**MVA '96** IAPR Workshop on Machine Vision Applications, November. 12-14, 1996, Tokyo, Japan

# A Focal Plane Array for Hot Punctual Target Identification and Tracking

Marius Vasiliu* and Francis Devos*
Fundamental Electronics Institute, Paris-Sud University


## Abstract

Hot punctual target identification and tracking must be usual tasks for many vision applications employed in defense systems, smart cars, or 3D scanning systems. Unfortunately, the signal-to-noise ratio is often low and the acquisition frame rate must be high. "Classical" vision systems cannot manage all specific constraints. We propose a new, adapted VLSI architecture for focal plane array, able to detect and to track numerous hot punctual targets at signal-to-noise ratios near to 0 dB. We validate the processing algorithm by performing a global simulation of the system and we find the best algorithm parameters with respect to qualitative criteria like target path detection rate and false detection probability.


## 1 Introduction

The real time identification and recognition of hot punctual targets, at high acquisition rate like 5000-10000 images per second and at signal-to-noise ratio close to 0 dB is a difficult task for any classical processing chain like CCD camera, numerical conversion followed by DSP processing [1]. The design of a compact system that could realize all these tasks inside one monolithic VLSI circuit has brought us to the study and to the realization of a focal plane array with a specialized architecture, called OPTICS (**"One Pixel Target Infrared Camera System"**).

OPTICS has to detect punctual targets on a slowly variable background (in time and space). Drawn to scale of the acquisition, targets are considered as attached to the background but the camera undergoes continually a complex motion as compared to the scene. Typically, the camera spins at 10 Hz on a slowly moving axis. Thus, a target never remains too long on the same pixel but its relative speed limits his motion to a jump between two neighbor pixels (in a 8-connected neighborhood).

The 2D implementation of a focal plane array generates "blind" gaps around each sensitive area. Our implementation needs about 25 % of each cell' surface for signal processing and communication. Thus, targets with a diameter lower than inter-pixel "blind" gap can became "invisible" for one frame (see the worst case in fig. 1 and the associated signal in fig. 2). When a small target crosses over a blind region, if we consider our typical parameters, the average signal attenuation is approximately equal to 5-15 dB.

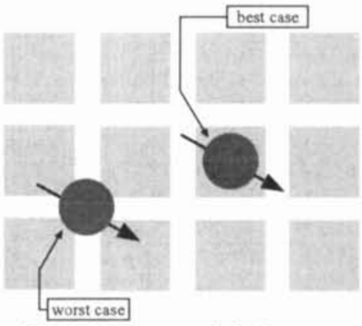



**Fig. 1.** The worst and the best case when a target cross a pixel

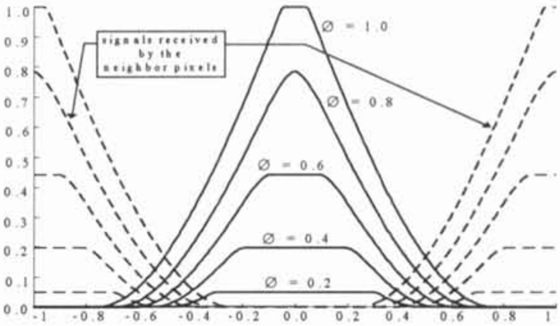



**Fig. 2.** Signal received by a pixel diagonally crossed by a target (relative sizes: pixel active area = 0.8x0.8, target diameter = Ø)

An other drawback due to technological spreads of components is the variable sensitivity between pixels. Sensitivity variations generate a noise type called Spatial Static Noise (SSN). We appreciate its value between 5 and 10 percents of the received signal peak. It is more important than temporal noise due to the quantum nature of photoconversion. Computer simulations shows us that for frame rates lower than 10000 Hz the temporal noise is negligible with respect of SSN.

*Address: Bat.220, Orsay 91405 France
E-mail: marius@ief-paris-sud.fr

## 2 Processing Algorithm

Processing algorithm must be easy to design, not very expensive in silicone surface (only 120 transistors per cell), fast enough (to process at least 5000 frames per second), and locally distributed (to minimize communication overloading). It has three levels: acquisition and SSN cleaning, complex filtering and binary decision. Our algorithm takes into account the unity of the spatio-temporal target path in the image flow. Acted treatments maximize the detection probability of a continuous path in the pixels space, even at low signal-to-noise ratios. At each frame, we will use the present and the last binary maps to "guess" the next target positions.

**SSN cleaning**:
In our VLSI implementation each sensor photocurrent depends linearly on received light intensity. Similar architectures are studied in [2] and [3].

$$I_C(E_{CELL}) = I_0(i,j) + \beta(i,j)E_{CELL}(i,j,t) \qquad (1)$$

where $I_0$ is the dark current,
$\beta$ is the light sensitivity of the sensor,
$E_{CELL}$ is the received light intensity.

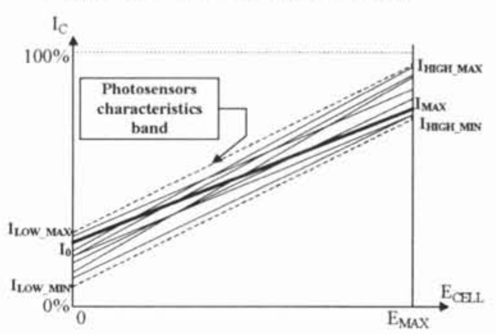



**Fig. 3.** Technological spread of photosensors characteristics

Practically, the SSN is generated by $I_0$ and $\beta$ variances between pixels (fig. 3). Using a linear correction for each photocurrent one can find the "correct" photocurrent $I_C^*$ if he can guess two correction parameters per pixel:

$$I_C^* = a(I_C - b) \qquad (2)$$

Static corrections like trimming methods [4] are unusable here because of temperature variations. A dynamic approach can use the smoothness of the scene and tries to minimize the global error energy $\varepsilon$ with respect to a and b:

$$\varepsilon = \sum_t \sum_{i,j} \sum_{l,k \in N(i,j)} [I_C^*(i,j,t) - I_C^*(l,k,t)]^2 \qquad (3)$$

where N(k,l) means the 8 direct neighbor pixels of pixel (i,j), in the 3-by-3 neighborhood.

Unfortunately, this may be too expensive. Local optimizations are much easier to implement. We studied a 3-by-3 pixels optimization and a single pixel optimization. For the first one we have:

$$\varepsilon_{ij} = \sum_t \left[ a_{ij}(I_C(i,j,t) - b_{ij}) - \frac{1}{8} \sum_{l,k \in N(i,j)} I_C^*(l,k,t) \right]^2 \qquad (4)$$

where the sum through time takes into account the last N frames (due to environmental conditions, N is approximately equal to 2000 frames).

Much simpler, the single pixel optimization suppose that statistically, each pixel has approximately the same minimal and mean photocurrent. This is more likely for neighbor pixels. As we will see later, the identification algorithm uses only the local information in a 3-by-3 neighborhood around each pixel.

$$a_{ij} = \frac{1}{\underset{t}{\mathrm{mean}}\{I_C(i,j,t)\} - b_{ij}} \qquad (5)$$

$$b_{ij} = \min_t \{I_C(i,j,t)\} = I_{MIN}(i,j) \qquad (6)$$

and so the normalized photocurrent:

$$I_C^* = \frac{I_C - I_{MIN}}{I_{MEAN} - I_{MIN}} \qquad (7)$$

Our plausible supposition was verified by simulation. An initial variance of $I_0$ and $\beta$ equal to 10% can be reduced by this way at only 2.5%. In order to keep a dynamical behavior of the SSN cleaning procedure, mean and minimum values are computed using recurrent formula:

$$I_{MEAN}(t) = \alpha_{MEAN} I_{MEAN}(t-1) + (1 - \alpha_{MEAN}) I_C(t) \qquad (8,9)$$

$$I_{MIN}(t) = \begin{cases} \alpha_+ I_{MIN}(t-1) + (1-\alpha_+) I_C(t) & \text{if } I_C(t) > I_{MIN} \\ \alpha_- I_{MIN}(t-1) + (1-\alpha_-) I_C(t) & \text{if } I_C(t) \le I_{MIN} \end{cases}$$

Due to the nature of the scene, we avoid using of local maximum values to normalize the photocurrents. When a hot punctual target crosses a pixel it modifies $I_{MAX}$ while $I_{MEAN}$ and $I_{MIN}$ are practically unchanged. This creates a strong asymmetry between the pixel and its neighbors, leading to eventual false detection.

**Complex filtering** estimates the global chances that a hot punctual target is projected on a given pixel (i,j). It includes the following components: **temporal filtering** to sharp pixel transition, **spatial filtering** to detect an isolated pixel hotter then its neighborhood and **path prediction** to take into account the spatio-temporal unity of target path on focal plane array.

**➔ Temporal filtering**: $S_T$
We studied first and second order temporal filtering, but the second order seems to be too expensive for our VLSI implementation (so $\alpha_2$ is actually equal to 0)

$$S_T(i,j,t) = I_C^*(i,j,t) - I_C^*(i,j,t-1) - \alpha_2 I_C^*(i,j,t-2) \qquad (10)$$

**➔ Spatial nonlinear filtering**: $S_S$
Easy to implement, the linear filtering can take a pointed corner of an hot object as punctual target:

$$S_S(i,j,t)=\frac{1}{8}\sum_{k,l\in N(i,j)}\left(I_C^{*}(i,j,t)-I_C^{*}(k,l,t)\right) \tag{11}$$

Hard nonlinear filtering is more adapted to punctual target. Unfortunately, it is affected by the SSN around the real target.

$$S_S(i,j,t)=I_C^{*}(i,j,t)-\max_{k,l\in N(i,j)}\left(I_C^{*}(k,l,t)\right) \tag{12}$$

A good compromise can be found by comparing the pixel value with the generalized mean of its neighbors:

$$S_S(i,j,t)=I_C^{*}(i,j,t)-\left(\frac{1}{8}\sum_{k,l\in N(i,j)}\left(I_C^{*}(k,l,t)\right)^{A}\right)^{\frac{1}{A}} \tag{13}$$

We found a good approximation of (13) as a combination of (11) and (12):

$$\begin{aligned}S_S(i,j,t)&=I_C^{*}(i,j,t)-\alpha_M\cdot\max_{k,l\in N(i,j)}\left(I_C^{*}(k,l,t)\right)\\&-(1-\alpha_M)\cdot\frac{1}{8}\sum_{k,l\in N(i,j)}I_C^{*}(k,l,t)\end{aligned} \tag{14}$$

➔ **Prediction map**: $S_{PM}(i, j, t)$

Initially equal to zero, this map holds an evaluation, for any pixel, of its chances to detect a target in the next frame. Typically, around a moving target, one must find positive values on pixels in front of the target and negative values behind the target. The anisotropy requested by this kind of prediction is expensive in silicon surface. We find an interesting decomposition of this prediction function in two isotropic functions (much easy to design): each binary detection of a target will add a 3-by-3 Gaussian mask **m** on the map and, symmetrically, a lost of target will subtract the same Gaussian mask from the prediction map. Thus, our algorithm needs a feed-back between the binary map and the prediction level. Figure 4 shows the result of a quick transition on the prediction map (lost of the target and new detection on the next pixel). To clean the map we introduce a "forgetting" factor $\alpha_F$, less but near to 1.

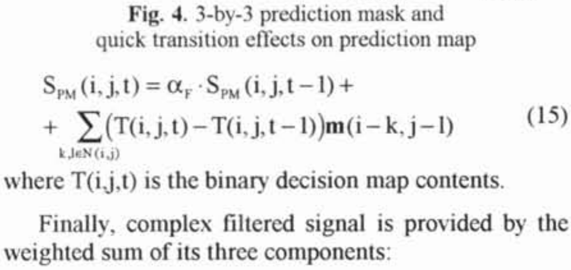



**Fig. 4.** 3-by-3 prediction mask and quick transition effects on prediction map

$$\begin{aligned}S_{PM}(i,j,t)&=\alpha_F\cdot S_{PM}(i,j,t-1)+\\&+\sum_{k,l\in N(i,j)}\left(T(i,j,t)-T(i,j,t-1)\right)\mathbf{m}(i-k,j-l)\end{aligned} \tag{15}$$

where $T(i,j,t)$ is the binary decision map contents.

Finally, complex filtered signal is provided by the weighted sum of its three components:

$$S_C(i,j,t) = \alpha_T S_T(i,j,t) + \alpha_S S_S(i,j,t) + S_{PM}(i,j,t) \tag{16}$$

**Binary decision:**

We used a Bayesian-like decision about the eventual target presence on each pixel. It must maximize the target path detection rate (TPDR) and minimize the false detection probability (see section 4 for a complete definition of this qualitative criteria). Practically, the target detection is made by a Winner-Takes-All competition between pixels in a 3-by-3 neighborhood, validated by a global threshold. Thus, the winning pixel filtered value must exceeds all his 8 direct neighbors and the global threshold $H_{WTA}$.

$$T(i,j,t)=\begin{cases}1 \ \textbf{if}\ S_C(i,j,t)>S_C(k,l,t)\ \ k,l\in N(i,j)\\ \quad\ \text{and}\ S_C(i,j,t)>H_{WTA}\\ 0\ \ \textbf{otherwise}\end{cases} \tag{17}$$

## 3 Focal Plan Array Architecture

The main part of signal processing architecture uses analog cells, due to circuit size and processing speed constraints. Proposed architecture (figure 5) includes three analog levels: a dynamic correction level to clean the SSN, a complex filtering level (temporal filtering, spatial nonlinear filtering and target path prediction) and a "Winner Take All" level that, using a dynamic threshold, takes a binary decision on the existence of the target in a local vicinity. Analog architecture uses less than 70 transistors per pixel. Digital maps follow the targets evolution and anticipate their paths by sensitizing pixels ahead of the detected target and by blinding pixels behind the target. Target detection and memorization, and I/O interface are realized with numerical operators.

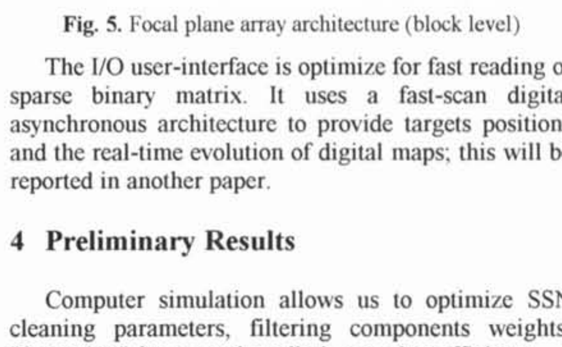



**Fig. 5.** Focal plane array architecture (block level)

The I/O user-interface is optimize for fast reading of sparse binary matrix. It uses a fast-scan digital asynchronous architecture to provide targets positions and the real-time evolution of digital maps; this will be reported in another paper.

## 4 Preliminary Results

Computer simulation allows us to optimize SSN cleaning parameters, filtering components weights, "forgetting" factor, and prediction mask coefficients.

To measure expected performance of the system (during simulation and validation steps) we have defined two qualitative criteria: the first concerns the target path detection rate (TPDR must be equal to 100 % for a perfect detection), and the second concerns the false detection probability (FDP) for any pixel in the focal plane array (to be detected as target in absence of an effective target). During simulation we compare output results with simulated target paths and we optimize internal parameters with respect to TPDR and FDP (fig. 6). Simulated with a typical scene parameters set, after optimization, OPTICS reaches a high rate of target path detection equal to 97% (fig. 7) while the false detection probability is less than $10^{-7}$ .

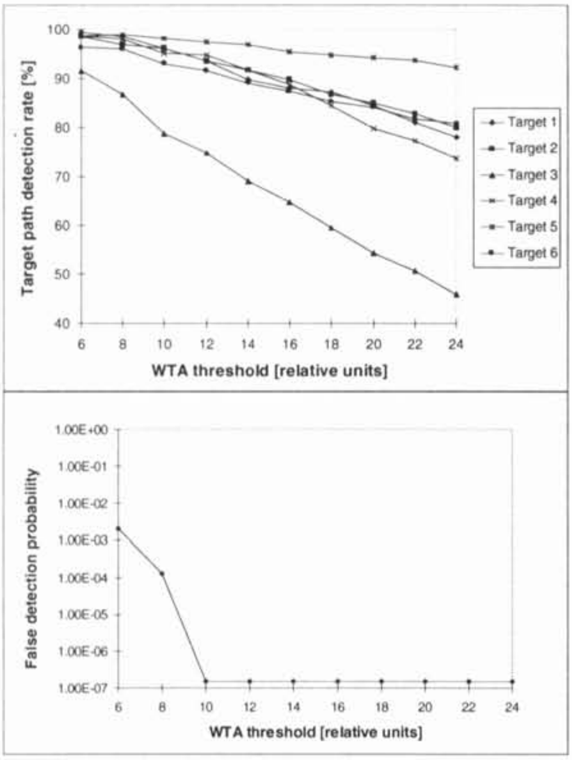



**Fig. 6.** TPDR and FDP towards WTA threshold

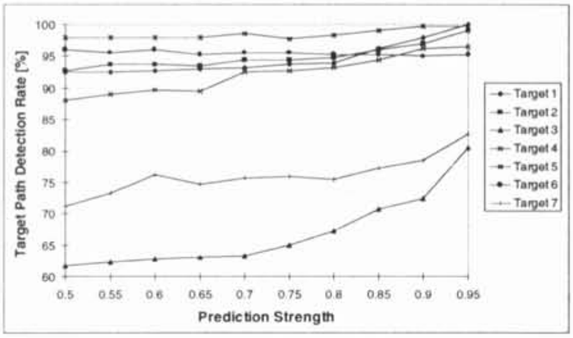
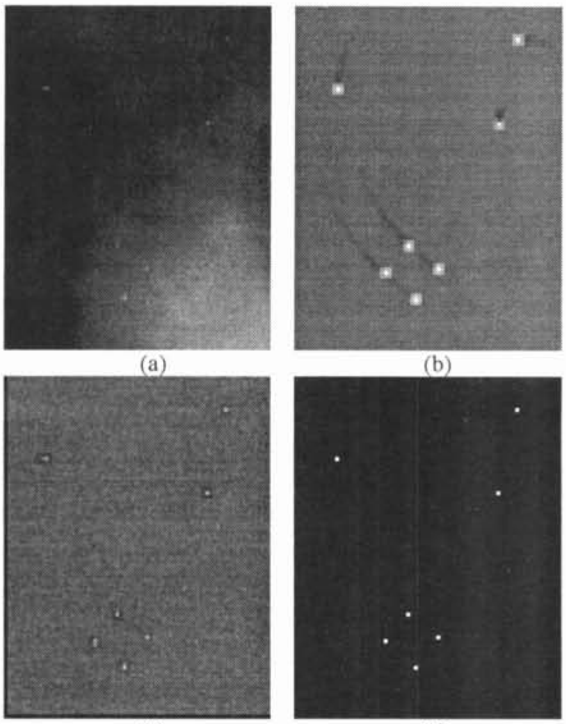



**Fig. 7.** Feed-back prediction increases target path detection rate

Optimized parameters like $\alpha_2$, $\alpha_F$, $\alpha_+$, $\alpha_-$, $\alpha_{MEAN}$, and **m** are frizzed in VLSI design while the others like $H_{WTA}$, $\alpha_S$ or $\alpha_T$ can be modified dynamically through I/O interface. Figure 8 shows the image processing chain while tracking seven hot punctual targets.



(c) (d)

**Fig. 8.** Image Processing Chain: Noisy Image (**a**), Fuzzy Prediction Map (**b**), Nonlinear Spatial Filtering Result (**c**) and Binary Detection Map (**d**)

## 5. Conclusions

This project uses a specific electronic approach to satisfy precise speed, quality, and reliability needs. The analog and digital operators mix is an objective necessity. Analog cells allow the increase of the local processing speed and the density of integration. This allows us to envisage a final architecture size like 128 by 128 pixels, using about one hundred transistors per cell.

## Acknowledgments

The authors would like to thank the CEA/DAM (*Commissariat à l'Energie Atomique*) for its support of this work through research contract U3-500-1156-EG.

## References

[1] Ni Y., Devos F., and Lavainne F. *An Analog Spot Position Coding Retina for Active Obstacle Detection*. International Journal of Robotics & Automation (IASTED) 8(2):77-83, 1993

[2] Deweerth S. and Mead C. *A Two-dimensional Visual Tracking Array* Proc '88 MIT Conf. on VLSI, 259-275; MIT Press Cambridge Mass

[3] Ni Y., Arion B., and Devos F. *CMOS compatible photoreceptor for high contrast car vision*, SPIE, International Symposium on Photonics for Industrial Applications, Boston, Nov. 1994

[4] Zhang M., Devos F., Ni Y., and Pone J. *Trimming smart imagers for an image converter using a non volatile analog memory* Sensors & Actuators A(46-47):456-459, 1995