In M. Goesele, S. Roth, A. Kuijper, B. Schiele, K. Schindler (Eds.): Pattern Recognition.
Lecture Notes in Computer Science, Vol. 6376, pp. 533–542, Springer, Berlin, 2010.
DAGM 2010 Best Paper Award.

From Box Filtering to Fast Explicit Diffusion

Sven Grewenig, Joachim Weickert, and Andrés Bruhn

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

Abstract. There are two popular ways to implement anisotropic diffusion filters with a diffusion tensor: Explicit finite difference schemes are simple but become inefficient due to severe time step size restrictions, while semi-implicit schemes are more efficient but require to solve large linear systems of equations. In our paper we present a novel class of algorithms that combine the advantages of both worlds: They are based on simple explicit schemes, while being more efficient than semi-implicit approaches. These so-called fast explicit diffusion (FED) schemes perform cycles of explicit schemes with varying time step sizes that may violate the stability restriction in up to 50 percent of all cases. FED schemes can be motivated from a decomposition of box filters in terms of explicit schemes for linear diffusion problems. Experiments demonstrate the advantages of the FED approach for time-dependent (parabolic) image enhancement problems as well as for steady state (elliptic) image compression tasks. In the latter case FED schemes are speeded up substantially by embedding them in a cascadic coarse-to-fine approach.

1 Introduction

Anisotropic diffusion filters with a diffusion tensor instead of a scalar-valued diffusivity offer additional degrees of freedom that allow to steer them according to a task at hand [1]: Coherence-enhancing diffusion filters, for example, are well-suited for processing seismic data sets [2], while edge-enhancing diffusion filters have attractive qualities for lossy image compression [3]. However, since such anisotropic diffusion filters require a diffusion tensor, their efficient implementation is much more difficult than for their isotropic counterparts with a scalar-valued diffusivity such as the Perona-Malik filter [4]. For the latter ones one can use e.g. additive operator splitting (AOS) schemes [5, 6], while there is no efficient full operator splitting in the general anisotropic case.

Although there has been a number of proposals for numerical schemes for anisotropic diffusion processes (see e.g. [7–9]), probably the two most popular ways to implement anisotropic diffusion filters are explicit and semi-implicit finite difference schemes. Explicit schemes are very simple to implement and allow a direct computation of the values at a new time level without solving linear or nonlinear systems of equations. However, they suffer from severe time step size restrictions which render them inefficient. Semi-implicit schemes, on the other hand, permit to use large time step sizes and can be more efficient than explicit approaches. Unfortunately, they are more difficult to implement and require to solve a large linear system of equations in each time step.

Our Contribution. The goal of the present paper is to show that it is possible to combine the advantages of explicit and semi-implicit schemes while avoiding their shortcomings. To this end we introduce a novel class of numerical schemes that we call *Fast Explicit Diffusion (FED) Schemes*. They perform cycles of explicit diffusion schemes with varying time step sizes. Since within each cycle up to 50 percent of all steps may violate the stability condition, one can achieve very large diffusion times. In this way one cycle can become even more efficient than one semi-implicit step. Moreover, we show that one can embed FED cycles within a coarse-to-fine strategy to solve stationary problems in an even more efficient way than with multigrid approaches. These findings are illustrated by applying the FED idea to edge- and coherence-enhancing diffusion filters. The starting point that has led us to the development of FED schemes was the observation that one can factorise a (stable) 1-D box filter into a cycle of explicit linear diffusion schemes with stable and unstable time step sizes. This idea can be generalised in a straightforward way to nonlinear and anisotropic problems in arbitrary dimensions.

Organisation of the Paper. Our paper is organised as follows: In Section 2 we derive the FED idea from the factorisation of a 1-D box filter into explicit linear diffusion steps, and we relate this approach to the so-called Super Time Stepping (STS) method of Gentzsch et al. [10, 11]. In Section 3 we show how FED can be generalised to arbitrary diffusion processes, and we show in Section 4 how this can be adapted to edge- and coherence-enhancing diffusion filters. After this, we perform numerical experiments in Section 5, and we conclude the paper in Section 6.

2 Filter Factorisation

2.1 Equivalence between 1-D Discrete Box Filtering and Linear FED

In order to motivate our FED approach, we restrict ourselves to the 1-D case first and consider linear diffusion processes. Since it is well-known that linear diffusion filtering is equivalent to Gaussian convolution and Gaussians can be approximated by iterated box filtering, we explore the connection between a box filter and explicit schemes for linear diffusion.

Let $\boldsymbol{f} = (f_i)_{i\in\mathbb{N}}$ be a discrete 1-D signal given on a grid with mesh size $h > 0$. We define the discrete box filter of length $(2n+1)h$, $n \in \mathbb{N}$, as well as the discrete second order derivative by

$$\left(B^h_{2n+1}(\boldsymbol{f})\right)_i := \frac{1}{2n+1}\sum_{k=-n}^{n} f_{i+k} \quad \text{and} \quad (\Delta_h \boldsymbol{f})_i := \frac{f_{i+1} - 2f_i + f_{i-1}}{h^2} \,. \tag{1}$$

The explicit discretisation of the linear heat equation for a function $u(x, t)$,

$$\partial_t u = \partial_{xx} u \; , \tag{2}$$

evaluated at a spatial-time-grid point (x_i, t_k) with $x_i := \left(i - \frac{1}{2}\right) h$ and $t_k := k\tau$, can then be formulated as

$$u_i^{k+1} = (I + \tau \Delta_h)\, u_i^k \; , \tag{3}$$

where I is the identity operator, $\tau > 0$ the time step size and $u_i^k \approx u(x_i, t_k)$ a numerical approximation.

The following theorem states a connection between 1-D discrete box filtering and explicit schemes with different time step sizes:

Theorem 1. *A discrete one-dimensional box filter* B^h_{2n+1} *is equivalent to a cycle with* n *explicit linear diffusion steps:*

$$B^h_{2n+1} = \prod_{i=0}^{n-1} (I + \tau_i \Delta_h) \;\; , \tag{4}$$

with the varying time step sizes

$$\tau_i = \frac{h^2}{4 \cos^2\left(\pi \frac{2i+1}{4n+2}\right)} \tag{5}$$

and corresponding stopping time

$$t_n \; := \; \sum_{i=0}^{n-1} \tau_i \; = \; \frac{h^2}{3} \binom{n+1}{2}. \tag{6}$$

The corresponding proof can be found in the Appendix.

We call one cycle of this novel scheme a *Fast Explicit Diffusion (FED)* cycle. Because of its equivalence to box filtering, FED is also stable. Interestingly, the time step sizes τ_i in Eq. (5) partially violate stability conditions. Table 1 shows both the smallest three and largest three time step sizes for different n. Since the stability restriction for the time step size of an explicit scheme in one dimension is given by $\tau \le \frac{h^2}{2}$, it is easy to show that the FED scheme consists of $\left\lceil \frac{n-1}{2} \right\rceil$ unstable time steps, where $\lceil a \rceil$ denotes the next largest integer $k \ge a$. Hence, for even n, half of the time steps are unstable. For $n \ge 3$, one FED cycle reaches the stopping time t_n faster than any other explicit scheme with stable time step sizes $\tau \le \frac{h^2}{2}$.

Since we want to approximate a diffusion process – or equivalently Gaussian convolution – one should use several iterated box filters – or equivalently FED cycles. Let $M \ge 2$ denote this number of FED cycles. This number M of outer cycles should not be confused with the number n of inner steps.

Before we explore extensions of FED to nonlinear, anisotropic and multidimensional problems, let us discuss some related work first.

Table 1: First three and last three step sizes of FED (1-D) with $h = 1$ (rounded). t_n denotes the stopping time of one FED cycle including n inner time steps

n	10	25	50	100	250	500	1000
τ_0	0.251404	0.250237	0.250060	0.250015	0.250002	0.250001	0.250000
τ_1	0.263024	0.252147	0.250545	0.250137	0.250022	0.250006	0.250001
τ_2	0.288508	0.256024	0.251518	0.250382	0.250061	0.250015	0.250004
$\vdots$							
τ_{n-3}	1.33	7.40	28.79	113.79	706.52	2820.19	11269.25
τ_{n-2}	2.88	16.55	64.68	255.93	1589.57	6345.33	25355.72
τ_{n-1}	11.25	65.97	258.48	1023.45	6358.01	25381.06	101422.61
t_n	18.33	108.33	425.00	1683.33	10458.33	41750.00	166833.33

2.2 Connection to Super Time Stepping

Our FED scheme uses different time step sizes, where some of them may violate stability limits. A similar method has been introduced under the name *Super Time Stepping (STS)* by Gentzsch et al. [10, 11]. Contrary to our derivation, they used a direct approach: Gentzsch et al. wanted to find a set of different time step sizes, which keeps stability after each cycle, and at the same time maximises the stopping time of such a cycle. Instead of factorising a box filter, one can show that their method intends to factorise the mask $\left(\frac{1}{2}, 0, \ldots, 0, \frac{1}{2}\right)$. Since this mask is very sensitive w.r.t. high frequencies, they have to introduce an additional damping parameter $\nu \geq 0$ that ensures better attenuation properties of high frequencies. This parameter can be seen as a trade-off between efficiency and damping quality, since larger values for ν scale down the stopping time. In our FED framework, such a damping parameter is not necessary.

While the ordering of the explicit diffusion steps does not matter in exact arithmetic, it can influence the result in practice due to numerical rounding errors when n is large. In order to improve robustness, Gentzsch et al. have proposed to rearrange the explicit steps within so-called $\varkappa$-cycles. We will also use this approach. For further details on STS, we refer to the above cited works and e.g. Alexiades et al., who have done an experimental evaluation [12].

3 Fast Explicit Diffusion (FED) for Arbitrary Problems

3.1 Extension to Arbitrary Diffusion Problems

While the FED scheme has been motivated in the 1-D setting with linear diffusion filtering, it is actually a general paradigm that can be applied to multidimensional, nonlinear and anisotropic diffusion processes. This can be seen as follows.

First, let us reconsider the 1-D diffusion equation (2) and its explicit discretisation (3). By assuming homogeneous Neumann boundary conditions and denoting $\boldsymbol{u}^k \in \mathbb{R}^N$ as the vector with entries u_i^k, Eq. (3) can be written as a matrix-vector product:

$$\boldsymbol{u}^{k+1} = (I + \tau A_h)\, \boldsymbol{u}^k \ , \tag{7}$$

with $\tau \leq h^2/2$. According to Gerschgorin's theorem, the eigenvalues of the matrix $A_h \in \mathbb{R}^{N \times N}$ lie in the interval $\left[-4/h^2, 0\right]$. These eigenvalues determine the stability in the Euclidean norm: A stable explicit step requires a time step size τ such that all eigenvalues of the matrix $I + \tau A_h$ lie in the interval $[-1, 1]$.

Keeping this in mind, it is straightforward to replace the matrix A_h by any negative semidefinite matrix P that results from a discretisation of a diffusion process. This process can be one- or multidimensional, linear or nonlinear, isotropic or anisotropic. In this case, one FED cycle is not any more equivalent to box filtering, but it corresponds to a first order approximation of the above-mentioned diffusion process. All one has to do is to adapt the time step size limit to the largest modulus of the eigenvalues of P. More precisely, let $\mu_i \leq 0$ be the eigenvalues of P and define $\mu_{\max} := \max_i |\mu_i|$. Then the explicit scheme in Eq. (7) with P instead of A_h is stable for time step sizes $\tilde{\tau} := c \cdot \tau$, where

$$c := \frac{4}{h^2 \cdot \mu_{\max}} \tag{8}$$

is the adjustment factor. Since $\mu_{\max}$ can easily be estimated using e.g. Gerschgorin's theorem, this adaptation is no problem at all in practice. Fig. 1 gives a summary of the general FED algorithm. Note that it is essentially an explicit scheme with some overhead that is not time critical.

3.2 Cascadic FED (CFED) for Stationary Problems

So far our FED scheme was designed for diffusion problems where we are interested in the temporal evolution. This refers to parabolic partial differential equations (PDEs) that are used for denoising and enhancement purposes.

However, in the case of inpainting and PDE-based compression problems, one is interested in the nontrivial steady state when Dirichlet boundary data are specified. The corresponding elliptic PDE results from the parabolic evolution for $t \to \infty$. To reach this steady state as quickly as possible, we embed our FED into a coarse-to-fine strategy [13], i.e. we use results computed on a coarse scale as an initialisation for a finer scale. Therefore, we scale down both the image and the reconstruction mask via area-based interpolation to a certain coarse level and apply the FED scheme on this image. Afterwards, we interpolate the corresponding solution and the mask to the next finer level and apply again FED on it. We apply this procedure recursively until the finest level is reached. To simplify matters, we always use the same parameter settings for the diffusion process on each level. We call this cascadic fast explicit diffusion approach *CFED*. It saves a lot of computational effort, since then a small or midsize stopping time is already sufficient on each level.

4 FED and CFED for Anisotropic Diffusion Filtering

In this section we review two specific two-dimensional anisotropic diffusion filters that we are going to use in our experiments as demonstrators for the potential of the FED and CFED algorithms.

1. **Input Data**:
   image $\boldsymbol{f}$, stopping time T, number M of outer FED cycles, and model parameters
2. **Initialisation**:
   (a) Compute the smallest n such that the stopping time t_n of one FED cycle fulfils $t_n \geq T/M$, and define $q := T/(M \cdot t_n) \leq 1$.
   (b) Compute the time step sizes $\tilde{\tau}_i := q \cdot c \cdot \tau_i$ with c according to (8), and τ_i according to (5).
   (c) Choose a suitable ordering for the step sizes $\tilde{\tau}_i$ according to [10].
   (d) If the diffusivity or diffusion tensor is constant in time, compute the corresponding matrix P.
3. **Filtering Loop**:
   (a) If the diffusivity or diffusion tensor is time-variant, update it and compute the corresponding matrix P.
   (b) Perform one FED cycle with the above ordering of the n explicit time steps $\tilde{\tau}_i$.
   (c) Go back to (a), if the stopping time T is not yet reached.

Fig. 1: General FED algorithm for diffusion filtering

4.1 Edge-Enhancing Diffusion (EED)

Edge-enhancing anisotropic diffusion inhibits diffusion across edges and instead prefers smoothing within the image regions [1]. It follows the evolution equation

$$\partial_t u \;=\; div\,(D\,(\nabla u_\sigma)\,\nabla u)\;, \tag{9}$$

where $D \in \mathbb{R}^{2\times 2}$ is the symmetric positive definite diffusion tensor, and u_σ is the image u convolved with a Gaussian of standard deviation σ. Its diffusion tensor is

$$D\,(\nabla u_\sigma) \;=\; g\left(|\nabla u_\sigma|^2\right) \cdot \frac{\nabla u_\sigma \nabla u_\sigma^\top}{|\nabla u_\sigma|^2} \;+\; 1 \cdot \frac{\nabla u_\sigma^\perp \nabla u_\sigma^{\perp\,\top}}{|\nabla u_\sigma^\perp|^2}\;, \tag{10}$$

where $\cdot^\top$ means the usual matrix transposition and $\binom{a}{b}^\perp := \binom{-b}{a}$. In our experiments we shall use the so-called Charbonnier diffusivity function

$$g\left(s^2\right) \;=\; \left(1 + s^2/\lambda^2\right)^{-1/2}\;. \tag{11}$$

It has proven to be highly useful for image interpolation purposes such as the compression method in [3]. In this case one computes the elliptic steady state solution.

We assume a uniform two-dimensional grid with the mesh sizes $h_x = h_y = 1$ and set the adjustment factor $c = 1/(2h^2)$, which is sufficient for stability with respect to the standard discretisation [14].

4.2 Coherence-Enhancing Diffusion (CED)

Coherence-enhancing diffusion filtering enhances line- and flow-like structures. Its diffusion tensor has the same eigenvectors as the so-called structure tensor

$$J_\rho\,(\nabla u_\sigma) \;:=\; K_\rho * \left(\nabla u_\sigma \nabla u_\sigma^\top\right)\;, \tag{12}$$

Fig. 2: Test image and reference image computed by a semi-implicit scheme. **Left**: Original image (finger, 300×300, rescaled to [0,255] for better visualisation). **Right**: CED-filtered reference image ($T = 300$, $\lambda = 1$, $\sigma = 0.5$, $\rho = 4$, $\alpha = 0.001$, $\tau = 0.1$), rescaled to [0,255]

where K_ρ is a Gaussian of standard deviation ρ, and its eigenvalues are given by

$$\lambda_1 := \alpha \tag{13}$$

$$\lambda_2 := \begin{cases} \alpha, & \text{if } \mu_1 = \mu_2 \ , \\ \alpha + (1-\alpha)\exp\left(\frac{-\lambda}{(\mu_1-\mu_2)^2}\right), & \text{else} \end{cases} , \tag{14}$$

where μ_1 and μ_2 are the eigenvalues of the structure tensor such that $\mu_1 \geq \mu_2$. For further details we refer to [1]. As a space discretisation for CED, we have used the one in [9]. It has low dissipativity and allows to use the same c as for the preceding EED scheme.

5 Experiments

In order to evaluate FED for parabolic problems, we enhance a fingerprint test image with CED. First we compute a reference solution by applying a semi-implicit scheme with very small time step sizes. The original image and the filtered result can be seen in Fig. 2.

Our error measure is the relative mean absolute error (RMAE), $\sum_i \frac{|u_i - r_i|}{\|r\|_1}$ with $\|r\|_1 := \sum_i |r_i|$. The filtered image is denoted by u, and r is the corresponding reference solution.

Table 2 shows that FED and the semi-implicit method yield comparable results with respect to the RMAE. In some cases, FED is even better than the semi-implicit scheme.

In order to show the efficiency of the novel FED compared to semi-implicit methods, we have conducted an experiment analysing the trade-off between the running time (CPU: Pentium 4, 3.2 GHz) and the RMAE. The result is depicted in Fig. 3. As one can see, the FED scheme shows a better trade-off, i.e. is more efficient than the usual semi-implicit scheme with a conjugate gradient (CG) solver.

Let us now consider an elliptic problem, where we evaluate the performance of our FED and CFED scheme. As a testbed we use an interpolation problem that is relevant for image compression with EED [3]. For the coarse-to-fine setting, we use three levels: $257 \times 257,\ 129 \times 129$ and 65×65 pixels.

Table 2: Comparison between FED and the semi-implicit method for different numbers of FED cycles/semi-implicit steps using the RMAE

cycles/steps	FED	semi-impl.
1	0.028106	0.020914
5	0.010587	0.009639
10	0.007206	0.006638
25	0.003922	0.003731
50	0.002074	0.002234
100	0.001063	0.001265

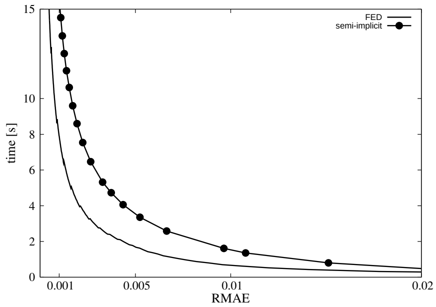

Fig. 3: CPU time (seconds) vs. RMAE

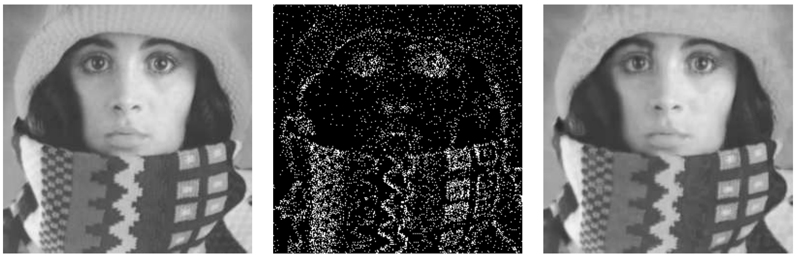

Fig. 4: Test setting for EED-based image reconstruction. **Left**: Original image (trui, 257×257). **Middle**: Inpainting mask where the pixels are specified. **Right**: Reconstruction with EED-based inpainting in the unspecified regions (semi-implicit, $T = 250000$, $\tau = 2.5$, $\lambda = 0.1$, $\sigma = 1.5$)

Fig. 4 depicts the test setting. We use the same error measure as above and compare our results to the reference reconstruction shown in Fig. 4. The comparison concerning the trade-off between the CPU time and the RMAE, which is illustrated in Fig. 5 for the stopping time $T = 5000$, emphasises the superior efficiency of FED and CFED respectively. In both cases, the corresponding semi-implicit schemes are less efficient. Moreover, CFED further improves the efficiency of FED. If one wants to have for example a solution whose RMAE is below 1%, CFED can manage this in less than a quarter of a second, because already a small stopping of $T = 100$ is sufficient.

6 Conclusions and Future Work

We have presented a new framework for explicit diffusion schemes, FED, which has been derived by the theory of one-dimensional box filters. This means we have established an interesting connection between a symmetric linear filter and an explicit scheme with varying time step sizes that partially violates stability limits. FED is very easy to implement, since existing explicit schemes with only few additional code lines can be used. Furthermore, we have successfully applied FED to anisotropic diffusion processes and PDE-based image reconstruction, where we have additionally used a coarse-to-fine strategy. Due to the large time step sizes, explicit schemes can become

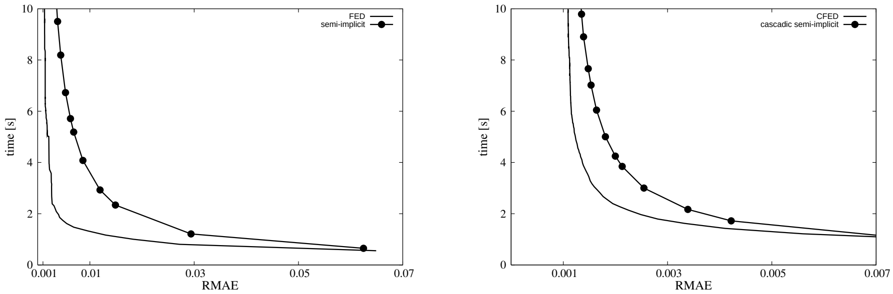

Fig. 5: CPU time (seconds) vs. RMAE for $T = 5000$. **Left**: FED and semi-implicit. **Right**: CFED and cascadic semi-implicit

more efficient than semi-implicit ones, as we have shown in the experimental section. The cascadic strategy CFED can even improve the results of FED with respect to inpainting applications.

In our ongoing work, we are currently working on parallelisation techniques as well as GPU-based implementations. With the help of them, it might be possible to yield even faster anisotropic diffusion filtering and real-time decoding with anisotropic diffusion via explicit schemes. Another research field are higher-dimensional problems, since semi-implicit schemes become cumbersome for such tasks due to the large neighbourhood structure. In this case, the benefit of FED is expected to increase even further.

Acknowledgements. Sven Grewenig gratefully acknowledges funding by the Deutsche Forschungsgemeinschaft (DFG), project WE 2602/7-1.

References

1. Weickert, J.: Anisotropic Diffusion in Image Processing. Teubner, Stuttgart (1998)
2. Höcker, C., Fehmers, G.: Fast structural interpretation with structure-oriented filtering. Geophysics **68**(4) (2003) 1286–1293
3. Galić, I., Weickert, J., Welk, M., Bruhn, A., Belyaev, A., Seidel, H.P.: Image compression with anisotropic diffusion. Journal of Mathematical Imaging and Vision **31**(2–3) (July 2008) 255–269
4. Perona, P., Malik, J.: Scale space and edge detection using anisotropic diffusion. IEEE Transactions on Pattern Analysis and Machine Intelligence **12** (1990) 629–639
5. Lu, T., Neittaanmäki, P., Tai, X.C.: A parallel splitting up method and its application to Navier-Stokes equations. Applied Mathematics Letters **4**(2) (1991) 25–29
6. Weickert, J., ter Haar Romeny, B.M., Viergever, M.A.: Efficient and reliable schemes for nonlinear diffusion filtering. IEEE Transactions on Image Processing **7**(3) (March 1998) 398–410
7. Drblíková, O., Mikula, K.: Convergence analysis of finite volume scheme for nonlinear tensor anisotropic diffusion in image processing. SIAM Journal on Numerical Analysis **46**(1) (2007) 37–60
8. Weickert, J., Scharr, H.: A scheme for coherence-enhancing diffusion filtering with optimized rotation invariance. Journal of Visual Communication and Image Representation **13**(1/2) (2002) 103–118

9. Welk, M., Steidl, G., Weickert, J.: Locally analytic schemes: A link between diffusion filtering and wavelet shrinkage. Applied and Computational Harmonic Analysis **24** (2008) 195–224
10. Gentzsch, W., Schlüter, A.: Über ein Einschrittverfahren mit zyklischer Schrittweitenänderung zur Lösung parabolischer Differentialgleichungen. ZAMM, Zeitschrift für Angewandte Mathematik und Mechanik **58** (1978) T415–T416
11. Gentzsch, W.: Numerical solution of linear and non-linear parabolic differential equations by a time discretisation of third order accuracy. In Hirschel, E.H., ed.: Proceedings of the Third GAMM-Conference on Numerical Methods in Fluid Mechanics, Friedr. Vieweg & Sohn (1979) 109–117
12. Alexiades, V., Amiez, G., Gremaud, P.A.: Super-time-stepping acceleration of explicit schemes for parabolic problems. Communications in Numerical Methods in Engineering **12** (1996) 31–42
13. Bornemann, F., Deuflhard, P.: The cascadic multigrid method for elliptic problems. Numerische Mathematik **75** (1996) 135–152
14. Weickert, J.: Nonlinear diffusion filtering. In Jähne, B., Haußecker, H., Geißler, P., eds.: Handbook on Computer Vision and Applications, Vol. 2: Signal Processing and Pattern Recognition. Academic Press, San Diego (1999) 423–450

Appendix: Proof of Theorem 1

After some calculations, one can represent the box filter as a finite operator series:

$$B^h_{2n+1} = \sum_{m=0}^{n} \frac{h^{2m}}{2m+1} \binom{n+m}{2m} \Delta_h^m \ , \tag{15}$$

where $\Delta_h^0 := I$ (identity operator). Replacing Δ_h by $(-z)$ in Eq. (15) defines the polynomial $p_n(z)$. It follows that p_n can be related to the Chebyshev polynomial of first kind,

$$C_{2n+1}(x) = \frac{2n+1}{2} \sum_{m=0}^{n} \frac{(-1)^m}{2n+1-m} \binom{2n+1-m}{m} (2x)^{2(n-m)+1} \ , \tag{16}$$

and it holds for $z > 0$:

$$p_n(z) = (-1)^n \cdot \frac{2C_{2n+1}\left(\frac{h\sqrt{z}}{2}\right)}{(2n+1)h\sqrt{z}} \ . \tag{17}$$

Hence, the roots z_i of p_n are related to the first n (positive) well-known roots of C_{2n+1}, namely $x_0, \ldots, x_{n-1}$:

$$z_i \ = \ \frac{4}{h^2} x_i^2 \ = \ \frac{4}{h^2} \cdot \cos^2\left(\pi \frac{2i+1}{4n+2}\right) > 0 \ . \tag{18}$$

Thus, we can represent p_n as a product of n linear factors $(1 - z/z_i)$, and by the back substitution $(-z) \rightarrow \Delta_h$ we finally get

$$B^h_{2n+1} = \prod_{i=0}^{n-1} \left(I + z_i^{-1} \Delta_h \right) \ . \tag{19}$$

This shows that B_{2n+1}^h is equivalent to an explicit linear diffusion scheme using the n time step sizes $\tau_i = z_i^{-1}$, and the stopping time t_n is equal to the coefficient of Δ_h ($m = 1$) in Eq. (15).