# Grid based high performance computing in satellite imagery. Case study — Perona–Malik filter

## B. Belean[1], C. Belean[2], C. Floare[1], C. Varodi[1], A. Bot[1], Gh. Adam[3,4,a]

[1]National Institute for R&D of Isotopic and Molecular Technolgies, 67-103 Donat St.,
PO 5 Box 700, Cluj-Napoca, 400293, Romania
[2]Babes-Bolyai University, Faculty of Mathematics and Computer Science, Department of Mathematics,
1 M. Kogalniceanu St., Cluj-Napoca, 400084, Romania
[3]Laboratory of Information Technologies, Joint Institute for Nuclear Research, 6 Joliot Curie St., Dubna,
141980, Russia
[4]Horia Hulubei National Institute for Physics and Nuclear Engineering (IFIN-HH),
30 Reactorului St., Magurele - Bucharest, 077125, Romania

E-mail: [a]adamg@jinr.ru

The present paper discusses an approach to the efficient satellite image processing which involves two steps. The first step assumes the distribution of the steadily increasing volume of satellite collected data through a Grid infrastructure. The second step assumes the acceleration of the solution of the individual tasks related to image processing by implementing execution codes which make heavy use of spatial and temporal parallelism. An instance of such execution code is the image processing by means of the iterative Perona–Malik filter within FPGA application specific hardware architecture.

# Сеточные высокопроизводительные вычисления в получении спутниковых изображний на примере фильтра Перона–Малик

**Б. Белеан**[1], **К. Белеан**[2], **К. Флоаре**[1], **К. Вароди**[1], **А. Бот**[1], **Г. Адам**[3,4]

[1]*Национальный институт исследования и развития технологии молекулярных изотопов (INCDTIM), Румыния, 400293, Клуж-Напока, Донат ул. 67-103, ПО 5, ПЯ 700*
[2]*Университет им. Бабеша-Бойаи, Факультет математики и информатики, отделение математики, Румыния, 400084, Клуж-Напока, Михаил Когэлничану ул. 1*
[3]*Лаборатория информационных технологий, Объединенный институт ядерных исследований, Россия, 141980, г. Дубна, ул. Жолио-Кюри, д. 6*
[4]*Национальный научно-исследовательский институт физики и ядерной технологии им. Хории Хулубея (IFIN-HH), Румыния, 077125, г. Мэгуреле-Бухарест, Реакторулуй ул., д. 30*

В данной работе рассматривается подход к эффективной обработке спутниковых изображений, который включает в себя два этапа. Первый этап заключается в распределении быстро взрастающего объема спутниковых данных, полученных через Грид-инфраструктуру. Второй этап включает в себя ускорение решения отдельных задач, относящихся к обработке изображений с помощью внедрения кодов, которые способствуют интенсивному использованию пространственно-временного параллелизма. Примером такого кода является обработка изображений с помощью итерационного фильтра Перона–Малик в рамках специального применения архитектуры аппаратного обеспечения ППВМ (FPGA).

Ключевые слова: фильтр Перона–Малик, обработка спутникового изображение, Грид, высокопроизводительные вычисления, ППВМ (ПЛИС), ЮНИТАР (UNOSAT)

# 1. Essentials of satellite imagery

The scanning of the Earth surface by satellites is done within a wide wavelength range of the reflected electromagnetic radiation coming from the Sun (microwave, infrared, visible, ultraviolet). The recorded electromagnetic waves are used to create images according to the pattern shown in figure 1. The radiation emitted by an energy source (1) covers a distance and interacts with the atmosphere (2) before reaching the target (3). Radiation is reflected or scattered to the satellite sensor (4), which registers it and then transmits the information on the reflected radiation to a receiving station (5) where the received input is transformed into digital images. The characterization of the target properties or the solution of some specific problem is performed based on the acquired images.

Satellite images are of fundamental interest in meteorology, agriculture, biodiversity conservation, geology, forestry, landscape, regional planning, education, intelligence and warfare. Commercial applications concern: *Insurance companies* — damage estimates based on acquired images before and after disaster; *Mass Media* — satellite imagery based news reports; *Software developers* — image incorporation in flight simulators, games; *Combined with GPS* — localization in geographic information systems, e.g., Google Earth and Google Earth Pro.
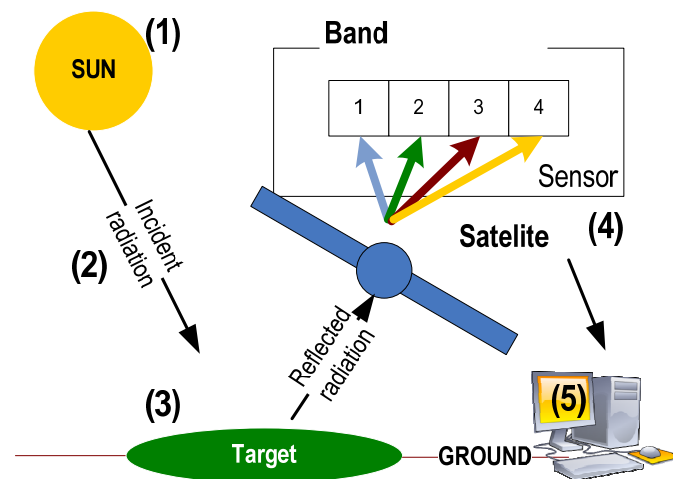


Fig. 1. Acquiring satellite images

**Resolution characterization of a satellite image**
Concerning the satellite images in remote sensing, there are four types of resolutions: spatial, spectral, temporal, and radiometric. Campbell [Campbell, 2002] defines them as follows. The ***spatial resolution*** represents the pixel size of an image of the surface area measured on the ground, determined by the sensor's instantaneous field of view. The ***spectral resolution*** is defined by the wavelength interval within the electromagnetic spectrum and the number of intervals measured by the sensor. (Examples: *visible images* come from sunlight reflected by the Earth surface; *infrared images* come from infrared sensor measurements of the temperature of the Earth surface; *water vapor images* come from infrared measurement of the temperature in an atmospheric layer about 6–10 km above the Earth surface.) The ***temporal resolution*** is defined by the time interval inbetween two subsequent image acquisitions for a given surface location. The ***radiometric resolution*** is defined by the amount of levels of brightness recorded by the imaging system.

Another resolution may be mentioned, namely, the ***geometric resolution*** which refers to the satellite sensor's ability to effectively image a portion of the Earth's surface in a single pixel and is typically expressed in Ground Sample Distance (GSD) units. GSD is a term containing the overall optical and systemic noise sources and is useful for comparing how well one sensor can "see" an object on the ground within a single pixel [Satellite imagery, 2014].

## 2. Grid based approach to satellite imagery

The increase of the number of high resolution satellites into orbit and the number of applications which use satellite images lead to "big data" to be processed, which cannot be  accommodated to the satellite's local computing infrastructures. The present paper focuses on high throughput computing strategies for processing satellite images which involve grid computing and application specific hardware architectures for high performance computing.

**UNOSAT** is the United Nations Institute for Training and Research (**UNITAR**) **O**perational **Sat**ellite Applications Programme. Created in 2000, it provides the worldwide users with high quality satellite imagery and Geographic Information System (GIS) services. These serve for planning sustainable development or monitoring natural disasters.

Excessive loads on the UNOSAT website, putting strong pressure on the computing and storage resources are frequently encountered. Instances:

*Use case 1*. During **natural catastrophes** and **disasters**.

*Use case 2*. **Web visualization from mobile devices** of the field workers (even though using compression and cropping when interrogating UNOSAT resources).

*Use case 3*. Need of storage and computing resources **from UNOSAT** by **users having slow internet connection.**

*Use case 4.* The **periodically performed updates** by the UNOSAT administrators (asked by image uploading in the databases; own search and processing tasks on the satellite image databases).

Since 2001, UNOSAT has been based at CERN and is supported by CERN's IT Department in the work it does. **CERN** **provides Grid approach** to the operations done on the satellite images: storage, processing, compression. The existing sequential codes for iterative solutions on multicore processors are, however, inefficient [Belean et al., 2013]. The usual approaches to speeding up the sequential solutions [Javier Gallego, 2005; Giuliani et al., 2011; Sarmah, Bhattacharyya, 2012] use **division** of the satellite images into **sub-images** in order to reduce the size to be processed. Each sub-image can be sent for processing to a different computing element (CE) within the grid. The most frequently encountered technique to divide the satellite images is based on the use of the **Thiessen polygons** [Javier Gallego, 2005].

The following limitations characterize the **sequential iterative algorithms:** **t**he computing power is limited to one CE while processing one sub-image; the CEs are General Purpose Processors (GPPs) (e.g. **Intel® Xeon® Processor E5**); the parallel processing strategies to be applied are limited by the available GPPs.

To secure smooth continuous functioning of the system, a vital condition is to produce **maximum relevant information** under **minimum system load**. The use of FPGA approach to the **Perona–Malik filter** offers such a tool.

## 3. Essentials of Perona–Malik filter for digital image processing

There are **two basic aims** of the Perona–Malik filter [Perona, Malik, 1987; Perona, Malik, 1990]: to eliminate disturbances, i.e., filter out noise; to retain and enhance the essential information.

The **basic idea** can be summarized as follows. The above aims can be achieved by solving a Neumann boundary problem for a generalized diffusion equation

$$\partial_t u(x,t) = \nabla(g(u,x,t)\nabla u(x,t)). \tag{1}$$

Where the diffusivity function $g$ is to be **anisotropic**, such that, along some directions $g \gg 1$ (strong diffusion) while along other directions $g \ll 1$ (weak smoothing).

The use of the equation (1) is motivated by the physical fact that, under $g \equiv 1$ Eq. (1) describes the **heat propagation.** The solution of the **heat equation** is a **convolution** of the initial value $u(x, 0) = f(x)$ with a **Gauss distribution function**. Since the latter is decreasing fast (both in the coordinate and the frequency spaces), the **fast oscillations are cut out**, hence (1) acts as an effective **low pass filter**.

The ***digital image processing*** based on the equation (1) is motivated by the following considerations. There is a ***bounded*** domain of the digital image, $\Omega \subset \mathbf{R}^n$ ($n = 2, 3$) of boundary $\partial\Omega$ of class $C^1$. The mapping $u: \Omega \rightarrow [0, 1]$ then achieves the correspondence from $\Omega$ to the ***gray level distribution*** (GLD) of a noisy image. The numerical investigation of the time evolution of the GLD, performed through an ***iterative approach***, results in ***successive instances*** attempting at solving the filtering tasks.

There are two contradictory features of the Gaussian smoothing associated to the heat propagation: ***efficient*** noise filtering, but ***edge smearing*** (image blurring) which results in ***quick loss*** of essential information contained in the original image.

The ***Perona–Malik hypothesis***: [Wielgus, 2010] the diffusivity coefficient in (1) is to be a function of the ***norm of the local gray level distribution gradient.*** This results into the diffusion Perona–Malik problem:

$$\partial_t u(x,t) = \nabla(g(|\nabla u|^2)\nabla u(x,t)) \text{ in } \Omega \times (0,+\infty),$$

$$\partial u / \partial n = 0 \text{ in } \partial\Omega \times (0,+\infty), \tag{2}$$

$$u(x,0) = f(x) \text{ in } \Omega.$$

***Basic features of the Perona–Malik problem***. It is an ***ill-posed*** problem, which ***does not*** admit a weak solution. A solution could, nonetheless, be defined in the sense of distributions. However, under ***spatial discretization by finite differences***, a ***well-conditioned problem*** arises, with an unexpectedly good filtering efficiency. This is usually called the ***Perona–Malik paradox*** in the digital image processing. An important quantity is ***the flux function***,

$$\Phi(s) = s \cdot g(s^2) > 0 \text{ for } s \in (0,+\infty),$$

which is asked to ***vary smoothly with*** $s$ and to have a maximum on $(0,+\infty)$ at some characteristic value $s_0 = \lambda > 0$.

The ***diffusivity function*** $g(s^2)$ enabling such $\Phi(s)$ should be infinitely continuous differentiable and to decrease monotonically from 1 to 0 while $s^2$ varies from 0 to $+\infty$.

An expression of $g(s^2)$ inspired by the Gaussian distribution function is

$$g(s^2) = \exp(-s^2 / 2\lambda^2), \lambda > 0. \tag{3}$$

Then $\Phi(s)$ has a maximum at $|s| = \lambda$, with $\Phi'(s) > 0$ for $|s| < \lambda$ and $\Phi'(s) < 0$ for $|s| < \lambda$.

An alternative choice of the diffusion function is

$$g(s^2) = 1/(1+s^2\lambda^2), \lambda > 0. \tag{4}$$

In the *two-dimensional case*, let $\xi$ and $\eta$ denote the local coordinates in directions perpendicular and parallel to $\nabla u$ respectively. Then the Perona–Malik equation can be rewritten as

$$\partial_t u = \nabla(g(|\nabla u|^2)u_{\xi\xi} + \Phi'(|\nabla u|)u_{\eta\eta}.$$

The coefficient of $u_{\xi\xi}$ is always positive, hence (2) acts as a smearing filter washing details along the contour lines of the function $u$. The coefficient of $u_{\eta\eta}$ may be both positive and negative, hence, in the perpendicular (gradient) direction slow gradient values are smeared out, while *large gradient values* (like *edges*) are sharpened instead of being blurred.

The experiments using Perona–Malik anisotropic diffusion were visually very impressive: edges remained stable over a very long time. It was demonstrated that edge detection based on this process clearly outperforms the linear Canny edge detector, even without applying non-maxima suppression and hysteresis threshold. This is due to the fact that diffusion and edge detection interact in one single process instead of being treated as two independent processes which are to be applied subsequently. Considering this advantage, this technique would improve edge detection in any image processing based applications.
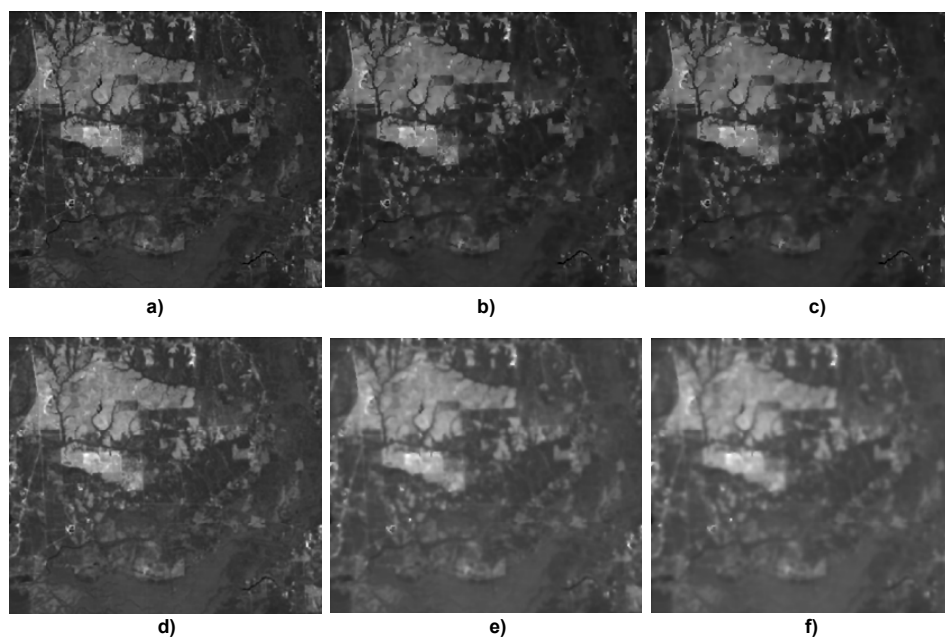
Fig. 2. Anisotropic diffusion showing edge enhancement in case of original image *Florida* [Florida…,2014], with choice (4) for diffusion function: a) *Num_iter* = 5, $\lambda$ = 10, b) *Num_iter* = 15, $\lambda$ = 10, c) *Num_iter* = 25, $\lambda$ = 10, d) *Num_iter* = 5, $\lambda$ = 30, e) *Num_iter* = 15, $\lambda$ = 30, f) *Num_iter* = 25, $\lambda$ = 30

Results of the conventional anisotropic diffusion (Perona & Malik) upon a gray scale image aiming edge enhancement are presented in Fig. 2. The number of iterations is denoted *Num_Iter*. The optimal parameter setup to preserve edge is found in figure 2.c, were all the details are preserved. Thus, it can be seen that for a number of 5 to 25 iterations (Fig. 2.a, 2.b and 2.c), the edge details are preserved. Meanwhile, for large values of $\lambda$, details are lost, see Fig. 2.d-f.

## 4. FPGA based hardware architectures for anisotropic diffusion

In order to propose hardware architecture for Perona–Malik filter, the computational steps of the anisotropic diffusions are independently presented next. Considering the iterative nature of the PDE filtering, the steps described previously are included in one filter iteration.

Given an initial image, to be processed within N iterations (empirically N = 10 to 20):

Steps 1, 2 and 3 shown in Fig. 3 are to be parallelized for efficient computation. Within the Field Programmable Gate Array (FPGA) architecture, both spatial parallelism (enabling the use of multiple computing units) and temporal parallelism (enabling the creation of a pipeline) were implemented. The main estimated advantages of the proposed architecture are as follows: within one iteration, each $p(i,j)$ resulted pixel value its delivered at each $3 \times T_{clk}$ cycles, with an initial delay of $T = 12$ $T_{clk}$ cycles. In the case of a general purpose processor, within iteration, for one output pixel there are necessary more than 30 $T_{clk}$ cycles. The exponential function can be implemented based on the approach presented in [Belean B. et al., 2012] for the log computation unit.

Computations done for a Virtex 5 FPGA and a general purpose processor are given in Fig. 4. A factor two acceleration of the computations, with good scalability, are obvious.

## Conclusions

- In future attempts to implement solutions for specific iterative algorithms, we intend to intensively use Field Programmable Gate Arrays

**for each** *p(i,j)*

    a ) **STEP 1** - compute finite differences $\eta_N$, $\eta_S$, $\eta_E$,, …$\eta_{NW}$ using

        the following computational masks

    hN = [0 1 0; 0 -1 0; 0 0 0]; hS = [0 0 0; 0 -1 0; 0 1 0];

      ………………………   hNW = [1 0 0; 0 -1 0; 0 0 0];

    *- equivalent with 8 additions*

    b) **STEP 2** -    compute diffusion function

$$c_N = e^{-(\eta_N/k)^2} \qquad c_S = e^{-(\eta_S/k)^2} \qquad ….. \qquad c_{NW} = e^{-(\eta_{NW}/k)^2}$$

    - *8 computations of the exponential function*

    c)    **STEP 3** - the p(*i,j*) pixel within the resulted image

$$p(i,j) = p(i,j) + c_N\eta_N + c_S\eta_S + c_E\eta_E + c_W\eta_W +$$
$$+ 0.5c_{NE}\eta_{NE} + 0.5c_{SE}\eta_{SE} + 0.5c_{NW}\eta_{NW} + 0.5c_{SW}\eta_{SW}$$

    - *8 multiplications*
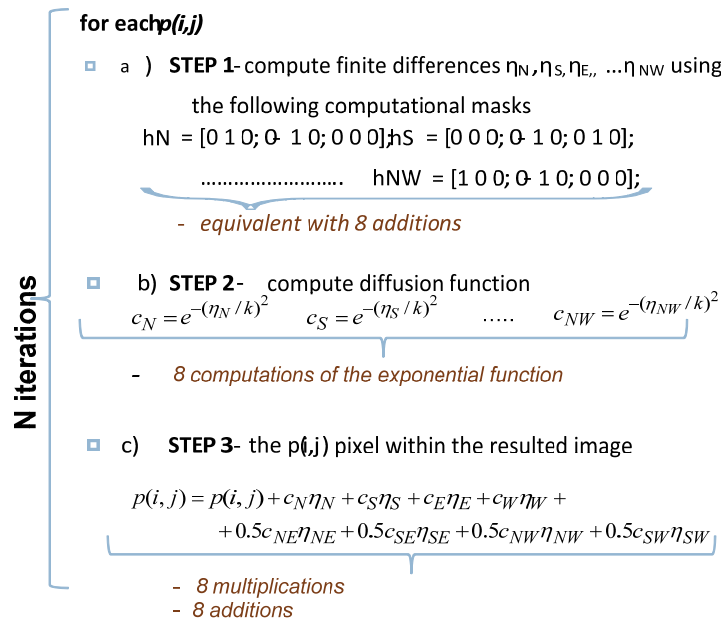    - *8 additions*

**N iterations**

Fig. 3. The three distinct computational steps identified within each of the iterations

- The reported results suggest that the general purpose processors are surpassed by Application Specific Hardware Architectures as it concerns the computing time
- In the next future, we plan to derive a GPU implementation of the Perona–Malik filter and to compare it with the FPGA implementation
- In the processing of grid Big Data sets, extensive use of Application Specific Hardware Architectures might offer efficient solutions.
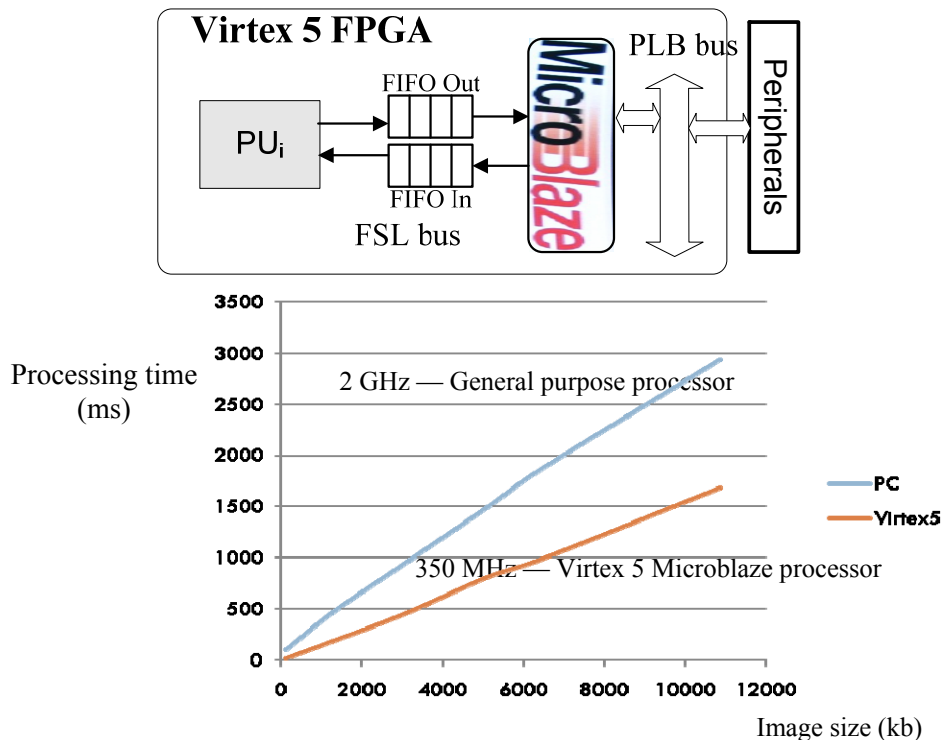
Fig. 4. Schematic presentation of the Virtex 5 FPGA (above) and comparison of the performance dependence of Virtex 5 FPGA with that of a 2 GHz general purpose processor

## References

*Belean B., Borda M., Bot A.* FPGA based hardware architectures for iterative algorithms implementations // 36th International Conference on Telecommunications and Signal Processing. Rome. — 2013.

*Belean B., Borda M., Le Gal B., Terebes R.* FPGA based system for automatic cDNA microarray image processing. // Computerized Medical Imaging and Graphics. —2012. — Vol. 36. — P. 419–429.

*Campbell J. B.* Introduction to Remote Sensing. New York London: The Guilford Press. — 2002.

Florida, Google Maps. — 2014. https://www.google.rp/maps/place/Florida

*Giuliani G., Ray N., Lehmann A.* Grid-enabled Spatial Data Infrastructure for environmental sciences: Challenges and opportunities // Future Generation Computer Syst. — 2011. —Vol. 27. — P. 292–303.

*Javier Gallego F.* Stratified sampling of satellite images with a systematic grid of points. // ISPRS Journal of Photogrammetry & Remote Sensing. — 2005. — Vol. 59. — P. 369–376.

*Perona P., Malik J.* Scale space and edge detection using anisotropic diffusion. // Proc. IEEE Computer Soc. Workshop on Computer Vision 16–22 (1987); Perona P., Malik J., Scale-Space and Edge Detection Using Anisotropic Diffusion, IEEE Transactions on Pattern Analysis and Machine Intelligence. — 1990. — Vol. 12. — P. 629–639.

*Sarmah S., Bhattacharyya D.K.* A grid-density based technique for finding clusters in satellite image // Pattern Recognition Letters. — 2012. — Vol. 33. — P. 589–604.

Satellite imagery. From Wikipedia, the free encyclopedia. — 2014.
http://en.wikipedia.org/wiki/Satellite_imagery

*Wielgus M.* // Perona–Malik equation and its numerical properties. Praca licencjacka na kierunku MATEMATYKA, Uniwersytet Warszawski, Lipiec. — 2010.