Morphological Classification of Particles Recorded by the Timepix Detector

Jan Bartovsky, Dominik Schneider, Eva Dokladalova, Petr Dokládal, Vjaceslav Georgiev, Mohamed Akil

Morphological Classification of Particles Recorded by the Timepix Detector

J. Bartovsky*‡, D. Schneider*, E. Dokladalova*, P. Dokladal†, V. Georgiev‡, and M. Akil*
* Laboratoire Informatique Gaspard Monge, CNRS-UMLV-ESIEE (UMR 8049), Paris-Est, France
emails: {j.bartovsky, d.schneider, e.dokladalova, m.akil}@esiee.fr
† Center of Mathematical Morphology (CMM), Mines ParisTech, 77305 Fontainebleau, France
email: petr.dokladal@mines-paristech.fr
‡ Faculty of Electrical Engineering, University of West Bohemia in Pilsen, Czech Republic
email: georg@kae.zcu.cz

***Abstract*—The Timepix is a pixel detector that records energy deposited by charged particles. Different particles leave a different trace. These traces can be analysed in order to identify the particles, and consequently, analyze the sources of the radiation.**

We propose an image processing approach to the classification of particles based on the shape of traces, using only a few basic morphological operations.

This method - implemented in an FPGA - achieves a performance and latency allowing a high acquisition rate. Embedded with Timepix, it can beneficially analyse radioactive fluxes of unknown sources and spectra.

I. Introduction

The Timepix device is a new generation of CMOS pixel detectors – derived from the Medipix2 detector and designed by CERN – allowing three types of measurements: i) "time of arrival" (ToA), ii) "time over threshold" (ToT) and iii) and counting [1]. In counting mode, the Timepix pixels increment the counter each time the deposited energy exceeds the threshold. In ToA mode the pixel records the timestamp of the hit. In ToT mode, the pixels value indicates the total deposited energy. Three operating modes supported by one chip enable a large scale of applications; from astronomical observations, X-ray fluorescence imaging to event reconstruction in physical numerous experiments (i.e., analyse radioactive fluxes from unknown radioactive sources). The device allows a high sampling rate; it can record up to 3000 frames per second (fps).

Behind the CMOS sensor, there is a threshold allowing to suppress the noise, retaining only the energy of the particles [2]. This eases the image processing, since those pixels that do not receive a sufficient energy deposit (above threshold) from a particle contain a zero value. Consequently, the obtained images are considered as noise-free.

The principle of the event reconstruction is based on the charged particle recognition and its class identification (e.g., α, γ, electrons, etc.), followed by some statistical measurements (i.e., major particles and their angle of incidence) [3]. The particles recognition requires to identify and analyse the *trace*, representing the particle's "signature", left by the particle whenever it strikes the Timepix detector. The different particles leave differently shaped traces in dependence on the type of the particle, its energy and incidence angle. Consequently, the shape and the energy deposited alongside every track can be used for identification of the particle [3].

In this paper, we concentrate on the application of the Mathematical Morphology (MM) tools to the particle classification. The MM has been introduced in late sixties and the particle classification is a standard issue addressed by the MM from its beginning [4], [5]. In addition, the attention paid to real-time implementations [6], [7] results in efficient and modular solutions on various kind of hardware platforms, such as CPU, GPU or FPGA.

The organization of the paper is as follows: Section II introduces a notion of classification and Section III outlines main principles of our method. The proposed hardware implementation is detailed in Section VI.

II. Classification Using Morphological Characteristics

The Timepix device records a sequence of grey-valued images I: $D \times t \rightarrow V$. The support $D \subset \mathbb{Z}^2$ is a rectangular 256×256 raster. The images are scalar-valued with the set of values V coded in 14 bits, with positive integer values from [0, 16383].

In the following, one *cluster* denotes a connected component of non-zero pixels. One cluster corresponds to the trace left by one particle (or more particles, if they overlap). In this work, though, we suppose that one cluster is left by only one particle.

The set of all traces observed at time t is defined as $CC\{(x, y) \mid I(x, y, t) > 0\}$, where CC denotes the connected components in a set, obtained with the 8-connectivity.

Each connected component can be associated with descriptors allowing to classify the particles into different classes. The most frequently used are morphological descriptors (based on the shape analysis), i.e. the area, the projected and the unrolled length, the skeleton, the geodesic diameter, the circularity, the tortuosity, etc. see [8].

The descriptor-based classification methods are very efficient; their drawback though is the computing complexity. They require the computation of connected components, labeling, skeletonization and reconstruction, even before the descriptors can be computed. Even if optimized [9], the

skeletonization and reconstruction are iterative, with data-dependent computational intensity. Such properties infer high memory requirements, undefined latency, and slow computation inapplicable in high-frame-rate applications.

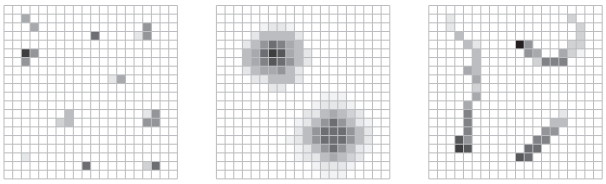
It is clear that the efficiency of the image processing bounds the sampling frequency of the image acquisition. At the same time, a high sampling frequency, limits the probability of overlapping traces.

This work presents a fast classification method based only on two descriptors, the thickness and the projected length. It uses only the morphological dilation and erosion, and simple arithmetic operators, and avoids all iterative, costly algorithms.

Implemented in a dedicated hardware, it takes advantage of recently proposed fast implementation of dilation and erosion. Thanks to the pipelining, the input image is classified in a single image scan at very high frame rate, reaching 738 fps. Furthermore, the proposed architecture is extensible in terms of the number of trace types to classify.

III. Method description

In this study, we consider three main classes of traces called blobs, dots and tracks, see Fig. 1. These names correspond to the nuclear physics terminology used in [3] and [2]. The dots are generated by, e.g., low-energy electrons or photons. The blobs are left by α or heavy ions. And the linear or curly tracks are produced by minimum ionizing particles or electrons.



(a) dots (b) blobs (c) tracks

Fig. 1. Examples of traces deposited by different particles.

A. Mathematical background

At first, let us to introduce the basic MM operators [5]. Let $f : \mathbb{Z}^2 \to \mathbb{R}$ denote a digital image. Then $\delta_B(f)$ denotes a dilation of f by a structuring element B, whereas ε_B denotes the erosion of f by B. In this paper, the structuring element $B \subset \mathbb{Z}^2$ is assumed rectangular and translation-invariant. The dilation and the erosion operations are defined by

$$\delta_B(f) = \bigvee_{b \in B} f_b \qquad \text{and} \qquad \varepsilon_B(f) = \bigwedge_{b \in \hat{B}} f_b \tag{1}$$

where f_b denotes the translation of f by some vector b. The $\hat{B}$ is the transposition of the structuring element (SE), equal to the set of reflection $\hat{B} = \{x | -x \in B\}$.

The concatenation of erosion and dilation forms a morphological opening

$$\gamma_B(f) = \delta_B \varepsilon_B(f) \tag{2}$$

B. Residual approach for particle classification

The particle extraction based on the shape features and measurements is a classical problem of Mathematical Morphology [10], [11].

Consider a family of shapes Ξ and an image I^{Ξ} containing objects from Ξ. The shape $\xi_i \in \Xi$ can be extracted from I^{Ξ} by opening γ_{ξ_i}

$$I^{\Xi'} = \gamma_{\xi_i} I^{\Xi} \tag{3}$$

where $\Xi' = \Xi \setminus \{\xi_i\}$, and $I^{\Xi'} = I^{\Xi} - I_i^{\xi}$.

This type of opening is commonly considered as algebraic opening. If Ξ is ordered, the shapes $\{\xi_i\}$ can be extracted one by one. This approach proceeds in a few steps each of which recognizes and extracts one type of particles retaining the other particles intact in the residual image. The following step extracts another particle and so on.

The algebraic opening γ_{ξ_i} from Eq. 3 can be constructed by morphological opening by reconstruction using the following steps:

1) Marker selection. It selects particles according to some criterion. A marker image $m : \mathbb{Z}^2 \to \mathbb{R}$ is commonly an image containing non-zero values intersecting the marked objects, and zero elsewhere. In the following, m^{ξ_i} will be used to mark objects of the shape ξ_i.
2) Object reconstruction (e.g. [12], [13]), recovers from the marker m the original values and shape from f. It is based, in general, on the geodesic dilation of m under f, $m < f$,

$$\delta^f(m) = \delta(m) \wedge f \tag{4}$$

hence from, by iteration

$$(\delta^f)^n(.) = \delta^f[(\delta^f)^{n-1}(.)] \tag{5}$$

we obtain the reconstruction

$$\mathcal{R}^f(m) = \lim_{n \to \infty} (\delta^f)^n(m) \tag{6}$$

Here we have a family of shapes $\Xi = \{\alpha, \gamma, \epsilon\}$. The process of separation based on a cascade of openings is a binary, decision tree classifier, see Fig. 2. First, we extract the thick dots (the α particles), second, the thin tracks (the ε particles). Finally, the last residual image will contain the dots (γ particles) only.

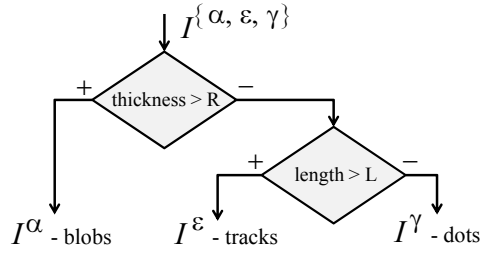


Fig. 2. Flowchart of residual approach. $I^{\{\alpha,\epsilon,\gamma\}}$ denotes the input image, and I^{α}, I^{ϵ} and I^{γ} the result images.

The reconstruction is an iterative process based on the geodesic dilation Eq. 4 with unitary geodesic ball as structuring element (omitted for brevity). We will show that these

stages can be approximated by a concatenation of basic morphological operators, erosion/dilation, and simple arithmetical operations. It can be computed only in one scan of the input image and "on the fly" without intermediate memory.

In the following section we describe in detail the proposed filter-based method of the morphological classification of particle traces.

IV. Implementation

As indicated above, the extraction of a shape ξ_i is done by algebraic opening γ_{ξ_i}, constructed by morphological opening by reconstruction.

Recall that the reconstruction is iterative process, iterated until idempotence. Given the restricted, and known family of shapes, we can approximate the reconstruction by only one geodesic dilation.

Hence, the stages consist of the following steps.

1) extraction of marker of the shape ξ by a morphological opening
$$m^{\xi} = \gamma_{B^{\xi}} I \tag{7}$$
2) geodesic dilation (approximating the reconstruction) of the marker under the image I
$$m' = \delta_B^I(m^{\xi}) \tag{8}$$
3) extraction of the image I^{ξ} containing the ξ-shaped objects
$$I^{\xi} = \begin{cases} I & \text{if } m' > 0 \\ 0 & \text{elsewhere} \end{cases} \tag{9}$$

Based on this scheme, Eqs. 7-9, the particle classification is done in the following order. (Refer to Tab. I for parameters of the structuring elements. The reconstruction step uses alike structuring element B for both shapes α and ε.)

1) *blobs* - First we obtain I^{α} from the initial image $I^{\{\alpha,\varepsilon,\gamma\}}$. The residual image is $I^{\{\varepsilon,\gamma\}} = I^{\{\alpha,\varepsilon,\gamma\}} - I^{\alpha}$.

2) *tracks* - Second, we obtain I^{ε}. The usual morphological approach to detect curvilinear objects is to use the supremum of openings $\gamma_{B^{\varphi}}$ by a rotating linear segment B^{φ}, oriented in φ. It is well known that a supremum of openings is itself an opening.
$$\gamma = \bigvee_{\varphi \in \Phi} \gamma_{B_{\varphi}} \tag{10}$$
The tracks are thin, curvilinear, oriented in arbitrary angle. This reguires a fine angular sampling of Φ, resulting in a high computational cost.
Here, to limit the number of discrete angles $\varphi \in \Phi$, we thicken the tracks by a dilation perpendicular to the opening. This allows to obtain satisfactory results with only two discrete angles, horizontal and vertical $\Phi = \{H, V\}$. Hence, using Eq. 10 for $\gamma_{B^{\xi}}$ in 7, with $\xi = \varepsilon$, we obtain
$$m^{\varepsilon} = \bigvee_{\varphi = H,V} \gamma_{B^{\varepsilon}_{\varphi}} \delta_{B'_{\varphi}} I^{\{\varepsilon,\gamma\}} \tag{11}$$
where $B_V = \mathrm{rot}(B_H)$, the copy of H rotated by 90^o, for both B^{ε} and B'.

3) *dots* - Finally, the residuum image $I^{\gamma} = I^{\{\varepsilon,\gamma\}} - I^{\varepsilon}$ contains the dots, i.e. the gamma particles.

TABLE I
Structuring element parameters.

Class	Blobs α	Tracks ε	
Marker selection	B^{α} [4,4]	B^{ε}_H [1,3]	B'_H [2,1]
Approximation of reconstruction	B [3,3]		

[H,W] denote the height and the width of a rectangular structuring element.

V. Experimental Results

To evaluate the proposed classification method, we have performed a statistical measuring of the traces' dimensions on randomly selected images of the Timepix database. The results suggests that the diameter of blob traces is at least 4 px. Therefore, the thickness criterion R equal to 4 (accords to B_{α}=$[4, 4]$) identifies the blobs.

On the other hand, the dot traces fit inside 2×2 bounding box. So the length parameter L equal to 3 (see B_H=$[1, 3]$) separates tracks from dots. The approximated reconstruction uses in both cases the SE of the marker's erosion plus one pixel in all directions. Such SE has the minimal surface necessary for the proper recovery of the original shape (cf. Table I).

The computed confusion matrix, see Table II, allows to appreciate the performance. The resulting errors are mainly due to: i) the border effects: particles touching the image border are sometimes misclassified, ii) the limit cases: the proposed method only approximates (with rectangular SE) the measurements of the particle trace thickness and projected length. The result misclassification of the method is below 7% of particles (each type of particles considered separately).

TABLE II
Confusion matrix computed for 100 images randomly selected from the database.

Input class	Blobs	Tracks	Dots
Number of particles	418	4627	12906
431 blobs classified as	418	13	0
4920 tracks classified as	0	4614	306
12600 dots classified as	0	0	12600

Extensions: Notice that the proposed method can be used to further analyse the three main classes by splitting them into sub-classes. The sub-classes definition is defined by the purpose of the particular physical measurement. We can illustrate this idea on the example of sorting the blobs with respect to their thickness. It requires to apply several consecutive blobs classification procedure with varying R.

Another example could be rough sorting of the track impact angles. The principle is to refine the angular sampling of Φ in Eq. 10.

VI. Hardware Architecture

We present the hardware implementation of the proposed particle classification in this section.

The overall architecture is displayed in Fig. 3. It consists of a several Recognition Units (RU), Control Unit, and optional Visualization Memory if results are to be displayed. The classification of a given type of particles is carried out in one RU block. The RU performs three tasks as described in the previous section: (i) the marker creation, (ii) the reconstruction, and (iii) the residual image. The RU outputs two images, I^{ξ} containing classified particles, and the residual image $I^{\Xi'}$ containing other particles.

In applications that need more types of particles to be recognized multiple RUs are instantiated in a pipeline (Fig. 3). It allows us to classify all types of particles concurrently on time-shifted data. The residual image of an RU is taken as an input by the following RU.

The control unit provides both controls and programmable parameters for each RU through *ctrl_ru x* signal (which is a set of $\{$*ctrl_mmb i*, *ctrl_alb j*$\}$; $i \in \{1:5\}$; $j \in \{1:2\}$). The classified particles I^{ξ_i} of any RU can be either read by a further block (RU, output, image compression, etc.) or stored in the global visualization memory.

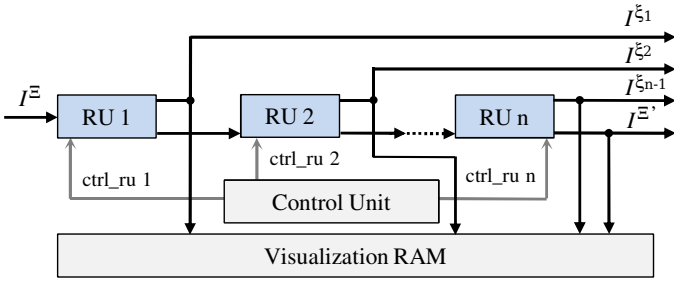


Fig. 3. Overview of the proposed hardware implementation.

A. Recognition Unit

The internal structure of the RU is shown in Fig. 4. First, the marker image m^{ξ} is to be created. It is made by processing the image according to 7. Both $\varphi = H$ and V in the supremum $\bigvee$ in (11) are independent and therefore separated in two parallel branches. Each branch computes one erosion and one dilation using two Mathematical Morphology Blocks (MMB1-4). The marker is completed from the parallel branches in Arithmetic Logic Block 1 (ALB1) that performs the $\bigvee$.

In the second step, the marker m^{ξ} is used in the approximated particle reconstruction. It consists of the marker dilation (MMB5) followed by threshold operation defined in eq. (9) with the input image. The result image I^{ξ} containing only the desired particles is obtained through comparison with the RU input image, see (9). Both previous operations are evaluated in ALB2.

Finally, the RU input image is split into two output images; I^{ξ} with classified particles, and the residual image $I^{\Xi'}$. This step is carried out in ALB2 as well. The FIFO memory connected between the input image and ALB2 must be sized properly to compensate the delay of the branch containing MMB$\{1{:}5\}$. For instance, let us consider that MMB$\{1{:}5\}$ infer total delay of 5 image lines due to δ, ε intrinsic latency. The intrinsic latency is inavoidable and defined by dimensions of B. Hence, the FIFO must be capable of storing at least 5 image lines as well.

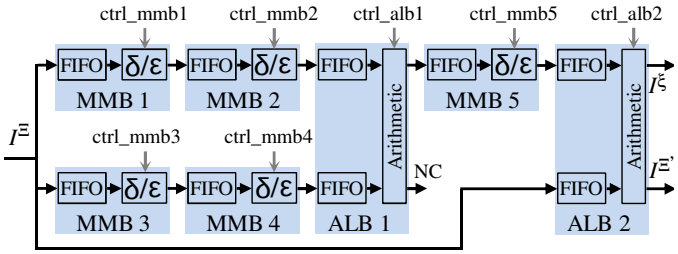
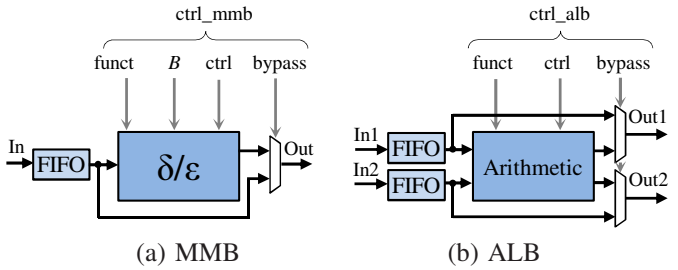


Fig. 4. Internal structure of the RU. I^{ξ} contains classified particles, $I^{\Xi'}$ is the residual image.

B. Morphological and Arithmetic Blocks

Both MMB and ALB are the basic elements of each RU. They have a few common properties. Both units are designed to work with the streamed data. Because the short-term processing rate of MMB may vary with data [7], each block integrates a front-end FIFO to balance currently different processing rates.

The units are fully run-time programmable, i.e., the parameters modifying the behavior can be changed between two frames without any overhead, as well as they can be entirely bypassed. The *bypass* input parameter is commonly used by marker m^{ξ} which needs only one MMB. When selected, a unit appears as a FIFO memory only. The *ctrl* contains the start and reset signals.



(a) MMB (b) ALB

Fig. 5. Morphological and Arithmetic blocks

The MMB performs either morphological dilation or erosion (it decides upon the *funct* select signal) on an input image by the structuring element B. The precedent version of the MMB was published at [7]. The *funct*, dimensions and origin position of B ($W \times H$, note definition of B in Section III), and *bypass* are programmable parameters.

The ALB is intended to perform several arithmetic operations. Besides the reconstruction and the residue process described previously, basic arithmetic operations as min(), max(), $<$, $>$, >0, <0, addition, or subtraction is selected via *funct* parameter.

TABLE III
TIMING RESULTS OF THE CLASSIFICATION.

Image type	Time [ms]	Latency [μs]	Rate [fps]
Best case	1.352	3.154	739
Worst case	1.678	39.4	594
Average Timepix	1.356	31.6	738

C. Application Demonstration

The application of particle traces classification into three types was implemented, see Fig. 6. It instantiates two RUs; the first I^{α} classifies blobs using the m^{α} marker image outputting dots and tracks in the residual image. The residual image of RU1 is read by the second RU2 that uses the marker m^{ε} to classify tracks I^{ε}. Hence, the residual image of RU2 $I^{\Xi'} = I^{\gamma}$ contains dots only. All three outputs are stored in on-chip Visualization Memory and displayed on a screen.

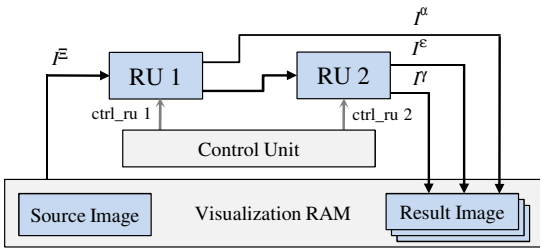


Fig. 6. Overview of application that classifies dot, blob, and track particle traces, see Section III.

D. Implementation Results

The proposed demonstration application has been targeted to Xilinx Virtex-5 FPGA (XC5VLX50T-1). Although the maximal synthesis frequency is about 160 MHz, the design is clocked by 100 MHz. The design (without the optional visualization) occupies the following hardware resources: 1405 registers, 4495 of 6-input look-up tables, and 9 36-kbit on-chip block RAMs.

The time benchmarks of the proposed design were performed on a set of Timepix images, each containing a mixture of all three kinds of particles. The results are outlined in Table III. All Timepix images were processed in almost the same time with minimal differences, so we use the average value. The worst case presents the lowest granted stream performance obtained on the most unpleasent gray-level image (artificial image containing monotonous gradient) whereas the best case conforms to the constant image. One can see that processing of Timepix image is very close to the best case since the Timepix image contains many zero-valued areas.

VII. CONCLUSIONS

In this paper, we presented the method of particle traces classification using the filter-based morphological markers instead of descriptors based on connected components, which are very computation intensive. The classification recognizes three main types of traces: dots, blobs, and tracks; and can be naturally extended.

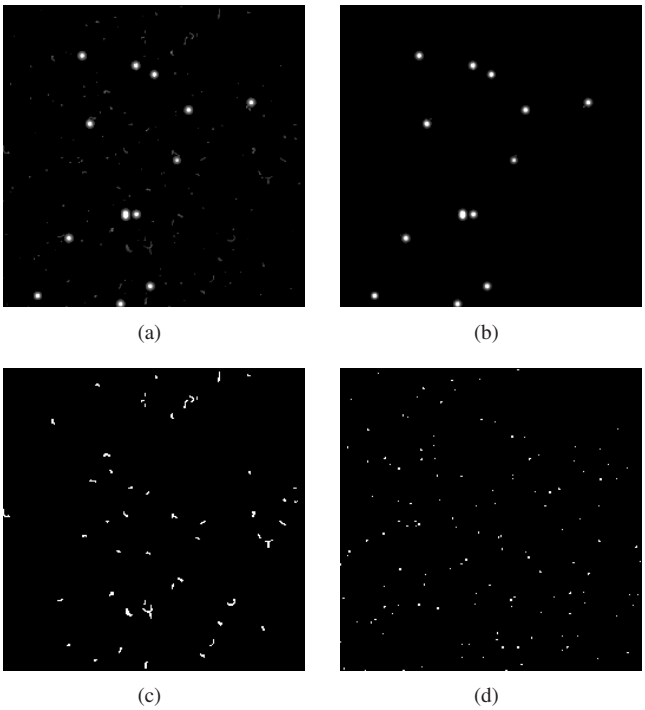


Fig. 7. Example of obtained results: a) experimental input image, b) classified as blobs, c) classified as tracks, d) classified as dots

We proposed the dedicated FPGA hardware architecture of the method. It processes the input image in a stream inferring the minimal latency. We achieved very high performance rate of 738 frames per second thanks the streaming pipeline structure. The high frame rate allows the Timepix detector to acquire images with a high sampling frequency, reducing thus the appearence of overlapping particles that can not be classified.

ACKNOWLEDGEMENT

The authors thank Ing. S. Pospíšil, DrSc. and Ing. J. Jakůbek, PhD. from IEAP[1], Czech Republic, for the initial discussion and the motivation of this work, and to Ing. Z. Vykydal for preparing the images. The authors are grateful for their permission to use the Timepix images (property of IEAP) in this publication.

REFERENCES

[1] X. Llopart, R. Ballabriga, M. Campbell, L. Tlustos, and W. Wong, "Timepix, a 65k programmable pixel readout chip for arrival time, energy and/or photon counting measurements," *Nucl. Instrum. Meth. A*, vol. 581, no. 1-2, pp. 485 – 494, 2007.

[2] Jan Jakubek, "Precise energy calibration of pixel detector working in time-over-threshold mode," *Nucl. Instrum. Meth. A*, vol. In Press, Corrected Proof, pp. –, 2010.

[3] J. Bouchami *et al.*, "Measurement of pattern recognition efficiency of tracks generated by ionizing radiation in a medipix2 device," *Nucl. Instrum. Meth. A*, vol. In Press, Corrected Proof, pp. –, 2010.

[4] J.C. Klein and J. Serra, "The texture analyser," *J. of Microscopy*, vol. 95, pp. 349–356, 1972.

[5] J. Serra, *Image Analysis and Mathematical Morphology*, Academic Press, Inc., Orlando, FL, USA, 1983.

[1] http://www.utef.cvut.cz

[6] Ch. Clienti, S. Beucher, and M. Bilodeau, "A system on chip dedicated to pipeline neighborhood processing for mathematical morphology," in *EUSIPCO 2008*, EURASIP, Ed., Lausanne, August 2008.

[7] J. Bartovsky, E. Dokladalova, P. Dokladal, and V. Georgiev, "Pipeline architecture for compound morphological operators," in *17th IEEE International Conference on Image Processing (ICIP)*, Sept. 2010, pp. 3765 –3768.

[8] P. Soille, *Morphological Image Analysis: Principles and Applications*, Springer-Verlag New York, Inc., Secaucus, NJ, USA, 2003.

[9] P. Matas *et al.*, "Parallel algorithm for concurrent computation of connected component tree," in *ACIVS 2008*, 2008, vol. LNCS 5259, pp. 230–241.

[10] M.F Devaux, P Robert, J.P Melcion, and F Le Deschault de Monredon, "Particle size analysis of bulk powders using mathematical morphology," *Powder Technology*, vol. 90, no. 2, pp. 141 – 147, 1997.

[11] J. Lee, M. Smith, L. Smith, and P. Midha, "A mathematical morphology approach to image based 3d particle shape analysis," *Machine Vision and Applications*, vol. 16, pp. 282–288, 2005, 10.1007/s00138-005-0181-x.

[12] Jean Serra and Luc Vincent, "An overview of morphological filtering," *Circuits Syst. Signal Process.*, vol. 11, no. 1, pp. 47–108, 1992.

[13] M. Schmitt and J. Mattioli, *Morphologie mathématique*, Masson: Paris, 1993.