as Fermi breaks), the problem of the smearing of these discontinuities due to a finite experimental resolution is another major issue of the data analysis.

Up to now, these two problems have been tackled separately. Many methods of reconstruction were considered by the 2D-ACAR experimenters. Some were borrowed from medical imaging, such as the inverse Fourier transform methods, the most commonly adopted one being the filtered-back-projection technique (FBP) [2]. More recently, Pecora [3] applied a basic idea of Majumdar [4] to expand the momentum densities in spherical harmonics and find the coefficients of the expansion on the basis of the projected data. In this method the crystal symmetry is explicitly inserted into the reconstruction in order to

0953-8984/98/4610517+12$19.50

reduce the degrees of freedom of the problem. Another treatment, proposed by Cormack [5] and applied to positron data by Kontrym-Sznaid [6], reconstructs the 3D momentum density on planes usually chosen to be perpendicular to the main symmetry axis. The reconstruction of the full 3D momentum density is then performed by piling up the reconstructed planes. This method, which is also based on polynomial expansion, was applied recently to obtain the Fermi surface of yttrium [7].

The problem of deconvolving the experimental data from the smearing of the experimental resolution is a recurring issue of the data analysis. Many recipes, including iterative, filtering, maximum entropy algorithms were attempted [8, 9]. The maximum entropy idea has recently been used in the 2D-ACAR field obtaining good results [10, 11]. The constraint to maximize the entropy has the benefit of reducing the noise-dependent artefacts which are often present in the deconvolution techniques. However, care must be exercised when one is trying to deconvolute the Fermi breaks. In fact, the entropy of discontinuities such as those generated by the interruptions of the occupation of the conduction bands (at the Fermi wavevectors, k_F) is much lower than that of the smooth function generated by the smearing of these breaks by action of the experimental resolution. Therefore, the maximum entropy algorithm might not be appropriate to recover the ideal FS discontinuities.

Artificial neural networks (ANNs) were developed for a wide variety of computational problems in pattern and language recognition, decision making and on line data analysis. The term 'neural network' is applied to a class of parallel computational units which are arranged in layers with a structure which imitates the architecture of the brain. The common properties between ANNs and biological systems consist of parallel structures, distributed memories and ability to learn from experience to solve new problems. The multiple interconnections (synapses) of the network have weights to be adjusted through a training procedure which is either supervised, when sample input output pairs are presented, or unsupervised, when the network self-organizes [12]. This preliminary implementation uses filtered projections as input data, trains the network assuming the crystal symmetry and trying at the same time to reduce the smearing effect of the experimental resolution. At this stage, the noise-free simulated images reconstructed via the trained network show a clear improvement compared to the FBP results.

2. Method

In general, the action of projecting an nD array over a discrete number of angles onto an $(n-1)$D detector can be described as

$$P_i = \sum_j^{N_a} T_{ij} S_j \tag{1}$$

where P_i is the projection counts in bin i, T_{ij} is the weight of the contribution of the nD array bin j to the projection bin i, S_j is the intensity of the bin j and N_a is the number of pixels of the nD array. In the notation of equation (1) the sequence of the projections and of the original array are viewed as single-column vectors which are interconnected via the matrix T_{ij}. The reconstruction problem consists then of finding the inverse matrix T_{ij}^{-1} which recovers the original array from its projections, according to the expression

$$S_i = \sum_j^{N \times N_{proj}} T_{ij}^{-1} P_j \tag{2}$$

where N_{proj} is the number of projections produced, each consisting of N pixels. Although in principle one could apply the method to 3D arrays and 2D projections, computer memory limits render practically impossible a direct 3D reconstruction. As for other reconstruction techniques, the process of reconstruction is split in the two-step procedure of reconstructing 2D planes and piling up the 2D planes to produce the original 3D array. In this case, if the linear size of a square 2D reconstructed matrix is N, the sizes of T_{ij}^{-1} are $N^2 \times N \times N_{proj}$.

The network is trained according to the generalized delta rule [13]. The approach adopted here assumed that input and output layers were directly interconnected, i.e. that hidden layers were absent. Under this assumption one could obtain a direct correspondence between the neural network topology and the analytic description of equation (1). According to the neural network general scheme, the response O_i of the output neurons are generated via the equation

$$O_i = f\left(\sum_{j}^{N \times N_{proj}} W_{ij} P_j + \theta_i \right) \tag{3}$$

where P_j is the input layer, (here the pixels of the projected data), W_{ij} are the interconnecting synapses and f can be a sigmoidal function, a step function or just the identity. The thresholds θ_i, yielding additional degrees of freedom, are normally inserted when the response of the output layer is not a linear function of the input layer. In the present case, the linearity of the problem (and the agreement with equation (2)) requires that the function f be the identity and that the thresholds θ_i be set to zero. For all the t labelled training pairs, consisting of some 2D patterns S_i^t and their set of projections P_j^t, the outputs O_i^t are generated via equation (3) and compared to the true patterns S_i^t. The weights are then iteratively updated as follows:

$$W_{ij}^{N+1} = W_{ij}^N + \Delta W_{ij} \equiv W_{ij}^N + \eta \sum_t \delta_i^t P_j^t \tag{4}$$

where W_{ij}^N and W_{ij}^{N+1} are the values of the weights connecting the projection bin counts P_j^t to the reconstructed pattern intensity O_i^t before and after the $(N+1)$th readjustment, respectively, η is the training coefficient, to be optimized empirically, and δ_i^t is the current difference between the actual and ideal outputs $(O_i^t - S_i^t)$. The correction ΔW_{ij} is obtained according to a feed-forward mechanism which uses the standard gradient descent to minimize the mean square error (MSE) between the ANN reconstructed and ideal patterns. The MSE is defined as

$$\mathrm{MSE} = \sum_t^{N_p} \left(\sum_i^{N^2} \left(\sum_k^{N \times N_{proj}} W_{ik} P_k^t - S_i^t \right)^2 \right) \tag{5}$$

where N_p is the number of training patterns.

The iterative correction to the weights which appears in equation (4) then becomes $\Delta W_{ij} = -\eta \partial \mathrm{MSE} / \partial W_{ij}$. The training should halt when the global MSE of the training set has converged. Often, the correction to the weights is supplemented by an auxiliary term, denoted as the momentum term, proportional to the correction to the weights of the previous iteration [14]. The purpose of keeping a direction in the W_{ij} space slightly different from that of the steepest descent is to reduce the chance of the $\mathrm{MSE}(W_{ij})$ function becoming trapped in local minima. The correction ΔW_{ij} in equation (4) was then modified to

$$\Delta W_{ij}^{N+1} = \eta \sum_t \delta_i^t P_j^t + \mu \Delta W_{ij}^N \tag{6}$$

where μ is the empirical proportionality constant of the momentum term.

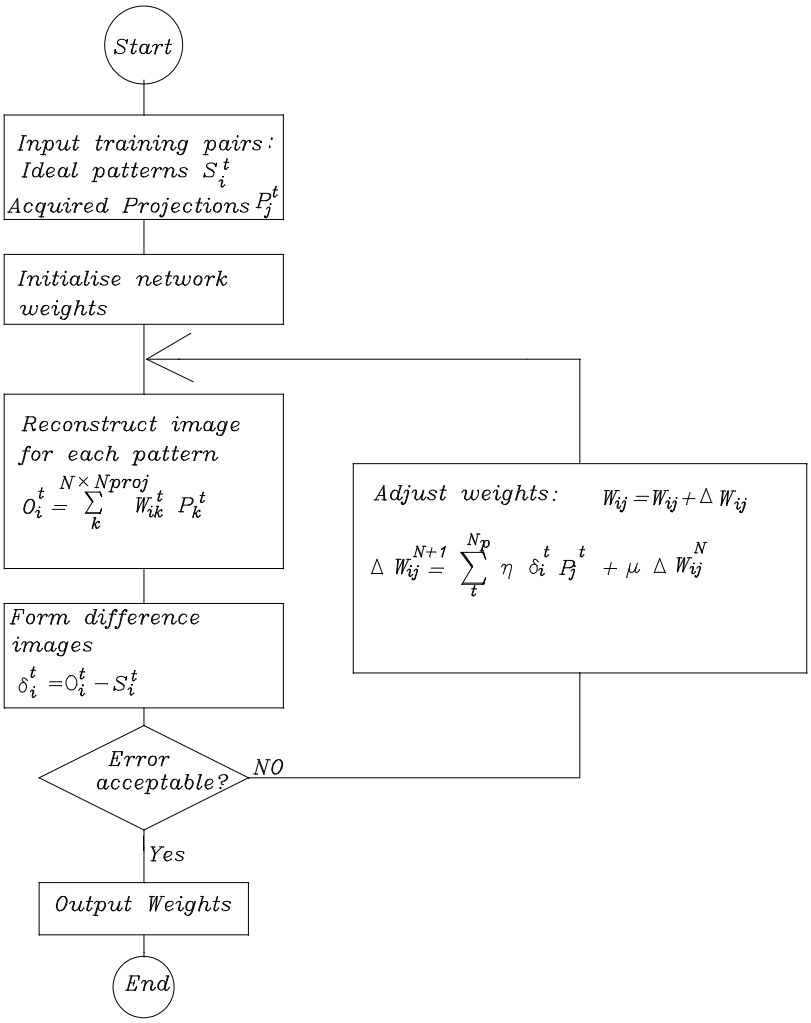

Figure 1. Flow chart of the training phase.

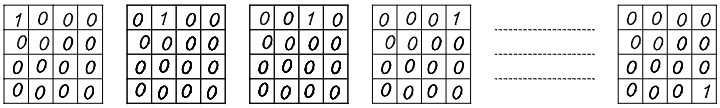

Figure 2. The *point-like* training matrices for a 4 × 4 bin case.

Figure 1 shows a flow chart of the training procedure. The training pairs of this study corresponded to point source matrices consisting of a single non-zero pixel (set to 1) in a predefined field of view and the corresponding set of projections. Figure 2 shows, as an example, the training matrices for a 4 × 4 bin case.

After the training has ended any $N \times N$ matrix can be reconstructed by applying the weights W_{ij} to its set of projections, according to equations (2) and (3). At first, the algorithm was tested for small arrays. They consisted of an 8 × 8 bin matrix and 16 bin

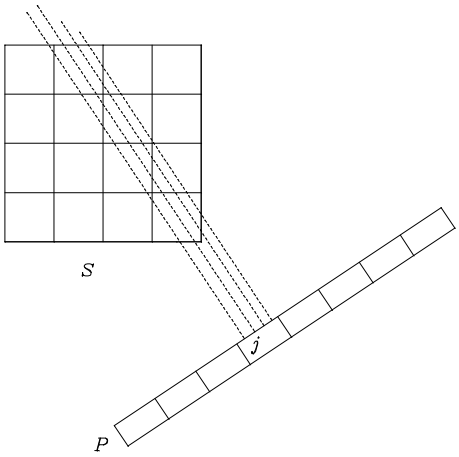

Figure 3. The construction of the projection vector at a given angle. For each projection pixel j, several rays intersect the matrix S and the pixels of interest (say l, m) contribute to $P(j)$ by an amount proportional to the intensity $S(l, m)$ times the average line segment across them.

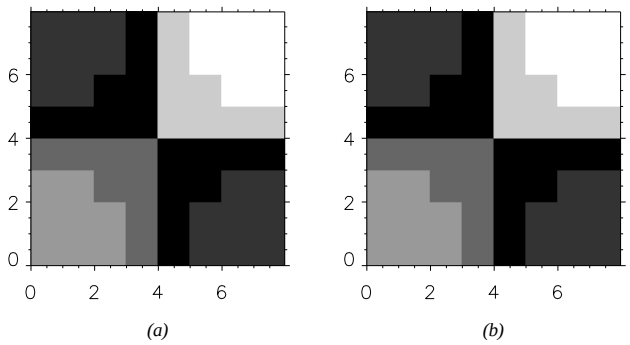

Figure 4. (a) A simulated 8×8 channel matrix. (b) The matrix reconstructed via ANN.

projection vectors for 12 projection angles. The field of view of the training pairs covered the whole matrix. When the bin number is so limited a special attention must be devoted to the projection operator. Therefore, a special algorithm was implemented: several straight lines (rays) per projection vector bin intersected the 2D matrix pixels and a fraction of the pixel intensity proportional to the average ray segment across was included in the projection bin in question (see figure 3). The operator was applied to generate the projections of the training patterns and of the 2D matrices which were tested for reconstruction after the training.

In this initial test the weights W_{ij} were all initialized to 0.1. The training coefficients η and μ which optimized the results were 0.001 and 0.9 respectively, in agreement with the values reported in the literature [15]. The iterative procedure was stopped at the 900th iteration, yielding an SME of $\approx$0.1. The quality of the reconstruction operated by the network was tested by monitoring the scatter between simulated matrices and the corresponding reconstructed counterparts. Figures 4 and 5 show two 8×8 bin examples. Whereas figure 4 consists of some kind of structured features, figure 5 contains pure noise. It appears that the reconstruction is almost perfect in trend and values of the pixel

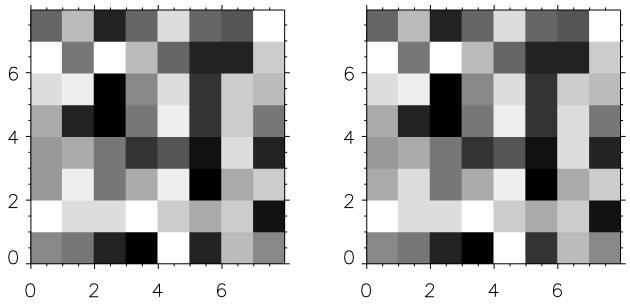

Figure 5. (a) An 8×8 channel matrix containing pure noise. (b) The matrix reconstructed via ANN.

Table 1. Listing of the precision parameter P_r, defined as $P_r = \mathrm{MSE}/\sum_i S_i$, where S_i are the simulated matrix pixels and MSE is the mean square error for the ANN or FBP reconstruction, for the examples considered in this work.

Figure number	P_r (ANN)	P_r (FBP)
4	0.0013	
5	0.006	
7	5.9	25
8	2.2	16
10	3.0	64

intensities. One can define a quantitative parameter P_r which describes the precision of the reconstruction as $P_r = \mathrm{MSE}\sum_i S_i$, where S_i are the simulated matrix pixels. The P_r parameters of all the reconstructions of this work are reported in table 1. Results of similar quality were obtained for any other 8×8 matrix tested for reconstruction. These examples show that with a sufficient number of projection pixels and ideal patterns the reconstruction via ANN could be attained perfectly. However, when the sizes of the matrix to reconstruct increase, the amount of CPU memory required overtakes the possibilities of many work-stations commonly used for the data analysis. The consequent unavoidable use of virtual memory increases enormously the time needed to reach convergence. For example, for a 64×64 matrix and 60 projections the sizes of both the W_{ij} matrix and the matrix which includes the projections of the patterns are $4096 \times 60 \times 128$ (up to a total $\approx 6.3 \times 10^7$ numbers). In practice, the 2D image is usually confined inside a circular domain. Therefore, the sizes of the projection vector need to be equal to the linear sizes of the 2D matrix as the necessary sample patterns are only those inside the circular domain. For a 64×64 2D array the weight sizes and projected pattern sizes could be reduced to $4096 \times 60 \times 64$ and $4096 \times (\pi/4) \times 60 \times 64$ respectively ($\approx 2.8 \times 10^7$ numbers), *at the expense of an imperfect reconstruction*. With this caveat, further studies concentrated on the attempt to improve the FBP results, reconstructing and deconvoluting the smearing caused by the resolution function.

The 2D matrix was assumed to have two inversion symmetry planes. In light of a future use for 3D momentum density reconstruction this should not be seen as a limitation. The crystal point symmetry group required should be at least $C_{2n\mathrm{v}}$ and the rotation should be performed about the C_{2n} axis (with $n = 1, 2, 3$). The vast majority of the crystal structures has symmetries equal or higher than $C_{2n\mathrm{v}}$.

The patterns considered were only those contained in one quarter of a circle and during the rotation their projections included also the contribution from the symmetrical pixels which unveil when the projection axis is not contained in the inversion symmetry planes.

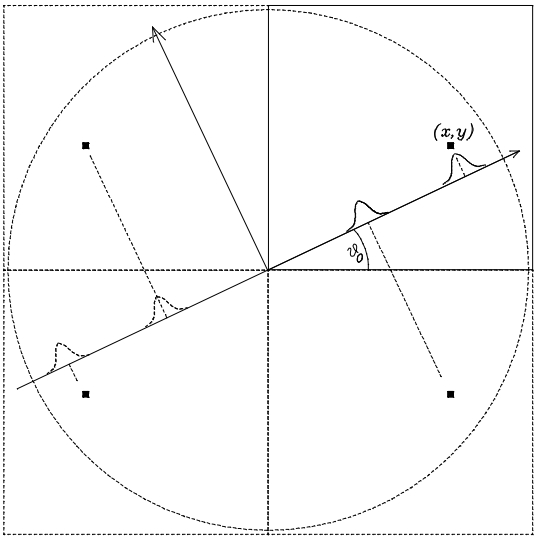

Figure 6. The 1D projection array for a specific rotation θ_0 of a training pattern characterized by having the pixel (x, y) different from zero. The contribution from symmetrical pixels consistent with a C_{2v} symmetry is included. The projection is convoluted with the experimental resolution.

The deconvoluting procedure consists simply of smearing the 1D projections of the point-like matrices with a one-dimensional slice of the experimental resolution. Indeed, the generalized delta rule, described by equations (4)–(6), aims at producing a network able to recover simultaneously all the N_p training matrices, independently of the degradation suffered by their projections. As each 2D-ACAR spectrum can be viewed as a superposition of delta-like functions at each $\boldsymbol{p}$-point, the network should be able to yield an approximate reconstruction and deconvolution of any spectrum after the iteration procedure has converged. Figure 6 shows, as an example, the 1D projection array, relative to a rotation angle θ_0, of the training pattern characterized by having the pixel (x, y) non-zero. Owing to the symmetry, only the positive part of the projection array need be used for the reconstruction. This method can easily be extended to higher symmetries. Moreover, to expedite the convergence during the training, the weights at position (i, j) were initialized to be non-zero only for (i, j) values consistent with the back-projection method and with the symmetry of the matrix. Indeed, the back-projection reconstruction can be obtained applying equation (2) to a matrix of unitary weights W_{ij} where the $[x + (n-1)y]$th row of W_{ij} differs from zero only at the positions j for which j equals $(|x|\cos\theta_n + |y|\sin\theta_n)$ [2]. Here (x, y) is the pixel of the matrix with C_{2v} symmetry to reconstruct and θ_n are the various rotation angles. The initialization had to be consistent also with the fact that the resolution broadens the response of the projection detector to a point source and with the decision to convolute the projection profiles in input with the Fourier transform of the ramp filter. The effect of this filtering, applied in the FBP technique, is to introduce negative (i.e. non-zero) intensities at pixels near those for which the relation $j = (|x|\cos\theta + |y|\sin\theta)$ holds.

The programs were written in the language IDL because of its high versatility and graphic power and run on an IBM RISC990 machine. The computing time was strongly dependent on the size of the matrices. For a 32×32 matrix to reconstruct and 33 projections the training took about 10 minutes per iteration. The time increased to more than one hour per iteration for a 48×48 channel matrix and 37 projections. As the computing time was so high the training was stopped after 100 iterations or so. Unlike the 8×8 initial test, a training coefficient η close to 100 was required. Such a large learning rate has previously been reported [16, 17].

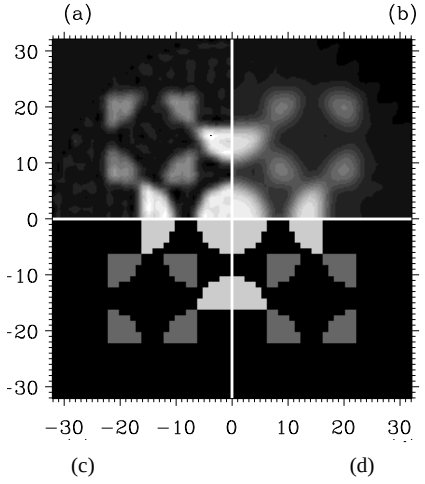

Figure 7. Test of reconstruction for a 64×64 matrix with C_{4v} symmetry: (a) ANN; (b) FBP; (c) and (d) original data. The (a) and (b) data were linearly interpolated to a size 128×128 matrix.

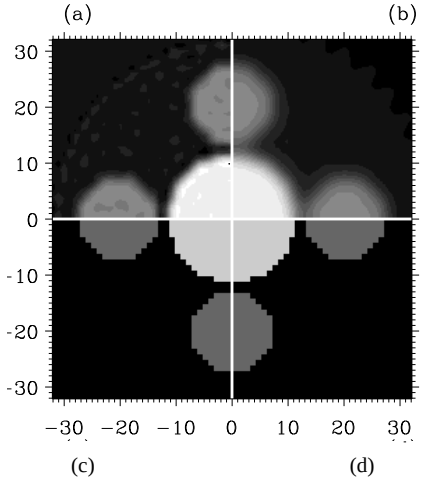

Figure 8. Test of reconstruction for a 64×64 matrix with C_{4v} symmetry: (a) ANN; (b) FBP; (c) and (d) original data. The (a) and (b) data were linearly interpolated to a size 128×128 matrix.

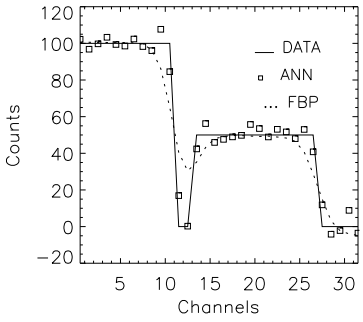

Figure 9. The slice at $y = 0$ of figure 8 (positive part). The ANN and FBP are compared to the original data (continuous lines).

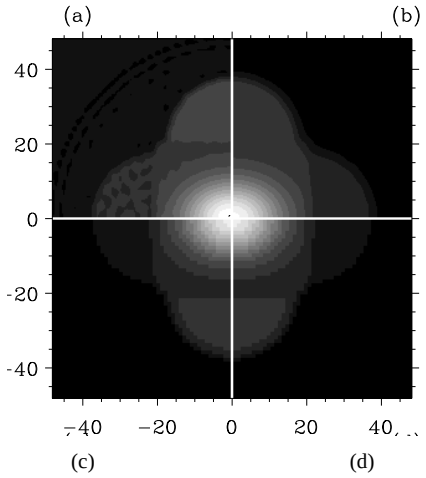

Figure 10. Test of reconstruction for a 96×96 matrix with C_{2v} symmetry: (a) ANN; (b) FBP; (c) and (d) original data. The (a) and (b) data were linearly interpolated to a size 144×144 matrix.

3. Results and discussion

The method could be applied in full for matrices up to 32×32 channels. Owing to the symmetry the weights obtained (a $32^2 \times 33 \times 32$ matrix) can reconstruct 64×64 channel matrices with inversion symmetry. A further attempt to increase the size of the matrix was possible by eliminating the momentum term in equation (6) to reduce the amount of memory of the program and apply the method to 48×48 matrices. A Gaussian resolution function, having a full width at half maximum of three channels, was assumed. In both trainings, possibly due to an insufficient number of degrees of freedom of the weights, the MSE converged to a value much higher than that attained in the 8×8 bin initial test. Therefore, it is useful to compare the MSEs to those obtained by applying the standard FBP reconstruction method with a ramp filter. Figures 7 and 8 show two examples for reconstructions of 64×64 matrices with C_{4v} symmetry. In both figures the lower part shows the simulated matrix and the upper parts show the matrix reconstructed via the ANN and FBP methods. The P_r parameters of all the examples considered, reported in table 1, are clearly in favour of the ANN scheme. However, despite the improvement in the MSE,

the quality of the ANN images is still only slightly better than the FBP ones. To illustrate more clearly the deconvoluting effect of the ANN procedure, figure 9 shows the slice at $y = 0$ of figure 8 comparing the simulated matrix with the ANN and FBP reconstructions. It should be noted that in figure 9 the FBP data had to be shifted upward by 10% of their maximum while the ANN data are raw. It appears that, at the expense of a modest increase of noise, the ANN procedure is quite efficient in recovering the discontinuity at channel 11. Figure 10 shows the comparison between the reconstructions by the ANN (figure 10(a)) and FBP (figure 10(b)) procedures for the 36×36 matrix with C_{2v} symmetry shown in figure 10(c) or 10(d). As might be expected, the improvement in resolution between ANN and FBP is less evident when the ratio between the sizes of the resolution function and the linear size of the matrix decreases. On the other hand, the deconvolution method applied in these tests brings in some kind of stability against the noise or faults introduced in the projection (typical quality of the human brain, denoted as fault tolerance). For example, unlike the case of the 8×8 matrix, the results were independent of the algorithm used to perform the projections. One drawback of the method is that the number of projections must be decided once and for all prior to the learning process.

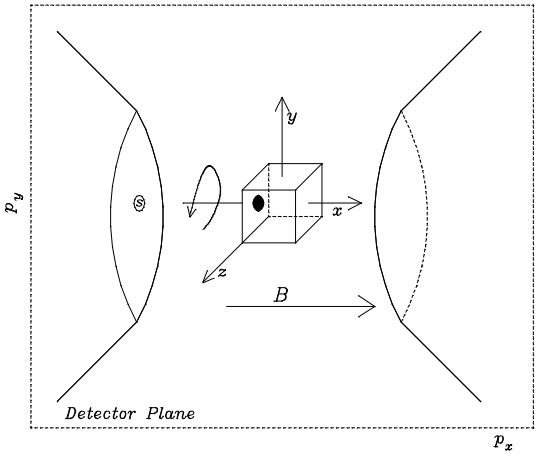

Figure 11. The projection procedure for 2D-ACAR spectrometers equipped with standard electromagnets: the positron source is indicated by the circle labelled by *S* on the pole of the electromagnet. The black spot on the sample represents the source image onto the sample. The sample is rotated about the *x* crystal axis, as indicated by the arrow. The dashed frame represents the detector plane, parallel to the *x*-direction and normal to the *z*-projection direction.

With regard to the deconvolution procedure it is worth mentioning the following: if it were possible to perform the training procedure directly from the 2D projections one could consider in full the smearing of a delta-like function at (p_{x0}, p_{y0}, p_{z0}) over the 2D projected matrix and teach the network how to recover the point-like sources. When one reconstructs the 3D array plane by plane the deconvolution procedure is incomplete because the reconstructed planes are considered independently. However, the features of the 2D-ACAR experiment reduce this problem. In the 2D-ACAR setups which adopt standard electromagnets to focus the positrons onto the sample, the geometry of the source–sample assembly determines uniquely the way the projections are collected: the sample must be rotated about the magnetic field ($\boldsymbol{B}$) direction (say x) which is in the detector plane and normal to the projection direction. It is well known that the resolution of the 2D-ACAR spectrometer is usually strongly asymmetric [1]: along the $\boldsymbol{B}$-direction the overall

resolution is determined by the spatial resolution of the 2D spectrometer (about 6.5 mm, for the Bologna spectrometer) convoluted with twice the (negligible) penetration of the positrons in the sample (0.2–0.4 mm). In the perpendicular direction (say y) the resolution is determined by the spatial resolution of the 2D spectrometer convoluted with twice the smallest between the sample size along the y-direction and the linear size of the source spot onto the sample (up to 5–10 mm). During the rotation the y- and z-directions mix, while the x-direction is unaffected. The reconstructed planes (p_{x0} = constant, p_y, p_z) are then piled up along the x-direction. Figure 11 clarifies the relevant directions of the projection procedure. Therefore, the one-dimensional deconvolution procedure operated by the network is applied to the p_y-direction which is mostly affected by the resolution.

4. Conclusion

The limited successful results presented here indicate that it is worth investigating the implementation of the ANN for the problem of the reconstruction of the electron–positron momentum density. The perfect reconstruction obtained in the simple 8×8 case, where a large number of degrees of freedom of the weights W_{ij} compared to the matrix size was available, needs to be applied to larger matrices keeping a similar ratio of the size of the matrix to the size of the weights and increasing the number of training pairs. This will only be possible by parallelizing the computer programs and by employing several parallel processors. By building a network able to reconstruct a 64×64 matrix consistent with inversion symmetry (equivalent to 128×128 bins) one could start applying the method to real data. Although modern spectrometers acquire with at least 256×256 bin matrices the loss of precision which results by using bins of double sizes should be balanced by the benefit of the deconvolution. Up to now, the present implementation for large matrices should be regarded merely as a tool to improve the FBP results. A positive feature of the method is that once the training process is completed, the computing time to obtain the reconstructed data (via equation (2)) is practically zero. Further work will consist in establishing how well the reconstruction matrices W_{ij}, produced by training pairs adopting noiseless projections, will perform when reconstructing noisy projected data.

The main philosophy of the neural network method is to acquire some expertise uniquely by analysing the behaviour of some representative examples. No knowledge of the physical process is required. The modifications due to the symmetry and the filtered projections in input were attempts to give the network some extra information to speed up and improve the learning procedure. A further benefit should come by inserting this extra information in the iterative scheme which features the weights W_{ij} rather than in the initialization and in the form of the input data.

References

[1] For a general introduction to the technique see, for example, West R 1995 *Proc. Int. School Phys. Enrico Fermi (1993)* ed A Dupasquier and A P Mills Jr (Amsterdam: IOS) p 75
[2] Brooks R A and Di Chiro G 1976 *Phys. Med. Biol.* **5** 692
[3] Pecora L M 1987 *IEEE Trans. Nucl. Sci.* **34** 642
[4] Majumdar C K 1971 *Phys. Rev.* B **4** 2111
[5] Cormack A C 1963 *J. Appl. Phys.* **34** 2722
Cormack A C 1964 *J. Appl. Phys.* **35** 2908
[6] Kontrym-Sznaid G 1990 *Phys. Status Solidi* a **117** 227
[7] Dugdale S B, Fretwell H M, Alam M A, Kontrym-Sznajd G, West R N and Badrzadeh S 1997 *Phys. Rev. Lett.* **79** 941

[8] Frieden B R 1979 *Picture Processing and Digital Filtering* ed T S Huang (New York: Springer) p 201
[9] Press W H, Teukolsky S A, Vetterling W T and Flannery B P 1986 *Numerical Recipes in Fortran: the Art of Scientific Computing* (Cambridge: Cambridge University Press)
[10] Hoffmann L, Shukla A, Peter M, Barbiellini B and Manuel A A 1993 *Nucl. Instrum. Methods* A **335** 276
[11] Dugdale D B, Alam M A, Fretwell H M, Biasini M and Wilson D 1994 *J. Phys.: Condens. Matter* **6** L435
[12] For a general introduction to the method the reader can consult Beale R and Jackson T 1990 *Neural Computing: an Introduction* (Bristol: Hilger)
Anderson J A and Rosenfeld E (eds) 1988 *Neurocomputing: Foundation Research* (Cambridge, MA: MIT Press)
[13] Ritter H and Schulten K 1987 *Proc. 1st IEEE Conf. on Neural Networks, ICNN (San Diego, CA, 1987)* (New York: IEEE) p 109
[14] McClelland and Rumelhart 1986 *Parallel Distributed Processing* (Cambridge, MA: MIT Press)
[15] Bologna G 1995 *Nuovo Saggiatore* **11** 50
[16] Floyd C E 1991 *IEEE Trans. Med. Imaging* **10** 485
[17] Munley M T, Floyd C E Jr, Bowsher J E and Coleman E 1994 *Med. Phys.* **21** 1890