Highly Accurate PDE-Based Morphology for General Structuring Elements

Michael Breuß and Joachim Weickert

Mathematical Image Analysis Group,
Faculty of Mathematics and Computer Science, Building E1.1
Saarland University, 66041 Saarbrücken, Germany
{breuss,weickert}@mia.uni-saarland.de

Abstract. Modelling the morphological processes of dilation and erosion with convex structuring elements with partial differential equations (PDEs) allows for digital scalability and subpixel accuracy. However, numerical schemes suffer from blur by dissipative artifacts. In our paper we present a family of so-called flux-corrected transport (FCT) schemes that addresses this problem for arbitrary convex structuring elements. The main characteristics of the FCT-schemes are: (i) They keep edges very sharp during the morphological evolution process, and (ii) they feature a high rotational invariance. Numerical experiments with diamonds and ellipses as structuring elements show that FCT-schemes are superior to standard schemes in the field of PDE-based morphology.

1 Introduction

Mathematical morphology is concerned with the analysis of shapes. Beginning with the works of Serra and Matheron [1, 2], it has evolved to a highly successful field in image processing. Many monographs and conference proceedings document this development, see e.g. [4, 6, 8, 18] and [17, 21, 22, 25], respectively.

In mathematical morphology two fundamental operations are employed, *dilation* and *erosion*. Many other morphological processes such as openings, closings, top hats and morphological derivative operators can be derived from them. While dilation/erosion are frequently realised using a set-theoretical framework, an alternative formulation is available via *partial differential equations (PDEs)* [10, 11, 13, 14, 15]. Compared to the set-theoretical approach, the latter offers the conceptual advantages of digital scalability and subpixel accuracy. However, a usual drawback of PDE-based algorithms is that they introduce blurring artefacts, especially at edges of dilated/eroded objects. In this paper we are addressing this problem by dealing with the proper numerical realisation of PDE-based dilation and erosion for general structuring elements. We show how a *flux-corrected transport (FCT) scheme* that gives a sharp resolution of dilated/eroded object edges combined with a high-rotational invariance can be

X.-C. Tai et al. (Eds.): SSVM 2009, LNCS 5567, pp. 758–769, 2009.
© Springer-Verlag Berlin Heidelberg 2009

used. It is not only easy to implement, but we also show in numerical experiments that it outperforms other schemes for PDE-based morphology.

Mathematical Formulation of Dilation and Erosion. Let us consider a grey-value image $f : \Omega \subset \mathbb{R}^2 \to \mathbb{R}$ and a so-called structuring element $B \subset \mathbb{R}^2$. The building blocks of morphological filters, dilation and erosion, are then defined by

$$\textit{dilation:} \qquad (f \oplus B)(\mathbf{x}) := \sup\{f(\mathbf{x}-\mathbf{z}),\ \mathbf{z}\in B\}, \tag{1}$$

$$\textit{erosion:} \qquad (f \ominus B)(\mathbf{x}) := \inf\{f(\mathbf{x}+\mathbf{z}),\ \mathbf{z}\in B\}. \tag{2}$$

Dilation/erosion are often realised in a set-theoretical framework. To this end, the structuring elements are given by masks defined in accordance to the discrete pixel grid in an image. For convex structuring elements, there exists an alternative formulation of dilation/erosion in terms of PDEs that guarantee the validity of the semigroup property of dilation/erosion operations [14, 15, 10, 11]. Here, a scaling parameter $t > 0$ is introduced within the structuring element which is then given as tB, achieving digital scalability. Especially, in Paragraph 4.2 of [14] it was shown that dilation/erosion can be realised by solving the PDEs

$$\textit{dilation:} \qquad \partial_t u(\mathbf{x},t) = \sup_{\mathbf{z}\in B} \langle \mathbf{z}, \nabla u(\mathbf{x},t)\rangle, \tag{3}$$

$$\textit{erosion:} \qquad \partial_t u(\mathbf{x},t) = \inf_{\mathbf{z}\in B} \langle \mathbf{z}, \nabla u(\mathbf{x},t)\rangle, \tag{4}$$

respectively. In (3)-(4), $\nabla = (\partial_x, \partial_y)^\top$ is the spatial nabla operator, and $\langle \mathbf{a}, \mathbf{b}\rangle$ denotes the Euclidean product of the vectors $\mathbf{a}$ and $\mathbf{b}$. Interpreting the scaling parameter t as an artificial time, the given image f serves as the initial condition for the temporal evolution described by the PDEs (3)-(4). As we deal with rectangular images of finite size, we also need to define boundary conditions. Thus, we employ homogeneous Neumann boundary conditions at the image boundary $\partial\Omega$, complementing the PDE-based problem description.

Set-Theoretical vs. PDE-Based Approach. As already mentioned, the PDE-based approach offers the advantages of digital scalability and subpixel accuracy compared to the set-theoretical formulation, while the PDE-based algorithms usually introduce blurring of edges. Let us note in addition, that round structuring elements such as circles or ellipses can not be represented conveniently in the set-theoretical approach, and they typically do not define a granulometric family [18]. Thus, conceptually the PDE-based approach is favourable.

Numerical Schemes. Let us first briefly comment on the nature of the evolutionary PDEs (3)-(4). By the first-order spatial derivatives these PDEs are hyperbolic, describing a wave propagation or transport behaviour, in analogy to Huygens' principle. Thereby, the shape of the evolving wavefront is determined by the shape of the scalable structuring element.

Thus, given the hyperbolic character of the dilation/erosion PDEs (3)-(4), it is natural that techniques from hyperbolic conservation laws are of importance for this work; see e.g. [23] for a general discussion of numerical methods

for hyperbolic PDEs. In the context of dilation/erosion, popular schemes are the *Osher-Sethian (OS) schemes* [24,9,20] and the *Rouy-Tourin (RT) scheme* [12,19]. In particular, let us note that one of the mentioned OS-schemes is a second-order high-resolution method. The use of a comparable high-resolution ansatz, specifically an essentially non-oscillatory (ENO) approach, was reported in [16]. In [26], Breuß and Weickert constructed a FCT-scheme for performing dilation/erosion with a disc of radius t as structuring element.

Our Contribution. We extend the applicability of the FCT-scheme introduced in [26] from discs to general structuring elements. As it turns out, this is feasible but involves technical difficulties, especially for the case of general ellipses as structuring elements we discuss here in detail. We validate experimentally that the attractive features discussed in [26], namely a sharp resolution of edges and high rotational invariance, do carry over to the general case. In order to compare the performance of the FCT-scheme to set-theoretical algorithms, we use a diamond-shaped structuring element. For a comparison relying completely on digitally scalable structuring elements, we use an ellipse as structuring element. We show experimentally that the FCT-scheme gives much more accurate results than other PDE-based schemes.

Paper Organisation. In Section 2, we briefly introduce classic numerical schemes important in this paper for the case of a diamond as structuring element. We also construct the FCT-scheme for the same structuring element there. After that, we elaborate in Section 3 on the FCT-construction for ellipses as structuring element. In Section 4, we present numerical results. The paper is finished by a conclusion and outlook in Section 5.

2 PDE-Based Algorithms for Diamonds

For the sake of brevity, we discuss only dilation in detail, as the corresponding scheme for erosion is easily obtained. Employing the structuring element

$$B := \left\{ \mathbf{z} \in \mathbb{R}^2,\ \|\mathbf{z}\|_1 \leq 1 \right\}, \tag{5}$$

the sought PDE (3) describing specifically dilation with a diamond is based on the dual norm to the norm used in (5). It reads as

$$\partial_t u = \|\nabla u\|_\infty \,, \tag{6}$$

where $\|\nabla u\|_\infty = \max\left(|\partial_x u|\,, |\partial_y u|\right)$. Now, we need to discretise the PDE (6). For this, we define a spatio-temporal grid with uniform mesh widths h_x, h_y and τ, respectively. For the formulae of numerical schemes, let us then introduce the notation $U_{i,j}^n$ via

$$U_{i,j}^n \;\approx\; u\left(ih_x, jh_y, n\tau\right) \,. \tag{7}$$

Also, for writing down our schemes let us define the following *finite difference operators*:

$$\begin{aligned} \textit{right-sided:} \qquad & D_+^x U_{i,j}^n := U_{i+1,j}^n - U_{i,j}^n \,, && (8)\\ \textit{left-sided:} \qquad & D_-^x U_{i,j}^n := U_{i,j}^n - U_{i-1,j}^n \,, && (9)\\ \textit{central:} \qquad & D_c^x U_{i,j}^n := U_{i+1,j}^n - U_{i-1,j}^n \,. && (10) \end{aligned}$$

In an analogous fashion, we use corresponding finite difference operators D_+^y, D_-^y and D_c^y for the y-direction.

2.1 The High-Resolution Osher-Sethian-Scheme

In what follows, we will refer to this method as the OS-scheme, as its simpler, first-order variant will not be considered here. For its definition, we employ the minmod-function (as it gives back the minimal modulus of its arguments) given as

$$\mathrm{mm}(a,b) := \begin{cases} \min(a,b) & \text{if } a>0 \text{ and } b>0\,, \\ \max(a,b) & \text{if } a<0 \text{ and } b<0\,, \\ 0 & \text{else}\ . \end{cases} \tag{11}$$

To keep the presentation of the OS-scheme short, let us define the following discrete derivative operators:

$$\delta_x^{\mathrm{OS}-} U_{i,j}^n := \frac{1}{h_x} \min\left(D_-^x U_{i,j}^n + \frac{1}{2}\mathrm{mm}\left(D_-^x D_+^x U_{i,j}^n,\, D_-^x D_-^x U_{i,j}^n\right),\, 0\right)\,, \tag{12}$$

$$\delta_x^{\mathrm{OS}+} U_{i,j}^n := \frac{1}{h_x} \max\left(D_+^x U_{i,j}^n - \frac{1}{2}\mathrm{mm}\left(D_+^x D_+^x U_{i,j}^n,\, D_-^x D_+^x U_{i,j}^n\right),\, 0\right)\,, \tag{13}$$

and we set analogously $\delta_y^{\mathrm{OS}-} U_{i,j}^n$ and $\delta_y^{\mathrm{OS}+} U_{i,j}^n$. Let us note that the basic idea behind the construction within (12)-(13) is to augment the first-order derivatives $D_-^x U_{i,j}^n$ and $D_+^x U_{i,j}^n$ by a higher-order correction given in terms discrete second-order derivatives. For a compact notation, let us then set

$$L\left(U^n, i, j\right) := \max\left(\left|\delta_x^{\mathrm{OS}-} U_{i,j}^n\right| + \left|\delta_x^{\mathrm{OS}+} U_{i,j}^n\right|, \left|\delta_y^{\mathrm{OS}-} U_{i,j}^n\right| + \left|\delta_y^{\mathrm{OS}+} U_{i,j}^n\right|\right)\,, \tag{14}$$

which realises the maximum norm on the discrete level. Let us briefly comment on the 'double' contributions of the discretised derivatives in (14), for instance in x-direction: $\left|\delta_x^{\mathrm{OS}-} U_{i,j}^n\right| + \left|\delta_x^{\mathrm{OS}+} U_{i,j}^n\right|$. For a strictly monotone grey-value profile in the points incorporating the indices $i-1, i, i+1$, there will only be *one* non-zero contribution from one of the summands; the other one will be zero. That is determined by the sign of the slope in a strictly monotone profile. Only at a local minimum $U_{i,j}^n$, both summands could be non-zero.

The OS-scheme is a second-order high-resolution scheme. As such, we need to employ a second-order time stepping scheme, for which we choose the well-known method of Heun which is a two-stage Runge-Kutta method [7]:

$$\begin{aligned} \bar{U}_{i,j}^{n+1} &= U_{i,j}^n \,+\, \tau\, L\left(U^n, i, j\right) \\ U_{i,j}^{n+1} &= \frac{1}{2} U_{i,j}^n \,+\, \frac{1}{2}\bar{U}_{i,j}^{n+1} \,+\, \frac{\tau}{2} L\left(\bar{U}^{n+1}, i, j\right)\ . \end{aligned} \tag{15}$$

2.2 The FCT-Scheme

Like the OS-scheme, the FCT-scheme is a predictor-corrector method. However, while this format arises in the case of the OS-scheme by use of a Runge-Kutta method for time integration, the FCT-construction works differently. As a predictor step, a first-order scheme is used for wave propagation. Thus, by the first-order error the predictor features desirable theoretical properties but also introduces much artificial dissipation. Then, by taking into account the so-called viscosity form of the predictor scheme, the dissipation can be quantified on a discrete level and is negated in a second step using stabilised inverse diffusion [27]. For details we refer to [26]. Let us note that the basic idea to negate dissipation by a corrector step was invented by Boris and Book [3,5]. However, the corrector step was realised technically quite differently in their original works. Following their procedure would lead to a different (and less attractive) scheme than with the approach followed here.

As a predictor step we use the dissipative scheme proposed by Rouy and Tourin [12]. In order to write this down, we use the abbreviation

$$\delta_x^{\mathrm{RT}} U_{i,j}^n \;:=\; \max\left(\frac{1}{h_x}\max\left(-D_-^x U_{i,j}^n, 0\right)\;,\; \frac{1}{h_x}\max\left(D_+^x U_{i,j}^n, 0\right)\right)\;, \tag{16}$$

and $\delta_y^{\mathrm{RT}} U_{i,j}^n$ is used accordingly. Then the RT-scheme is in our case defined as

$$\bar{U}_{i,j}^{n+1} = U_{i,j}^n \;+\; \tau\;\max\left(\left|\delta_x^{\mathrm{RT}} U_{i,j}^n\right|, \left|\delta_y^{\mathrm{RT}} U_{i,j}^n\right|\right)\;. \tag{17}$$

The FCT scheme then consists of a subsequent application of (17) and a corrector step negating the artificial dissipation of the RT scheme, reading in total as

$$\begin{aligned} \bar{U}_{i,j}^{n+1} &= U_{i,j}^n \;+\; \tau\;\max\left(\left|\delta_x^{\mathrm{RT}} U_{i,j}^n\right|, \left|\delta_y^{\mathrm{RT}} U_{i,j}^n\right|\right) \\ U_{i,j}^{n+1} &= \bar{U}_{ij}^{n+1} + C_h\left(\bar{U}^{n+1}\right) - C_d\left(\bar{U}^{n+1}\right)\;. \end{aligned} \tag{18}$$

Let us consider the corrector step and especially the functions C_h ('h' for high-order part) and C_d ('d' for dissipative part) in some detail.

As indicated, the first step of the FCT procedure is to split the dissipative part of the scheme from the non-dissipative second-order part. The latter part of the scheme can be described via central differences as in (10), since central differences do not feature dissipation in the leading-order part of the truncation error. Thus, the discretisation of the dilation PDE (6) using central differences only,

$$\bar{U}_{i,j}^{n+1} \;=\; U_{ij}^n + \max\left(\left|\frac{\tau}{2h_x} D_c^x U_{i,j}^n\right|, \left|\frac{\tau}{2h_y} D_c^y U_{i,j}^n\right|\right)\;, \tag{19}$$

incorporates no numerical dissipation in the approximation of spatial derivatives.

Employing *predicted data* as arguments in the formulae of the corrector step, we can identify the high-order part within the predictor formula (which was absent before adding it) as

$$C_h\left(\bar{U}^{n+1}\right) \;:=\; +\max\left(\left|\frac{\tau}{2h_x} D_c^x \bar{U}_{i,j}^{n+1}\right|, \left|\frac{\tau}{2h_y} D_c^y \bar{U}_{i,j}^{n+1}\right|\right) \tag{20}$$

by adding zero via adding/subtracting

$$\max\left(\left|\frac{\tau}{2h_x}D_c^x\bar{U}_{i,j}^{n+1}\right|,\left|\frac{\tau}{2h_y}D_c^y\bar{U}_{i,j}^{n+1}\right|\right)\,. \tag{21}$$

Let us now stress that the remaining terms of the predictor formula plus the contribution due to (20)-(21), with data $\bar{U}^{n+1}$ as arguments, define the discrete dissipation C_d. However, since we want to subtract C_d in (18), we aim for a *backward dissipation*. Thus, we need to stabilise this contribution with help of a straightforward extension of the minmod-function from (11) to three arguments:

$$G_{i+1/2,j} := \mathrm{mm}\left(D_-^x\bar{U}_{i,j}^{n+1},\ \frac{\tau}{2h_x}D_+^x\bar{U}_{i,j}^{n+1},\ D_+^x\bar{U}_{i+1,j}^{n+1}\right), \tag{22}$$

$$G_{i,j+1/2} := \mathrm{mm}\left(D_-^y\bar{U}_{i,j}^{n+1},\ \frac{\tau}{2h_y}D_+^y\bar{U}_{i,j}^{n+1},\ D_+^y\bar{U}_{i,j+1}^{n+1}\right). \tag{23}$$

Let us note that the left and right arguments in (22)-(23) are supposed to prevent overshoots, while the middle argument is determined in accordance to (16). For details concerning this procedure and an analysis of stabilised inverse diffusion, see [26,27], respectively. For the dissipative correction term $C_d\left(\bar{U}^{n+1}\right)$ we employ the stabilised fluxes from (22)-(23), yielding

$$\delta_x^{bd}\bar{U}_{i,j}^{n+1} := \frac{\tau}{2h_x}\left|D_c^x\bar{U}_{i,j}^{n+1}\right| + G_{i+1/2,j} - G_{i-1/2,j}, \tag{24}$$

$$\delta_y^{bd}\bar{U}_{i,j}^{n+1} := \frac{\tau}{2h_y}\left|D_c^y\bar{U}_{i,j}^{n+1}\right| + G_{i,j+1/2} - G_{i,j-1/2}\,, \tag{25}$$

and finally:

$$C_d\left(\bar{U}^{n+1}\right) \;=\; \max\left(\left|\delta_x^{bd}\bar{U}_{i,j}^{n+1}\right|,\left|\delta_y^{bd}\bar{U}_{i,j}^{n+1}\right|\right)\,. \tag{26}$$

3 The FCT-Scheme for General Ellipses

The key for obtaining dilation with a general ellipse is to consider the *normal form* of an ellipse in the x-y-plane which can be written for our purpose as

$$a^2x^2 + b^2y^2 \;=\; 1\,. \tag{27}$$

This equation describes the location of the *front* of the solution of the evolutionary PDE

$$\partial_t u \;=\; \sqrt{a^2\left(\partial_x u\right)^2 + b^2\left(\partial_y u\right)^2} \tag{28}$$

at time $t = 1$, starting from the center $(x, y)^T = (0, 0)^T$. For $a = b = 1$, one obtains a circle, retrieving a disc as structuring element.

Note that we should be able to handle a PDE like (28) easily, while implementing *directly* an algorithm for ellipses with a general orientation of the principal axis poses difficulties.

The General Idea. Let us briefly outline the procedure. In order to finally solve the PDE (28), we collect, for each pixel individually, grey values from positions

corresponding to a rotated grid. As these will not be located exactly at pixel centers, they will in general not coincide with the given grey values and need to be interpolated. With these interpolated data we solve pointwise the PDE (28).

Having thus described the general proceeding, we begin its realisation for general ellipses as structuring elements by implementing a rotation of the coordinate system. For a more detailed explanation of this, we need to fix some geometric properties of the ellipse defining the structuring element. In order to simplify the presentation, we set $h_x := h_y := 1$.

First, let us calibrate the length of the principal axis to 1, i.e. the final ellipse is a subset of the unit disc. In order to use a PDE of the form of (28), we have to rotate the grid. Let us note that for $h_x = h_y = 1$, all points within the stencil of the Rouy-Tourin scheme (16)-(17) are on the unit sphere if we center this at $(ih_x, jh_y)^T$.

Then we rotate the local Euclidean coordinate system centered at $(ih_x, jh_y)^T$ by an angle α with $0 \leq \alpha \leq \pi/2$. Making use of elementary trigonometry, the values rotated now onto the knots of our finite difference stencil are grey values from the points given by $(\cos\alpha_k, \sin\alpha_k)^T$, $\alpha_k := \alpha + k \cdot \frac{\pi}{2}$, $k = 0, 1, 2, 3$. Let us note that in using this procedure, we effectively consider an ellipse where the angle between x-axis and principal axis is $-\alpha$.

Let us stress, that we can obtain via $0 \leq \alpha \leq \pi/2$ *all* possible ellipses, as we can switch at any time the roles of a and b in (28) that define the principal axis. It is just practical to impose $0 \leq \alpha \leq \pi/2$ since this helps to give a suitable interpolation formula, which is the next step.

Obviously, we need at each pixel the grey values after rotation for defining our finite difference scheme. We wish to achieve second-order accuracy because the second-order high-resolution OS-scheme will serve as the comparison scheme for the procedure. Thus, we use standard bilinear interpolation for this purpose as the error of this approach is formally of the same order.

In order to show how the computation wroks, we now clarify the details for the values in the first quadrant. As $0 \leq \alpha \leq \pi/2$, the grey value we need at the knot $((i+1)h_x, jh_y)^T$ is located at $(\cos\alpha_0, \sin\alpha_0)^T = (\cos\alpha, \sin\alpha)^T$. Because of $h_x = h_y = 1$, we can use the general formula for bilinear interpolation of some function g over the rectangle $[0,1] \times [0,1]$ reading as

$$\begin{aligned} g(x,y) \approx\ & g(0,0)(1-x)(1-y) + g(1,0)x(1-y) \\ & + g(0,1)(1-x)y + g(1,1)xy\,, \qquad x, y \in [0,1]. \end{aligned} \tag{29}$$

Plugging in our values within the first quadrant, we obtain the rotated grey value $\tilde{U}^n_{i+1,j}$ as

$$\begin{aligned} \tilde{U}^n_{i+1,j} :=\ & U^n_{i,j}(1-\cos\alpha)(1-\sin\alpha) + U^n_{i+1,j}\cos\alpha(1-\sin\alpha) \\ & + U^n_{i,j+1}(1-\cos\alpha)\sin\alpha + U^n_{i+1,j+1}\cos\alpha\sin\alpha\,. \end{aligned} \tag{30}$$

Analogously, we can compute the other members of our stencil after rotation of our local coordinate system.

The resulting formulae are:

$$\tilde{U}^n_{i,j} := U^n_{i,j}\,, \tag{31}$$

$$\begin{aligned}\tilde{U}^n_{i,j+1} := {} & U^n_{i,j}(1-\cos\alpha)(1-\sin\alpha) + U^n_{i,j+1}\cos\alpha(1-\sin\alpha)\\ & +U^n_{i-1,j}(1-\cos\alpha)\sin\alpha + U^n_{i-1,j+1}\cos\alpha\sin\alpha\,,\end{aligned} \tag{32}$$

$$\begin{aligned}\tilde{U}^n_{i-1,j} := {} & U^n_{i,j}(1-\cos\alpha)(1-\sin\alpha) + U^n_{i-1,j}\cos\alpha(1-\sin\alpha)\\ & +U^n_{i,j-1}(1-\cos\alpha)\sin\alpha + U^n_{i-1,j-1}\cos\alpha\sin\alpha\,,\end{aligned} \tag{33}$$

$$\begin{aligned}\tilde{U}^n_{i,j-1} := {} & U^n_{i,j}(1-\cos\alpha)(1-\sin\alpha) + U^n_{i,j-1}\cos\alpha(1-\sin\alpha)\\ & +U^n_{i+1,j}(1-\cos\alpha)\sin\alpha + U^n_{i+1,j-1}\cos\alpha\sin\alpha\,.\end{aligned} \tag{34}$$

In terms of these values we now give the main formulae for the FCT-scheme. Comparing especially with (18), (20) and (26) from Paragraph 2.2 shows what needs to be done:

$$\begin{aligned}\bar{U}^{n+1}_{i,j} &= \tilde{U}^n_{i,j} + \tau\sqrt{a^2\left(\delta^{\mathrm{RT}}_x\tilde{U}^n_{i,j}\right)^2 + b^2\left(\delta^{\mathrm{RT}}_y\tilde{U}^n_{i,j}\right)^2}\\ U^{n+1}_{i,j} &= \bar{U}^{n+1}_{ij} + C_h\left(\bar{U}^{n+1}\right) - C_d\left(\bar{U}^{n+1}\right)\,,\end{aligned} \tag{35}$$

with

$$C_h\left(\bar{U}^{n+1}\right) = +\frac{\tau}{2h_x}\sqrt{a^2\left(D^x_c\bar{U}^{n+1}_{i,j}\right)^2 + b^2\left(D^y_c\bar{U}^{n+1}_{i,j}\right)^2}\,, \tag{36}$$

$$C_d\left(\bar{U}^{n+1}\right) = \sqrt{a^2\left(\delta^{bd}_x\bar{U}^{n+1}_{i,j}\right)^2 + b^2\left(\delta^{bd}_y\bar{U}^{n+1}_{i,j}\right)^2}\,. \tag{37}$$

Note that for the arguments of the minmod-function used within $C_d\left(\bar{U}^{n+1}\right)$, we also need to compute rotated grey values $\tilde{U}^{n+1}_{i\pm2,j\pm2}$ from the data set $\bar{U}^{n+1}$. This can be done in the same fashion as in (30)-(34). Also note that because of the callibration of ellipses, we always have in our experiments $a = 1$ and $b \in [0, 1]$.

4 Numerical Experiments

The main disadvantage in using PDE-based algorithms is the occurence of dissipative discretisation artefacts. The resulting blurring is especially observable at edges of dilated/eroded objects.

The Diamond Experiment. In this experiment, we solve the dilation PDE (6), comparing the FCT-scheme with the set-theoretical approach. For convenience, we always employ $h_x = h_y = 1$.

For the fully discrete, set-theoretical approach, we employ the usual 5-point-structuring element defined centered in $(0, 0)^T$ with vertices

$$(1, 0)^T,\ (0, 1)^T,\ (-1, 0)^T,\ (0, -1)^T\,. \tag{38}$$

In Figure 1, we observe the outcome of this experiment, where we have inverted the grey values. As input image, we use an image of size 129×129, where we

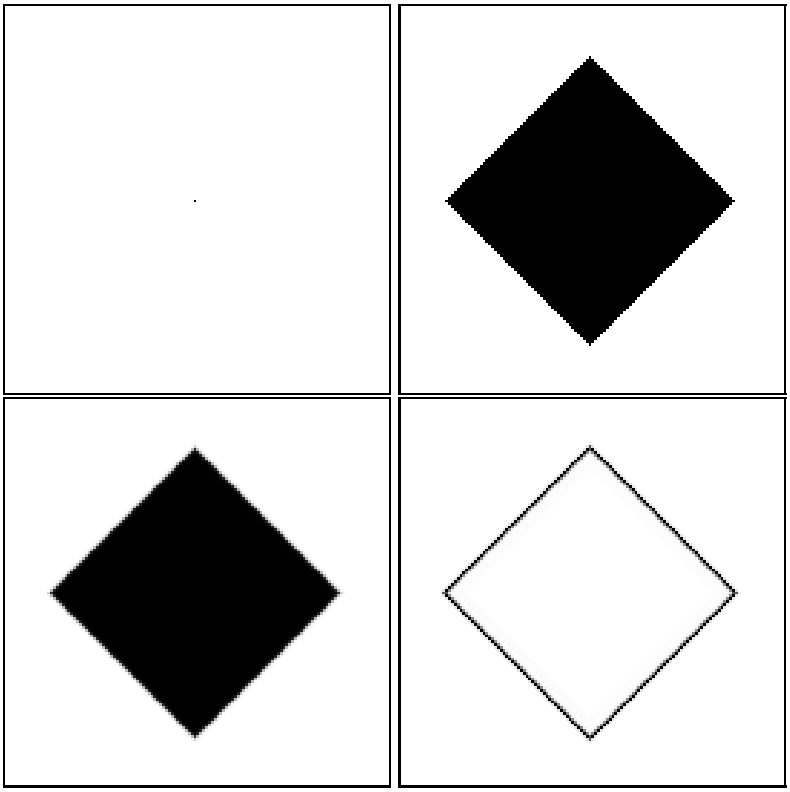

Fig. 1. Comparison of dilation with a diamond using inverted grey values. **Top.** Left: Initial image (a). Right: The set-based result (b). **Bottom.** Left: The FCT-result (c). Right: Scaled difference (d). The average difference visualised in (d) is of the grey value 1.502.

have exactly one pixel in the center of the image which is dilated. We perform 100 time steps with $\tau = 0.5$ for dilation with FCT, and 50 iterations with the set-based algorithm, respectively.

We observe that the FCT-result (c) is visually nice, with sharp diamond edges. Compared with the set-based result (b), we observe that there is some difference at the edges, which can be seen in the scaled difference map (d). Note that the average (unscaled) difference amounts to a grey value of 1.502. However, let us also note that the solution of the PDE is digitally scalable, so that the set-based solution is not intended to be the true solution of the dilation PDE.

The Ellipse Experiment. We now show computational results for ellipses as structuring elements. In order to first give an impression of what quality one may expect, we first consider an ellipse where the principal axis is aligned with the grid. For this experiment, we define the structuring element via $a = 1$, $b = 0.25$, compare (27). For the numerical experiment, we use $\tau = 0.5$, and we perform 100 time steps. The results of the OS-scheme together with the result of the

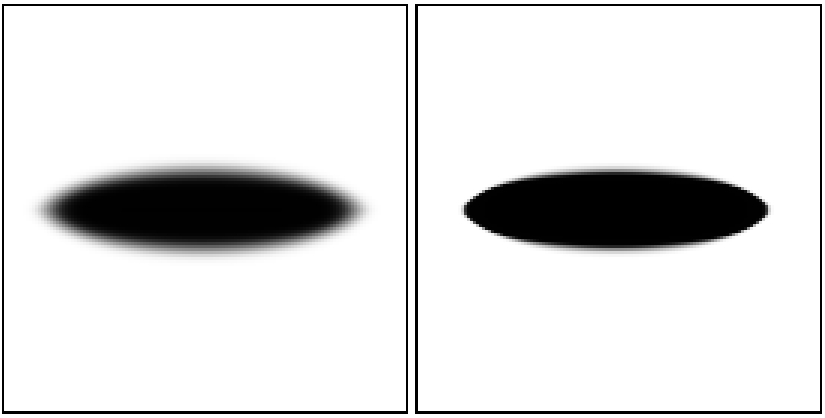

Fig. 2. Dilation comparison with inverted grey values. Left: OS-result (a). Right: FCT-result (b).

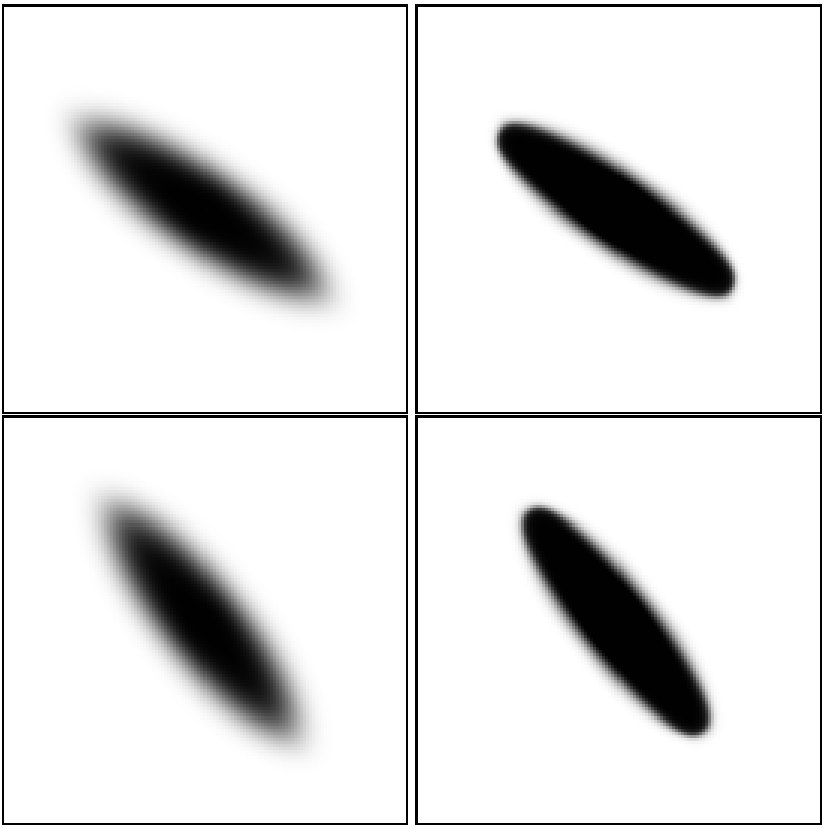

Fig. 3. Dilation comparison with inverted grey values. Rotated ellipses with (top) $\alpha = 0.6$ and (bottom) $\alpha = 0.9$. Bilinear interpolation was used to rotate the grid in each time step. **Left column:** OS-results. **Right column:** FCT-results.

FCT-scheme are displayed in Figure 2. While the result of the OS-scheme is quite blurry, we observe a mixed behaviour of the FCT-scheme. While the left and right front travelling with the largest signal velocity in this example are well-resolved, there is some blurring on the slow-moving upper and lower part of the edge of the ellipse.

Let us now consider the rotated case employing bilinear interpolation. In a setting analogous to the non-rotated case, but with $a = 1$, $b = 0.25$ and (a) $\alpha = 0.6$, (b) $\alpha = 0.9$. We obtain the results displayed in Figure 3. We observe that due to the interpolation there is some more blurring in using both schemes, however, the general qualitative relationship between results of these schemes is the same as in the non-rotated case.

5 Conclusion and Outlook

The main message of this paper is twofold:

- The FCT-methodology is readily applicable in the context of general structuring elements. With the exception of suitable interpolation formulae, there is no more technical effort than in the case of a disc-shaped structuring element.
- The quality of FCT-results is better than the quality of results using other schemes with respect to edge resolution.

The current paper represents one of the most advanced numerical approaches to continuous-scale morphology. For our future work, we aim to improve the quality of numerical schemes in this field even further.

References

1. Serra, J.: Echantillonnage et estimation des phénomènes de transition minier. PhD thesis, University of Nancy, France (1967)
2. Matheron, G.: Eléments pour une théorie des milieux poreux. Masson, Paris (1967)
3. Boris, J.P., Book, D.L.: Flux corrected transport. I. SHASTA, a fluid transport algorithm that works. Journal of Computational Physics 11(1), 38–69 (1973)
4. Matheron, G.: Random Sets and Integral Geometry. Wiley, New York (1975)
5. Boris, J.P., Book, D.L.: Flux corrected transport. III. Minimal error FCT algorithms. Journal of Computational Physics 20, 397–431 (1976)
6. Serra, J.: Image Analysis and Mathematical Morphology, vol. 1. Academic Press, London (1982)
7. Hairer, E., Norsett, S., Wanner, G.: Solving Ordinary Differential Equations. I: Nonstiff Problems. Springer Series in Computational Mathematics, vol. 8. Springer, New York (1987)
8. Serra, J.: Image Analysis and Mathematical Morphology, vol. 2. Academic Press, London (1988)
9. Osher, S., Sethian, J.A.: Fronts propagating with curvature-dependent speed: Algorithms based on Hamilton–Jacobi formulations. Journal of Computational Physics 79, 12–49 (1988)

10. Brockett, R.W., Maragos, P.: Evolution equations for continuous-scale morphology. In: Proc. IEEE International Conference on Acoustics, Speech and Signal Processing, San Francisco, CA, March 1992, vol. 3, pp. 125–128 (1992)
11. van den Boomgaard, R.: Mathematical Morphology: Extensions Towards Computer Vision. PhD thesis, University of Amsterdam, The Netherlands (1992)
12. Rouy, E., Tourin, A.: A viscosity solutions approach to shape-from-shading. SIAM Journal on Numerical Analysis 29, 867–884 (1992)
13. Sapiro, G., Kimmel, R., Shaked, D., Kimia, B.B., Bruckstein, A.M.: Implementing continuous-scale morphology via curve evolution. Pattern Recognition 26(9), 1363–1372 (1993)
14. Alvarez, L., Guichard, F., Lions, P.L., Morel, J.M.: Axioms and fundamental equations in image processing. Archive for Rational Mechanics and Analysis 123, 199–257 (1993)
15. Arehart, A.B., Vincent, L., Kimia, B.B.: Mathematical morphology: The Hamilton–Jacobi connection. In: Proc. Fourth International Conference on Computer Vision, Berlin, pp. 215–219. IEEE Computer Society Press, Los Alamitos (1993)
16. Siddiqi, K., Kimia, B.B., Shu, C.W.: Geometric shock-capturing ENO schemes for subpixel interpolation, computation and curve evolution. Graphical Models and Image Processing 59, 278–301 (1997)
17. Heijmans, H.J.A.M., Roerdink, J.B.T.M. (eds.): Mathematical Morphology and its Applications to Image and Signal Processing. Computational Imaging and Vision, vol. 12. Kluwer, Dordrecht (1998)
18. Soille, P.: Morphological Image Analysis, 2nd edn. Springer, Berlin (2003)
19. van den Boomgaard, R.: Numerical solution schemes for continuous-scale morphology. In: Nielsen, M., Johansen, P., Olsen, O.F., Weickert, J. (eds.) Scale-Space 1999. LNCS, vol. 1682, pp. 199–210. Springer, Heidelberg (1999)
20. Sethian, J.A.: Level Set Methods and Fast Marching Methods, 2nd edn. Cambridge University Press, Cambridge (1999)
21. Goutsias, J., Vincent, L., Bloomberg, D.S. (eds.): Mathematical Morphology and its Applications to Image and Signal Processing. Computational Imaging and Vision, vol. 18. Kluwer, Dordrecht (2000)
22. Talbot, H., Beare, R. (eds.): Proc. Sixth International Symposium on Mathematical Morphology and its Applications, Sydney, Australia (April 2002), `http://www.cmis.csiro.au/ismm2002/proceedings/`
23. LeVeque, R.J.: Finite Volume Methods for Hyperbolic Problems. Cambridge University Press, Cambridge (2002)
24. Osher, S., Fedkiw, R.P.: Level Set Methods and Dynamic Implicit Surfaces. Applied Mathematical Sciences, vol. 153. Springer, New York (2002)
25. Ronse, C., Najman, L., Decencière, E. (eds.): Mathematical Morphology: 40 Years On. Computational Imaging and Vision, vol. 30. Springer, Dordrecht (2005)
26. Breuß, M., Weickert, J.: A shock-capturing algorithm for the differential equations of dilation and erosion. Journal of Mathematical Imaging and Vision 25, 187–201 (2006)
27. Breuß, M., Welk, M.: Analysis of staircasing in semidiscrete stabilised inverse linear diffusion algorithms. Journal of Computational and Applied Mathematics 206, 520–533 (2007)