Journal of ELECTRICAL ENGINEERING, VOL. 51, NO. 12/s, 2000, 21–25

A MODEL AND NUMERICAL SCHEME FOR PROCESSING OF COLOR IMAGES

Zuzana Krivá — Karol Mikula *

We propose a model for processing of RGB images based on regularized (in the sense of Catté, Lions, Morel and Coll) Perona-Malik nonlinear image selective smoothing equation. The model is represented by a system of nonlinear different equations with a common diffusion coefficient given by a synchronization of the information coming from all three channels. For the numerical solution we adjust a finite volume computational method given by Mikula and Ramarosy ([5]) and propose a coarsening strategy to reduce the number of unknowns in the linear system to be solved at each discrete scale step of the method.

K e y w o r d s: image processing, RGB image, nonlinear partial differential equations, numerical solution, finite volume method, adaptivity, grid coarsening.

2000 *Mathematics Subject Classification*: 35K55, 65P99

1 INTRODUCTION

A greyscale image can be modelled as a real function $u_0(x)$ representing values of greylevel intensity, defined in some rectangular subdomain $\Omega \subset I\!R^d$ (in practice $d = 2$ or 3). Applying the evolutionary partial differential equation (PDE) to $u_0(x)$, we can solve many basic tasks of image processing and computer vision. Such an approach is known as *image multiscale analysis* ([1], [3], [6]), since the initial image $u_0(x) = u(0, x)$ is associated with a sequence of images $u(t, x)$, a solution of PDE, depending on an abstract parameter $t > 0$ called the scale.

A RGB image can be viewed as a composition of three greyscale images, representing levels of intensity for red, green and blue colors and thus it corresponds to three real functions $u_i^0(x) = u_i(0, x)$, $i = 1, 2, 3$. These "initial" functions can be associated with solutions $u_i(t, x)$, $t > 0$ applied to each $u_i^0(x)$.

One of the well known examples of PDE in image processing is image selective smoothing represented by "edge enhancing" nonlinear diffusion. In this paper we are dealing with Perona-Malik type ([7]) system of equations, regularized in the sense of Catté, Lions, Morel and Coll ([4]) which we adapt to RGB image. In our model we do not apply Perona-Malik like equation to each channel (which would be the simplest approach) but we synchronize the diffusion in each channel by computing a common diffusion coefficient depending on information coming from all three colors (see also [8], [9], dealing with similar techniques in vector valued diffusion and color image processing).

Thus we propose the following system of nonlinear partial differential equations

$$\partial_t u_i - \nabla \cdot (d \nabla u_i) = 0\,, \quad i = 1, 2, 3 \tag{1.1}$$

in $Q_T \equiv I \times \Omega$, where

$$d = g\Big(\sum_{i=1}^{3} |\nabla G_\sigma * u_i|\Big), \tag{1.2}$$

together with zero Neumann and initial conditions in each channel

$$\partial_\nu u_i = 0\,, \quad i = 1, 2, 3, \quad \text{on } I \times \partial\Omega\,, \tag{1.3}$$

$$u_i(0, \cdot) = u_i^0, \quad i = 1, 2, 3, \quad \text{in } \Omega\,. \tag{1.4}$$

In (1.1)–(1.4), $\Omega \subset I\!R^d$ is a rectangular domain, $I = [0, T]$ is a scaling interval, and

$g(s)$ is a decreasing smooth function,

$$g(0) = 1\,,\ 0 < g(s) \to 0 \text{ for } s \to \infty\,, \tag{1.5}$$

$G_\sigma \in C^\infty(I\!R^d)$ is a smoothing kernel with

$$\int_{I\!R^d} G_\sigma(x)\mathrm{d}x = 1 \quad \text{and} \tag{1.6}$$

$G_\sigma(x) \to \delta_x$ for $\sigma \to 0$, δ_x — Dirac function at point x,

$$u_i^0 \in L^2(\Omega), \quad i = 1, 2, 3\,. \tag{1.7}$$

The basic idea of Perona and Malik consists in controlling diffusion (smoothing) of the image by the shape of the diffusion coefficient in nonlinear parabolic equation by means of its dependence on ∇u which is in a sense an edge indicator. Catté, Lions, Morel and Coll considered $\nabla G_\sigma * u$, the *Gaussian gradient*, for decision where there is un-spurious/spurious edge.

In the case of (1.1)–(1.4), if an un-spurious edge is presented in all three channels, g returns a smaller value

* Department of Mathematics and Descriptive Geometry, Faculty of Civil engineering STU, Radlinského 11, 813 68 Bratislava 1
E-mail: kriva@vox.svf.stuba.sk, mikula@vox.svf.stuba.sk

ISSN 1335-3632

than in a case when the channels are processed independently and thus the edge is better preserved. If noise is present in only one of the channels, the model works in the same way as for the greyscale image. If the noise is present in all three channels at the same time, smoothing can be slower at the beginning but with the increasing scale the difference diminishes.

For the numerical solution of (1.1)–(1.4), we adjust a technique suggested and analysed in [5]. It is based on a semi-implicit discretization in scale and on the so-called finite volume method in space. Recently, the finite volume method (FVM) has been widely used in computational sciences and engineering since it is based on physical principles as conservation laws, it is local and easily implemented. Moreover, in the FVM the discrete approximations of a solution of partial differential equation are considered to be piecewise constant in control volumes (cells) which in image processing corresponds to pixel structure of a discrete image. From the conceptional point of view such an approach seems to be *the most natural for image processing.*

As the solution tends to be more flat with the increasing scale in large regions of the image, we can improve the efficiency of the method using adaptivity because it is not necessary to consider the same fine resolution in the whole spatial domain. This approach reduces the computational effort because coarsening of the computational grid reduces the number of unknowns in the linear system to be solved at the discrete scale step of the method. Since the whole information about the image is contained in the initial grid and there is no spatial movement of the edges, no refinement is needed and we work just with grids, elements of which are obtained by merging of pixels. This process is called *coarsening* in numerical methods for solving PDEs. In this paper, we present a coarsening strategy for rectangular grids, join such strategy with the finite volume method for (1.1)–(1.4) and solve by such a technique our nonlinear selective smoothing system of equations for color images.

The rest of this paper is organized as follows. In Section 2 we deal with solution of (1.1)–(1.4) on a regular grid. Section 3 is devoted to the finite volume method joined with adaptive grid and Section 4 shows some numerical experiments.

R e m a r k 1. If we have some apriori knowledge about the image, e.g. that for all three channels of the original version gradients near edges are of the same sign (do not cancel each other), we can use a simplified model, which enables us to speed up the calculation of the diffusion coefficient

$$d = g\Big(\big|\sum_{i=1}^{3} \nabla G_\sigma * u_i\big|\Big). \tag{1.8}$$

2 FINITE VOLUME SCHEME ON A UNIFORM GRID

In this chapter we introduce the finite volume computational method for solving (1.1)–(1.4) on a nonadaptive regular grid. The finite volume scheme for Perona-Malik equation given in [5] is generalized to the system (1.1)–(1.4). Instead of applying the original scheme to every color channel straightforwardly with diffusion coefficients different for each channel, we take into account information from all color channels and by summing the gradients (i.e. expressions $\nabla G_\sigma * u_i$) in a way described later we obtain a diffusion coefficient which is common for all three channels.

Let τ_h be a uniform mesh of Ω with cells p of measure $m(p)$ (we assume rectangular cells here). For every cell p we consider set of neighbours $N(p)$ consisting of all cells $q \in \tau_h$ for which common interface of p and q, denoted by e_{pq}, is of non-zero measure $m(e_{pq})$.

In the numerical scheme we will provide computations in the series of discrete scale steps starting with $\overline{u}^0_{i_p}$, $p \in \tau_h$, corresponding to given intensities on the pixel structure of the initial discrete image. We assume

$$\overline{u}^0_{i_p} = \frac{1}{m(p)} \int_p u_i^0(x)\mathrm{d}x\,, \quad p \in \tau_h\,, \tag{2.9}$$

i.e, the discrete image intensity represents the average cell value of the continuous intensity function $u_i^0(x)$. In the FVM, in every subsequent discrete scale step we get again a piecewise constant approximation $\overline{u}^n_{i_p}$, $p \in \tau_h$, $n = 1, 2, \dots$ of the continuous solution (with the same interpretation as cell averages). Convergence of such approximations to a weak solution of (1.1)-(1.4) for the greyscale image, provided the length of the discrete scale step and the size of the pixel tends to zero, is given in [5]. In [5], it is assumed that for every p, there exists a representative point $x_p \in p$ such that for every pair p, q, $q \in N(p)$, the vector $\frac{x_q - x_p}{|x_q - x_p|}$ is equal to unit vector n_{pq} which is normal to e_{pq} and oriented from p to q (Let us note that this assumption is not fulfilled for adaptive grids given by the coarsening algorithm). In the simple case of a uniform grid we can take x_p just as center of the pixel. Let x_{pq} be the point of e_{pq} intersecting the segment $\overline{x_p x_q}$. Then we define coefficients

$$T_{pq} := \frac{m(e_{pq})}{|x_q - x_p|} \tag{2.10}$$

and

$$g^{\sigma,n}_{pq} := g\Big(\sum_{i=1}^{3} \big|\nabla G_\sigma * \tilde{u}_i\,(x_{pq})\big|\Big) \tag{2.11}$$

where $\tilde{u}_i$ is a periodic extension of discrete color channel computed in n-th scale step. The finite volume scheme on uniform grid is then written as follows:

Let $0 = t_0 \le t_1 \le \cdots \le t_{N_{\max}} = T$ denote the scale discretization steps with $t_n = t_{n-1} + k$, where k is a discrete scale step. For $i = 1, 2, 3$ and $n = 0, \dots, N_{\max} - 1$

we look for $\overline{u}_{i_p}^{n+1}$, $p \in \tau_h$, *satisfying the system of linear equations*

$$\Big(\frac{m(p)}{k} + \sum_{q\in N(p)} g_{pq}^{\sigma,n} T_{pq}\Big)\overline{u}_{i_p}^{n+1} - \sum_{q\in N(p)} g_{pq}^{\sigma,n} T_{pq} \overline{u}_{i_q}^{n+1} = \frac{m(p)}{k}\overline{u}_{i_p}^{n}. \quad (2.12)$$

This scheme is linear **semi-implicit** in scale, since the scale derivative is replaced by the backward difference and nonlinear terms of equation (1.1) are treated from the previous scale step while the linear terms are discretized on the current scale level. After such scale discretization, (2.12) is derived by integrating the corresponding elliptic equation over the cell, applying the divergence theorem and approximating the normal derivative on the boundary of the cell by $\frac{\overline{u}_{i_q} - \overline{u}_{i_p}}{|x_q - x_p|}$.

In the scheme (2.12) we must compute the term (2.11), i.e. the vector

$$\nabla G_\sigma * \tilde{u}_i\,(x_{pq}) = \Big(\frac{\partial (G_\sigma * \tilde{u}_i)}{\partial x}(x_{pq}), \frac{\partial (G_\sigma * \tilde{u}_i)}{\partial y}(x_{pq})\Big),$$

which is an input of the Perona-Malik function g. For that goal, we use the following property of the convolution

$$\frac{\partial (G_\sigma * \tilde{u})}{\partial x}(x_{pq}) = \Big(\frac{\partial G_\sigma}{\partial x} * \tilde{u}\Big)(x_{pq})\,.$$

Then one gets

$$\nabla G_\sigma * \tilde{u}_i(x_{pq}) = \Big(\frac{\partial G_\sigma}{\partial x} * \tilde{u}_i(x_{pq}), \frac{\partial G_\sigma}{\partial y} * \tilde{u}_i(x_{pq})\Big)$$
$$= \Big(\sum_r \overline{u}_{i_r}^n \int_r \frac{\partial G_\sigma}{\partial x}(x_{pq} - s)\mathrm{d}s, \sum_r \overline{u}_{i_r}^n \int_r \frac{\partial G_\sigma}{\partial y}(x_{pq} - s)\mathrm{d}s\Big).$$

Using (1.8)

$$\Big(\frac{\partial G_\sigma}{\partial x} * \sum_{i=1}^{3} \tilde{u}_i\Big)(x_{pq}) = \int_{\mathbb{R}^d} \frac{\partial G_\sigma}{\partial x}(x_{pq} - s) \sum_{i=1}^{3} \tilde{u}_i(s)\mathrm{d}s$$
$$= \sum_r \sum_{i=1}^{3} \overline{u}_{i_r}^n \int_r \frac{\partial G_\sigma}{\partial x}(x_{pq} - s)\mathrm{d}s\,. \quad (2.13)$$

Now the sum is restricted to the control volumes r inside $B_\sigma(x_{pq})$, the ball centered at x_{pq} with radius σ. The ball B_σ is given either by a support of the compactly supported smoothing kernel or it can represent a "numerical support" (a domain in which values of a function are above some threshold given e.g. by a computer precision) of the Gauss function. In any case, just a finite sum in (2.13) is evaluated and coefficients of this sum, namely $\int_r \nabla G_\sigma(x_{pq} - s)\mathrm{d}s$ can be precomputed in advance using a computer algebra system, e.g. Mathematica. Moreover, we can see that computing of the diffusion coefficients is significantly faster in the synchronized model because they are computed only once and this is particularly desirable, when we work with σ covering several pixels because it considerably reduces number of multiplication operations. Further advantages of the model will be demonstrated in the section dealing with numerical experiments.

3 FINITE VOLUME SCHEME ON THE ADAPTIVE GRID

The initial image is given as a set of discrete grey values on pixels of a uniform grid. At the beginning, and especially with the increasing scale, we can merge cells using some coarsening criterion and instead on the regular grid we can work on the irregular adaptive structure. For its construction we chose an approach based on quadtrees. Moreover, to simplify creating the matrix of the linear system we require that the ratio of sides of two neighboring squares be $1:1$, $1:2$ or $2:1$. We will call such a structure *balanced*.

Without lost of generality, let us have an image with $2^n \times 2^n$ pixels. The idea of constructing the quadtree is the following. We divide the original image into four quadrants and test if the coarsening criterion is fulfilled. We consider the following coarsening criterion:

The cells are merged if the difference in intensities is bellow a prescribed tolerance ε *in every color channel.*

If the cells in the quadrant can not be merged, we continue in recursive subdivision of the quadrant in four new ones. At the end, we either have quadrants with pixels merged due to the coarsening criterion (new intensity value of merged cells is set to pixels' average) or quadrants of the size 1×1, i.e. pixels of the original image. These quadrants represent the cells of the adaptive grid which is common for all three channels.

As we have already noticed, it is not possible to apply the previous scheme (2.12) straightforwardly to an adaptive non-uniform grid obtained by the coarsening algorithm. However, it is possible to modify it. For that goal, we will change the meaning of x_{pq} in (2.11) and the definition (2.10) of T_{pq}. Let in the sequel x_{pq} be the middle point of the common boundary of two neighboring cells (with possibly non-equal measures). The definition of $g_{pq}^{\sigma,n}$ will remain the same. The only practical difference will be that the sum (2.13) will be evaluated over non-equal control volumes. However, one can precompute all possible coefficients of the sum again in advance for every candidate larger cell on the higher level of hierarchy. In the definition of T_{pq} in (2.10), the value $|x_p - x_q|$ represents the distance used for approximation $\frac{\overline{u}_{i_q} - \overline{u}_{i_p}}{|x_q - x_p|}$ of the normal derivative. Of course, in the case of uniform rectangular grid with unite size of cells, T_{pq} is equal 1. In the case of non-uniform rectangular grids, we can adjust this parameter in several ways [10]. Here we will consider the following one

$$T_{pq} = \min\{l_p, l_q\} \quad (3.14)$$

where l_p and l_q are the lengths of sides of two adjacent cells p, q (of possibly non-equal measure). It is like we

Fig. 1. Image enhancement by the synchronized model (1.1)–(1.4) (see Example 1)

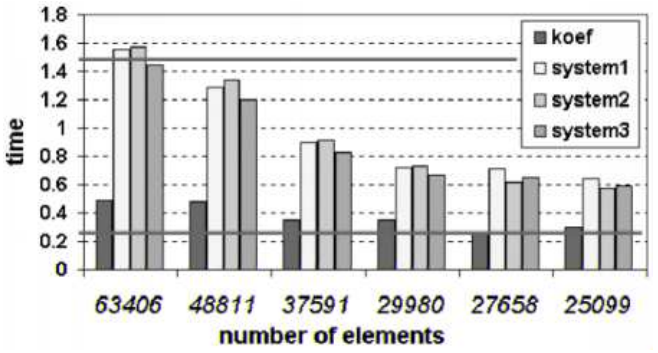

Fig. 2. Graph comparing adaptive and nonadaptive algorithms.

assume an exchange of intensity between neighbouring cells just in a strip of unit thickness along boundary of cell. This adjustment can be used for any grid but in the case of a balanced grid, we can observe the following: if the coarsening process creates a uniform grid with cells of size l, the scheme works like (2.12) but with the scale step enlarged l times and thus diffusion is "faster" than in the original nonadaptive scheme. To make computing of coefficients faster, we can use the fact that T_{pq} is always equal to 1 or $\frac{1}{2}$ for the balanced grid.

Finally, as our *adaptive finite volume scheme* we will consider system (2.12) where x_{pq} represents the middle point of the common boundary of two neighboring cells and T_{pq} is given by (3.14). In every discrete scale step, the scheme gives a linear system which is symmetric and strictly diagonally dominant (with a positive diagonal and negative numbers out of the diagonal) which guarantee the existence of its unique solution and for which also L_∞ stability can be easily proved.

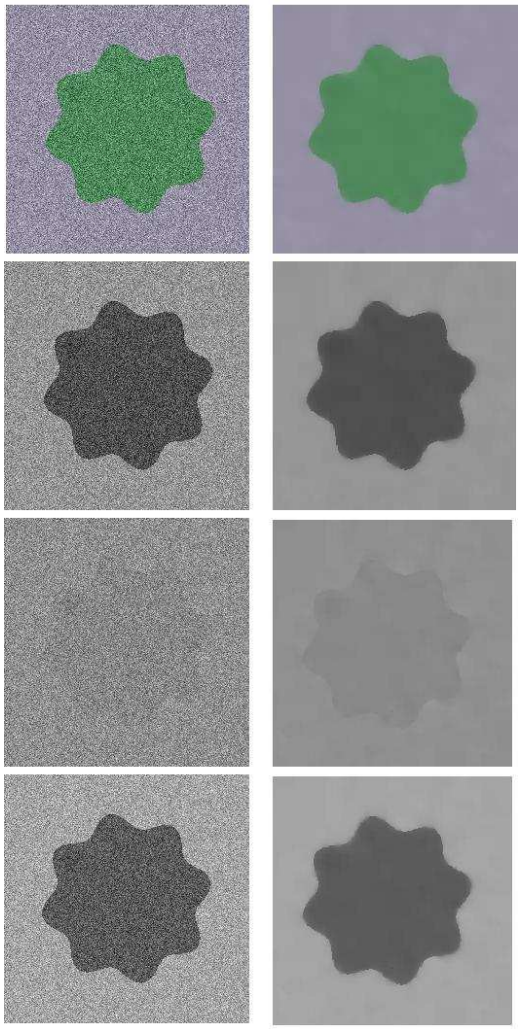

Fig. 3. Denoising of an artificial color image.

4 NUMERICAL EXPERIMENTS

In this section we present experiments with some real as well as artificial images perturbed by various types of noise. In simulations, we use the function

$$g(s) = \frac{1}{1 + Ks^2}$$

with $K > 0$ and the convolution is realized with the kernel

$$G_\sigma(x) = \frac{1}{Z} e^{\frac{|x|^2}{|x|^2 - \sigma^2}},$$

where the constant Z is chosen so that G_σ has a unit mass. In order to compute the diffusion coefficient $g_{pq}^{\sigma,n}$ we use the concept given in (2.13). In numerical experiments we have chosen $\sigma = \frac{1}{2}$, i.e. half size of the cell on the finest level, which is the fastest and simplest approach. Figures and a graph document the results of multiscale analysis (iterative filtering) as well as adaptive computational grids. All experiments were done on AMD K6-2 (266 MHz) with linux operating system.

Fig. 4. Removing of "moiré" from the color image.

Example 1. This numerical experiment demonstrates the feature of our model mentioned in the introduction — better preserving of edges. In the top of Fig. 1, on the left we show the original noisy image and on the right the image obtained by independent smoothing of the channels. Bellow, on the left there is an image obtained by our model using a nonadaptive scheme and on the right we show a visually very similar adaptive version with a corresponding adaptive grid. It is clear that the synchronized model denoises and enhances the image successfully.

The graph in Fig. 2 compares computational times in discrete scale steps of the nonadaptive and adaptive algorithms. The legend, graph columns and the x-axis inform about 6 scale steps of the adaptive algorithm, the bold horizontal lines show the average time for computation of one linear system (top line) and for computation of coefficients (bottom line) in one scale step of the nonadaptive algorithm. The initial number of elements is 65536, the threshold value for coarsening criterion has been set to 0.025, $K = 10$.

Example 2. In the example documented in Fig. 3 with the size 256×256 we process artificial image pixels. Every channel of the original image is a double-valued image $\hat{u}_i(x)$ with intensity difference set to 70.7 and 75 for particular color channels. The initial functions $u_{0_i}(x)$, representing noisy channels of the color image, are given by

$$u_{0_i}(x) = MIN(255, MAX(0, \hat{u}_i(x) - C + \psi)/255\,.$$

where ψ is a random function generating integer values in $[0, 2C]$, in our example $C = 50$. In the top of Figure 3 the color noisy image and its smoothed version are plotted. Bellow there are red, green and blue noisy and processed channels. The purpose of this example is to demonstrate, that although the green channel is almost lost for individual smoothing, the synchronized diffusion allows to recover it thanks to the information from the other channels. The example shows the results of the adaptive algorithm after 10 scale steps. We have chosen $\varepsilon = 0.025$, $K = 10$. The number of elements after 10th step is 4185. While, at the beginning, one system solving takes about 1.2 seconds, at the end only 0.07 seconds. The achieved total CPU time is 24.27 seconds. The nonadaptive algorithm needs 40.05 seconds for 10 scale steps, but the comparable smoothing results were obtained after 15 steps, with the time 62.96 seconds.

Example 3. The experiment documented in Fig. 4 was performed on the image of the size 512×402 pixels. The picture is a result of scanning and has a significantly damaged blue channel (top of Fig. 4). With the help of the other two channels, which are of much better quality, the synchronized smoothing recovered the blue channel to the form shown by the picture in the bottom of Fig. 4.

References

[1] ALVAREZ, L.—GUICHARD, F.—LIONS, P. L.—MOREL, J. M.,: Axioms and Fundamental Equations of Image Processing, Arch. Rat. Mech. Anal. **123** (1993), 200–257.

[2] ALVAREZ, L.—LIONS, P. L.—MOREL, J. M.: Image Selective Smoothing and Edge Detection by Nonlinear Diffusion II, SIAM J. Numer. Anal. **29** (1992), 845–866.

[3] ALVAREZ, L.—MOREL, J. M.: Formalization and Computational Aspects of Image Analysis, Acta Numerica (1994), 1–59.

[4] CATTÉ, F.—LIONS, P. L.—MOREL, J. M.—COLL, T.: Image Selective Smoothing and Edge Detection by Nonlinear Diffusion, SIAM J.Numer.Anal. **29** (1992), 182–193.

[5] MIKULA, K.—RAMAROSY, N.: Semi-Implicit Finite Volume Scheme for Solving Nonlinear Diffusion Equations in Image Processing, to appear in Numerische Mathematik.

[6] LIONS, P. L.: Axiomatic Derivation of Image Processing Models, Mathematical Models and Methods in Applied Sciences **4** (1994), 467–475.

[7] PERONA, P.—MALIK, J.: Scale Space and Edge Detection Using Anisotropic Diffusion, Proc. IEEE Computer Society Workshop on Computer Vision (1987).

[8] WHITACKER, R.—GERIG, G.: Vector-Valued Diffusion, *in B.M.t.M.Romemy(Ed): Geometry Driven Diffusion in Computer Vision, Kluwer* (1994).

[9] WEICKERT, J.: Coherence-Enhancing Diffusion of Colour Images, Image and Vision Computing **17** (1999), 201–212.

[10] KRIVÁ, Z.—MIKULA, K.: An Adaptive Finite Volume Scheme for Solving Nonlinear Diffusion Equations in Image Processing, *submitted.*

Received 31 May 2000

Zuzana Krivá (RNDr) is a graduate student of applied mathematics at the Faculty of Mathematics and Physics of Comenius University, Bratislava.

Karol Mikula (Doc, RNDr, CSc) is an associate professor at the Department of Mathematics, Slovak University of Technology, Bratislava.