Universität des Saarlandes

Fachrichtung 6.1 – Mathematik

Preprint Nr. 236

Highly Accurate Schemes for PDE-Based Morphology with General Structuring Elements

Michael Breuß and Joachim Weickert

Saarbrücken 2009

Fachrichtung 6.1 – Mathematik
Universität des Saarlandes

Preprint No. 236
submitted: June 9, 2009

Highly Accurate Schemes for PDE-Based Morphology with General Structuring Elements

Michael Breuß

Saarland University
Department of Mathematics
P.O. Box 15 11 50
66041 Saarbrücken
Germany
breuss@mia.uni-saarland.de

Joachim Weickert

Saarland University
Department of Mathematics
P.O. Box 15 11 50
66041 Saarbrücken
Germany
weickert@mia.uni-saarland.de

Edited by
FR 6.1 – Mathematik
Universität des Saarlandes
Postfach 15 11 50
66041 Saarbrücken
Germany

Fax: + 49 681 302 4443
e-Mail: preprint@math.uni-sb.de
WWW: http://www.math.uni-sb.de/

Abstract

The two fundamental operations in morphological image processing are dilation and erosion. These processes are defined via structuring elements. It is of practical interest to consider a variety of structuring element shapes. The realisation of dilation/erosion for convex structuring elements by use of 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 degree of rotational invariance. We validate the FCT-scheme theoretically by proving consistency and stability. 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 [30, 22], it has evolved to a highly successful field in image processing. Many monographs and conference proceedings document this development, see e.g. [23, 31, 32, 35] and [19, 17, 37, 27, 3], 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)* [1, 2, 10, 13, 38]. 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 constructed for this general setting. It requires some efforts 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 [1, 2, 10, 38]. Here, a scaling parameter $t>0$ is introduced within the structuring element which is then given as tB, achieving digital scalability. The computation of dilation/erosion can then 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. We employ homogeneous Neumann boundary conditions at the image boundary $\partial\Omega$, i.e. $\langle \nabla u, \mathbf{n}\rangle = 0$ where $\mathbf{n}$ denotes the unit normal vector pointing to the outside of the domain, 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. Note in addition, that round structuring elements such as circles or ellipses cannot be represented conveniently in the set-theoretical approach. For instance, the arising masks do not define a granulometric family [35]. 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. In the context of dilation/erosion, the location of the evolving wavefront is determined by the shape of the scalable structuring element. We refer to [21] for a general discussion of hyperbolic PDEs and their numerical solution, and to [20] for a detailed account of hyperbolic PDEs in computer vision.
Popular numerical schemes for solving these PDEs are the *Osher-Sethian (OS) schemes* [24, 25, 33] and the *Rouy-Tourin (RT) scheme* [28, 39]. These schemes are often considered in a basic, first-order accurate format that introduces dissipative numerical artefacts. However, it is possible to construct so-called high-resolution schemes with a better accuracy. One of the OS-schemes is a second-order high-resolution method [25]. The use of a comparable high-resolution ansatz, specifically an essentially non-oscillatory (ENO) approach, was reported in [34]. In [7], Breuß and Weickert introduced a *Flux Corrected Transport (FCT) scheme* that combines high-resolution with particularly sharp evolving wavefronts. This method has later been successfully applied in the framework of tensor-valued morphology [11].
The structuring element considered in the mentioned works for the numerical solution of morphological dilation/erosion is usually a disc of radius t. The reason is surely that it simulates one of the historically first morphological evolutions called grassfire flow [4], see e.g. [29] for an extensive discussion. Moreover, since PDEs allow for rotationally invariant processes, it is natural to consider disc-shaped structuring elements for PDE-based morphology.
On the other hand, often it would also be desirable to incorporate additional flexibility in the shape of structuring elements, e.g. by choosing ellipses that are adapted to the local image structure. For these cases, not many sophisticated numerical schemes have been proposed so far. The goal of our paper is to fill this gap.

Our Contribution. We extend the applicability of the FCT-scheme introduced in [7] from discs to general convex 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 [7], namely a sharp resolution of edges and a high degree of rotational invariance, do carry over to the general case. In order to compare the performance of the FCT-scheme as well as of other PDE-based schemes 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.
This paper represents a significant extension of our conference paper [9]. Compared to that work, and in addition to the contents just stated above, we have added a detailed numerical study of important properties of the method: for instance, we investigate its accuracy depending on the anisotropy of the structuring element, and the influence of the interpolation formula used as part of the method on the quality of results. Also on the theoretical side we extend here the paper [9] considerably, as we prove that the scheme is consistent as well as stable under a mild restriction on the time step size which is very important for practical applications. Moreover, we give more details concerning the construction of our scheme.

Paper Organisation. In Section 2, we introduce classic numerical schemes important in this paper as well as the FCT-scheme for the case of a diamond as structuring element. Then we construct the FCT-scheme for general ellipses in Section 3. In Section 4 we give a theoretical discussion of the latter method. This is followed by a presentation of numerical results in Section 5. The paper is finished by a conclusion and outlook.

2 PDE-based Algorithms for Diamonds

For the sake of brevity, we discuss only dilation in detail, as the corresponding schemes for erosion are easily obtained.

Dilation via PDEs. Employing the set

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

as *structuring element*, the sought PDE describing 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 $\|\nabla u\|_\infty = \max\left(|\partial_x u|\,, |\partial_y u|\right)$.

Discretisation Basics. 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, we 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 *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.
For discretising the Neumann boundary condition $\langle \nabla u, \mathbf{n} \rangle = 0$, the corresponding difference operator from (8)-(10) is set to zero at a boundary point if it needs a pixel outside of the computational domain. This yields directional derivatives equal to zero in the direction of the outer unit normal at $\partial\Omega$.

2.1 The First-Order Scheme of Rouy and Tourin

This method, constructed in [28], is a classic scheme in the field. It is based on the concept of *upwinding* which is very important for solving hyperbolic PDEs.
We briefly illustrate the upwind concept in 1-D, at hand of the dilation PDE

$$\partial_t u \;=\; |\partial_x u| \;. \tag{11}$$

In a first step, the time derivative is discretised via a Euler forward difference, as usual for so-called explicit schemes we deal with in this paper:

$$\partial_t u\big|_{(x=ih,t=n\tau)} \;\approx\; \frac{U_i^{n+1} - U_i^n}{\tau} \,. \tag{12}$$

Then it remains to discretise $|\partial_x u|$ at the point $(ih, n\tau)$. Applying the upwind concept, we use at $(x = ih, t = n\tau)$ one-sided finite differences depending on the correct transport direction. If $\partial_x u \geq 0$, i.e. $U_{i-1}^n \leq U_i^n \leq U_{i+1}^n$, the discretisation

$$|\partial_x u| \;=\; \partial_x u \;\approx\; \frac{U_{i+1}^n - U_i^n}{h} \tag{13}$$

is correct in the sense of dilation, since a larger grey value as here U_{i+1}^n is be transported to the 'pixel' at ih. In the case $\partial_x u \leq 0$, i.e. $U_{i-1}^n \geq U_i^n \geq U_{i+1}^n$, the discretisation

$$|\partial_x u| \;=\; -\partial_x u \;\approx\; \frac{U_{i-1}^n - U_i^n}{h} \tag{14}$$

is correct in the sense of dilation, since here the larger grey value U_{i-1}^n is be transported to the 'pixel' at ih. Moreover, at a local maximum U_i^n nothing

should change, while at a local minimum we employ the largest one-sided difference. The whole procedure can be encoded in compact form in 1-D as

$$U_i^{n+1} = U_i^{n+1} + \max\left(\max\left(\frac{U_{i+1}^n - U_i^n}{h}, 0\right), \max\left(\frac{U_{i-1}^n - U_i^n}{h}, 0\right)\right) . \quad (15)$$

The formula (15) is exactly the Rouy-Tourin (RT) scheme.
In order to generalise the latter formula and employ it in 2-D for diamond structuring elements, we define

$$\hat{D}_-^{x,-} U_{i,j}^n := \frac{1}{h_x} \max\left(-D_-^x U_{i,j}^n, 0\right) , \quad (16)$$

$$\hat{D}_+^{x,+} U_{i,j}^n := \frac{1}{h_x} \max\left(D_+^x U_{i,j}^n, 0\right) , \quad (17)$$

and we use the abbreviation

$$\delta_x^{\mathrm{RT}} U_{i,j}^n := \max\left(\hat{D}_-^{x,-} U_{i,j}^n , \hat{D}_+^{x,+} U_{i,j}^n\right) , \quad (18)$$

$\delta_y^{\mathrm{RT}} U_{i,j}^n$ being defined accordingly. Then the RT-scheme reads in our case of a diamond-shaped structuring element as

$$U_{i,j}^{n+1} = U_{i,j}^n + \tau \max\left(\left|\delta_x^{\mathrm{RT}} U_{i,j}^n\right|, \left|\delta_y^{\mathrm{RT}} U_{i,j}^n\right|\right) . \quad (19)$$

2.2 The High-Resolution Osher-Sethian Scheme

The basis of the high-resolution method of Osher and Sethian is given by a first-order method very similar to the RT-scheme. In fact, for discretising a 1-D dilation PDE as elaborated on when introducing the RT-scheme, only the treatment of local minima works differently in the first-order OS scheme which amounts to add up both one-sided differences in that case. For constructing the high-resolution OS-scheme, the one-sided differences $D_-^x U_{i,j}^n$ and $D_+^x U_{i,j}^n$ are augmented by a suitable higher-order correction. Also, a more accurate discretisation than Euler forward is used for time integration. In what follows, we will refer to this augmented method as the OS-scheme.

Spatial Discretisation. For the definition of the scheme, we employ the *minmod function*. The minmod function gives back the minimal modulus of its arguments:

$$\mathrm{mm}(a,b) := \begin{cases} \min(a,b) & \text{if} \quad a > 0 \text{ and } b > 0 , \\ \max(a,b) & \text{if} \quad a < 0 \text{ and } b < 0 , \\ 0 & \text{else} . \end{cases} \quad (20)$$

This function is used to compute a cautious discrete version of second-order derivatives via:

$$\hat{D}^{xx,-}_{i,j} := \mathrm{mm}\left(D^x_- D^x_+ U^n_{i,j},\, D^x_- D^x_- U^n_{i,j}\right) , \tag{21}$$

$$\hat{D}^{xx,+}_{i,j} := \mathrm{mm}\left(D^x_+ D^x_+ U^n_{i,j},\, D^x_- D^x_+ U^n_{i,j}\right) . \tag{22}$$

Then, within the following discrete derivative operators, the high-resolution correction of first-order finite differences $D^x_- U^n_{i,j}$ and $D^x_+ U^n_{i,j}$ is given:

$$\delta^{\mathrm{OS}-}_x U^n_{i,j} := \frac{1}{h_x} \min\left(D^x_- U^n_{i,j} + \frac{1}{2}\hat{D}^{xx,-}_{i,j},\, 0\right) , \tag{23}$$

$$\delta^{\mathrm{OS}+}_x U^n_{i,j} := \frac{1}{h_x} \max\left(D^x_+ U^n_{i,j} - \frac{1}{2}\hat{D}^{xx,+}_{i,j},\, 0\right) , \tag{24}$$

and we set analogously $\delta^{\mathrm{OS}-}_y U^n_{i,j}$ and $\delta^{\mathrm{OS}+}_y U^n_{i,j}$.
Now, we can put together the components of the OS discretisation of the right hand side of (6). For a compact notation we set

$$\begin{aligned} L\left(U^n, i, j\right) := \max\big(&\left|\delta^{\mathrm{OS}-}_x U^n_{i,j}\right| + \left|\delta^{\mathrm{OS}+}_x U^n_{i,j}\right| , \\ &\left|\delta^{\mathrm{OS}-}_y U^n_{i,j}\right| + \left|\delta^{\mathrm{OS}+}_y U^n_{i,j}\right|\big) , \end{aligned} \tag{25}$$

which realises the maximum norm on the discrete level. Let us briefly comment on the 'double' contributions of the discretised derivatives in (25), for instance in x-direction: $\left|\delta^{\mathrm{OS}-}_x U^n_{i,j}\right| + \left|\delta^{\mathrm{OS}+}_x U^n_{i,j}\right|$. 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 $U^n_{i,j}$, both summands could be non-zero.

Temporal Discretisation. Because of the second-order corrections in (23)-(24), 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 [18]:

$$\begin{aligned} \bar{U}^{n+1}_{i,j} &= U^n_{i,j} + \tau\, L\left(U^n, i, j\right) \\ U^{n+1}_{i,j} &= \frac{1}{2}U^n_{i,j} + \frac{1}{2}\bar{U}^{n+1}_{i,j} + \frac{\tau}{2} L\left(\bar{U}^{n+1}, i, j\right) . \end{aligned} \tag{26}$$

Note that this is not an arbitrary choice for second-order time stepping: Especially in the context of hyperbolic PDEs as discussed here, the method of Heun preserves desirable stability properties, see e.g. [15, 16].

2.3 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 two-stage Runge-Kutta method for time integration, the construction of the FCT method works differently.
As a predictor step, a first-order scheme is used for wave propagation. Thus, the predictor features desirable theoretical properties but also introduces much artificial dissipation. Then, by taking into account the so-called *viscosity form* [36] of the predictor scheme, the dissipation can be quantified on a discrete level and is negated in a second step using stabilised inverse diffusion [8]. For details we refer to [7].
The basic idea to negate dissipation by a corrector step goes back to Boris and Book [5, 6]. However, in their works the corrector step was constructed based on an analytical approach, using an analysis of the local truncation error of the predictor scheme. In contrast, by relying on the viscosity form we employ a fully discrete approach. Following their procedure would lead to a different and less attractive scheme than with our construction, since our scheme not only negates dissipation, it automatically incorporates a nonlinear mechanism that keeps evolving fronts sharp. The latter component is effectively a small additional stabilised backward dissipation. One may show this easily by a local truncation error analysis of the predictor and the corrector step. We refrain to elaborate on the use of this standard technique in detail here.

The Predictor Step. As a predictor step we use the first-order-accurate scheme proposed by Rouy and Tourin [28], see (19). For the convenience of the reader, we write it down again for a diamond-shaped structuring element:

$$\bar{U}_{i,j}^{n+1} = U_{i,j}^{n} \; + \; \tau \; \max\left(\left|\delta_x^{\mathrm{RT}} U_{i,j}^{n}\right|, \left|\delta_y^{\mathrm{RT}} U_{i,j}^{n}\right| \right) \; . \tag{27}$$

Here $\bar{U}^{n+1}$ denotes the set of predicted data.

The Corrector Step. The FCT-scheme consists of a subsequent application of (27) and a corrector step, reading in the total as

$$\left\{ \begin{array}{l} \bar{U}_{i,j}^{n+1} \; = \; U_{i,j}^{n} \; + \; \tau \; \max\left(\left|\delta_x^{\mathrm{RT}} U_{i,j}^{n}\right|, \left|\delta_y^{\mathrm{RT}} U_{i,j}^{n}\right| \right) \; , \\ U_{i,j}^{n+1} \; = \; \bar{U}_{ij}^{n+1} + \bar{C}_h - \bar{C}_d \, . \end{array} \right. \tag{28}$$

Let us consider the corrector step and especially the functions $\bar{C}_h$ ('h' as it arises by identifying a high-order component within the dissipation of the

RT scheme) and $\bar{C}_d$ ('d' by identifying it via the dissipative basis scheme) 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): The local truncation error of the central finite difference operator $D_c^x U_{i,j}^n$ reveals that no first-order error and therefore no dissipation is present. Thus, the discretisation of the dilation PDE (6) using central differences only, via

$$\|\nabla u\|_\infty \approx \max\left(\left| \frac{\tau}{2h_x} D_c^x U_{i,j}^n \right|, \left| \frac{\tau}{2h_y} D_c^y u_{i,j}^n \right| \right), \tag{29}$$

incorporates no numerical dissipation in the approximation of spatial derivatives.
We now undertake a step which seems at first glance redundant, but it is an important step in the FCT strategy. We add and subtract

$$C_h := \max\left(\left| \frac{\tau}{2h_x} D_c^x U_{i,j}^n \right|, \left| \frac{\tau}{2h_y} D_c^y u_{i,j}^n \right| \right) \tag{30}$$

on the right hand side of the RT formula (27), rewriting in this way what needs to be added for the update as

$$\begin{aligned} &\tau \ \max\left(\left|\delta_x^{\mathrm{RT}} U_{i,j}^n\right|, \left|\delta_y^{\mathrm{RT}} U_{i,j}^n\right| \right) \\ &\quad = \ \tau \ \max\left(\left|\delta_x^{\mathrm{RT}} U_{i,j}^n\right|, \left|\delta_y^{\mathrm{RT}} U_{i,j}^n\right| \right) - C_h + C_h \,. \end{aligned} \tag{31}$$

We now have a closer look at (31), aiming to decompose the right hand side into a higher order and a dissipative part. To state the idea behind this clearly, we thus decompose indirectly the left hand side of (31), that is identical to the update in (27), into a higher order term and into a term which is *identical* to a dissipative error.
By the non-dissipative properties of (29), we can identify a high-order part of formula (31) – and thus of the predictor scheme – as C_h. Consequently, the numerical dissipation of the predictor is basically contained in the part

$$\tau \ \max\left(\left|\delta_x^{\mathrm{RT}} U_{i,j}^n\right|, \left|\delta_y^{\mathrm{RT}} U_{i,j}^n\right| \right) - C_h \tag{32}$$

of the update formula.
As indicated at the beginning of this paragraph, we now want to negate the dissipation in the scheme identified by (32) in order to realise the FCT idea. This means that we want to perform now a *backward diffusion* step.
The first crucial point in this is that the predicted data $\bar{U}^{n+1}$ inhibit the numerical dissipation after the predictor step. Thus, in the corrector step we

aim to subtract (32) evaluated using data from the set $\bar{U}^{n+1}$ instead of the set U^n. For the second term in (32), this can be done in a straight forward fashion, setting

$$\bar{C}_h := \max\left(\left| \frac{\tau}{2h_x} D_c^x \bar{U}_{i,j}^{n+1} \right|, \left| \frac{\tau}{2h_y} D_c^y \bar{U}_{i,j}^{n+1} \right| \right) , \tag{33}$$

see (30) and the corresponding contribution in (28).
Secondly, as discussed in detail in [7, 8], the backward diffusion step needs to incorporate a *stabilisation*, as otherwise it creates oscillations and is ill-posed. From the discussion in [7] it is known how the stabilisation should be constructed. We briefly illustrate here the idea in the same 1-D setting as used in Subsection 2.1. One may rewrite the RT-scheme for dilation in 1-D, (15), in terms of its *viscosity form*:

$$U_i^{n+1} = \begin{cases} U_i^n \quad \text{for} \quad U_i^n - U_{i-1}^n \geq 0,\, U_{i+1}^n - U_i^n \leq 0, \\ \text{or else:} \\ U_i^n + \dfrac{\tau}{2h} \left| U_{i+1}^n - U_{i-1}^n \right| \\ \qquad + \dfrac{\tau}{2h} \left(U_{i+1}^n - U_i^n \right) - \dfrac{\tau}{2h} \left(U_i^n - U_{i-1}^n \right) . \end{cases} \tag{34}$$

The latter formula incorporates a central difference which is second-order accurate. This means, one can identify by the so-called viscous fluxes

$$+\frac{\tau}{2h} \left(U_{i+1}^n - U_i^n \right) \text{ and } -\frac{\tau}{2h} \left(U_i^n - U_{i-1}^n \right) \tag{35}$$

terms that are responsible for a dissipation due to the discretisation. Exactly these viscous fluxes we negate in our FCT-approach. Using in 1-D the predicted values $\bar{U}^{n+1}$, we correct $\bar{U}_i^{n+1}$ by a stabilised version of

$$-\frac{\tau}{2h} \left(\bar{U}_{i+1}^n - \bar{U}_i^n \right) \text{ and } +\frac{\tau}{2h} \left(\bar{U}_i^n - \bar{U}_{i-1}^n \right) , \tag{36}$$

respectively. For stabilisation, the minmod function is employed which limits the viscous fluxes in (36) so that no over- and undershoots are generated.
We now fix the stabilisation step in the 2-D setting. Extending the minmod function from (20) in a straightforward way to three arguments, we set

$$G_{i+1/2,j} := \operatorname{mm}\left(D_-^x \bar{U}_{i,j}^{n+1},\, \bar{Q}_{ij}^x,\, D_+^x \bar{U}_{i+1,j}^{n+1} \right) , \tag{37}$$

$$G_{i,j+1/2} := \operatorname{mm}\left(D_-^y \bar{U}_{i,j}^{n+1},\, \bar{Q}_{ij}^y,\, D_+^y \bar{U}_{i,j+1}^{n+1} \right) , \tag{38}$$

where

$$\bar{Q}_{ij}^x := \frac{\tau}{2h_x} D_+^x \bar{U}_{i,j}^{n+1} , \quad \bar{Q}_{ij}^y := \frac{\tau}{2h_y} D_+^y \bar{U}_{i,j}^{n+1} . \tag{39}$$

The left and right arguments in (37)-(38) are supposed to prevent over- and undershoots by limiting the backward viscous fluxes that are identical to the middle arguments by the local grey value differences. For defining now the second component of the correction term $-\bar{C}_d$ identical to a stabilised version of

$$-\tau \max\left(\left|\delta_x^{\mathrm{RT}} \bar{U}_{i,j}^{n+1}\right|, \left|\delta_y^{\mathrm{RT}} \bar{U}_{i,j}^{n+1}\right| \right) \tag{40}$$

we employ these stabilised fluxes, yielding

$$\delta_x^{bd} \bar{U}_{i,j}^{n+1} := \frac{\tau}{2h_x} \left|D_c^x \bar{U}_{i,j}^{n+1}\right| + G_{i+1/2,j} - G_{i-1/2,j}, \tag{41}$$

$$\delta_y^{bd} \bar{U}_{i,j}^{n+1} := \frac{\tau}{2h_y} \left|D_c^y \bar{U}_{i,j}^{n+1}\right| + G_{i,j+1/2} - G_{i,j-1/2}\,, \tag{42}$$

and finally:

$$\bar{C}_d := \max\left(\left|\delta_x^{bd} \bar{U}_{i,j}^{n+1}\right|, \left|\delta_y^{bd} \bar{U}_{i,j}^{n+1}\right| \right)\,. \tag{43}$$

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

$$\left(1/a^2\right) x^2 + \left(1/b^2\right) y^2 \;=\; 1\,. \tag{44}$$

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

$$\partial_t u \;=\; \sqrt{a^2 \left(\partial_x u\right)^2 + b^2 \left(\partial_y u\right)^2} \tag{45}$$

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

The General Idea of the Algorithm. In order to solve for dilation with a general ellipse via (45), we need to consider in each individual pixel the *translation* and *rotation* of the corresponding ellipse to its normal form representation. It is elementary that by these two operations all possible ellipses can be addressed.

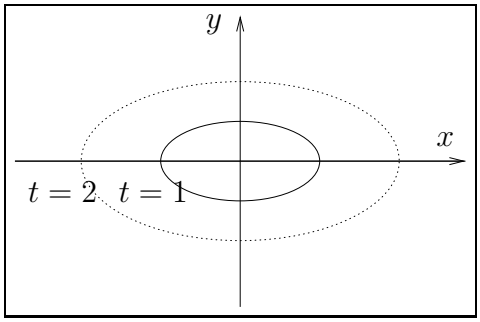

Figure 1: **Continuous line.** Normal form of an ellipse, $a = 1$, $b = 1/2$, with principal axis aligned with the x-axis, compare (44). **Dotted line.** Snapshot of corresponding front evolution.

Consequently, the basic algorithm collects 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 (45). By this procedure, we realise *(i)* the translation of each pixel center to $(0, 0)^T$ and *(ii)* the rotation of the principal axis of the ellipse in each point to its normal representation.

Geometric Implementation. Having outlined the basic idea, we now consider for general ellipses as structuring elements the 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 (45), 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)-(27) 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 $(\cos \alpha_k, \sin \alpha_k)^T$, $\alpha_k := -\alpha + k \cdot \frac{\pi}{2}$, $k = 0, 1, 2, 3$. In using this procedure, we effectively consider an ellipse where the angle between x-axis and principal axis is α. See Figure 2 for a graphical representation.
Via $0 \leq \alpha \leq \pi/2$ we can obtain *all* possible ellipses, as we can switch at any time the roles of a and b in (44) that define the principal axis. It is just

practical to impose $0 \leq \alpha \leq \pi/2$ since this helps to give suitable interpolation formulae, which is the next step.

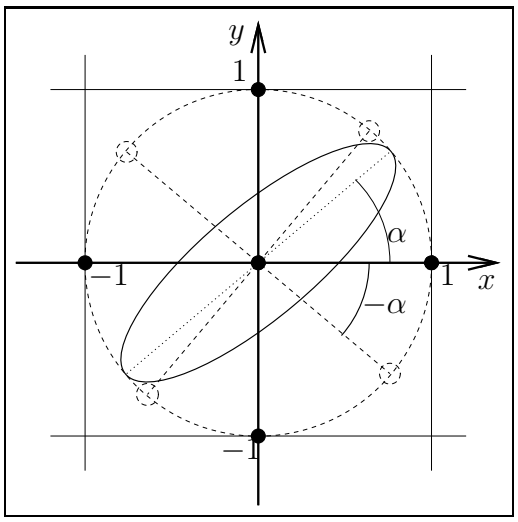

Figure 2: **Local geometry.** The dotted circle represents the unit circle with radius $h_x = h_y = 1$. The ellipse represents the boundary of an ellipsoid structuring element. The major axis of the ellipse is indicated by dashed lines.
Computational points. The stencil points of the RT scheme are given by black circles. The points from which we need input values for the predictor scheme are indicated by dashed spheres.

Interpolation. Obviously, we need at each pixel the grey values after rotation for defining our finite difference scheme. In Figure 3 (left) it is illustrated which values are needed within the stencil of the OS-scheme as well as in the corrector step of the FCT finite difference method.
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. Also, we test bicubic interpolation in order to study the influence of the interpolation formula on the quality of our numerical solution, and to obtain a clear impression of the rotational invariance of our schemes.
In order to show how the computation works, we now exemplify the details for the values in the fourth quadrant, see Figure 3 (right). 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

$$\begin{pmatrix} \cos\alpha_0 \\ \sin\alpha_0 \end{pmatrix} = \begin{pmatrix} \cos(-\alpha) \\ \sin(-\alpha) \end{pmatrix} = \begin{pmatrix} \cos\alpha \\ -\sin\alpha \end{pmatrix} . \tag{46}$$

Because of $h_x = h_y = 1$, we can use the general formula for bilinear interpolation of some function $g(x, y)$ over the rectangle $[0, 1] \times [0, 1]$ reading as

$$\begin{aligned} g(x,y) \;\approx\; & g(0,0)(1-x)(1-y) + g(1,0)x(1-y) \\ & + g(0,1)(1-x)y + g(1,1)xy\,. \end{aligned} \tag{47}$$

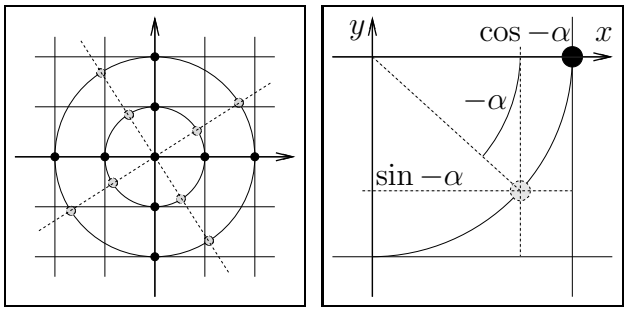

Figure 3: **(a) Left.** Sketch of the points in the stencil of the tested methods (black circles) and points needed for interpolation (grey circles). **(b) Right.** Visualisation of bilinear interpolation example in the fourth quadrant.

Considering our values within the fourth quadrant, we obtain for use in (47):

$$\tilde{x} \;:=\; \cos\alpha\,, \quad \tilde{y} \;:=\; -\sin(-\alpha)\,. \tag{48}$$

Note that $\tilde{y}$ should be non-negative for use within (47). Also, since we consider the fourth quadrant, we need to be a bit careful in choosing which value is plugged in at the corners of the rectangle, as the rectangle is practically $[0, 1] \times [-1, 0]$ there, see again Figure 3 (right). We then compute the rotated grey value $\tilde{U}^n_{i+1,j}$ as

$$\begin{aligned} \tilde{U}^n_{i+1,j} \;:=\; & U^n_{i,j}\tilde{x}\tilde{y} + U^n_{i+1,j}\,(1-\tilde{x})\,\tilde{y} \\ & + U^n_{i,j-1}\tilde{x}\,(1-\tilde{y}) + U^n_{i+1,j-1}\,(1-\tilde{x})\,(1-\tilde{y})\,. \end{aligned} \tag{49}$$

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

employ follows the same principle in rotating the grid locally. The technique itself is standard as is the bilinear interpolation, see e.g. [14] for a detailed exposition. We refrain from giving the bicubic formulae here.
A comparison with Subsection 2.3 shows what needs to be done. We summarise the algorithm as follows:

Step 1

1. From the set of given data U^n compute the rotated grey values
$$\left\{\tilde{U}^n_{i,j}, \tilde{U}^n_{i+1,j}, \tilde{U}^n_{i-1,j}, \tilde{U}^n_{i,j+1}, \tilde{U}^n_{i,j-1}\right\} \tag{50}$$
as outlined above.

2. Perform the predictor step
$$\bar{U}^{n+1}_{i,j} = \tilde{U}^n_{i,j} \,+\, \tau \,\sqrt{a^2\left(\delta_x^{\mathrm{RT}}\tilde{U}^n_{i,j}\right)^2 + b^2\left(\delta_y^{\mathrm{RT}}\tilde{U}^n_{i,j}\right)^2} \tag{51}$$

Step 2

1. Analogously to the outlined procedure compute a set of rotated grey values
$$\left\{\tilde{U}^{n+1}_{i+k,j+l}\,\middle|\; k,l \in \{-2,-1,0,1,2\}\right\} \tag{52}$$
from the set of predicted data $\bar{U}^{n+1}$.

2. Perform the corrector step
$$U^{n+1}_{i,j} \;=\; \bar{U}^{n+1}_{ij} + \tilde{C}_h - \tilde{C}_d \tag{53}$$
with
$$\tilde{C}_h \;=\; \frac{\tau}{2h_x}\sqrt{a^2\left(D_c^x\tilde{U}^{n+1}_{i,j}\right)^2 + b^2\left(D_c^y\tilde{U}^{n+1}_{i,j}\right)^2}, \tag{54}$$
$$\tilde{C}_d \;=\; \sqrt{a^2\left(\delta_x^{bd}\tilde{U}^{n+1}_{i,j}\right)^2 + b^2\left(\delta_y^{bd}\tilde{U}^{n+1}_{i,j}\right)^2}. \tag{55}$$

Note that for the arguments of the minmod-function used within $\tilde{C}_d$, we also need to compute rotated grey values $\tilde{U}^{n+1}_{i\pm2,j\pm2}$ from the data set $\bar{U}^{n+1}$, see Figure 3 (left). This results in the need to consider all the points in (52). The computation can be done in the same fashion as presented above using bilinear or bicubic interpolation.
The presented procedure involves some technical efforts. However, it was shown in a matrix-valued setting yet with a simpler basic scheme that an additional effort is necessary in order to capture anisotropic evolution components with the numerics [26].

4 Theoretical Analysis

In this section we collect useful theoretical assertions that can be verified for the FCT-scheme for general ellipses as structuring elements. By the methodology of the proofs, it is clear that analogous assertions can be shown for other structuring elements, too.

Proposition[Consistency]The FCT-scheme for general ellipses is consistent to the PDE of this problem.

Proof. We only sketch the proof as the individual steps are easy to follow. Let us note, that consistency means that the sought PDE is satisfied by the discrete representation up to an error of $O(h)$ or smaller, where h is a characteristic grid parameter, usually the maximum of h_x, h_y and τ.
The proof is performed in three steps.

(i) By interpolating on the rotated grid, the data are represented correctly with the exception of the interpolation error, so that consistency is kept.

(ii) The predictor scheme relies on upwinding, so that it is consistent to the PDE formulated in rotated coordinates (45), compare [28].

(iii) By its derivation in [7] the stabilised backward diffusion negates the local truncation error introduced by upwinding. Any amount of stabilised backward dissipation generated in this step that deviates from the local truncation error is an error in $O(h)$.

Thus, the complete scheme is consistent by construction.

Of much practical interest is the choice of the time step size related to the stability of the scheme.

Theorem[Extremum Principle]The FCT-scheme for general ellipses making use of bilinear interpolation is stable, i.e. it satisfies a discrete maximum principle under the restriction on the time step size

$$\tau \leq \frac{1}{\sqrt{2}}\,. \tag{56}$$

Proof. By the investigations in [7], one can easily infer that the stability restriction on the time step size due to the predictor step dominates the stability bound arising by the stabilised backward diffusion step. Thus, we focus on the predictor method, i.e. the RT-scheme.
For the *first step* in proving stability we slightly rearrange the update formula of the RT-scheme (51) as

$$\bar{U}_{i,j}^{n+1} - \tilde{U}_{i,j}^{n} \;=\; \tau \sqrt{a^2 \left(\delta_x^{\mathrm{RT}} \tilde{U}_{i,j}^{n}\right)^2 + b^2 \left(\delta_y^{\mathrm{RT}} \tilde{U}_{i,j}^{n}\right)^2} \tag{57}$$

The idea is, **if** we can assure that the difference between old and new value, i.e. $\bar{U}_{i,j}^{n+1} - \tilde{U}_{i,j}^{n}$, is smaller than the the maximal local data difference which we define as

$$\begin{aligned} \delta_{max}\tilde{U}_{i,j}^{n} \;\; := \;\; \max \Bigg(& \max_{k\in\{-1,1\}} \left| \tilde{U}_{i+k,j}^{n} - \tilde{U}_{i,j}^{n} \right| , \\ & \max_{l\in\{-1,1\}} \left| \tilde{U}_{i,j+l}^{n} - \tilde{U}_{i,j}^{n} \right| \Bigg) , \end{aligned} \tag{58}$$

then

$$\bar{U}_{i,j}^{n+1} \;\in\; \mathrm{conv}\left(\tilde{U}_{i,j}^{n}, \tilde{U}_{i-1,j}^{n}, \tilde{U}_{i+1,j}^{n}, \tilde{U}_{i,j-1}^{n}, \tilde{U}_{i,j+1}^{n} \right) \tag{59}$$

holds, where conv denotes the convex hull of the arguments. The latter formula implies the validity of a discrete maximum principle with respect to the data set $\tilde{U}^n$ since it guarantees that no over- or undershoots are generated. We now proceed along this line. Using (57), the sought relation $\bar{U}_{i,j}^{n+1} - \tilde{U}_{i,j}^{n} \leq \delta_{max}\tilde{U}_{i,j}^{n}$ is equivalent to

$$\tau \sqrt{a^2 \left(\delta_x^{\mathrm{RT}} \tilde{U}_{i,j}^{n} \right)^2 + b^2 \left(\delta_y^{\mathrm{RT}} \tilde{U}_{i,j}^{n} \right)^2} \;\leq\; \delta_{max}\tilde{U}_{i,j}^{n} \, . \tag{60}$$

At first, we consider the argument of the root on the left hand side of (60). By the formulae (16)-(18) we can estimate

$$\begin{aligned} & a^2 \left(\delta_x^{\mathrm{RT}} \tilde{U}_{i,j}^{n} \right)^2 + b^2 \left(\delta_y^{\mathrm{RT}} \tilde{U}_{i,j}^{n} \right)^2 \\ & \leq \;\; \left(a^2 + b^2 \right) \max \left(\left(\delta_x^{\mathrm{RT}} \tilde{U}_{i,j}^{n} \right)^2 , \left(\delta_y^{\mathrm{RT}} \tilde{U}_{i,j}^{n} \right)^2 \right) \\ & \leq \;\; \left(a^2 + b^2 \right) \left(\frac{1}{\min\left(h_x, h_y\right)} \delta_{max}\tilde{U}_{i,j}^{n} \right)^2 , \end{aligned} \tag{61}$$

so that

$$\begin{aligned} & \tau \sqrt{a^2 \left(\delta_x^{\mathrm{RT}} \tilde{U}_{i,j}^{n} \right)^2 + b^2 \left(\delta_y^{\mathrm{RT}} \tilde{U}_{i,j}^{n} \right)^2} \\ & \leq \;\; \tau \sqrt{a^2 + b^2} \frac{\delta_{max}\tilde{U}_{i,j}^{n}}{\min\left(h_x, h_y\right)} \end{aligned} \tag{62}$$

Comparing (62) to the relation (60) we are heading to, the *sufficient condition*

$$\tau \;\leq\; \frac{\min\left(h_x, h_y\right)}{\sqrt{a^2 + b^2}} \tag{63}$$

arises. By the geometry of our ellipses, we always have in our set-up $a = 1$ and $b \in [0, 1]$, or vice versa, so that $a^2 + b^2 \leq 2$. Employing $h_x = h_y = 1$ we obtain the sought formula (56).
To conclude the proof, we need to confirm that the interpolated data $\tilde{U}^n$ used up to now also obey a discrete extremum principle. This holds since the bilinear interpolation formula (47) is given by a convex combination of values in the corners of the unit rectangle. Thus, the FCT-scheme satisfies at (i, j) locally a discrete extremum principle with respect to the set $\big\{U^n_{i+k,j+l} \mid k, l \in \{-1, 0, 1\}\big\}$.

Remarks. *(i)* For $a = b = 1$, i.e. dilation with a circle, we retrieve from (55) the condition shown in [7]. *(ii)* Selecting without a loss of generality $a = 1$ in (44) corresponding to the major axis $b \leq 1$ follows. This means, the anisotropy of the structuring element does not influence formally the allowed time step size. This is an important property of the method: It confirms the generality of the approach. However, for extremely anisotropic structuring elements the accuracy (not the stability) deteriorates if the time step size is chosen close to the theoretical limit. *(iii)* For higher-order interpolation such as cubic interpolation, we cannot verify an analogous result, as cubic splines may create over- or undershoots which prevents in general the validity of a discrete extremum principle.

5 Numerical Experiments

In this section we show that the FCT-scheme gives better results for PDE-based morphology than other schemes in the field. First, we compare it with a standard algorithm arising by the set-theoretic approach. Thereby, we also show how the first-order RT-scheme and the second-order OS-scheme perform in comparison. Secondly, the FCT-scheme is compared with the second-order OS-scheme for ellipses as structuring elements. After that we perform a study showing the influence of the time step size and the interpolation formula in use, and we also consider strongly anisotropic structuring elements.

Comparison with Set-based Algorithm. In this experiment, we solve the PDE (7) realising dilation with a diamond, comparing the PDE-based schemes with the set-theoretical approach. For convenience, we always employ $h_x = h_y = 1$.

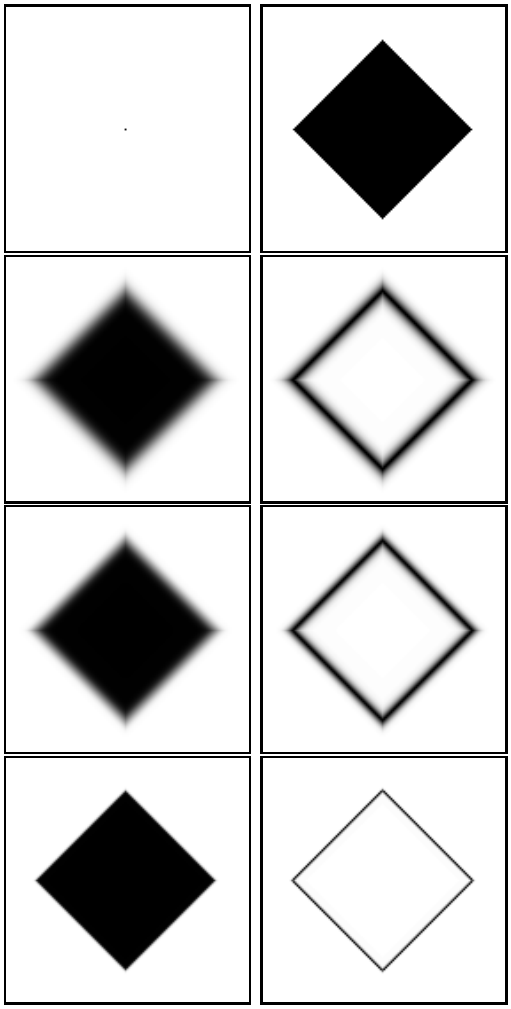

Figure 4: Comparison of schemes for dilation with a diamond using inverted grey values. **(a) Top Row.** Left: Initial image. Right: Set-based result. **(b) Second Row.** Left: RT-result. Right: Scaled difference. The average difference to the result of the set-based method is of the grey value 12.421. **(c) Third Row.** Left: OS-result. Right: Scaled difference. The average difference here is 9.486. **(d) Bottom Row.** Left: FCT-result. Right: Scaled difference. The average difference visualised here is 1.502.
The images in rows (a) and (d) are adopted from [9], the other tests are new.

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

$$(1,0)^T,\ (0,1)^T,\ (-1,0)^T,\ (0,-1)^T\,. \tag{64}$$

In Figure 4, the outcome of this experiment is shown, where we have inverted the grey values. As input image, an image of size 129×129 is used where we 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 the PDE-based schemes, and 50 iterations with the set-based algorithm, respectively.
We observe that the RT-result (b) is very blurry, as expected since it is a first-order method. The result obtained with the second-order OS-scheme shown in (c) is, however, only a little bit better. The FCT-result (d) is visually nice, with very sharp diamond edges. In comparison with the set-based result (a), we observe that there is some difference at the edges, which can be seen in the scaled difference map in (d). Note that the average (unscaled) difference amounts to a barely noticable grey value of 1.502 in the case of the FCT-scheme. However, let us also note that the exact solution of the PDE is digitally scalable, so that the set-based solution here is not identical to the true solution of the dilation PDE. This experiment shows that the FCT-scheme is capable to give results of comparable quality to set-based methods, while the other tested PDE-based schemes perform significantly weaker.

Comparison with the OS-scheme. We now show computational results for ellipses as structuring elements.
In order to give an impression of what quality one may expect, we first consider an ellipse where the principal axis is aligned with the grid. In this case no interpolation for using a rotated grid is needed. The structuring element is defined here via $a = 1$, $b = 0.2$, compare (44). For the numerical experiment, we use again $\tau = 0.5$, and we perform 150 time steps using the initial image from Figure 4(a). The results of the OS-scheme together with the result of the FCT-scheme are displayed in Figure 5. 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 speed in this example are well-resolved, there is some blurring on the slow-moving upper and lower part of the edge of the ellipse.
We now consider also the rotated case employing bilinear interpolation. For this experiment we use the same parameters as before, but (a) $\alpha = 0.15$, (b) $\alpha = 0.3$. We obtain the results displayed in Figure 6. 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.

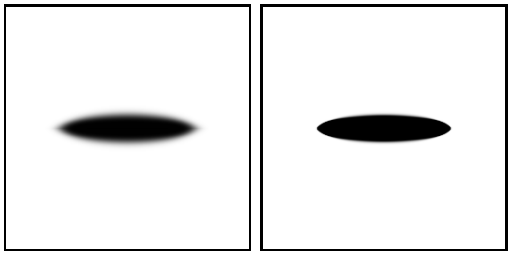

Figure 5: Dilation comparison with inverted grey values. **(a) Left.** OS-result. **(b) Right.** FCT-result. A similar test was done in [9].

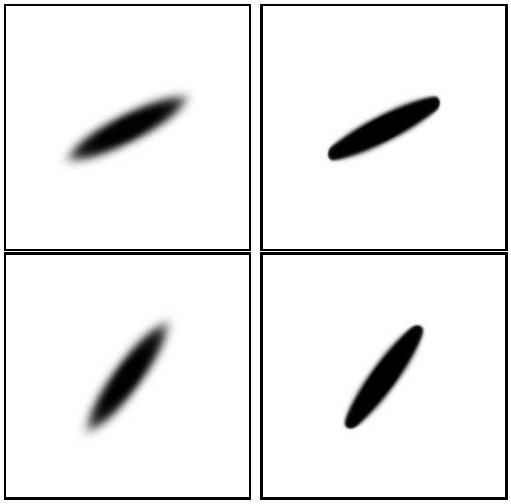

Figure 6: Dilation comparison with inverted grey values. Rotated ellipses with (top) $\alpha = 0.15$ and (bottom) $\alpha = 0.3$. Bilinear interpolation was used to rotate locally the grid in each time step. **(a) Left column.** OS-results. **(b) Right column.** FCT-results. A similar test was done in [9].

Discussion of Computational Settings for FCT. Setting up the FCT-scheme, an important issue is the influence of several choices that need to be made: *(i)* the order of the interpolation formula, *(ii)* the time step size, and *(iii)* the allowed anisotropy of structuring elements. We now elaborate on exactly these topics which was not done in [9]. Performing the computations, it turns out that it makes sense to illuminate relationships between these points.

We begin with considering the *interpolation formula*. As shown by our Theorem, bilinear interpolation is useful for proving the stability of the FCT-scheme. Also noted in Section 4, bicubic interpolation may generally give interpolation values not in the convex hull of interpolated data, so that the

stability of the scheme in terms of a discrete extremum principle can not be guaranteed. The question arises, if there is nevertheless an accuracy benefit in using a higher-order interpolation formula such as bicubic interpolation. For an answer to the latter question, we compare in two experiments FCT results using bicubic and bilinear interpolation, respectively.
In the first experiment we use again the ellipse parameters $a = 1$, $b = 0.2$, $\alpha = 0.15$. However, this time we perform 100 time steps with $\tau = 0.5$, and zoom onto the solution in order to show more details. The corresponding dilation results with inverted grey values are shown in Figure 7. We clearly observe a quality gain using the bicubic interpolation.
However, generating over- or undershoots within the bicubic interpolation step translates into larger computed spatial gradients, and this translates into larger updates when compared to bilinear interpolation, see especially (50)-(51). This again means that over- or undershoots are generated in the numerical result. The effect of potentially larger values from the spatial gradient computation used for the update can be partially off-set by choosing a smaller time step size, compare (51). This logic needs to be applied especially in the context of images with a fine structure leading to rapidly varying gradients. It also holds for very anisotropic structuring elements, since in this case even simple initial images may lead to fine structures after a few iterations.

Figure 7: Comparison of interpolation order used in the scheme, images are of size 115×115. Parameters are $a = 1$, $b = 0.2$, $\alpha = 0.15$; results are given after 100 time steps with $\tau = 0.5$. **(a) Left.** Bilinear interpolation. **(b) Right.** Bicubic interpolation.
New test not performed in [9].

We employ these considerations when considering now an extremely *anisotropic structuring element* defined via $a = 1$ and $b = 0$, i.e. we effectively aim for a directional dilation along a single line, see Figure 8. The images are of size 256×256. While we observe some blurring of the desired line, especially for bilinear interpolation, the accuracy is quite allright since the algorithm was

not designed specifically for this extreme purpose. In order to obtain a reasonable quality gain by bicubic interpolation in Figure 8, we need to employ about one tenth of the step size used in the bilinear setting. Otherwise large oscillations and a useless numerical solution are generated.

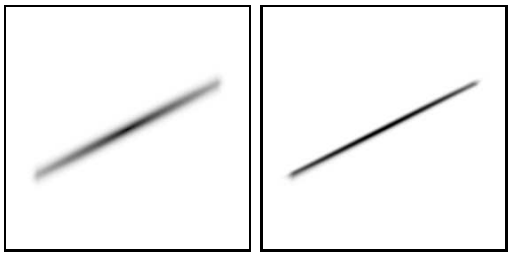

Figure 8: Comparison of interpolation order at hand of directional dilation displayed via inverted grey values via the parameters $a = 1$, $b = 0$. **(a) Left.** Bilinear interpolation, $\tau = 0.1$. **(b) Right.** Bicubic interpolation, $\tau = 0.01$. New test not performed in [9].

Staying with the bilinear interpolation in FCT, it remains to ask for *suitable time step sizes*. It is of practical interest to consider time step sizes as large as possible, since the number of time steps directly relate to the computational time. As our stability theorem shows, the theoretical maximum is $\tau = 1/\sqrt{2}$. At hand of an anisotropic example we explore the effect of a time step size close to this value vs. a small time step size, see Figure 9. We choose $a = 1$, $b = 0$, $\alpha = 0.15$, and evaluate the dilation process at $t = 40$. Also, we zoom onto the solutions, giving images of size 79×80.
We observe for the large time step size artefacts due to the bilinear interpolation at the ends of the 'ellipse'. This behaviour is not observable for small time step sizes. Let us note that the interpolation artefacts disappear for time step sizes $\tau = 0.5$ or smaller, while the solution is about the same as in Figure 9(b) if we employ smaller time step sizes.

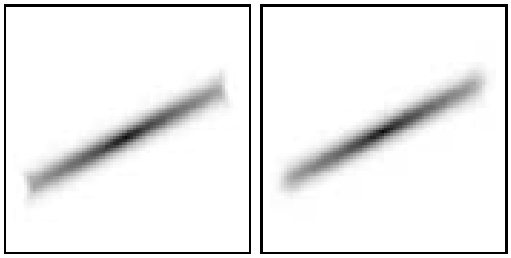

Figure 9: Comparison of time step sizes at hand of directional dilation displayed via inverted grey values. **(a) Left.** By $60 \times \tau \approx 40$, with $\tau = 0.6667$. **(b) Right.** By $400 \times \tau = 40$, with $\tau = 0.1$.
New test not performed in [9].

6 Conclusion and Outlook

We summarise the main extensions to [9] documented in this work, drawing corresponding conclusions:

- We have shown the consistency and stability of the FCT-scheme for dilation/erosion with general convex structuring elements.

  Conclusion. Especially, there is no particular restriction on the time step size depending on a structuring element in use. This shows that it is possible to apply the FCT-scheme in a general setting with varying structuring elements. This is important for many applications.

- We have performed detailed numerical tests, comparing FCT with other PDE-based schemes and a set-based method.

  Conclusion. The quality of FCT-results is better than the quality of results using other PDE-based schemes with respect to edge resolution in all experiments.

- We have discussed in detail important issues like the interpolation formula, anisotropic structuring elements and a practical choice of the time step size.

  Conclusions. The interpolation formula can be of importance for the quality of results. If the structuring element is very anisotropic, a general set-up with FCT as described here performs reasonable, but it can be improved by using schemes more specialised to an extreme set-up. The FCT-scheme behaves very robust with respect to the time step size.

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] 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)

[2] Arehart, A.B., Vincent, L., Kimia, B.B.: Mathematical morphology: The Hamilton–Jacobi connection. In: Proc. Fourth International Conference on Computer Vision, pp. 215–219. IEEE Computer Society Press, Berlin (1993)

[3] Banon, G.J.F., Barrera, J., Braga-Neto, U.d.M., Hirata, N.S.T. (eds.): Proceedings of the 8th International Symposium on Mathematical Morphology: Volume 1 - Full Papers, vol. 1. Instituto Nacional de Pesquisas Espaciais (INPE), São José dos Campos (2007). URL http://urlib.net/dpi.inpe.br/ismm@80/2007/05.29.15.58

[4] Blum, H.: Biological shape and visual science. Journal of Theoretical Biology **38**, 205–287 (1973)

[5] 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)

[6] Boris, J.P., Book, D.L.: Flux corrected transport. III. Minimal error FCT algorithms. Journal of Computational Physics **20**, 397–431 (1976)

[7] 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)

[8] Breuß, M., Welk, M.: Analysis of staircasing in semidiscrete stabilised inverse linear diffusion algorithms. Journal of Computational and Applied Mathematics **206**, 520–533 (2007)

[9] Breuß, M., Weickert, J.: Highly accurate PDE-based morphology for general structuring elements. In: X.-C. Tai et al. (Eds.): Scale-Space and Variational Methods in Computer Vision, *Lecture Notes in Computer Science*, vol. 5567, pp. 758–769. Springer, Berlin (1999)

[10] Brockett, R.W., Maragos, P.: Evolution equations for continuous-scale morphology. In: Proc. IEEE International Conference on Acoustics, Speech and Signal Processing, vol. 3, pp. 125–128. San Francisco, CA (1992)

[11] Burgeth, B., Breuß, M., Didas, S., Weickert, J.: PDE-based morphology for matrix fields: Numerical solution schemes. In: S. Aja-Fernandez, R. de Luis-Garcia, D. Tao, X. Li (Eds.): *Tensors in Image Processing and Computer Vision*, Advances in Pattern recognition, pp. 125–150. Springer, London, 2009

[12] Burgeth, B., Breuß, M., Pizarro, P., Weickert, J.: PDE-driven adaptive morphology for matrix fields. In: X.-C. Tai et al. (Eds.): Scale Space and Variational Methods in Computer Vision, *Lecture Notes in Computer Science*, Vol. 5567, pp. 247–258. Springer, Berlin, 2009

[13] Butt, M.A., Maragos, P.: Comparison of multiscale morphology approaches: PDE implemented via curve evolution versus Chamfer distance transform. In: P. Maragos, R.W. Schafer, M.A. Butt (eds.) Mathematical Morphology and its Applications to Image and Signal Processing, *Computational Imaging and Vision*, vol. 5, pp. 31–40. Kluwer, Dordrecht (1996)

[14] Farin, G.:: Curves and Surfaces for CADG. Morgan-Kaufmann, (2002)

[15] Gottlieb, S., Shu, C.-W.: Total variation diminishing Runge-Kutta schemes. Mathematics of Computation **67**, No. 221, 73–85 (1998)

[16] Gottlieb, S., Shu, C.-W., Tadmor, E.: Strong stability-preserving high-order time discretisation methods. SIAM Review **43**, No. 1, 89–112 (2001)

[17] 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)

[18] Hairer, E., Norsett, S., Wanner, G.: Solving Ordinary Differential Equations. I: Nonstiff Problems, *Springer Series in Computational Mathematics*, vol. 8. Springer, New York (1987)

[19] 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)

[20] Kimia, B.B., Tannenbaum, A., Zucker, S.W.: Shapes, shocks, and deformations i: The components of two-dimensional shape and the reaction-diffusion space. International Journal of Computer Vision **15**, 189–224 (1995)

[21] LeVeque, R.J.: Finite Volume Methods for Hyperbolic Problems. Cambridge University Press, Cambridge, UK (2002)

[22] Matheron, G.: Eléments pour une théorie des milieux poreux. Masson, Paris (1967)

[23] Matheron, G.: Random Sets and Integral Geometry. Wiley, New York (1975)

[24] Osher, S., Fedkiw, R.P.: Level Set Methods and Dynamic Implicit Surfaces, *Applied Mathematical Sciences*, vol. 153. Springer, New York (2002)

[25] 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)

[26] Pizarro, L., Burgeth, B., Breuß, M., Weickert, J.: A directional Rouy-Tourin scheme for adaptive matrix-valued morphology. To appear in Proc. Ninth International Symposium on Mathematical Morphology (ISMM 2009), Groningen, The Netherlands, August 2009

[27] Ronse, C., Najman, L., Decencière, E. (eds.): Mathematical Morphology: 40 Years On, *Computational Imaging and Vision*, vol. 30. Springer, Dordrecht (2005)

[28] Rouy, E., Tourin, A.: A viscosity solutions approach to shape-from-shading. SIAM Journal on Numerical Analysis **29**, 867–884 (1992)

[29] Sapiro, G., Kimmel, R., Shaked, D., Kimia, B.B., Bruckstein, A.M.: Implementing continuous-scale morphology via curve evolution. Pattern Recognition **26**, 1363–1372 (1993)

[30] Serra, J.: Echantillonnage et estimation des phénomènes de transition minier. Ph.D. thesis, University of Nancy, France (1967)

[31] Serra, J.: Image Analysis and Mathematical Morphology, vol. 1. Academic Press, London (1982)

[32] Serra, J.: Image Analysis and Mathematical Morphology, vol. 2. Academic Press, London (1988)

[33] Sethian, J.A.: Level Set Methods and Fast Marching Methods, second edn. Cambridge University Press, Cambridge, UK (1999). Paperback edition

[34] 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)

[35] Soille, P.: Morphological Image Analysis, second edn. Springer, Berlin (2003)

[36] Tadmor, E.: Numerical viscosity and the entropy condition for conservative difference schemes. Mathematics of Computation **43**, 369–381 (1984)

[37] Talbot, H., Beare, R. (eds.): Proc. Sixth International Symposium on Mathematical Morphology and its Applications. Sydney, Australia (2002). Http://www.cmis.csiro.au/ismm2002/proceedings/

[38] van den Boomgaard, R.: Mathematical morphology: Extensions towards computer vision. Ph.D. thesis, University of Amsterdam, The Netherlands (1992)

[39] van den Boomgaard, R.: Numerical solution schemes for continuous-scale morphology. In: M. Nielsen, P. Johansen, O.F. Olsen, J. Weickert (eds.) Scale-Space Theories in Computer Vision, *Lecture Notes in Computer Science*, vol. 1682, pp. 199–210. Springer, Berlin (1999)