

We are IntechOpen, the world's leading publisher of Open Access books Built by scientists, for scientists

4,900

Open access books available

124,000

International authors and editors

140M

Downloads

Our authors are among the

154

Countries delivered to

TOP 1%

most cited scientists

12.2%

Contributors from top 500 universities



WEB OF SCIENCE™

Selection of our books indexed in the Book Citation Index
in Web of Science™ Core Collection (BKCI)

Interested in publishing with us?
Contact book.department@intechopen.com

Numbers displayed above are based on latest data collected.
For more information visit www.intechopen.com



Applying SOFM and Its FPGA Implementation on Event Processing of PET Block Detector

Dongming Hu

Siemens Medical Solutions, Molecular Imaging
USA

1. Introduction

This chapter introduces an application case of self-organizing feature map (SOFM) on event processing of Positron Emission Tomograph (PET) detector. The Siemens Inveon Dedicated PET scanner uses an array of 64 20x20 segmented LSO detectors in a tomographic arrangement to allow PET imaging of small animals. A detector consists of 20x20 LSO crystal pixels coupled to a Position Sensitive Photomultiplier Tube (PSPMT) through tapered light guide (Fig. 1 (a)). The 511 keV gamma ray enters LSO pixel, deposits its energy and emits visible light through scintillation process. The light is guided through light guide and detected by PSPMT. The position of scintillation (X, Y) is calculated by Anger logic. The flood image (position profile) of an Inveon PET detector is shown in Fig. 1 (b). The white clouds indicate the accumulation of scintillation events.

The matching and calibration of the system electronics to the detector arrays involves three stages. The first stage identifies the individual crystal elements (pixel) from the raw X and Y values. That is to build a mapping relation between the X, Y value to the crystal index. The current implementation of crystal identification uses a hill-climbing algorithm to find the exact position of every crystal by locating the peaks in position profile. Crystal Lookup Tables (CLTs) are generated as the results of the crystal identification process. Although this method achieves around 95% accuracy, it involves intense human interactions for crystal look-up table corrections. The complex algorithm can only be implemented on a PC, which makes the crystal identification an off-line procedure.

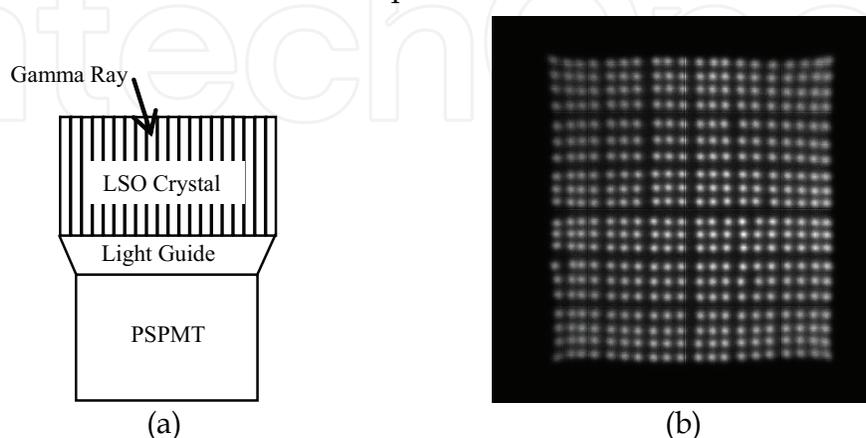


Fig. 1. Inveon PET detector and calibration. (a) Inveon PET detector. (b) Position profile of flood image

This chapter introduces the study on a neural network based algorithm for crystal identification. A modified unsupervised SOFM is trained by the incoming scintillation events to construct a CLT. The algorithm is implemented in a Field Programmable Gate Array (FPGA) chip in the Siemens Inveon Event Processing Module (EPM) electronics, which significantly reduces the training time and brings feasibility to detector real-time training. The preliminary training result shows that SOFM can be used effectively in CLT construction with excellent accuracy. The simulations show that SOFM has a good performance on real-time crystal peak tracking, which brings a good solution to eliminate crystal identification errors caused by gain drifting of PMT or Avalanche Photo Diode (APD).

2. Neural network structure (Hagan et al., 1996)

A block position profile from which a CLT is built contains millions of events, randomly detected by all pixels in a detector block. A supervised neural network is not practical in this case. However, a self-organizing (unsupervised) network can solve the problem effectively. Kohonen's self-organizing feature map is a competitive network that contains a weight vector matrix W (neurons) and a competitive layer C (Fig. 2).

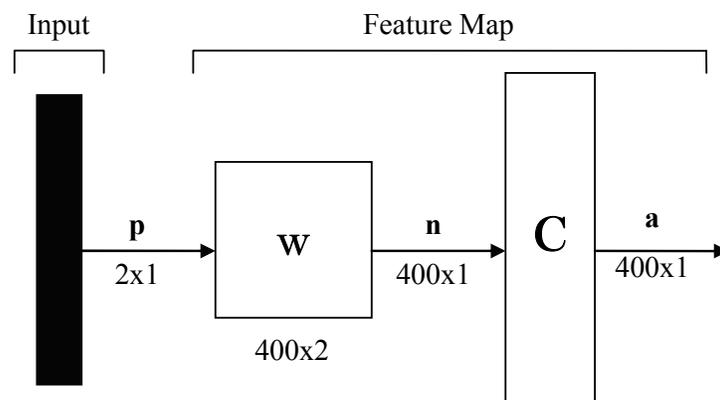


Fig. 2. Structure of self-organizing feature map

The SOFM first determines the winning neuron from n using the competitive layer.

$$\begin{aligned} \mathbf{a} &= \text{compet}(\mathbf{n}) \\ \mathbf{n} &= \mathbf{W}\mathbf{p} \end{aligned} \quad (1)$$

where \mathbf{p} is input vector with dimension of 2×1 representing the (X, Y) position of an event. \mathbf{W} is weight matrix with dimension of 400×2 representing neurons positions. Each neuron stands for a detector pixel. So for an Inveon 20×20 detector, the number of neurons is 400. In the normalized case, $\mathbf{W}\mathbf{p}$ can be regarded as the distances between \mathbf{p} and neurons weight vectors \mathbf{W} . Vector \mathbf{a} is the output from the competitive layer indicating the winning neuron. Only the neuron with the closest weight vector to \mathbf{p} wins the competition each time. Next, the weight vectors for all neurons within a certain neighborhood of the winning neuron are updated using the Kohonen rule,

$$\mathbf{w}(q) = (1 - \alpha)\mathbf{w}(q-1) + \alpha \mathbf{p}(q) \quad (2)$$

where α is learning rate, q is training index number, \mathbf{w} is the neuron in the neighborhood of the winning neuron, which has the same dimension of \mathbf{p} . Here, neighborhood 1 is used, so only the winning neuron is updated.

When a vector \mathbf{p} is presented, the position of the winning neuron will move a small distance toward \mathbf{p} . The result is that, after many presentations of \mathbf{p} from incoming events, the neuron will have learned the vectors that cluster together, which indicates a pixel location in position profile. After training, the positions of the neurons in the weight matrix represent the position of the detector pixels. A CLT can then be easily built from the neurons.

3. Simulation on SOFM training (Hu et al., 2006)

The SOFM training is first simulated using a Matlab program to verify the algorithm feasibility. The 400 neurons are evenly distributed in the interested area as initial weight vectors. Fig. 3 (a) shows a detector position profile and the neurons' initial positions (round dots). The initial weight vectors can be randomly arranged. However, since we know that the detector pixels are assembled in a grid pattern, applying this constraint in the arrangement of initial weight vectors will speed up the self-organizing process and make it reliable.

The position profile is histogrammed from a list mode file that records the sequence of all events, randomly detected by all pixels of the detector. The list mode file is acquired with a wide open energy window. Reading the (X, Y) position of an event sequentially from the list mode file as an input vector ensures that a training vector is randomly picked up from the position profile. In this simulation, three million events are used as training vectors, which indicates the weight matrix is updated 3 million times. In the Matlab program, the training algorithm is modified and the learning rate is varied over time to achieve better training solutions.

The training result is shown in Fig. 3 (b). In this simulation, most of the neurons find the right pixel location. Only two neurons are misplaced among 400 neurons, which yield 99.5% location accuracy. The total training time for 3 million events is around 8 hours on a Pentium IV 1.6GHz PC. All the neurons are involved in the competition, so training time increases with the number of neurons.

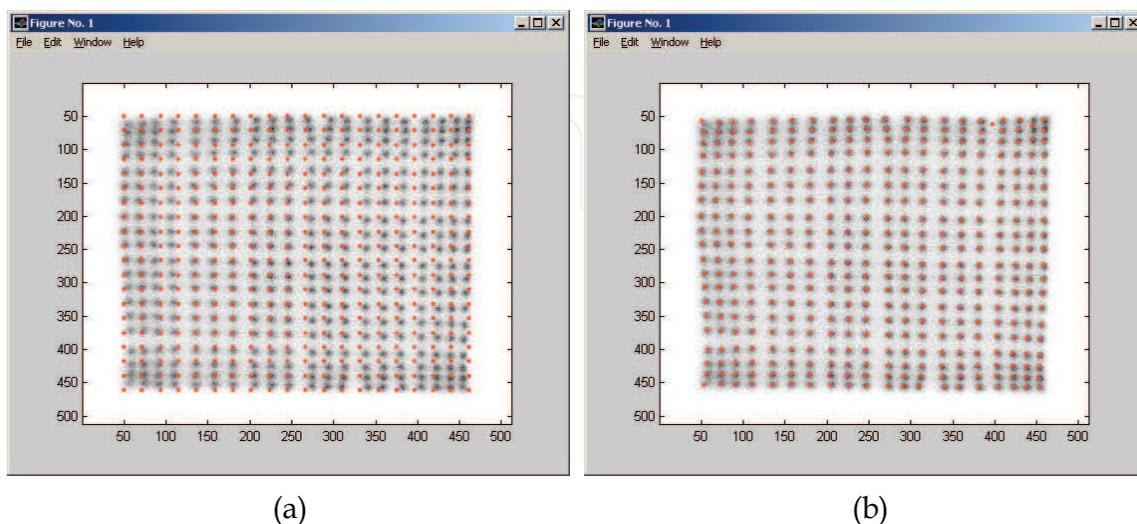


Fig. 3. Matlab simulation on SOFM training. (a) Position profile and initial positions of neurons in simulation. (b) Matlab training results

4. FPGA implementation

To make the SOFM a practical method for crystal identification, the issue of long training time must be addressed. Since the SOFM algorithm is pretty compact and simple, it can be implemented in the FPGA chip in the Inveon EPM electronics to accelerate the training process. The schematic of the FPGA implementation is shown in Fig. 4. It includes a large scale FPGA chip and several Static Random Access Memory (SRAM) chips. The FPGA implementation consists of a front end module, a SRAM addressing module, a competition pipeline and a neuron training module. A position profile is acquired first with a short period of time. Initial weight vectors of the 400 neurons based on the position profile are calculated and downloaded to the SRAM before training. The initial neurons are evenly distributed in the interested area of the position profile as discussed before. Since multiplication is involved in competition and learning calculations, and integer does not provide enough resolution, a fixed point data format with 9 integer bits and 7 fractional bits is used. So the (X, Y) position of each neuron occupies 4 bytes, with range of 0 - 512 and resolution of 2^{-7} .

A lookup table is also generated based on the initial weight vectors of the neurons, and stored in the SRAM. The lookup table defines regions indexed by the (X, Y) value of the event. Only the neurons in the same region rather than all 400 neurons compete to each other. The adjacent regions overlap each other to ensure that all necessary neurons are picked up for competition when an event comes in. The lookup table reduces the size of the competitive layer and the pipeline processing time. So the required FPGA resource and processing cycle does not increase with the number of neurons.

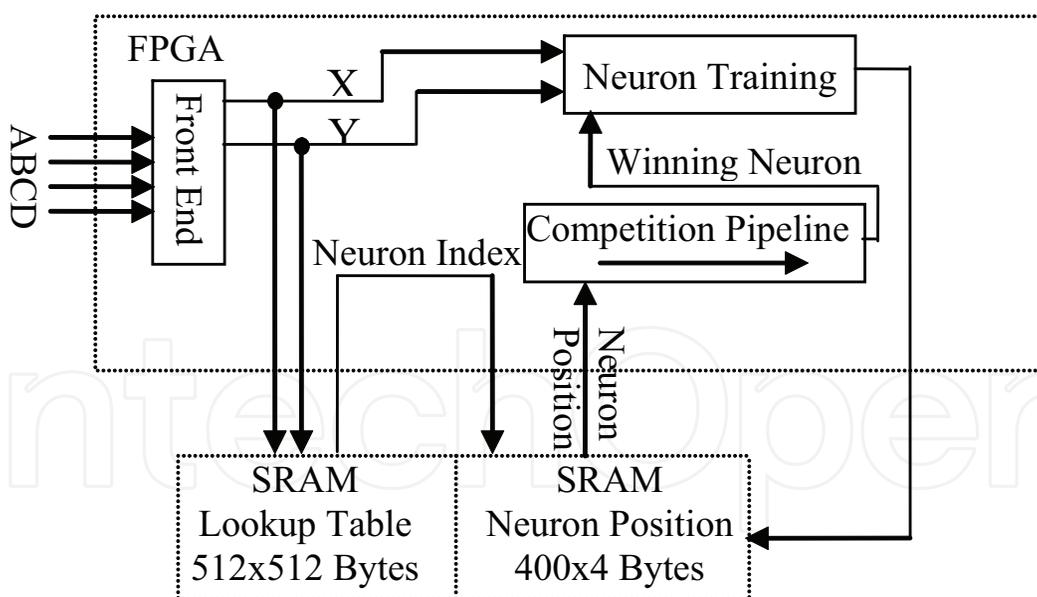


Fig. 4. SOFM FPGA implementation schematic

When an event triggers, the digitized ABCD values are sampled in the front end, and the (X, Y) position of the event is calculated by Anger logic. The (X, Y) value is used as an address to retrieve the region number from the lookup table in the SRAM. The positions of all the neurons in the region are read from the SRAM and sent to the competition pipeline sequentially. The FPGA contains multiple 18 bit x 18 bit multiplier blocks. Due to the pipeline implementation, only one multiplier is needed. The winner of the competition is the

neuron to be updated by the (X, Y) value of the incoming event using the Kohonen rule. Each time an event happens, only one neuron is updated. The updated neuron is then stored back to the SRAM. And the training logic is ready for the next event. The neurons' positions are uploaded from the SRAM after the training is completed. A CLT is built based on the peak positions represented by the neurons' positions.

During training, a 511keV weak source is placed in front of the detector. In 3 minutes, the SOFM is trained by more than 4.5 million events. The FPGA training logic is running at a clock with a period of 16ns. A training cycle only consumes 688ns. If a hot source is used, the training can be accomplished in seconds. In this training, all neurons find their corresponding pixel location.

5. Real-time crystal peak tracking (Hu et al., 2007)

Using the FPGA implementation, the neuron positions can be always trained or updated in the acquisition. Once the neurons are trained to the positions of clustered events, which indicate the positions of crystal peaks, they are trapped around these positions in the further training. When the positions of clustered events move slowly due to the drifting of gains, the neurons are capable of tracking the position changes of crystal peaks in real-time.

The feasibility of real-time crystal peak tracking by SOFM is verified through simulations on an APD detector, the SOFM is first trained to get the correct neuron positions. Then the bottom-left corner of the position profile in Fig. 5 (a) is pushed to the right slowly to simulate APD gain drifting. Fig. 5(b) shows that the position profile of the detector is changed because of the gain drifting. If the original crystal lookup table is used for crystal decoding, wrong crystal id is generated and the resolution of the scanner is degraded. In the simulation, the neurons are trained simultaneously while the gain is drifting. Fig. 5(b) shows the drifted position profile, and the responses of the neurons. It can be seen that the neurons are capable of tracking the position changes of the crystal peaks in real-time. A dynamic crystal lookup table can be generated based on the updated neuron positions. It brings a good solution to eliminate crystal decoding errors due to the APD gain drifting caused by temperature variations.

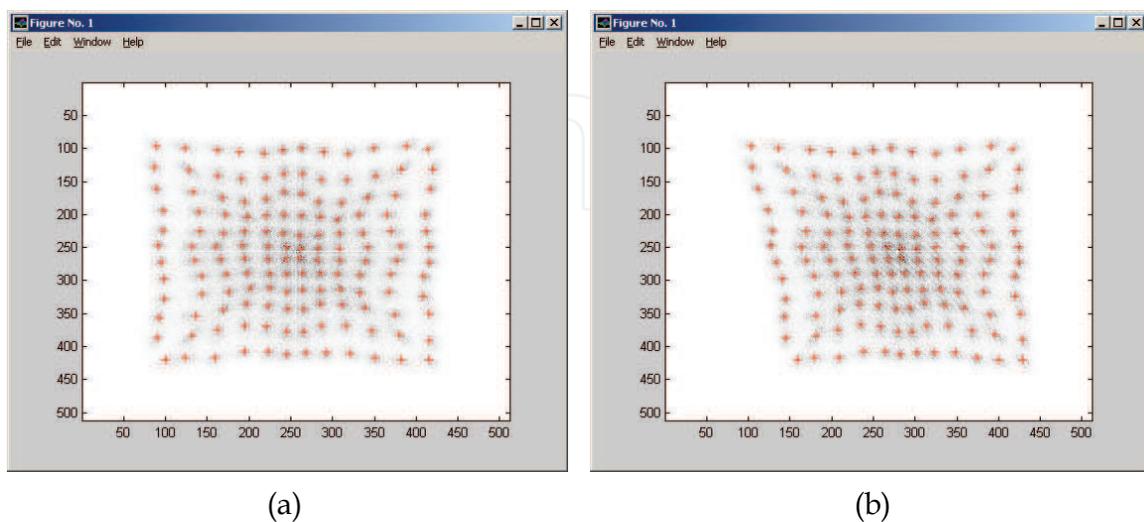


Fig. 5. Matlab simulation on SOFM training. (a) SOFM training results for an APD detector. (b) SOFM neuron response to APD gain drifting

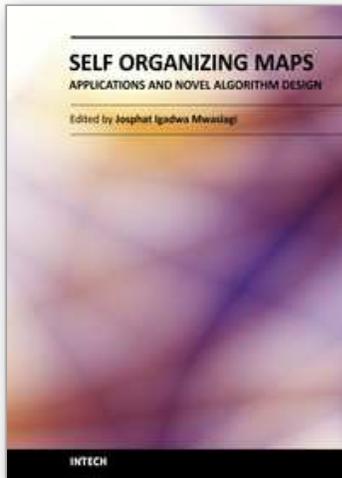
6. Conclusions

The SOFM based algorithm for CLT building yields higher pixel identification accuracy than the currently used method. The algorithm is compact and simple, and can be easily implemented in the FPGA chip present in the Inveon EPM electronics. The required FPGA resources do not increase with the number of pixels in the detector. The successful FPGA implementation makes this algorithm practical in training time. By applying FPGA implementation, the SOFM is capable of tracking slow position changes of crystal peaks in real-time, which brings a good solution to eliminate crystal identification errors caused by gain drifting, especially the gain changes of APD detectors due to the temperature variations.

7. References

- Hagan, M.T.; Demuth, H. B. & Beale M. (1996). *Neural Network Design*, PWS Publishing Company, ISBN 0-5349-4332-2, Boston, USA.
- Hu, D.; Atkins, B. & Lenox, M. (2006). A Neural Network Based Algorithm for Building Crystal Lookup Table of PET Block Detector, *2006 IEEE Nuclear Science Symposium Conference Record*, pp. 2458-2461, ISBN 1-4244-0561-0, San Diego CA, USA, Oct. 2006, IEEE, San Diego
- Hu, D. & Gremillion, T. (2007). Verification of Neural Network Based Algorithm for Crystal Identification of PET Block Detector, *2007 IEEE Nuclear Science Symposium Conference Record*, pp. 2847-2850, ISBN 1-4244-0923-3, Honolulu HI, WSA, Oct. 2007, IEEE, Honolulu

IntechOpen



Self Organizing Maps - Applications and Novel Algorithm Design

Edited by Dr Josphat Igadwa Mwasiagi

ISBN 978-953-307-546-4

Hard cover, 702 pages

Publisher InTech

Published online 21, January, 2011

Published in print edition January, 2011

Kohonen Self Organizing Maps (SOM) has found application in practical all fields, especially those which tend to handle high dimensional data. SOM can be used for the clustering of genes in the medical field, the study of multi-media and web based contents and in the transportation industry, just to name a few. Apart from the aforementioned areas this book also covers the study of complex data found in meteorological and remotely sensed images acquired using satellite sensing. Data management and envelopment analysis has also been covered. The application of SOM in mechanical and manufacturing engineering forms another important area of this book. The final section of this book, addresses the design and application of novel variants of SOM algorithms.

How to reference

In order to correctly reference this scholarly work, feel free to copy and paste the following:

Dongming Hu (2011). Applying SOFM and Its FPGA Implementation on Event Processing of PET Block Detector, Self Organizing Maps - Applications and Novel Algorithm Design, Dr Josphat Igadwa Mwasiagi (Ed.), ISBN: 978-953-307-546-4, InTech, Available from: <http://www.intechopen.com/books/self-organizing-maps-applications-and-novel-algorithm-design/applying-sofm-and-its-fpga-implementation-on-event-processing-of-pet-block-detector>

INTECH
open science | open minds

InTech Europe

University Campus STeP Ri
Slavka Krautzeka 83/A
51000 Rijeka, Croatia
Phone: +385 (51) 770 447
Fax: +385 (51) 686 166
www.intechopen.com

InTech China

Unit 405, Office Block, Hotel Equatorial Shanghai
No.65, Yan An Road (West), Shanghai, 200040, China
中国上海市延安西路65号上海国际贵都大饭店办公楼405单元
Phone: +86-21-62489820
Fax: +86-21-62489821

© 2011 The Author(s). Licensee IntechOpen. This chapter is distributed under the terms of the [Creative Commons Attribution-NonCommercial-ShareAlike-3.0 License](#), which permits use, distribution and reproduction for non-commercial purposes, provided the original is properly cited and derivative works building on this content are distributed under the same license.

IntechOpen

IntechOpen