# ANISOTROPIC DIFFUSION USING POWER WATERSHEDS

*Camille Couprie*, Leo Grady†, Laurent Najman*, Hugues Talbot**

\* Université Paris-Est, Lab. Informatique Gaspard Monge, Equipe A3SI,
ESIEE Paris (93160 Noisy-le-Grand, France)
c.couprie@esiee.fr, l.najman@esiee.fr, h.talbot@esiee.fr

† Siemens Corporate Research,
(Princeton, N.J. 08540. USA)
leo.grady@siemens.com

## ABSTRACT

Many computer vision applications such as image filtering, segmentation and stereo-vision can be formulated as optimization problems.Whereas in previous decades continuous-domain, iterative procedures were common, recently discrete, convex, globally optimal methods such as graph cuts have received a lot of attention. However not all problems in computer vision are convex, for instance $L_0$ norm optimization such as seen in compressive sensing. Recently, a novel discrete framework encompassing many known segmentation methods was proposed : power watershed. We are interested to explore the possibilities of this minimizer to solve other problems than segmentation, in particular with respect to unusual norms optimization. In this article we reformulate the problem of anisotropic diffusion as an $L_0$ optimization problem, and we show that power watersheds are able to optimize this energy quickly and effectively. This study paves the way for using the power watershed as a useful general-purpose minimizer in many different computer vision contexts.

*Index Terms*— Combinatorial optimization, image processing, denoising, mathematical morphology, watersheds.

## 1. INTRODUCTION

The most common assumption about image noise is that the noise is high frequency. Therefore, a natural choice of image filter is a lowpass filter which dampens power in the high-frequency range. Unfortunately, a lowpass filter has the undesirable effect of *blurring* object boundaries, since these boundaries have a high-frequency. This dilemma was recognized early in the image processing literature and various approaches were proposed to solve it, with the goal of smoothing the image content *internal* to an object, but preserving image discontinuities across boundaries.

An approach to discontinuity-preserving filtering was proposed by Perona and Malik (PM) [1] who modeled image filtering as an anisotropic diffusion process that smoothed image intensities inside an object while preserving the intensity discontinuity between objects. Later, the anisotropic diffusion approach was shown by Black *et al.* [2] to be interpreted as a gradient descent method for optimizing a robust error function model which had been proposed previously. Anisotropic diffusion has been a very successful filtering algorithm, in part because it is easy to implement. However, this algorithm does have the problem of setting two parameters, the robust estimator parameter and the diffusion time. Practical usage of anisotropic diffusion requires a choice between long computation times or blurry boundaries.

Recently, Couprie *et al.* introduced the power watershed (PW) method [3]. Although this technique was introduced in the context of image segmentation, the method can be used as an optimization method for some functionals. In this paper we show that PW is well-suited to address the robust estimator filtering model and therefore provide an alternative to anisotropic diffusion for the optimization of this model. There are several advantages of PW as compared to anisotropic diffusion in the context of optimizing the robust estimator filtering model. First, we remove the robust estimator parameter. Second, we preserve discontinuities without blurring while also allowing for a fast optimization based on PW. Third, no time step needs to be determined (i.e., no risk of a divergent solution). Therefore, our optimization via PW removes the false choice presented by anisotropic diffusion between fast optimization and sharp object boundaries.

Other models for discontinuity-preserving filtering are also well-known, including total variation [4], the Mumford-Shah functional [5] and the piecewise constant Mumford-Shah (Chan-Vese) model [6]. The total variation model is convex, allowing for a relatively efficient solution [7, 8] while the Mumford-Shah and Chan-Vese models are more difficult to optimize. However, recent work [9, 10] has provided more efficient optimization methods. Despite this progress in the optimization of other discontinuity-preserving filtering methods, very little work has been done to improve the speed/blur tradeoff that is necessary in the anisotropic diffusion algorithm. Rather than promoting one discontinuity-preserving model over another, we simply show here how the PW may be used to very efficiently optimize the robust estimator model underlying anisotropic diffusion such that we no longer have to choose between fast optimization and blurred boundaries. Our optimization will be achieved at the cost of a few iterations of the efficient PW, which is simple to implement.

## 2. FORMULATION

Black *et al.* showed that anisotropic diffusion could be viewed as the minimization of a robust estimator. We employ this same robust estimator formulation of anisotropic diffusion, but show that the power watershed may be used to iteratively optimize the formulation.

Since the power watershed is defined on a graph, we begin by casting the anisotropic diffusion algorithm in discrete terms. A graph consists of a pair $G = (V, E)$ with vertices $v \in V$ and edges $e \in E \subseteq V \times V$. An edge, $e$, spanning two vertices, $v_i$ and $v_j$, is denoted by $e_{ij}$. A weighted graph assigns a (typically non-negative and real) value to each edge called a weight. The weight of an edge $e_{ij}$ is denoted by $w_{ij}$. We define the edge-node incidence matrix $A$ of a graph $G$ as

$$A_{e_{ij}v_k} = \begin{cases} +1 & \text{if the node } i = k, \\ -1 & \text{if the node } j = k, \\ 0 & \text{otherwise.} \end{cases}$$

The incidence matrix $A$ is used here as the combinatorial analogue of the continuous gradient operator.

Given these definitions, the anisotropic diffusion algorithm is defined as

$$\frac{dx}{dt} = A^T g(Ax) Ax, \tag{1}$$

where $x$ is the image intensities of a filtered image, and $x$ at time 0 equals the input, unfiltered image. The function $g(x)$ is used to prevent blurring over edges and may be of any decreasing form. One possibility for $g(x)$ suggested in [1] is

$$g(x) = \exp\left(-\alpha x^2\right), \tag{2}$$

where $\alpha$ is a free parameter.

The typical method for solving the anisotropic diffusion equation in (1) is via a forward Euler method in which the iteration

$$x^{k+1} = x^k + dt A^T g(Ax^k) Ax^k,$$

is applied, using a time step $dt$ which is designed to obey the CFL conditions and ensure stability of the solution. In practice, the number of iterations applied to produce the filtered image is a free parameter which is set to define the desired level of smoothing.

Black *et al.* [2] showed that the anisotropic diffusion equation given by (1) could be viewed as the gradient of the energy

$$E(x) = \sigma(Ax), \tag{3}$$

where $\sigma(x)$ is a *robust estimator* or an *M-estimator*. A gradient of the Black *et al.* energy in (3) is given by

$$\frac{dE}{dx} = A^T \sigma'(Ax) Ax. \tag{4}$$

Therefore, when $\sigma'(Ax) = g(Ax)$, then the gradient of the Black *et al.* energy, (4) is the same as the anisotropic diffusion equation in (1). In other words, *anisotropic diffusion is*

*gradient descent minimization of* (3). Further, the $\sigma(z)$ corresponding to the PM weighting function in (2) is given by the Welsch function $\sigma(z) = 1 - \exp\left(-\alpha z^2\right)$.

Black *et al.* explored the effect of different robust estimator forms of $\sigma(z)$. However, a common characteristic of all robust estimators is that they exhibit linear or quadratic growth near $z = 0$, but provide a nearly constant output as $z \to \infty$. The transition between quadratic growth and constant growth is controlled by the parameter $\alpha$. If $\alpha \to \infty$, then the robust estimator ultimately achieves its constant growth phase for any $z \neq 0$. In this way, the $\sigma(z)$ function may be viewed as a differentiable approximation to the $|| \cdot ||_0$ norm in which $||0||_0 = 0$ and $||z||_0 = 1$ for any $z \neq 0$. Consequently, the anisotropic diffusion filtering algorithm may be viewed as gradient descent on an energy functional which is an approximation to the $|| \cdot ||_0$ norm.

Previous filtering algorithms that explicitly formulated filtering from the standpoint of optimizing the $|| \cdot ||_0$ norm took the form

$$E(x) = \sigma(Ax) + \lambda h(x, f), \tag{5}$$

in which $f$ represents the intensities of the input unfiltered image, $h(x, f)$ represents a loss function and $\lambda$ is a free parameter. When $h(x, f) = ||x - f||_2^2$, the gradient of (5) becomes $\frac{dE}{dx} = A^T \sigma'(Ax) Ax + \lambda(f - x)$, which achieves a stable point when

$$\left(A^T \sigma'(Ax) A + \lambda I\right) x = \lambda f. \tag{6}$$

Since (6) may be viewed as a *backward Euler* solution for the anisotropic diffusion equation (1) when $\lambda = \frac{1}{dt}$, then the anisotropic diffusion algorithm may be seen as the optimization of a robust estimator of the image gradient balanced against a loss function of the form $||x - f||_2^2$ where the tradeoff between gradient smoothness and data fidelity is governed by $\lambda = \frac{1}{dt}$. Therefore, by setting a fixed *time* for the anisotropic diffusion, we may view the solution obtained for this time as a steady-state optimization of our second energy functional (5) with a corresponding $\lambda$ parameter. Viewing the time parameter as a loss function for the fidelity of the filtered image with the original, we may freely alter the loss function.

### 2.1. Anisotropic diffusion and $L_0$ norm

As said in the previous section, it is possible to alter the loss function. For example, we could optimize the fully robust energy

$$E(x) = \sigma(Ax) + \lambda \sigma(x - f),$$

with gradient given by

$$\frac{dE}{dx} = 2\alpha A^T \sigma'(Ax) Ax + 2\alpha\lambda \sigma'(x - f) x.$$

If, at any iteration, we fix the values of $x$ inside the robust error function, then we have

$$\frac{dE}{dx^{k+1}} = A^T \sigma'(Ax^k) Ax^{k+1} + \lambda \sigma'(x^k - f) x^{k+1}. \tag{7}$$

This energy may be written as the steady-state optimization of the energy functional

$$E_{k+1} = x^{{k+1}^T} A^T \sigma'(Ax^k) Ax^{k+1} + \lambda x^{{k+1}^T} \sigma'(x^k - f) x^{k+1},$$

that may also possibly be written

$$E_{k+1} = \sum_{e_{ij}} \sigma'(Ax^k) \left(x_i^{k+1} - x_j^{k+1}\right)^2$$
$$+ \lambda \sum_{v_i} \sigma'(x^k - f) \left(x^{k+1} - f\right)^2. \quad (8)$$

This expression for the energy to compute a minimum step is of a form that may be optimized by the power watershed (PW) [3] if the parameter $\alpha \to \infty$ for $g(x)$.

The generalized PW energy is given by

$$\min_x \sum_{e_{ij} \in E} w_{ij}{}^p |x_i - x_j|^q + \sum_{v_i \in V} w_i{}^p |x_i - y_i|^q \quad (9)$$

where $y$ represents a measured configuration and $x$ represents the target configuration. In this equation, the first term essentially forces $x$ to vary smoothly within an object, and the second term enforces data fidelity. We can see that Eq. 8 is of the same form as Eq. 9, with $q = 2$, $p = \alpha$.

Therefore, we suggest the following filtering algorithm :

---

**Algorithm 1**: Anisotropic diffusion using PW

**Data**: An image $f$, an initial solution $x^0$, $\lambda \in \mathbb{R}_+^*$
**Result**: A **filtered image** $x^k$
Set $k = 0$. Build the graph described in Fig. 1.
**repeat**

 Generate the pairwise weights $\exp -(Ax^k)^2$, and unary weights $\exp -(x^k - f)^2$.
 Use PW with $y = f$ to optimize (8) to obtain $x^{k+1}$.
 $k = k + 1$;
**until** $||x^{k+1} - x^k||_2 < \epsilon$

---

This technique can be initialized by the image $f$, or, in the case of very noisy images by a smoother version of the image, for instance by the application of a Gaussian or a median filter.

In the initialization step of the algorithm, we need to build a graph in linking each pixel node with its neighbors, for example its 4 neighbors for 4-connectivity. The values of those nodes will be the $x^k$. Noting $N$ the number of pixels in the image, we add $N$ nodes to this graph in linking each pixel node $v_i$ by an edge to the new node $v_{i+N}$. Those additional nodes are set to constant values of $f$. A description of the graph on a example is shown in Fig. 1. The algorithm using power watershed for anisotropic diffusion is summarized in Alg. 1.

## 3. RESULTS

We now demonstrate the performance of the power watershed algorithm for anisotropic diffusion in presenting results on
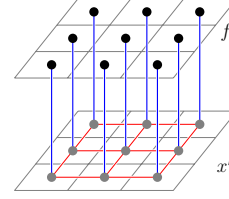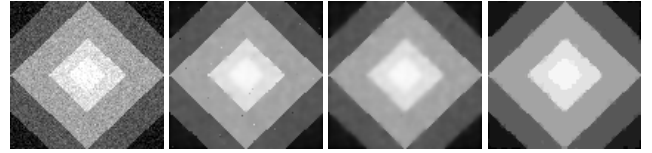


**Fig. 1**. Graph necessary to build for applying the power watershed diffusion algorithm to a $3 \times 3$ image. The unary weights in blue and pairwise weights in red are updated at each step.



(a) Noisy image, (b) PM, PSNR = (c) PM, PSNR = (d) PW, PSNR = PSNR = 24.24dB  34.03dB          30.46dB          31.54dB

**Fig. 2**. Comparison of Perona-Malik(PM), and power watershed(PW) algorithms for denoising a synthetic image. (b) PM used with 80 iterations $\alpha = 0.0015$, leading to a good PSNR but with remaining isolated noisy pixels. (c) PM, best compromise found for this image to remove the isolated pixels with 50 iterations and $\alpha = 0.0005$. (d) A median filtered image as initialization and $\lambda = 0.975$ allows to obtain a better PSNR while removing isolated noisy pixels.

synthetic and real images. A first, the test in Fig. 2 is performed on a synthetic image corrupted with a Gaussian noise of standard deviation $\sigma = 16$. For each result, we may compute the peak signal-to-noise ratio (PSNR) relatively to the original image. Typical values for the PSNR in denoising lie between 20 and 40 dB where higher is better. In comparison with PM algorithm, the power watershed algorithm tends to produce piecewise constant results. The result obtained at Fig. 2(d) shows a good compromise between noise removing and edge preservation. However there is a choice to make for PM algorithm (Fig. 2(b) and Fig. 2(c)) between complete denoising and good contrast edge restoration.

Examples on real images in Fig. 3 and Fig. 4 show that the power watershed algorithm may be useful as a filtering step

|  |  | Perona-Malik |  | PW |
|---|---|---|---|---|
| Fig. 2, $104 \times 100$ | Nb iter. | 50 | 80 | 5 |
|  | Time (s) | 0.19 | 0.30 | 0.17 |
| Fig. 4, $250 \times 300$ | Nb iter. | 50 | 80 | 6 |
|  | Time (s) | 1.38 | 2.14 | 1.78 |
| Fig. 3, $299 \times 364$ | Nb iter. | 50 | 80 | 6 |
|  | Time (s) | 1.95 | 3.08 | 2.43 |

**Table 1**. Number of iterations of Perona-Malik, and power watershed algorithms on the image of Fig. 2, Fig. 4, and Fig. 3

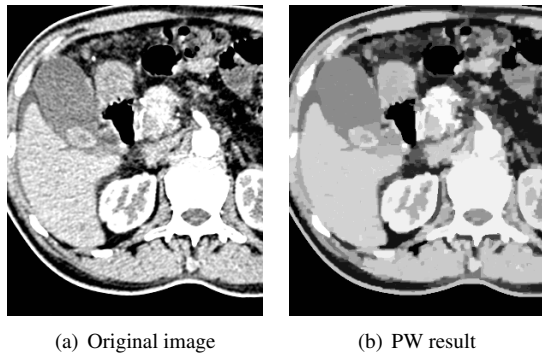(a) Original image      (b) PW result

**Fig. 3**. Filtering of a liver image by power watershed. Here noise and small vessels are both removed, leading to a result which may be used as a first step before segmentation.
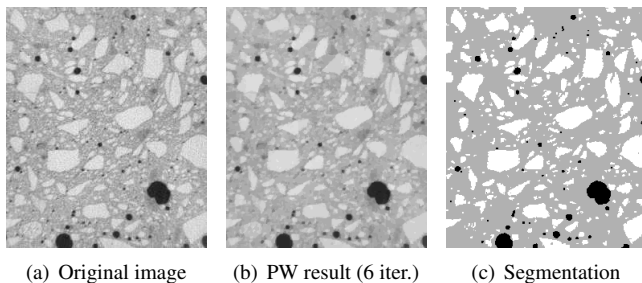


(a) Original image    (b) PW result (6 iter.)    (c) Segmentation

**Fig. 4**. The segmentation of concrete images in three classes (bubbles in black, stones in light grey) is useful for the study of the material's mechanical properties. The segmentation is obtained by two thresholds of the filtered image by PW.

before segmentation. In terms of computation time, each iteration of the power watershed algorithm for anisotropic diffusion operates in quasi-linear time in practice. We noticed empirically that the convergence is very fast (fewer than 10 iterations). The result obtained at the first iteration is generally close to the final result as shown in Fig. 4. We give some timing with necessary number of iterations to reach convergence at Table 1.

## 4. CONCLUSION

The power watershed (PW) algorithm was originally presented as a technique applied to image segmentation. In this work, we showed that PW provides an optimization procedure that allows us to optimize robust error measures which are operated in a particular parameter range.

Black *et al.* showed how the anisotropic diffusion method could be viewed as optimization of a robust error filtering model. We showed that PW could be used to optimize the same robust error filtering model, leading to an alternative optimization procedure.

An aspect of the PW optimization in this context is that it applies to the optimization of the robust error filtering model when operated in a particular parameter range which effectively models the image as being piecewise constant. For cases in which this model was reasonable (such as the synthetic images), the algorithm exhibited a strong denoising capability. When this denoising algorithm was applied to real images, the effect of imposing a piecewise constant model was to effectively quantize images into a small number of grayscale levels (Fig. 3 and 4). This transformation of an image into a piecewise constant form may be helpful as a pre-processing step for segmentation or recognition.

We used the energy minimization aspect of PW to expand the traditional use of watersheds from a segmentation algorithm to an image filtering algorithm. Although our algorithm optimizes the filtering objective as a real-valued optimization problem, we showed that we still retain the fast speed of the watershed algorithm. However, the limit of PW as an energy optimization strategy remains unclear — What energy functions can be minimized via power watersheds? Future work will address this issue and continue to demonstrate applications which can benefit from the speed of power watershed.

## 5. REFERENCES

[1] P. Perona and J. Malik, "Scale-space and edge detection using anisotropic diffusion," *IEEE Transactions on PAMI*, vol. 12, no. 7, pp. 629–639, July 1990.

[2] M. J. Black, G. Sapiro, D. H. Marimont, and D. Heeger, "Robust anisotropic diffusion," *IEEE Trans. on Image Proc.*, vol. 7, no. 3, pp. 421–432, March 1998.

[3] C. Couprie, L. Grady, L. Najman, and H. Talbot, "Power watersheds: A new image segmentation framework extending graph cuts, random walker and optimal spanning forest," in *Proc. of ICCV*, Sept. 2009, pp. 731–738.

[4] L. Rudin, S. Osher, and E. Fatemi, "Nonlinear total variation based noise removal algorithms," *Physica D*, vol. 60, no. 1-4, pp. 259–268, 1992.

[5] D. Mumford and J. Shah, "Optimal approximations by piecewise smooth functions and associated variational problems," *Comm. Pure and Appl. Math.*, vol. 42, pp. 577–685, 1989.

[6] L. Vese and T. Chan, "A multiphase level set framework for image segmentation using the Mumford and Shah model," *Int. J. of Comp. Vis.*, vol. 50, no. 3, pp. 271–293, 2002.

[7] A. Chambolle, "An algorithm for total variation minimization and applications," *JMIV*, vol. 20, no. 1–2, pp. 89–97, 2004.

[8] J. Darbon and M. Sigelle, "Image restoration with discrete constrained total variation part I: Fast and exact optimization," *JMIV*, vol. 26, no. 3, pp. 261–276, Dec. 2006.

[9] X. Bresson, S. Esedoglu, P. Vandergheynst, J. Thiran, and S. Osher, "Fast global minimization of the active contour/snake model," *JMIV*, vol. 28, no. 2, pp. 151–167, 2007.

[10] L. Grady and C. Alvino, "The piecewise smooth Mumford-Shah functional on an arbitrary graph," *IEEE Trans. on Image Proc.*, vol. 18, no. 11, pp. 2547–2561, Nov. 2009.