Vladimír Chalupecký; Michal Beneš

Some applications of parabolic differential equations of Allen-Cahn type in image processing

In: Jaromír Kuben and Jaromír Vosmanský (eds.): Equadiff 10, Czechoslovak International Conference on Differential Equations and Their Applications, Prague, August 27-31, 2001, [Part 2] Papers. Masaryk University, Brno, 2002. CD-ROM; a limited number of printed issues has been issued. pp. 83--101.

Persistent URL: http://dml.cz/dmlcz/700339

Equadiff 10, August 27–31, 2001
Prague, Czech Republic

Equadiff 10 CD ROM
Papers, pp. 83–101

Some Applications of Parabolic Differential Equations of Allen-Cahn type in Image Processing ⋆

Vladimír Chalupecký[1] and Michal Beneš[2]

[1] Department of Mathematics, Czech Technical University in Prague, Trojanova 13, 120 00 Praha, Czech Republic
Email: chalupecky@km1.fjfi.cvut.cz

[2] Department of Mathematics, Czech Technical University in Prague, Trojanova 13, 120 00 Praha, Czech Republic
Email: benes@km1.fjfi.cvut.cz

Abstract. In this contribution, we present several algorithms for image processing based on some modifications of the Allen-Cahn equation. These algorithms include noise removal, pattern recovery and shape recovery and are motivated by similar models based on the level set formulation of motion by mean curvature. The equations are solved by semi-implicit finite difference method. Also, results of some numerical experiments are included.

MSC 2000. 35K55, 35K65, 65M06, 68U10

Keywords. Image processing, finite difference method, mean curvature, Allen-Cahn equation

1 Introduction

In this contribution, we will present some algorithms for image processing by means of modifications of the Allen-Cahn equation, which is closely connected with notion of mean curvature motion.

⋆ This work has been partly supported by grants of the Grant Agency of Czech Republic number 201/01/0676 and of Ministry of Education of Czech Republic number MSM 98/210000010.

This is an overview article summarizing the degree thesis of the first author.

The family of curves $\Gamma^t : S \to \mathbb{R}^2, t \in (0,T), S \subset \mathbb{R}$, undergoes motion in the normal direction and the velocity is a function of curvature κ. The isotropic motion law then is

$$v_\Gamma = -\kappa + F. \tag{1}$$

Equations of this type have recieved a lot of attention during recent years, both from the theoretical and practical point of view, as there is a wide range of possible applications. They can be solved by directly discretizing the parameterized curve and these methods are then called *direct*.

Another approach are indirect methods, where the evolving curve is a level set of a higher dimensional function, that is the solution of a partial differential equation. *Level set methods* are based on a degenerate parabolic equation, that can be directly derived from the equation (1). Level sets of the solution of this equation move in the normal direction with speed a function of curvature, as it was introduced in the work of Osher and Sethian [20]. In order to solve this equation numerically, some kind of regularization needs to be introduced.

The Allen-Cahn equation [3] can be considered as another possible regularization of level-set methods. It originates from the theory of phase transitions [3,4,5,6,7,8], where the evolving curve represents an interface separating solid and liquid phases. The dependence of normal velocity of this interface is given by surface tension effects. With level set methods, singularities such as corners, splitting and merging of curves seem to be handled automatically, which is a great advantage. However, from the computational point of view, the level set methods are more computationally expensive than numerical methods based on the direct approach due to the discretization of a 2D domain.

The range of applications of the Allen-Cahn in image processing includes among other noise removal, image segmentation, shape recovery and morphing. The goal of this work is to investigate properties of this equation within the context of image processing. Several models for image processing based on the Allen-Cahn equation will be shown followed by the description of numerical approximations of these models. Results of numerical experiments will be demonstrated on some artificial and real images.

This work is organised as follows. In the Section 2, we will shortly describe the Allen-Cahn equation and its different forms. In the Section 3, a number of possible applications of the equation in some of the tasks in image processing will be described. The Section 4 is devoted to numerical schemes for solving the Allen-Cahn equation based on semi-implicit finite difference methods. Some remarks on used iterative solvers together with results of several numerical experiments will be presented in the Section 5.

2 Allen-Cahn equation

The Allen-Cahn equation is a well-known reaction-diffusion equation, that originates from the theory of phase transitions (see [3,4,5,6,7,8] and references therein). It gives rise to a sharp interface between two domains Ω_0^t and Ω_1^t (both depending

on time), that moves by its mean curvature according to the law (1). The general form of this equation with zero Neumann boundary condition is following

$$\xi^2 \frac{\partial p}{\partial t} = \xi^2 \Delta p + f(p;\xi), \quad x \in \Omega, \quad t > 0, \tag{2}$$

$$\frac{\partial p}{\partial n}\Big|_{\partial\Omega} = 0, \tag{3}$$

$$p|_{t=0} = p_0, \tag{4}$$

where ξ is a given constant and n denotes the outer normal to the domain Ω. The term ξ plays a special role in the model. It determines the width of the interface domain between the domains Ω_0^t and Ω_1^t, which is of order ξ. For $\xi \to 0_+$, the motion law for this interface converges to the law (1). Each form of this equation then differs in the choice of the function $f(p;\xi)$ and three possible choices will be described now.

Model 1 This is the original form of the function $f(p;\xi)$ and it has the following form [8]

$$f(p;\xi) = ap(1-p)(p-0.5) + \xi F, \tag{5}$$

where a and F are given constants.

However, this model has certain limitations. The behaviour of the solution is given by three zeros of the polynom $f(p;\xi)$. For certain values of ξ and F this polynom looses these three zeros and thus the correct behaviour (see the figure 2).

Another limitation is that the range of the solution is not $\langle 0, 1\rangle$.

Model 2 Another form of the function $f(p;\xi)$, that was proposed in [18], is

$$f(p;\xi) = ap(1-p)(p-0.5+\xi F). \tag{6}$$

The solution of the Allen-Cahn equation with this form of $f(p;\xi)$ stays between 0 and 1 for all times. However, the constants ξ and F must be chosen so that the term $p - 0.5 + \xi F$ has a zero between 0 and 1.

Model 3 Better properties has the Allen-Cahn equation with the following form of the function $f(p;\xi)$, which was proposed in [4,5]

$$f(p;\xi) = ap(1-p)(p-0.5) + \xi^2 F \cdot |\nabla p|. \tag{7}$$

This form is motivated by the level set formulation of the motion law (1). The solution of this equation should stay within $\langle 0, 1\rangle$ for all t and it approximates the motion law (1) more accurately. Comparison of this model with the corresponding sharp interface law was performed in [7]. In the subsequent sections, the Allen-Cahn equation solely in this form will be used.

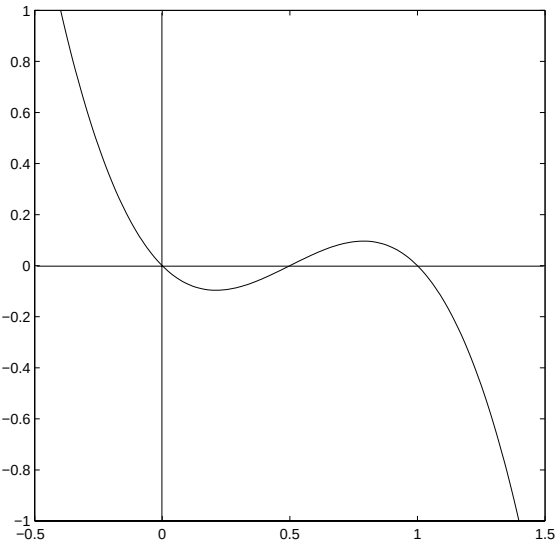

Fig. 1. Function $f_0(p) = 2p(1-p)(p-0.5)$

3 Allen-Cahn equation in image processing

In this section, we describe applications of the Allen-Cahn equation in a stage of image processing that is usually called *image pre-processing*. The aim of image pre-processing is to improve an image and enhance features of objects in it, so that any further analysis, such as object detection and recognition, is more reliable or even possible. A typical field of human activity, that uses tools of image processing extensively, is medicine. Visual data from methods like echocardiography or MRI are often disturbed by noise and some image enhacement is thus necessary. Noise removal therefore makes very important part of image pre-processing, but not a sole one.

Another important task encountered in image processing is segmentation. The goal of image segmentation is to divide the rectangular domain of the image into finitely many regions in which a certain property has constant value. This property can be value of intensity function or a specific high-frequency pattern, often called a *texture*. Boundaries of these regions are *edges*, along which the intensity gradient is by definition "large". We will address this problem in the Sections 3.1 and 3.2.

In the Section 3.3, we will describe how the Allen-Cahn equation can be applied in image morphing. By image morphing we mean a task with two given images, the initial image and the final image, and we seek a continous transformation between these two images. Apparently, there are infinitely many such transformations and it is up to the user to choose the appropriate one. Our model performs transformation, in which the level sets of the solution move by mean curvature.

In the following we suppose, that the input image is a real function $p_0(x, y)$

$$p_0 : \Omega \to \langle 0, 1 \rangle,$$

representing intensity of a gray-scale image. $\Omega \subset \mathbb{R}^2$ represents a spatial domain (rectangular in practice). The multiscale image analysis (as introduced in [1]), associates with $p_0(x, y)$ a family $p(x, y, t)$ of transformed images depending on an abstract parameter t, that is called a *scale*. As it has been proved in [1], under basic assumptions the family can be represented as a solution of a second order parabolic partial differential equation

$$\frac{\partial p}{\partial t} = F(p, \nabla p, \nabla^2 p, t) \tag{8}$$

with the initial condition

$$p(x, y, 0) = p_0(x, y). \tag{9}$$

In our case, the equation (8) will be represented by the Allen-Cahn equation. For other possible forms of equation (8) representing multiscale analysis see [1,2].

3.1 Noise removal

In this Section, when referring to the Allen-Cahn equation, we consider the following form

$$\begin{aligned} \xi \frac{\partial p}{\partial t} &= \xi \Delta p + \frac{1}{\xi} a p(1-p)(p-0.5) + \xi F \cdot |\nabla p|, \\ \frac{\partial p}{\partial n}\Big|_{\partial\Omega} &= 0, \\ p|_{t=0} &= p_0. \end{aligned} \tag{10}$$

We already know, that the Alle-Cahn equation has some very special properties. With increasing scale t, the Allen-Cahn equation forms regions, in which the solution is close to 0 or 1, and an interface domain of a small width (determined by the parameter ξ), where the solution changes rapidly. This interface moves in the normal direction at a speed that is proportional to its mean curvature. We can use these properties in image processing as follows.

Noise in an image is a disturbation, whose level sets are curves with high mean curvature. By a level set of a function $p(x, t)$ at level c we mean the following set

$$\mathcal{L} = \{x \in \mathbb{R}^2 \,|\, p(x) = c\}.$$

Thus, if we were able to move these level sets by their mean curvature, the features in an image with high curvature (such as noise) would rapidly shrink to a point, while at the same time important features with lower mean curvature would change only a little. From the properties of the Allen-Cahn equation follows, that this can be succesfully accomplished by applying this equation on the noisy image as an

initial condition. Moreover, we are able to control, which features in the image will be kept by means of the forcing term F. If we look at the motion law (1)

$$v_\Gamma = -\kappa + F,$$

we can see, that if the term F is equal to mean curvature, the normal velocity is zero. This means that such an object in the image will be left unchanged, while objects with higher curvature will shrink and objects with lower curvature grow.

More precisely, let us consider a simple example with a circle. A circle has curvature equal to $1/r$, where r is its radius. Thus, if we put $F = 1/r$, this circle will remain unchanged during the whole evolution of the equation. For $F > 1/r$, the circle will shrink and vice versa. With $F = 0$, the radius of the circle with initial radius r_0 at time t is given by

$$r(t) = \sqrt{{r_0}^2 - 2t}$$

The process of shrinking of a circle can be seen at Figure (3.1). The previous

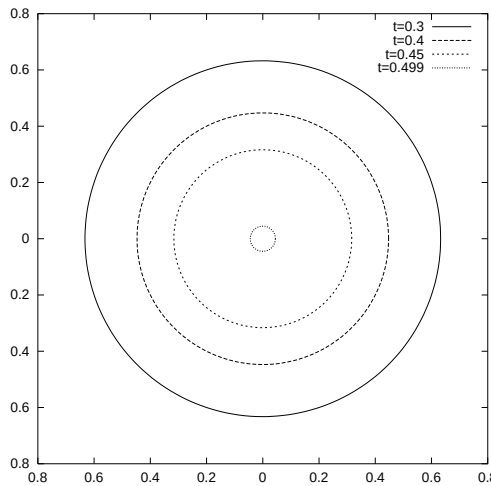

Fig. 2. Shrinking of a circle with $r_0 = 1$ by its mean curvature.

relation implies a certain disadvantage. As the system evolves, not only the noise is removed, but also other objects in the image, that might be important. In time, every object in the image becomes convex due to the influence of mean curvature and then shrinks to a point. In the end, the system gets to the steady state, which in the case of images would be a black or white image, depending on the intial image and value of the parameter F. This implies, that the evolution has to be stopped after a certain time. However, no stopping criterion is known, which means, that the stopping time has to be determined experimentally. Another disadvantage is

implied by the properties of the Allen-Cahn equation itself. Its solution tends to values 0 and 1 for any initial condition and any choice of parameters, thus making the equation unsuitable for processing of gray-scale images. For computational experiments, see Section 5.

3.2 Shape recovery

After removing the noise, the subsequent aim is often detection of objects in the image or recovery of their boundaries. Geometric active contour models turn out to be a very useful tool for edge detection in gray-scale images (e.g. [10,11,12,16,17]). The idea behind these methods is quite simple. Given an initial curve, that encloses all the objects to be recovered, we want to move this curve in the direction of its normal vector field so that it adheres to the edges in the image. Besides edge detection, active contour models have been used for segmentation, shape modelling and visual tracking.

A number of approaches have been proposed for active contour models in the past few years. Caselles et al. [10] and Malladi et al. [19] proposed the following model based on the level set formulation

$$\frac{\partial u}{\partial t} = g(x,y)|\nabla u| \left(\nabla \frac{\nabla u}{|\nabla u|} + \nu \right). \tag{11}$$

The function $g(x,y)$ depends on the image I, in which we want to find boundaries, and is used as a "stopping term". In [10,19], the term

$$g = \frac{1}{1 + \lambda |\nabla G_\sigma * I|^n}$$

was chosen (in [10] $n = 1$, in [19] $n = 2$), where G_σ is the Gaussian function. The level sets of u move in their normal direction with speed a function of curvature as it was introduced in [20]. The initial curve given as a level set of the initial condition u_0 is put inside or outside the object to be recovered. The stopping term g should ensure, that the speed of shrinking of the curve is small near a boundary, thus stopping the curve at the boundary. However, due to the term ν, this model does not guarantee to stop the motion at the boundary and the edge can be crossed.

An improved model has been proposed by Kichenassamy et al. [17]. The model they propose is

$$\frac{\partial u}{\partial t} = g|\nabla u| \left(\nabla \frac{\nabla u}{|\nabla u|} + \nu \right) + \nabla g \cdot \nabla u. \tag{12}$$

Again, the shrinking (or growing) of the edge-seeking curve is an inhomogeneous process. The curve in those parts of the image I, where there is no edge, shrinks due to the term ν. When the curve approaches an edge in the image I, the function g is "small", thus suppressing the influence of curvature. The convection term $\nabla g \cdot \nabla u$ attracts the curve to the edge and its form causes, that the curve will not go past an edge, to which it has adhered (see Figure 3.2).

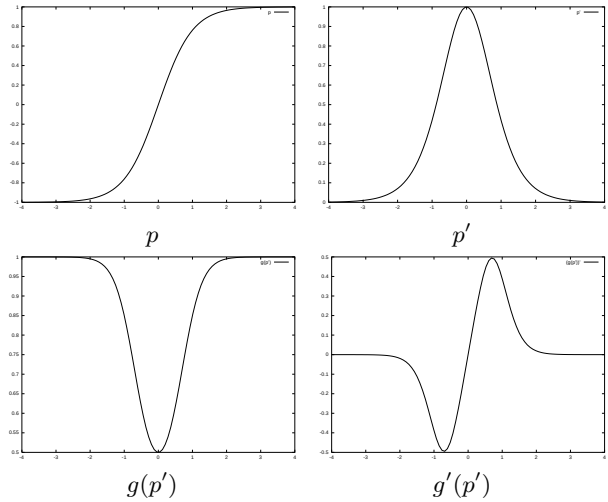

Fig. 3. Explanation of the forcing term in the shape recovery model in one-dimensional case.

Motivated by the just described models, let us now describe our model based on the Allen-Cahn equation

$$\begin{aligned} \xi\frac{\partial p}{\partial t} &= g\big(\xi\Delta p + \frac{1}{\xi}f_0(p)\big) + \xi K, \\ \frac{\partial p}{\partial n}\Big|_{\partial\Omega} &= 0, \\ p|_{t=0} &= p_0, \end{aligned} \tag{13}$$

where

$$K = \begin{cases} F\cdot|\nabla p| & \text{if} \quad \nabla g\cdot\nabla p < \beta \\ \nabla p\cdot\nabla g & \text{if} \quad \nabla g\cdot\nabla p \geq \beta \end{cases} \tag{14}$$

$$g = \frac{1}{1+\lambda|\nabla G_\sigma * I|^2}.$$

We modify the Allen-Cahn equation by introducing the function g and by modifying the forcing term F in the same way as in the level set model so that, as the edge-seeking curve evolves, it is attracted to the boundaries in the image, in which we want to recover boundaries (we have denoted this image as I). We supply the initial edge-seeking curve as a boundary of $C \subset \Omega$, that contains all objects to be recovered. The initial condition for the modified Allen-Cahn equation then is $p_0 = 1 - \chi_C$. Let us explain the terms appearing in this equation in more detail. With the choice for the forcing term given by (14), two cases may arise. The parameter β plays a role of a "switch"; it determines, which form of the forcing term should be used:

1. $\nabla g \cdot \nabla p \geq \beta$ – the edge-seeking curve is given by the level set at 0.5 of the solution p. Then, this case arises, when this curve gets "close" to an edge in the image I. To explain the form of the forcing term in more detail, let us demonstrate the idea in the one-dimensional case as shown at Figure (3.2). As it can be seen from the last figure, on each side of the edge the forcing term has a different sign. As the curve approaches a region with higher gradient in image I, it is attracted by the edge. Moreover, the form of $\nabla g \cdot \nabla p$ in this case does not allow the curve to go past this edge. If it crossed the edge, the opposite sign of $\nabla g \cdot \nabla p$ would guarantee, that the curve would be "dragged" back. From these facts, the role of the function g is apparent – at a boundary with high curvature, we need to suppress its influence. In the other case, the force keeping the curve at the boundary would be smaller than the influence of curvature and the curve would cross the edge. Then, the boundary could not be recovered.
2. $\nabla g \cdot \nabla p < \beta$ – in this case, the edge-seeking curve does not approaches an edge in I and the forcing term has the same form as in the Allen-Cahn equation. This has two reasons. Firstly, if the curve moved only by its mean curvature, the evolution might be too slow. Secondly, during the evolution of the solution, some parts of the curve may become straight. If the term K had the same form as in the previous case, it would be equal or close to zero. Mean curvature in such a part would be zero as well and thus there would not be any force, that would move the curve to the desired boundary. This would result in a situation, when the exact shape of the object could not be recovered. On the other hand, this feature might be sometimes desirable, for instance in the case we wanted to recover a missing part of the boundary.

3.3 Pattern recovery

Pattern recovery in our context essentially differs from shape recovery. In shape recovery, we have one image in which we want to find boundaries of objects in it. In pattern recovery, by means of the Allen-Cahn equation with a modified forcing term we want to obtain a continuous transformation of one image into another. Obviously, there are infinitely many such transformations, but this model chooses the one, in which the level sets move by mean curvature.

The model we propose is

$$\begin{aligned} \xi \frac{\partial p}{\partial t} &= \xi \Delta p + \frac{1}{\xi} f_0(p) + \xi F(p; p_1) \cdot |\nabla p|, \\ \frac{\partial p}{\partial n}\Big|_{\partial\Omega} &= 0, \\ p|_{t=0} &= p_0, \\ F(p; p_1) &= b(p_1 - p), \quad b > 0, \end{aligned} \tag{15}$$

where p_0 denotes the initial image (initial condition), p_1 is the final image and $b > 0$ is a constant. The equation for pattern recovery consists in modifying the

forcing term in the type 3 of the Allen-Cahn equation (2). The evolution is then an inhomogeneous process - the forcing term is modified so that is depends on the solution p and on the final image p_1. With this setting, level sets of the solution move to level sets in the final image in the apropriate direction, depending on whether the value of the solution is greater or less than the value of the final image in a given point. By means of the constant b, we can control speed of the transformation.

This model can be easily extended to processing of color images, the equation can be applied to each color channel separately.

4 Numerical schemes

In this section, numerical schemes for the models presented in the Section 3 will be discussed. These schemes have been used to compute the numerical results presented in the Section 5.

To approximate the parabolic eqution from the Section 3, the finite difference method with the semi-implicit time discretization has been used. The nonlinearity introduced into the models by the polynom $f_0(p)$ was treated from the previous discrete time step, while the linear terms are handled implicitly. In this way, we obtain a linear algebraic system, that can be efficiently solved by some iterative method. By using an explicit scheme, we would have to face the stability problems and by using a fully implicit scheme we would have to solve a system of nonlinear equations.

Notations Let us first introduce notations used throughout this section. Let h and τ be discrete space and time (in our context this variable is called a scale) steps. By Ω we denote the rectangular domain of the image, by n_x and n_y we denote the number of pixels in the direction of x and y axis respectively and by n_T the number of discrete scale steps. Let us denote

$$
\begin{aligned}
p^j = p(\cdot, j\tau), \quad \delta p^j = \frac{p^j - p^{j-1}}{\tau}, \quad p_{k,s} = p(kh, sh),\\
\omega_h = \{[ih, jh] | i = 1, \dots, n_x - 2; j = 1, \dots, n_y - 2\},\\
\bar{\omega}_h = \{[ih, jh] | i = 0, \dots, n_x - 1; j = 0, \dots, n_y - 1\},\\
p_{\bar{x},ks} = \frac{p_{ks} - p_{k-1,s}}{h}, \quad p_{x,ks} = \frac{p_{k+1,s} - p_{ks}}{h},\\
p_{\bar{y},ks} = \frac{p_{ks} - p_{k,s-1}}{h}, \quad p_{y,ks} = \frac{p_{k,s+1} - p_{ks}}{h},\\
p_{\bar{x}x,ks} = \frac{1}{h^2}(p_{k+1,s} - 2p_{ks} + p_{k-1,s}),
\end{aligned}
$$

and

$$
\begin{aligned}
\nabla_h p = [1/2(p_{\bar{x}} + p_x), 1/2(p_{\bar{y}} + p_y)],\\
\Delta_h p = p_{\bar{x}x} + p_{\bar{y}y},
\end{aligned}
$$

as mappings from ω_h to $\mathbb{R}^2$ or $\mathbb{R}$, respectively.

4.1 Allen-Cahn equation

The semi-implicit scheme for the Allen-Cahn equation, used for the numerical experiments, is

$$\xi\delta p_h^j = \xi\Delta_h p_h^j + \frac{1}{\xi} f_0(p_h^{j-1}) + \xi F \cdot |\nabla_h p_h^{j-1}| \quad \text{on} \quad \omega_h, \tag{16}$$

with initial condition

$$p_h^0 = p_0.$$

The solution is a map $p_h : \{1, \ldots, n_T\} \times \bar{\omega}_h \to \mathbb{R}$.

The zero Neumann boundary condition is handled by reflecting the image at the boundaries, for instance at the left boundary of the image holds

$$\frac{p_{1,s}^j - p_{-1,s}^j}{2h} = 0 \qquad \Rightarrow \qquad p_{1,s}^j = p_{-1,s}^j \quad s = 0, \ldots, n_y - 1.$$

If we number the pixels by rows from left to right, we can write the scheme using matrix notation

$$\mathbf{A}p^j = g^{j-1}, \quad \mathbf{A} \in \mathbb{R}^{n,n} \quad j = 1, \ldots, n_T, \tag{17}$$

where $n = n_x \cdot n_y$. In this case, p^j denotes the vector of unknows and g^{j-1} is the right-hand side of the linear algebraic system, made of terms from the previous scale step. The matrix $\mathbf{A}$ is a sparse positive definite matrix with a special penta-diagonal structure. The boundary condition causes, that the matrix is not symmetric. However, if we multiply the rows of (17) corresponding to the pixels at boundaries by $1/2$ and the rows corresponding to the pixels at corners by $1/4$, the matrix $\mathbb{A}$ becomes symmetric, which allows us to use efficient iterative solvers like conjugate gradient method.

4.2 Shape recovery

The semi-implicit scheme for the shape recovery equation is

$$\begin{aligned}
\xi\delta p_h^j &= \xi\Delta_h p_h^j + \frac{1}{\xi} f_0(p_h^{j-1}) + \xi K \quad \text{on} \quad \omega_h, \\
K &= \begin{cases} F \cdot |\nabla_h p_h^j| & \text{if} \quad \nabla_h p_h^j \cdot \nabla_h g_h < \beta \\ \nabla_h p_h^j \cdot \nabla_h g_h & \text{if} \quad \nabla_h p_h^j \cdot \nabla_h g_h \geq \beta \end{cases} \\
g_h &= \frac{1}{1 + \lambda (G_\sigma * I)_h^2},
\end{aligned}$$

with initial condition

$$p_h^0 = p_0.$$

The image, in which we want to recover boundaries, is denoted by I.

4.3 Pattern recovery

The semi-implicit FDM scheme of this model is

$$\xi\delta p_h^j = \xi\Delta_h p_h^j + \frac{1}{\xi} f_0(p_h^{j-1}) + \xi F(p_h^{j-1})|\nabla_h p_h^{j-1}| \quad on \quad \omega_h,$$
$$F(p) = b(p_1 - p), \qquad b > 0,$$

with initial condition

$$p_h^0 = p_0.$$

5 Computational results

As it has been mentioned in the previous section, the semi-implicit FDM discretization leads to solving a large sparse algebraic system. In most cases, the matrix of this system is symmetric, which allows one to use efficient iterative solvers like conjugate gradient method, possibly with some kind of preconditioning. In the case of the shape recovery model, the matrix is not symmetric, which influences the choice of the solver. For instance, possible choices then could be bi-conjugate gradient method or classical relaxation methods like Gauss-Seidel or SOR. In fact, in most cases, the use of the Gauss-Seidel method is sufficient due to the special properties of the Allen-Cahn equation. With lowering the parameter ξ (and thus increasing the accuracy of the model), one needs to lower the discrete scale step as well. The solution between two scale levels does not differ much and the solution from the previous scale level can be used as a good initial guess for the solver. Number of required iterations then is not large, thus the convergence rate even of the Gauss-Seidel method is very good. Compared to the CG method, the advantage of the Gauss-Seidel method is in lower memory requirements and in simplicity of the algorithm.

# of iter	1st step	5th step	20th step	50th step	Total time
Gauss-Seidel	23	23	23	22	153.22
CG No precond	28	28	26	24	250.45
CG Jacobi	21	20	17	17	252.85
CG IC(0)	17	17	16	15	373.38

Table 1. Comparison of iterative solvers for the scheme (16); gray-scale image size: 512x512; implicit scheme

Despite the fact, that for the presented numerical computations the Gauss-Seidel method has been used, it is interesting to compare this method with CG

without preconditioning and with Jacobi preconditioner and the IC(0) preconditioner. See Table (1) for comparison of these two types of preconditioners. The results in this table were computed by solving the Allen-Cahn equation of the type 3 on AMD 650MHz with 128MB memory. In each column there is a number of solver iterations at a given scale. The total time, shown in the last column, is amount of time spent in the solver during the evolution.

5.1 Computational experiments

Example 1 Allen-Cahn equation with gradient term. Gray-scale image (size 512x 512 pixels), contamined with salt and pepper noise. Parameters: $h = 0.05$, $\tau = 0.0005$, $\xi = 0.05$, $a = 2$, $F = 0$. Image in the upper left corner is the original image, followed by images after 10, 20 and 30 iterations. Two bottom images show level set at 0.5 in the original and final image.

Example 2 Allen-Cahn equation with gradient term. Binary image (size 180x180 pixels), contamined with salt and pepper noise. Parameters: $h = 0.05$, $\tau = 0.0005$, $\xi = 0.05$, $a = 2$, $F = 10$. Image in the upper left corner is the original image contamined with additive noise, followed by images after 4, 8 and 12 iterations. Two bottom images show level set at 0.5 in the original and final image. It is apparent, that the Allen-Cahn equation gives excellent results on binary images, even if their structure is quite complicated. The noise vanishes very quickly.

Example 3 Pattern recovery model. Initial condition: binary image (size 180x180 pixels), final condition: binary image (size 180x180 pixels). Parameters: $h = 0.05$, $\tau = 0.0005$, $\xi = 0.05$, $a = 2$, $b = 100$. The sequence consists of images after 10, 20, 30, 40 and 50 iterations.

Example 4 Shape recovery of a binary image (size 180x180 pixels) contamined by salt and pepper noise. Parameters: $h = 0.05$, $\tau = 0.0005$, $\xi = 0.07$, $a = 2$, $b = 100$, $\lambda = 1000$, $\beta = 10^{-10}$, $F = 15$. In the upper left corner, the image I is shown and in the upper right corner, level set at 0.5 of the final image is shown. Then, initial condition together with images after 300, 600, 800 iterations is depicted. We can see, that a part of the letter Q with high curvature has not been recovered exactly.

Example 5 Shape recovery of multiple objects in a binary image (size 256x256 pixels) contamined by salt and pepper noise. Parameters: $h = 0.05$, $\tau = 0.0005$, $\xi = 0.07$, $a = 2$, $b = 100$, $\lambda = 5000$, $\beta = 10^{-9}$, $F = 15$. In the upper left corner, the image I is shown and in the upper right corner, level set at 0.5 of the final image is shown. Then, initial condition together with images after 400, 800, 1200 iterations is depicted. We can conclude, that the presented model gives satisfactory results for binary images even in the presence of noise.

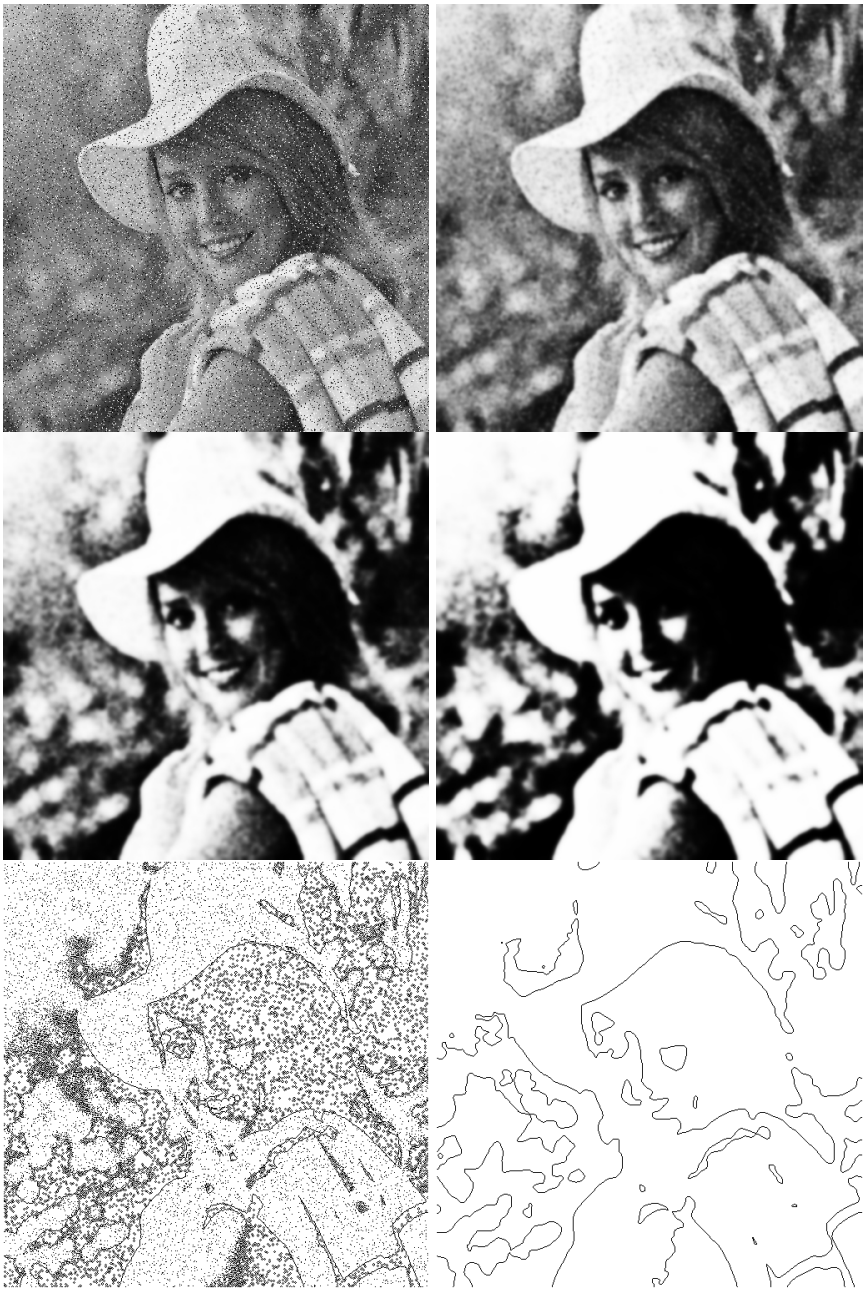

Fig. 4. Example 1

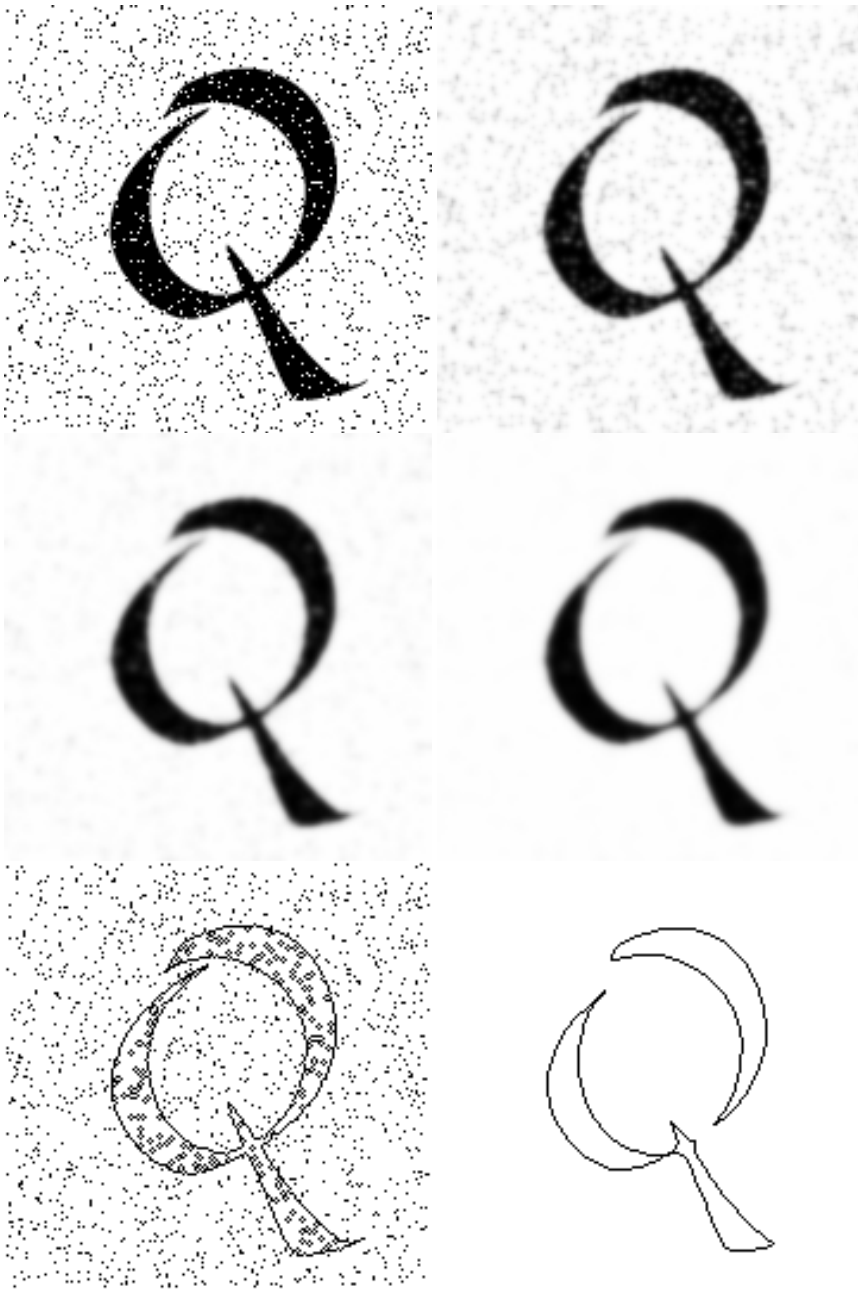

Fig. 5. Example 2

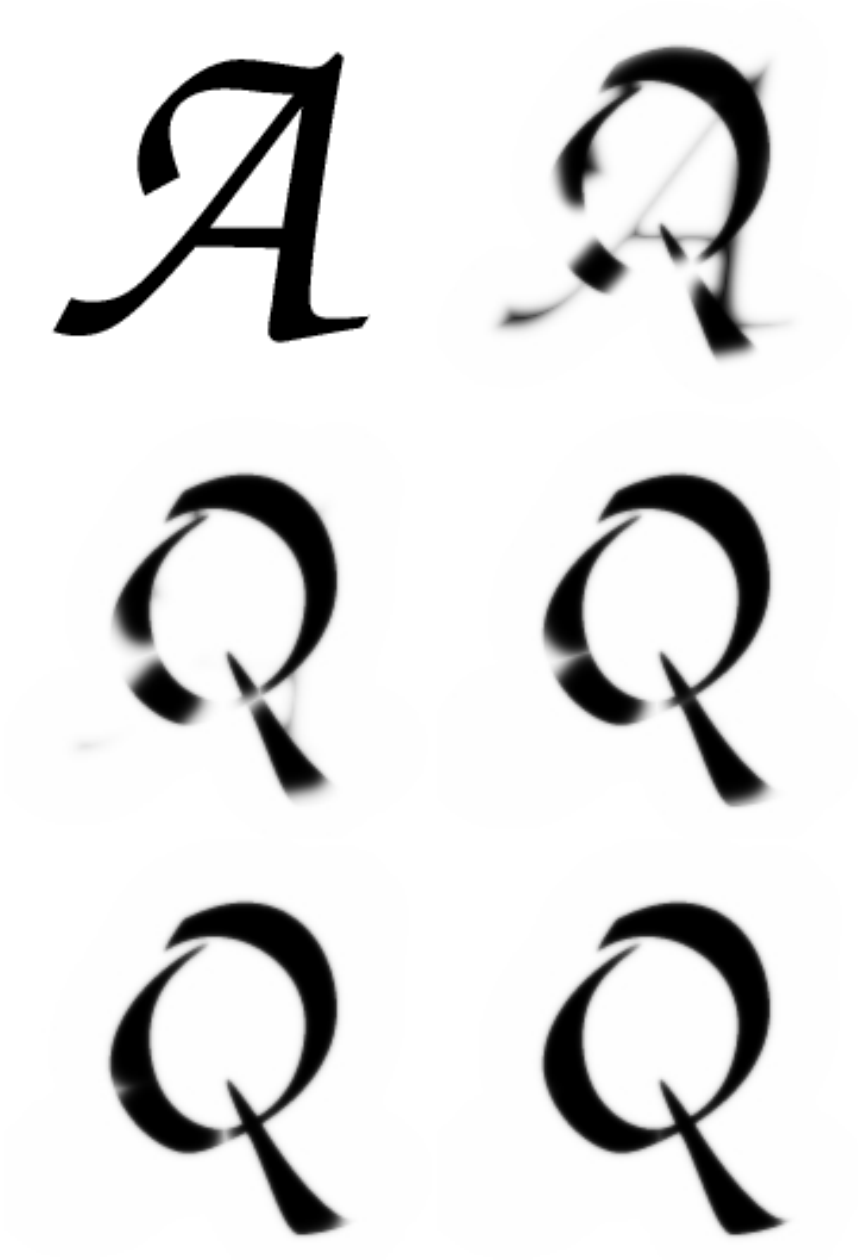

Fig. 6. Example 3

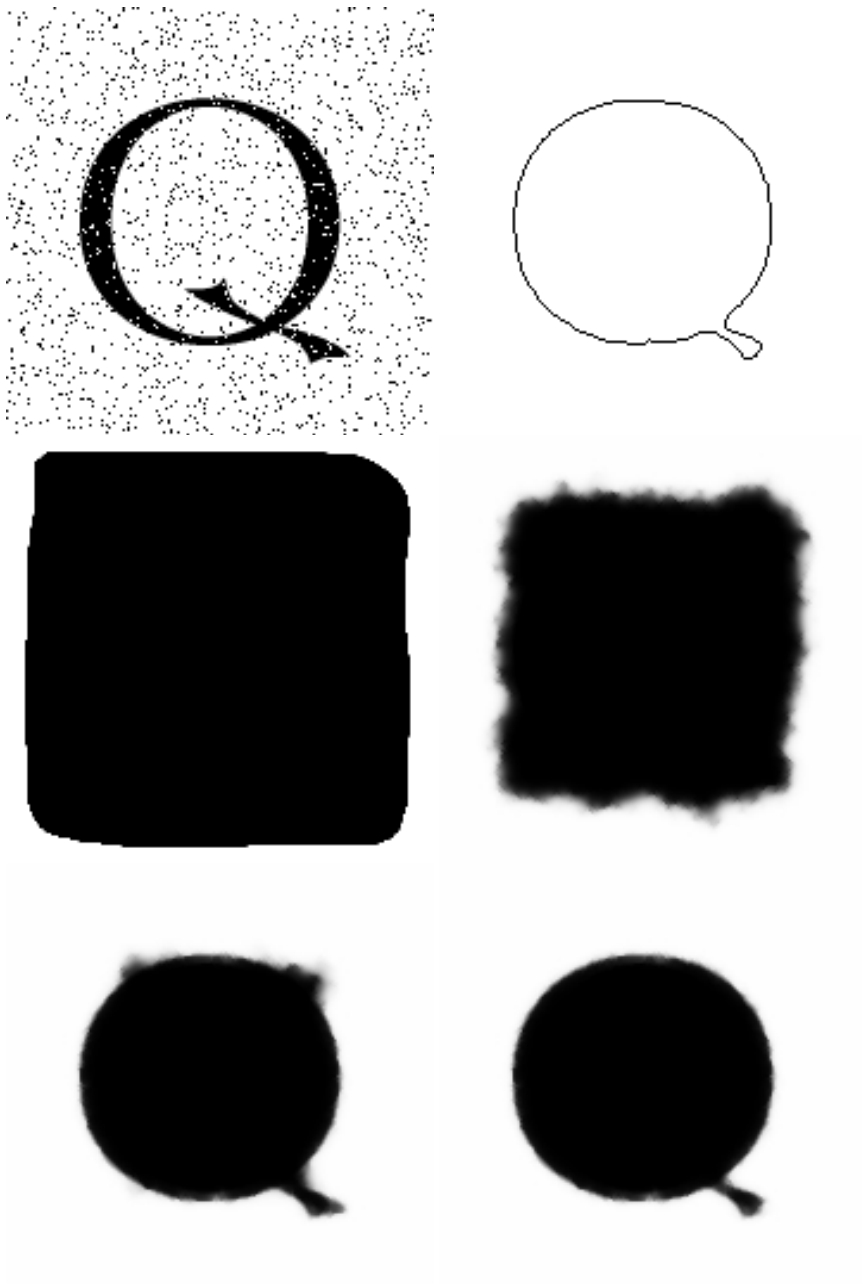

Fig. 7. Example 4

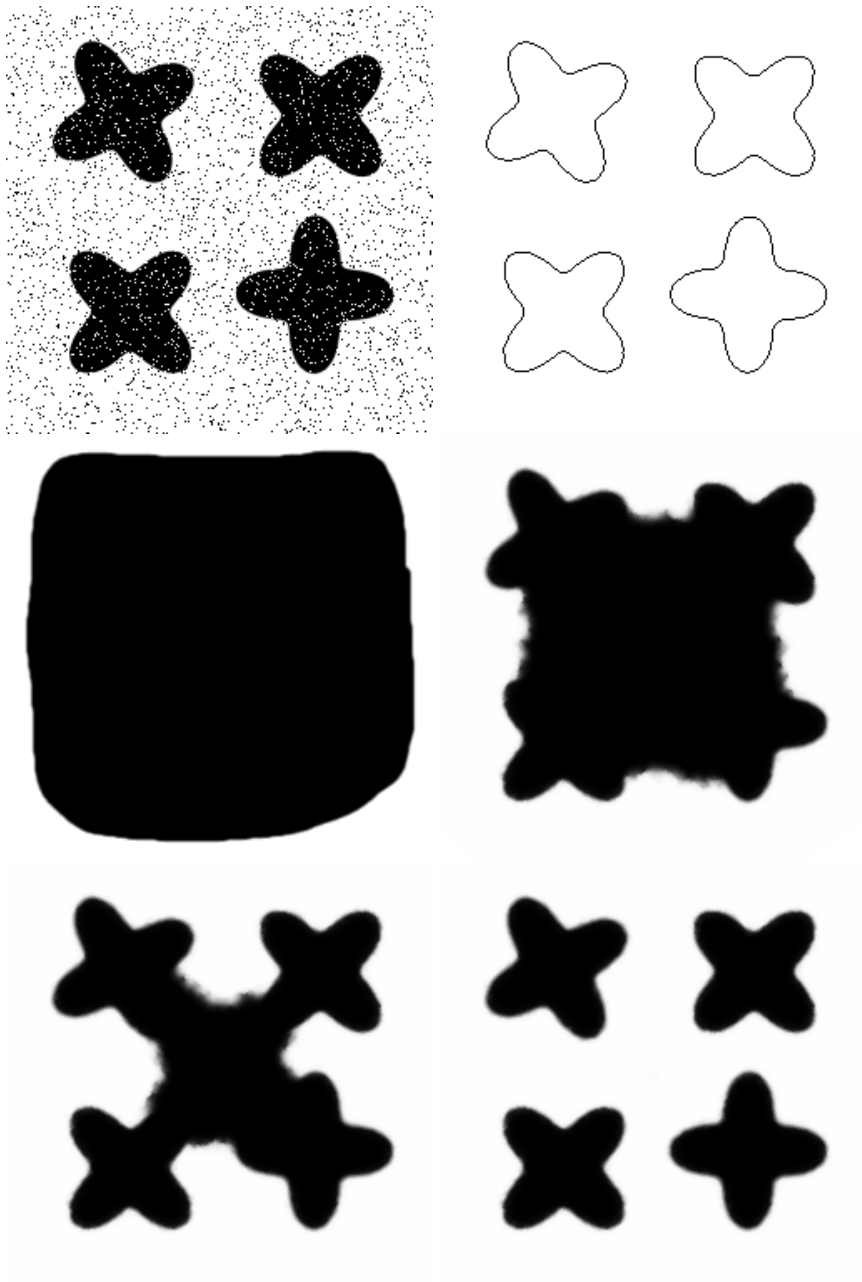

Fig. 8. Example 5

References

1. L. Alvarez, F. Guichard, P.-L. Lions and J.-M. Morel, *Axioms and fundamental equations of image processing*, Arch. Rational Mech. Anal. **123** (1993), 199–257.
2. L. Alvarez,P. L. Lions and J. M. Morel, *Image selective smoothing and edge detection by nonlinear diffusion. II*, SIAM J. Numer. Anal. **29** (1992), 845–866.
3. S. M. Allen and J. W. Cahn, *A microscopic theory for antiphase boundary motion and its application to antiphase domain coarsening*, Acta Metall. **27** (1979), 1085–1095.
4. M. Beneš, *On a computational comparison of phase-field and sharp-interface model of microstructure growth in solidification*, Acta Techn. CSAV **41** (1996), 597–608.
5. M. Beneš, *Quantitative analysis of phase-field model of phase transitions*, Proceedings on Prague Mathematical Conference 1996, 1996, pp. 23–28.
6. M. Beneš, *Analysis of equations in the phase-field model*, Proc. of Equadiff 9, Brno, (1997), 17–35.
7. M. Beneš and K. Mikula, *Simulation of anisotropic motion by mean curvature - comparison of phase field and sharp interface approaches*, Acta Math. Univ. Comenianae **LXVII** (1998), 17–42.
8. G.Caginalp, *An analysis of a phase field model of a free boundary*, Arch. Rational Mech. Anal. **92** (1986), 205–245.
9. G. Caginalp, *The dynamics of a conserved phase field system: Stefan-like, Hele-Shaw, and Cahn-Hilliard models as asymptotic limits*, IMA J. Appl. Math. **44** (1990), 77–94.
10. V. Caselles, F. Catté, T. Coll and F. Dibos, *A geometric model for active countours*, Numer. Math. **66** (1993), 1–31.
11. V. Caselles and B. Coll, *Snakes in movement*, SIAM J. Numer. Anal. **33** (1996), 2445–2456.
12. V. Caselles, R. Kimmel and G. Sapiro, *Geodesic active contours*, Int. J. Comp. Vision. (1997).
13. L. C. Evans and J. Spruck, *Motion of level sets by mean curvature I.*, J. Diff. Geom. **33** (1991), 635–681,
14. A. Handlovičová, K. Mikula and F. Sgallari, *Variational numerical methods for solving nonlinear diffusion equations arising in image processing*, Journal of Visual Communication and Image Representation (2001) (to appear).
15. K. Mikula and J. Kačur *Solution of nonlinear diffusion appearing in image smoothing and edge detection*, Appl. Numer. Math. **17** (1995), 47–59.
16. M. Kass, A. Witkin and D. Terzopoulos, *Snakes: active contour models*, Int. J. Comp. Vision **1** (1987) 321–331.
17. S. Kichenassamy, A. Kumar, P. Olver, A. Tannenbaum, and A. Yezzi, Jr., *Conformal curvature flows: from phase transitions to active vision*, Arch. Rational Mech. Anal. **134** (1996), 275–301.
18. R. Kobayashi, *Modeling and numerical simulations of dendritic crystal growth*, Physica D **63** (1993), 410–423.
19. R. Malladi, J. A. Sethian and B. Vemuri, *Shape modelling with front propagation: a level set approach*, IEEE Trans. on Pattern Analysis and Machine Intelligence **17** (1995), 158–175.
20. S. J. Osher and J. A. Sethian, *Front propagating with curvature dependent speed: algorithms based on the Hamilton-Jacobi formulation*, J. Comp. Phys. **79** (1988), 12–49.