

## A generalization of the Perona-Malik anisotropic diffusion method using restricted dissimilarity functions

C. Lopez-Molina<sup>1,2</sup>, B. De Baets<sup>2</sup>, J. Cerron<sup>1</sup>, M. Galar<sup>1</sup>, H. Bustince<sup>1</sup>

<sup>1</sup> *Dpto. Automática y Computación, Universidad Publica de Navarra,  
31006 Pamplona, Spain,*

*E-mail: carlos.lopez@unavarra.es*

<sup>2</sup> *Dept. of Mathematical Modelling, Statistics and Bioinformatics, Ghent University  
Coupure links 653,  
9000 Gent, Belgium*

*E-mail: bernard.debaets@ugent.be*

Received 7 August 2012

Accepted 17 August 2012

### Abstract

There exists a large number of techniques for content-aware smoothing. Despite its simplicity, the Perona-Malik Anisotropic Diffusion method is among the most employed ones. In this work we study this method in detail and propose a generalization of its diffusion scheme using restricted dissimilarity functions to measure the intensity differences between neighbouring pixels. This generalization permits a better adaptation of the diffusion process to the characteristics of the images.

*Keywords:* Image processing, Anisotropic diffusion, Edge detection, Restricted dissimilarity function.

### 1. Introduction

Natural images usually contain a large number of imperfections due to factors such as noise or bad illumination of the scene. In order to minimize their impact, most of the automated tasks in image processing require some kind of image preprocessing. In the specific case of edge detection, this preprocessing generally intends to remove small imperfections or unimportant artifacts from the image, reducing the number of false positives <sup>1</sup>.

The most basic (and employed) preprocessing techniques for edge detection are the so-called content-unaware smoothing (CUS) techniques. These techniques perform spatial regularization at every region in the image, regardless of

its content. A prominent CUS technique is Gaussian linear filtering, which has excellent properties of signal conservation and computational efficiency <sup>2,3</sup>. CUS is effective in removing imperfections of the image, as well as attenuating the visual impact of high frequencies and textures, but also has undesirable effects. Since the smoothing is performed at every region (including those having edges), the position of the edges might be altered, hindering their accurate localization. Moreover, smoothing potentially leads to excessively blurred images, on which some edges are undetectable. This trade-off between the regularization of non-meaningful regions of the image and the overblurring of the edges has been studied in the literature <sup>4,5</sup>, but it has not yet been solved. An important example of these efforts is the

study of the Gaussian Scale-Space (GSS)<sup>6</sup>, a framework on which the changes undergone by an image filtered with Gaussian filters can be studied (see<sup>7,8</sup>).

Content-aware smoothing (CAS) techniques behave differently depending upon local features of each region. In the case of edge detection, these techniques aim to combine the regularization of non-meaningful image variations with the preservation of regions containing edges. One of the pioneering CAS techniques is Anisotropic Diffusion (AD), proposed by Perona and Malik<sup>9</sup>. AD methods consider an image to be a grid of bodies, each of them having a heat level equal to the intensity of the corresponding pixel. These methods simulate the process of heat diffusion in the image, inhibiting heat transfer at those positions of the image where the local intensity differences are large. In this way, AD combines the regularization inside the objects (where the local intensity differences are expected to be small) with the preservation of the true edges (across which these differences are large). Perona and Malik propose a very simplistic, yet effective, scheme for diffusion, avoiding most of the implicit problems in discrete spatio-temporal models. We refer to this method as the Perona-Malik AD method (PMAD). Many other AD methods have been presented thereafter, aiming at either edge preservation<sup>10,11</sup>, structural reinforcement<sup>12</sup> or noise removal<sup>13,14</sup>. In this work, we focus on PMAD and propose a generalization which tackles the way in which the local intensity differences are computed and provides more flexibility when performing the heat transfer inhibition.

The results obtained by CAS techniques heavily depend upon the adequate measurement of local features, since they comprise all the information used to determine the local behaviour. In the case of PMAD, the heat transfer at each pixel is determined by the difference between its intensity (heat level) and that of its neighbours. However, proposals other than the difference have been suggested in the literature to express the dissimilarity of two values. In this work, we focus on Restricted Dissimilarity Functions (RDFs)<sup>15</sup>. Initially created to express the dissimilarity (distance) between two membership degrees, they can also be used to express the

dissimilarity between the intensity of two pixels. Indeed, they have been successfully applied to image processing<sup>15,16</sup>. We generalize PMAD using RDFs to measure the dissimilarity between neighbouring pixels. In this way we introduce a new framework where the diffusion process varies depending upon the RDF used to measure the local intensity differences, so that we can adapt their interpretation to the characteristics of the image.

The remainder of this paper is organized as follows. In Section 2 we recall the concept of an RDF. In Section 3, we recall PMAD and present our generalization using RDFs. Section 4 reports on the experiments comparing the performance of PMAD with that of several variants of the proposed generalization. To conclude, conclusions are drawn in Section 5.

## 2. Restricted dissimilarity functions

RDFs attempt to quantify the dissimilarity between two membership degrees (*i.e.*, two values in  $[0, 1]$ )<sup>17</sup>.

**Definition 2.1**<sup>15</sup> *A restricted dissimilarity function is a mapping  $r : [0, 1]^2 \rightarrow [0, 1]$  satisfying*

- (1)  $r(x, y) = r(y, x)$  for all  $x, y \in [0, 1]$ ;
- (2)  $r(x, y) = 1$  if and only if  $x = 0$  and  $y = 1$  or  $x = 1$  and  $y = 0$ ;
- (3)  $r(x, y) = 0$  if and only if  $x = y$ ;
- (4) For all  $x, y, z \in [0, 1]$ , if  $x \leq y \leq z$ , then  $r(x, y) \leq r(x, z)$  and  $r(y, z) \leq r(x, z)$ .

**Proposition 2.1** *If  $\varphi_1, \varphi_2$  are two automorphisms of the unit interval, then the mapping  $r : [0, 1]^2 \rightarrow [0, 1]$  defined by*

$$r(x, y) = \varphi_1(|\varphi_2(x) - \varphi_2(y)|) \quad (1)$$

*is a restricted dissimilarity function.*

**Example 1.** Consider the automorphisms  $\varphi_1(x) = x^2$  and  $\varphi_2(x) = x$  for all  $x \in [0, 1]$ . Proposition 2.1 yields  $r(x, y) = |x - y|^2$ .

**Corollary 2.1** *Let  $r$  be an RDF constructed as in Prop. 2.1, and let  $n$  be a strong negation constructed as  $n(x) = \varphi_2^{-1}(1 - \varphi_2(x))$ . Then,  $r(x, y) = r(n(x), n(y))$  holds for all  $x, y \in [0, 1]$ .*

The latter property is of special interest for image processing, since it implies that the dissimilarity between the intensity of any two pixels in a given image is the same as in the negative image obtained using  $n(x) = \varphi_2^{-1}(1 - \varphi_2(x))$ .

Automorphisms can be generated using the expression  $\varphi(x) = x^\alpha$ , with  $\alpha \in ]0, \infty[$ . In this work we consider different pairs of automorphisms for the construction of RDFs (Prop. 2.1). In Table 1 we list the RDFs used in this work, whose graphical representation is included in Fig. 1. Note that, although RDFs have common features due to the properties in Def. 2.1, they can produce very divergent interpretations of the dissimilarity of certain pairs  $(x, y)$ .

### 3. Anisotropic diffusion

#### 3.1. The Perona-Malik anisotropic diffusion method

AD stems from the application of the heat diffusion paradigm to image processing. Initially, if no inhibition were imposed to the heat transfer, the whole image would reach a stable state with homogeneous heat (intensity value) at each pixel. AD consists of inhibiting heat transfer at the edges of the objects in the image, so that the regularization process within each object is independent from that in the rest of the image. Ideally, each object should reach a stable state (*i.e.*, each pixel within each object should have the same intensity), but the edges should still be recognizable, since each object is likely to reach a different heat level.

Heat diffusion in an image  $I$  is a flux  $\phi$  modelled by Fick's equation:

$$\phi = -D\nabla I, \quad (2)$$

where  $D$  is a symmetric, positive definite matrix and  $\nabla I$  represents the local conductivity of  $I$ . Assuming  $I$  is a heat loss-free scenario, at time  $t$  we have

$$\delta_t I = -\operatorname{div} \phi, \quad (3)$$

where  $\operatorname{div}$  is the divergence operator. From (2) and (3), we have

$$\delta_t I = \operatorname{div}(D\nabla I). \quad (4)$$

Perona and Malik approximate the evolution of the image with

$$\delta_t I = \operatorname{div}(g(|\nabla I|^2)\nabla I), \quad (5)$$

where  $g$  is a decreasing function modulating the amount of heat transfer depending upon the magnitude of the gradient<sup>9</sup>.

Vectorial expressions (such as Eq. (5)) are prone to generate problems in discrete environments such as digital images. Even the computation of the gradient  $\nabla I$  is an ill-posed problem<sup>18</sup>. In order to avoid these problems (which affect both the calculation of  $\nabla I$  and the subsequent application of the flux  $\delta_t I$ ), Perona and Malik develop a simple, scalar approximation. In PMAD the implementation of Eq. (5) consists of calculating the heat transfer between each pixel and its four direct neighbours (see Fig. 2). The rest of the pixels in the neighbourhood is assumed to be unconnected to the central one. From this diagram, the authors propose to calculate 4 values:

$$\begin{aligned} \Psi_N I_{i,j} &= I_{i-1,j} - I_{i,j} \\ \Psi_W I_{i,j} &= I_{i,j-1} - I_{i,j} \\ \Psi_E I_{i,j} &= I_{i,j+1} - I_{i,j} \\ \Psi_S I_{i,j} &= I_{i+1,j} - I_{i,j} \end{aligned} \quad (6)$$

where  $N, W, E$  and  $S$  indicate the direction in which the difference is computed.

Any diffusion problem is driven by an initial state and a diffusion scheme. In the case of PMAD, the initial state ( $I^{[0]}$ ) is the original image. Then, the image evolves iteratively so that the value of the pixel at the position  $(i, j)$  at time  $t + 1$  is given by:

$$I_{i,j}^{[t+1]} = I_{i,j}^{[t]} + \lambda \left( z_N \cdot \Psi_N I_{i,j}^{[t]} + z_W \cdot \Psi_W I_{i,j}^{[t]} + z_E \cdot \Psi_E I_{i,j}^{[t]} + z_S \cdot \Psi_S I_{i,j}^{[t]} \right) \quad (7)$$

where  $\lambda \in ]0, 0.25]$  is required for the numerical scheme to be stable. Smaller values of  $\lambda$  lead to more stable (and slower) diffusion. The conductivity coefficients  $z$  represent the amount of diffusion that takes place between the pixel at the position  $(i, j)$  and its neighbours. The authors propose to use

$$z_\gamma = g(|\Psi_\gamma I_{i,j}|) \quad (8)$$

Table 1: List of RDFs used in this work.

Name	Automorphisms	Expression
$r_1$	$\varphi_1(x) = x, \varphi_2(x) = \sqrt{x}$	$r_1(x, y) =  \sqrt{x} - \sqrt{y} $
$r_2$	$\varphi_1(x) = x, \varphi_2(x) = x$	$r_2(x, y) =  x - y $
$r_3$	$\varphi_1(x) = x, \varphi_2(x) = x^2$	$r_3(x, y) =  x^2 - y^2 $
$r_4$	$\varphi_1(x) = \sqrt{x}, \varphi_2(x) = \sqrt{x}$	$r_4(x, y) = \sqrt{ \sqrt{x} - \sqrt{y} }$
$r_5$	$\varphi_1(x) = \sqrt{x}, \varphi_2(x) = x$	$r_5(x, y) = \sqrt{ x - y }$
$r_6$	$\varphi_1(x) = \sqrt{x}, \varphi_2(x) = x^2$	$r_6(x, y) = \sqrt{ x^2 - y^2 }$

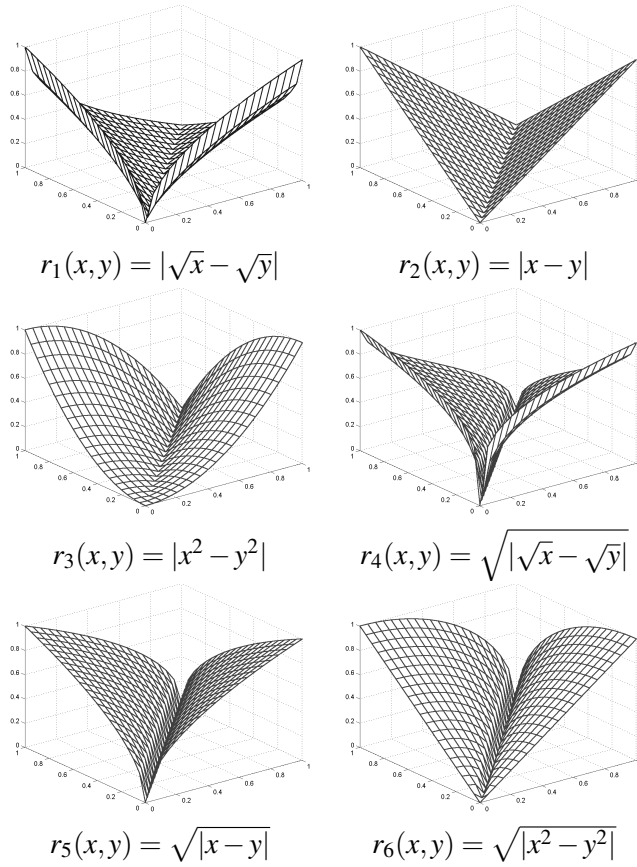


Figure 1: RDFs obtained using different automorphisms of  $[0, 1]$ , as listed in Table 1.

for each orientation  $\gamma \in \{N, W, E, S\}$ , where  $g$ , as in Eq. (5), is a decreasing function that modulates the amount of diffusion depending upon the absolute difference between the intensity of the pixels. We refer to the functions  $g$  as *conductivity functions*. As a conductivity function, Perona and Malik propose to use either

$$g_1(x) = e^{-(\frac{x}{K})^2} \quad (9)$$

or

$$g_2(x) = \frac{1}{1 + (\frac{x}{K})^2}, \quad (10)$$

where  $K$  is a threshold such that the diffusion is inhibited whenever  $|\Psi_\gamma I_{i,j}| > K$ . In Fig. 3 we plot the functions  $g_1$  and  $g_2$  with two different values of  $K$ . In this figure we observe how, for a fixed  $K$ ,  $g_2$  permits more diffusion than  $g_1$ . More importantly, when  $x \approx 1$ ,  $g_2(x)$  is above 0, which means that there is some diffusion even when the difference between pixels is extremely large. A deep study on the interpretation of conductivity functions was performed by Whitaker and Pizer<sup>19</sup>, while Black *et al.* proposed and studied alternative conductivity functions in<sup>20</sup>.

PMAD has been criticized in the literature, mainly because of the scalar treatment of the diffusivity<sup>21,10,19</sup>. However, it can also be seen as vectorial, being restrictive on the orientation of the fluxes. The flux does not happen *at the pixel*, but at each of the four sides of the pixel (as depicted in Fig. 2). The simplicity of the scheme, especially regarding the connectivity restrictions in the neighbourhood, was already recognized by Perona and Malik<sup>9</sup>. They acknowledge that the discretization is not *exact*, but justify the choice by its *simplicity*, and claim that *less crude approximations of the gradient yielded perceptually similar results at the price of increased computational complexity*. Despite its simplicity, PMAD holds interesting properties, namely:

- Energy preservation: The global energy in the environment (image) is preserved.
- Causality: No new local maxima or minima are generated in the diffusion process.

Note that, unlike other AD methods, PMAD is not bound to the time scale. Using Eq. (7) a sequence of versions of the image are obtained, but

time is not taken into consideration for purposes other than iteration counting.

### 3.2. Generalizing Perona-Malik anisotropic diffusion using restricted dissimilarity functions

As we have shown in Section 2, RDFs are useful to measure the dissimilarity between the intensity of two pixels (Fig. 1). The fact is that the perceptual dissimilarity between two intensity values need not to be linearly proportional to their absolute difference. As a consequence, functions that are able to represent appropriately the dissimilarity between two pixels can be considered. We generalize PMAD by replacing the absolute difference between pixels by an RDF. In this way, the measurements in Eq. (6) turn into

$$\begin{aligned} \Psi_N I_{i,j} &= \text{sign}(I_{i-1,j} - I_{i,j}) \cdot r(I_{i-1,j}, I_{i,j}) \\ \Psi_W I_{i,j} &= \text{sign}(I_{i,j-1} - I_{i,j}) \cdot r(I_{i,j-1}, I_{i,j}) \\ \Psi_E I_{i,j} &= \text{sign}(I_{i,j+1} - I_{i,j}) \cdot r(I_{i,j+1}, I_{i,j}) \\ \Psi_S I_{i,j} &= \text{sign}(I_{i+1,j} - I_{i,j}) \cdot r(I_{i+1,j}, I_{i,j}) \end{aligned} \quad (11)$$

where  $r$  is an RDF. Our method generalizes PMAD, since Eq. (6) is computationally equivalent to Eq. (11) using  $r_2(x, y) = |x - y|$ . We refer to this AD method as the Generalized Perona-Malik Anisotropic Diffusion method using an RDF ( $\text{GPMAD-}r$ ).

The role of  $r$  in  $\text{GPMAD-}r$  is to measure the dissimilarity between neighbouring pixels. Ideally, an RDF must produce small dissimilarity values for pixels within the same object; but at the same time, it should be able to produce large values for pixels belonging to different objects. Since the expected behaviour of the diffusion process may vary among different images, the selection of the most appropriate RDF is problem dependent.

In order to show the influence of the RDF in the diffusion process, Fig. 4 contains the diffusion coefficients ( $z_\gamma(x, y) = g(r(x, y))$ ) obtained combining the RDFs in Table 1 and the conductivity functions  $g_1$  and  $g_2$ . For each conductivity function two different values of the threshold  $K$  have been used. Recall that the case of PMAD is recovered using  $r_2$ . The amount of diffusion between two pixels is linearly

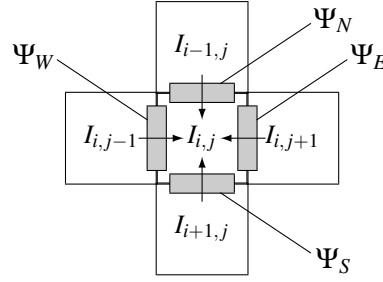


Figure 2: Discrete scheme for diffusion used in PMAD.

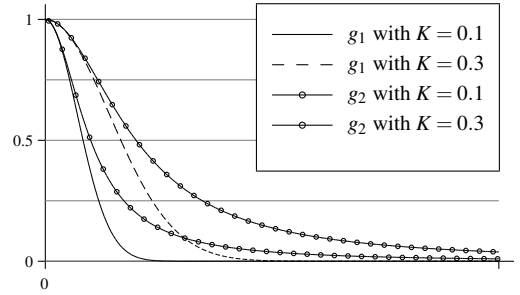


Figure 3: Conductivity functions  $g$  proposed by Perona and Malik.

proportional to  $z_\gamma(x,y)$ , and hence combinations of  $g$  and  $r$  producing greater  $z_\gamma(x,y)$  empower the heat transfer between pixels. We observe two different behaviours, depending upon the automorphism  $\varphi_1$  used to generate the RDF. Those RDFs generated with  $\varphi_1(x) = x$ , *i.e.*,  $r_1$ ,  $r_2$  and  $r_3$ , permit significantly more diffusion than those using  $\varphi_1(x) = \sqrt{x}$ . As expected from Fig. 3,  $g_2$  yields more diffusion than  $g_1$ , but the most influential factor in  $z_\gamma(x,y)$  is the selection of the automorphism  $\varphi_1$ . An important divergence between  $g_1$  and  $g_2$  occurs when  $|x - y| \approx 1$ , since the latter leads to a certain amount of diffusion, even for relatively small values of  $K$ . Interestingly, the behaviour of the function strongly depends upon the overall brightness of  $x$  and  $y$ , not only upon the difference between them. For example,  $r_1$  produces more diffusion than  $r_3$  when  $x \approx y \approx 1$ , but exactly the opposite happens with  $x \approx y \approx 0$ . Therefore,  $r_1$  is a better choice when we intend to regularize bright regions, while inhibiting heat transfer in dark ones.

### 3.3. Examples of image regularization using GPMAD- $r$

In order to illustrate the results that can be obtained using different RDFs, we have applied GPMAD- $r$  to the images in Fig. 5 using the RDFs in Table 1. The first image is a  $100 \times 100$  image contaminated with Gaussian additive contamination ( $\sigma = 0.2$ ). The second one is a natural image with  $256 \times 256$  pixels. The diffusion has been carried out with fixed  $\lambda = 0.1$  and variable  $K$ , which is calculated at each iteration using the histogram-based technique proposed in <sup>9</sup>.

In Fig. 6, the results on the first image are presented. We observe that the choice of  $r$  severely affects the amount of noise that GPMAD- $r$  is able to remove, as well as the number of iterations it needs to do so. Those RDFs producing more diffusion ( $r_1$ ,  $r_2$  and  $r_3$ ) lead to smoother versions of the image. In the case of  $r_1$ , the resulting image is excessively blurred, to the point that the edges are hardly recognizable. When using  $r_2$  or  $r_3$ , the diffusion process is able to remove most of the noise, but at the same time preserves the true edges. Those RDFs yield-

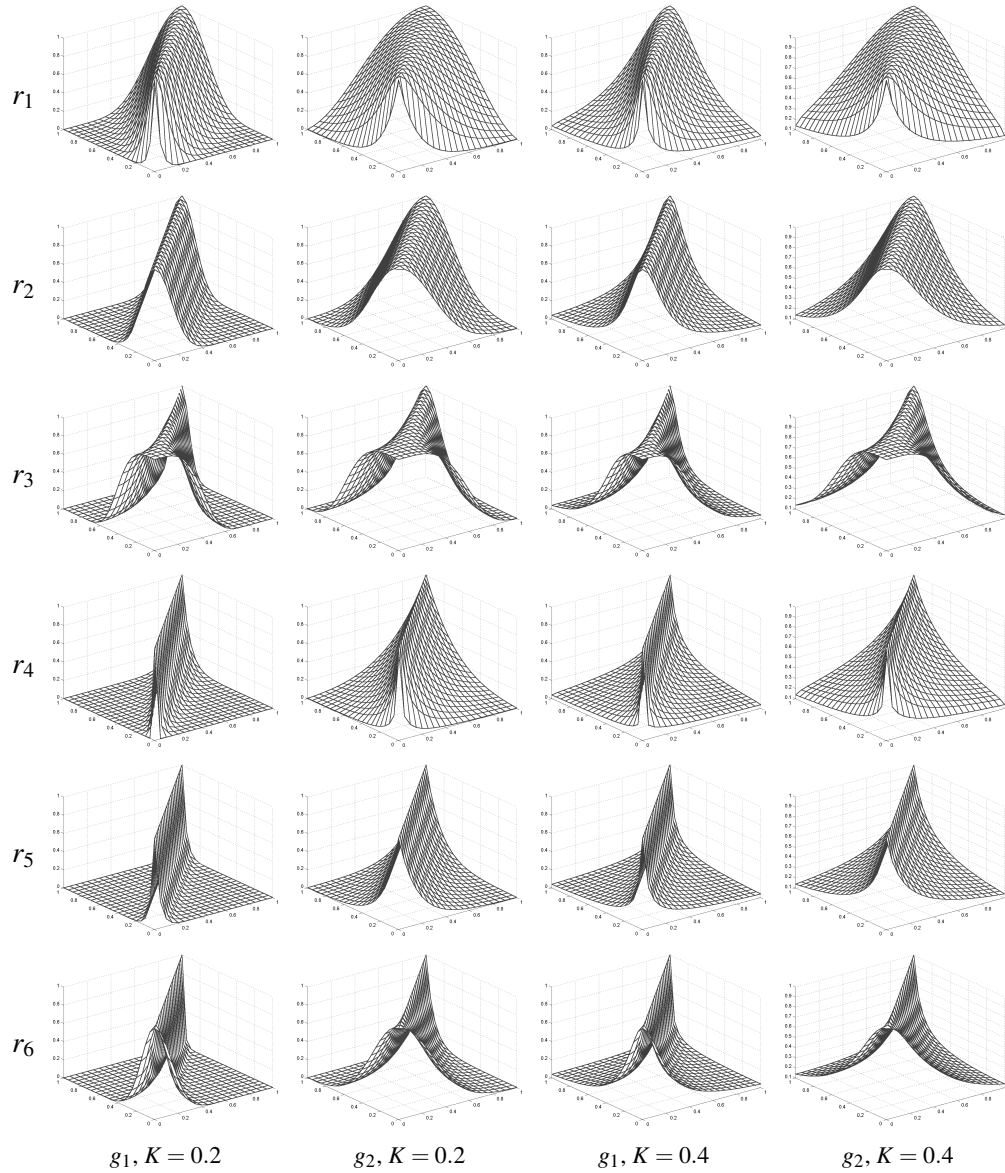


Figure 4: Diffusion coefficients ( $z(x,y) = g(r(x,y))$ ) obtained with different combinations of RDFs and conductivity functions.



Figure 5: Images used for the visual evaluation of GPMAD- $r$ .

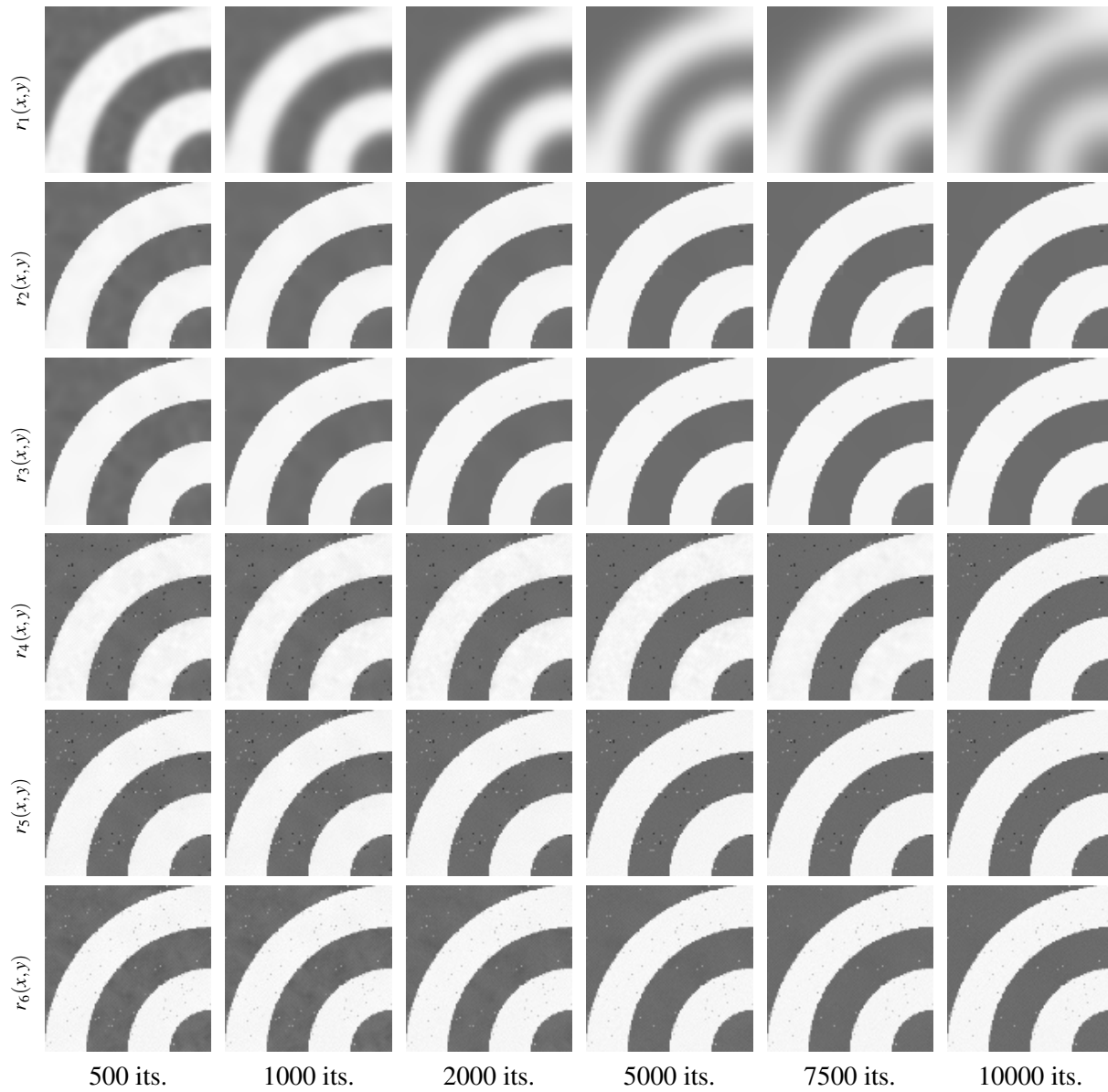


Figure 6: Example of the application of the GPMAD- $r$  on the first image in Fig. 5 with different RDFs. Note that the use of  $r_2$  is computationally equivalent to PMAD.



ing more restricted diffusion ( $r_4$ ,  $r_5$  and  $r_6$ ) produce similar results: they perfectly preserve the edges, but they are unable to regularize the image at the positions where the noise contamination is severe. As a consequence, a significant amount of noise is also preserved. In this image, due to the combination of high-contrast edges and heavy Gaussian contamination,  $r_2$  and  $r_3$  are the RDFs producing the best results.

The results obtained on the second image are displayed in Fig. 7. In this case, the less conservative RDFs (in particular  $r_1$ ) produce excessive diffusion, leading to the removal of important edges, such as the lower side of the middle pepper. Relatively high-contrasted edges are also destroyed, such as the stalk of the leftmost pepper. Alternatively, the RDFs  $r_4$ ,  $r_5$  and  $r_6$  offer a good trade-off between preservation of the edges and intra-region regularization. For this specific image,  $r_4$  seems to produce the best results. However, studying the images in detail we observe that some weak edges in the original image gain contrast as the number of iterations increases. In this way, smooth transitions due to the lighting or shading become sharper, looking like true edges, and potentially producing false positive edge detections. This effect is referred to as *staircasing*, and it might induce errors in further processing of the image<sup>22</sup>.

Despite the fact that intra-region regularization (together with the preservation of the edges) is not easily noticeable on a visual example, is very important for edge detection. In order to illustrate the improvement obtained, we display in Fig. 8 the local contrast\* of the images in the rightmost column of Fig. 7, together with the local contrast of the images produced with Gaussian linear filters with different standard deviations  $\sigma$ . We observe how GPMAD- $r$  outperforms the commonly used Gaussian smoothing in the regularization within the objects, while not blurring the edges. However, this comes at the cost of removing some meaningful edges (leading to false negative detections) and the reinforcement of non-meaningful objects (producing false positives).

The most interesting conclusion drawn from Figs. 6 and 7 is that the results of GPMAD- $r$  are

directly related to the characteristics of the RDF employed to quantify the dissimilarity of the intensity of two pixels. In this way, we can predict the results by analyzing the diffusion coefficients  $z_\gamma(x, y) = g(r(x, y))$ , which depend upon the characteristics of  $r$  and  $g$ .

## 4. Experimental study

The qualitative example in Section 3.2 illustrates the performance of GPMAD- $r$ , but there is a need for a quantitative evaluation of its usefulness for image regularization. In this section, we study the performance of the well-known Canny method for edge detection after regularizing the images using GPMAD- $r$ , with different RDFs  $r$ . We also intend to find out whether there are significant differences in the performance of GPMAD- $r$  when using different RDFs. Since we include  $r_2$  in the comparison, we also cover PMAD. In this experiment the parameter  $\lambda$  is set to 0.05, while  $K$  is updated at each iteration using the same technique as in Section 3.3.

### 4.1. Edge detection algorithm

For the experiment we have selected the Canny method<sup>4</sup>, which is commonly taken as a standard for edge detection<sup>5,23</sup>. In this edge detection method the images are preprocessed using Gaussian linear filters, which we replace by GPMAD- $r$ . The Canny operators for gradient extraction are generated with  $\sigma = 1.0$ <sup>4,24</sup>. Then, the binarization process consists of (a) thinning using Non-Maximum Suppression (NMS)<sup>25</sup> and (b) binarizing using hysteresis<sup>4</sup>. The values of the thresholds in the hysteresis have been set using the technique by Medina-Carnicer *et al.*<sup>26</sup>, which is based on computing the properties of the edge images generated with a large set of threshold candidates.

### 4.2. Quantifying the results of an edge detection method

In order to quantify the quality of a given edge image we use Baddeley's Delta Metric (BDM)<sup>27</sup>,

\*The local contrast at each pixel is defined as the difference between the brightest and darkest pixel in its  $3 \times 3$  neighbourhood.



Figure 7: Example of the application of the GPMAD- $r$  on the second image in Fig. 5 with different RDFs. Note that the use of  $r_2$  is computationally equivalent to PMAD.

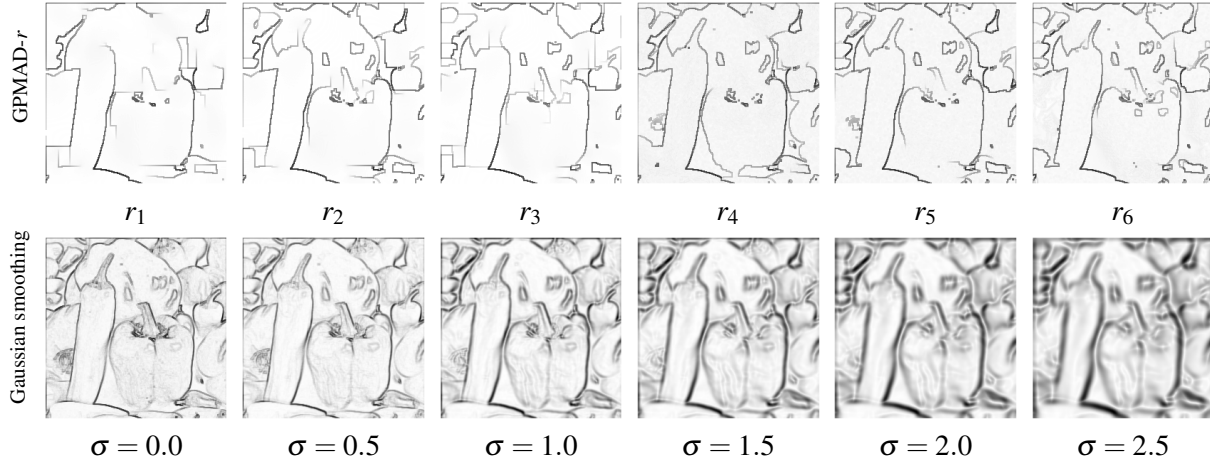


Figure 8: Local contrast of the images generated performing 20000 iterations of GPMAD- $r$  with different RDFs (first row) and the images generated using Gaussian smoothing with different standard deviation ( $\sigma$ ).

which permits to quantify its distance to a *perfect* or *ground-truth* solution. Let  $A$  and  $B$  be two binary edge images of size  $M \times N$ , and let  $S = \{1, \dots, M\} \times \{1, \dots, N\}$  be the set of their positions. Their  $\beta$ -BDM, also denoted  $\Delta_w^\beta(A, B)$ , is defined as

$$\Delta_w^\beta(A, B) = \left[ \frac{1}{|S|} \sum_{s \in S} |w(d(s, A)) - w(d(s, B))|^\beta \right]^{\frac{1}{\beta}} \quad (12)$$

where  $d(s, A)$  represents the distance from the position  $s$  to the closest edge point of the image  $A$  and  $w : \mathbb{R} \rightarrow \mathbb{R}$  is a concave, increasing function used for weighing. We use the trimmed distance  $d(p, q) = \min(10, d_{\text{euc}}(p, q))$ , where  $d_{\text{euc}}$  is the Euclidean distance. In this way, the maximum penalization produced by a pixel is restricted to 10 (about 2% of the image diagonal). The other parameters take values  $\beta = 2$  and  $w(x) = x$ , as in <sup>28,29</sup>.

Let  $A$  be the output of the Canny method in combination with GPMAD- $r$ , with a given RDF  $r$ . Given a set of ground-truth images provided by different human experts,  $\{B_1, \dots, B_n\}$ , the error of the RDF  $r$  in that image is  $\min_{i \in \{1, \dots, n\}} \Delta^2(A, B_i)$ , *i.e.*, the distance of the edge image to the closest solution provided by a human expert.

We have used the images in the Berkeley Segmentation Dataset (BSDS) to test our proposal, consisting of 100 images. This dataset offers a wide

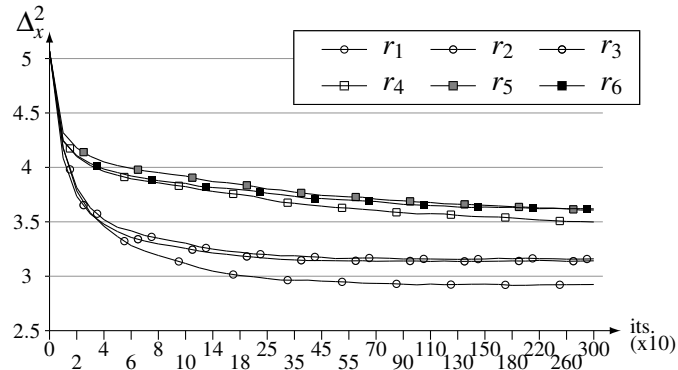
variety of natural images, each of them coming with 5 to 9 hand-made segmentations. The boundaries of the regions of these segmentations can be taken as ideal solutions of the edge detection problem. The images in the BSDS are in grayscale and have dimensions  $321 \times 481$  or  $481 \times 321$ .

### 4.3. Analysis of the results

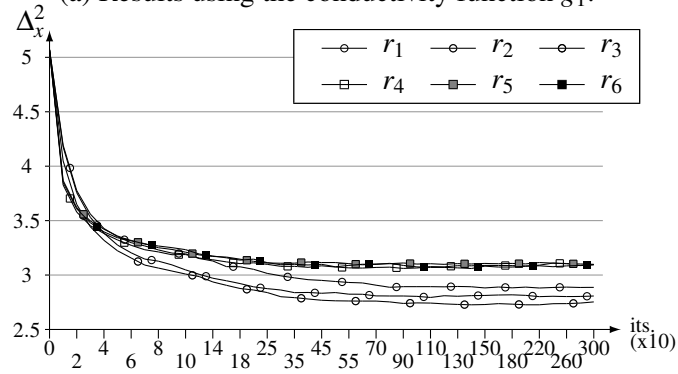
In Fig. 9 we display the results obtained by each of the RDFs in Table 1, *i.e.* the average performance of the Canny method after applying GPMAD- $r$ , depending on the number of iterations. In Fig. 9(a) the results obtained with the conductivity function  $g_1$  are shown, whereas Fig. 9(b) includes those obtained with  $g_2$ . The scale used for the iterations is not linear, so that the evolution of the image in the first iterations can be properly noticed.

First, we observe that most of the improvement gained using GPMAD- $r$  takes place in the first iterations. The improvement after 100 iterations is about 80% of the total gain after 3000 iterations. Second, the error never rises when the number of iterations increases. In this way, we can overestimate the number of iterations needed to perform the diffusion without having the risk of hindering the performance.

In Fig. 9 it can be seen that the performance of GPMAD- $r$  strongly depends on the automorphism



(a) Results using the conductivity function  $g_1$ .



(b) Results using the conductivity function  $g_2$ .

Figure 9: Performance of the Canny edge detection method in combination with GPMAD- $r$  measured in terms of  $\Delta^2$ .

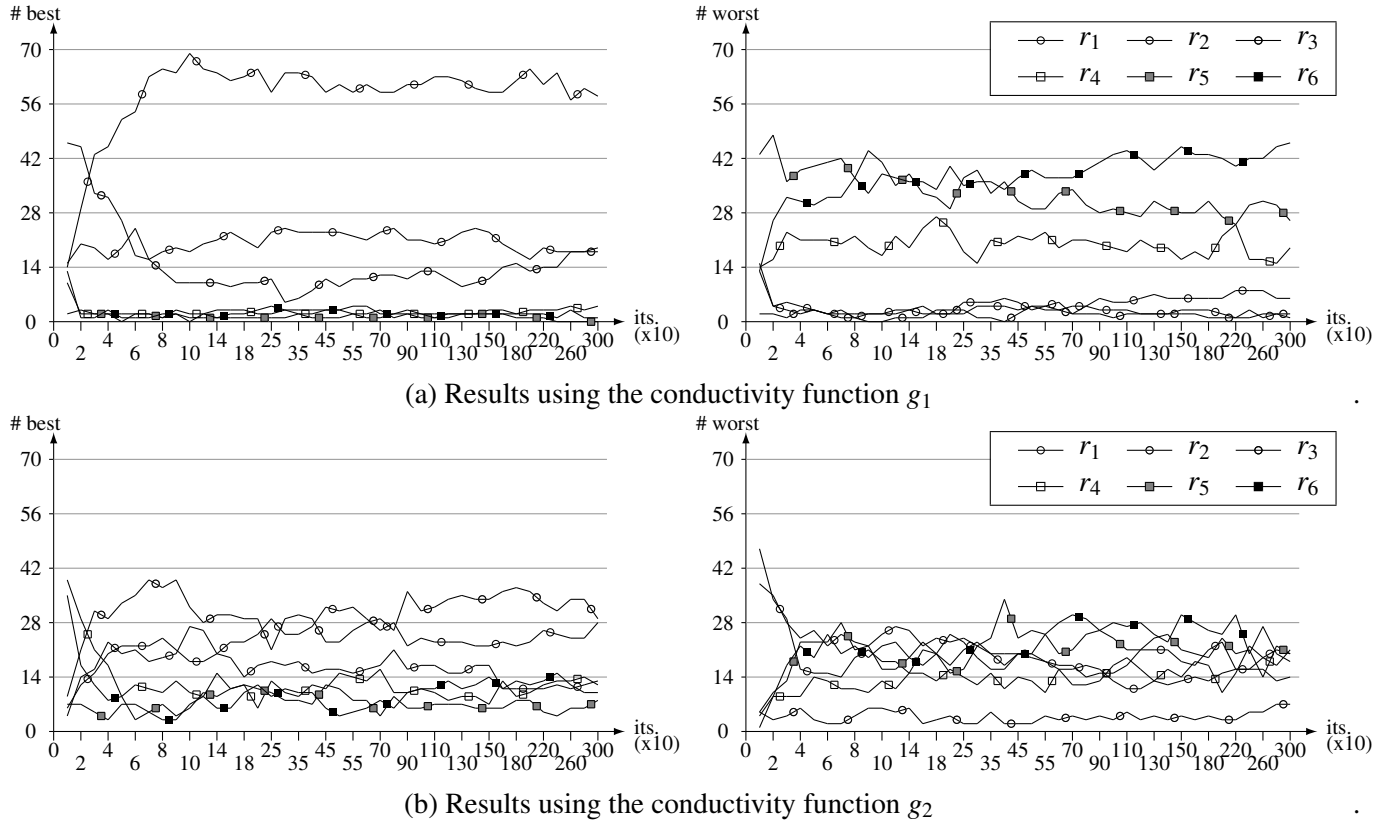


Figure 10: Number of images in the test set of the BSDS for which the results produced by each RDF  $r$  are the best and the worst among the candidate RDFs.

$\varphi_1$ , as we anticipated in Section 3.3. The RDFs based on  $\varphi_1(x) = x$  (i.e.  $\alpha = 1$ ) obtain the best average results, always outperforming those based on  $\varphi_2(x) = \sqrt{x}$ . There are slight differences in performance depending upon  $\varphi_2$ , but they are not as significant as those generated by the selection of  $\varphi_1$ . Regarding the comparison between conductivity functions, we observe that both perform in a similar way. As expected,  $g_2$  produces a faster improvement of the results, due to the larger amount of diffusion (see Figs. 3 and 4), which leads to a faster removal of noise and textures. Note also that the combinations of the most conservative conductivity function ( $g_1$ ) and the most conservative conductivity coefficients (those based on  $r_4$ - $r_6$ ) do not produce good results, since the reduction of  $\Delta^2$  is relatively slow.

In Fig. 10 we display the number of images for which each RDF  $r$  is the best and the worst performer, after a fixed number of iterations. The results using  $g_1$  are included in Fig. 10(a), while those with  $g_2$  are in Fig. 10(b). When using  $g_1$  we observe clear differences between  $r_1 - r_3$  and  $r_4 - r_6$ . It is remarkable that  $r_1$  accounts for a large portion of the best results, although the average performance is not much better than that of  $r_2$  or  $r_3$ . When using  $g_2$ , the differences in performance are much smaller, as seen in Fig. 9(b). It is noteworthy how GPMAD- $r_2$ , despite having a similar average  $\Delta^2$  to that of GPMAD- $r_1$  or GPMAD- $r_3$ , is a much safer choice, in the sense that it almost never produces the worst possible result, regardless of the conductivity function  $g$ .

We observe that the incorporation of different RDFs into PMAD does not result in an increase of the performance on average. However, it can result in a clear improvement when the conditions of the images fit the characteristics of the RDF. From these results we conclude that (a) any RDF  $r$  can perform well when the conditions of the image fit its interpretation of the intensity dissimilarities and (b) GPMAD- $r_2$  is the safest choice, in the sense that it yields acceptable average performance and hardly ever produces very bad results. We recommend to select the specific RDF based upon the characteristics of the images in which the diffusion is going to take place. However, in case no information is avail-

able about the image, GPMAD- $r_2$  (which is equivalent to PMAD) is the best option.

## 5. Conclusions

In this work, we have introduced GPMAD- $r$ , a generalization of PMAD. Our proposal consists of replacing the absolute difference by an RDF to measure the dissimilarity of the intensity two contiguous pixels, which determines the amount of diffusion that takes place at each position of the image. We have studied the behaviour of different RDFs and, using visual examples, we have shown that the success of a diffusion process depends upon the selection of an adequate RDF. Finally, we have carried out a quantitative analysis of the improvement an edge detection method can experience by using our method with different RDFs.

The use of RDFs provides more flexibility in the interpretation of the difference between pixels, offering the possibility to adapt AD to the conditions of an image, and consequently leading to better results from a qualitative and quantitative point of view. Moreover, GPMAD- $r$  is simple enough to establish a clear connection between the RDFs and the final results. Ideally, we would be able to automatically adapt the choice (or even the creation) of the RDF to the specific conditions of an image. This process could be semi-supervised, using tagged regions provided by a human expert, or fully unsupervised, using noise and texture estimators. For this reason, in future works we aim to study the automation of the RDF selection depending upon the characteristics of each individual image, or even depending on the region where the RDF is applied.

## Acknowledgement

This work was partially funded by the Spanish Ministry of Science, project TIN2010-15055, and by the Research Services of the Universidad Publica de Navarra.

## Bibliography

1. C.-M. Pun and W.-Y. Ho, "An edge-based Macao license plate recognition system," *International Journal*

- of *Computational Intelligence Systems*, vol. 4, no. 2, pp. 244–254, 2011.
2. J. Babaud, A. P. Witkin, M. Baudin, and R. O. Duda, “Uniqueness of the Gaussian kernel for scale-space filtering,” *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 8, no. 1, pp. 26–33, 1986.
3. J. Canny, “Finding edges and lines in images,” Massachusetts Institute of Technology, Cambridge, MA, USA, Tech. Rep., 1983.
4. —, “A computational approach to edge detection,” *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 8, no. 6, pp. 679–698, 1986.
5. T. Lindeberg, “Edge detection and ridge detection with automatic scale selection,” *International Journal of Computer Vision*, vol. 30, no. 2, pp. 117–156, 1998.
6. A. P. Witkin, “Scale-space filtering,” in *Proc. of the International Joint Conference on Artificial Intelligence*, vol. 2, Karlsruhe, 1983, pp. 1019–1022.
7. A. L. Yuille and T. A. Poggio, “Scaling theorems for zero crossings,” *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 8, pp. 15–25, 1986.
8. L. Florack and A. Kuijper, “The topological structure of scale-space images,” *Journal of Mathematical Imaging and Vision*, vol. 12, pp. 65–79, 2000.
9. P. Perona and J. Malik, “Scale-space and edge detection using anisotropic diffusion,” *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 12, no. 7, pp. 629–639, 1990.
10. F. Catté, P.-L. Lions, J.-M. Morel, and T. Coll, “Image selective smoothing and edge detection by nonlinear diffusion,” *SIAM Journal on Numerical Analysis*, vol. 29, pp. 182–193, 1992.
11. J. Song and H. R. Tizhoosh, “Fuzzy anisotropic diffusion based on edge detection,” *Journal of Intelligent and Fuzzy Systems*, vol. 17, no. 5, pp. 431–442, 2006.
12. J. Weickert, “Coherence-enhancing diffusion filtering,” *International Journal of Computer Vision*, vol. 31, no. 2-3, pp. 111–127, 1999.
13. Y. Yu and S. Acton, “Speckle reducing anisotropic diffusion,” *IEEE Trans. on Image Processing*, vol. 11, no. 11, pp. 1260–1270, 2002.
14. K. Krissian, C.-F. Westin, R. Kikinis, and K. Vosburgh, “Oriented speckle reducing anisotropic diffusion,” *IEEE Trans. on Image Processing*, vol. 16, no. 5, pp. 1412–1424, 2007.
15. H. Bustince, E. Barrenechea, and M. Pagola, “Relationship between restricted dissimilarity functions, restricted equivalence functions and normal EN-functions: Image thresholding invariant,” *Pattern Recognition Letters*, vol. 29, no. 4, pp. 525–536, 2008.
16. M. Galar, J. Fernandez, G. Beliakov, and H. Bustince, “Interval-valued fuzzy sets applied to stereo matching of color images,” *IEEE Trans. on Image Processing*, vol. 20, no. 7, pp. 1949–1961, 2011.
17. J. M. Merigó and M. Casanovas, “A new Minkowski distance based on induced aggregation operators,” *International Journal of Computational Intelligence Systems*, vol. 4, no. 2, pp. 123–133, 2011.
18. V. Torre and T. Poggio, “On edge detection,” *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 8, no. 2, pp. 147–163, 1984.
19. R. T. Whitaker and S. M. Pizer, “A multi-scale approach to nonuniform diffusion,” *CVGIP: Image Understanding*, vol. 57, no. 1, pp. 99–110, 1993.
20. M. Black, G. Sapiro, D. Marimont, and D. Heeger, “Robust anisotropic diffusion,” *IEEE Trans. on Image Processing*, vol. 7, no. 3, pp. 421–432, 1998.
21. M. Nitzberg and T. Shiota, “Nonlinear image filtering with edge and corner enhancement,” *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 14, no. 8, pp. 826–833, 1992.
22. A. Buades, B. Coll, and J.-M. Morel, “The staircasing effect in neighborhood filters and its solution,” *IEEE Trans. on Image Processing*, vol. 15, no. 6, pp. 1499–1505, 2006.
23. P.-L. Shui and W.-C. Zhang, “Noise-robust edge detector combining isotropic and anisotropic Gaussian kernels,” *Pattern Recognition*, vol. 45, no. 2, pp. 806–820, 2012.
24. M. Basu, “Gaussian-based edge-detection methods- A survey,” *IEEE Trans. on Systems, Man, and Cybernetics, Part C: Applications and Reviews*, vol. 32, no. 3, pp. 252–260, 2002.
25. A. Rosenfeld and M. Thurston, “Edge and curve detection for visual scene analysis,” *IEEE Trans. on Computers*, vol. 20, no. 5, pp. 562–569, 1971.
26. R. Medina-Carnicer, F. Madrid-Cuevas, A. Carmona-Poyato, and R. Muñoz-Salinas, “On candidates selection for hysteresis thresholds in edge detection,” *Pattern Recognition*, vol. 42, no. 7, pp. 1284–1296, 2009.
27. A. J. Baddeley, “An error metric for binary images,” in *Robust Computer Vision: Quality of Vision Algorithms*, W. Förstner and S. Ruwiedel, Eds. Karlsruhe: Wichmann Verlag, 1992, pp. 59–78.
28. R. Medina-Carnicer and F. Madrid-Cuevas, “Unimodal thresholding for edge detection,” *Pattern Recognition*, vol. 41, no. 7, pp. 2337–2346, 2008.
29. C. Lopez-Molina, B. De Baets, and H. Bustince, “Generating fuzzy edge images from gradient magnitudes,” *Computer Vision and Image Understanding*, vol. 115, no. 11, pp. 1571–1580, 2011.