A NEW NUMERICAL METHOD FOR LEVEL SET MOTION IN NORMAL DIRECTION USED IN OPTICAL FLOW ESTIMATION

PETER FROLKOVIC AND VIERA KLEINOVA

Faculty of Civil Engineering, Slovak University of Technology
Department of Mathematics and Descriptive Geometry
Radlinského 11, 810 05 Bratislava, Slovak Republic

Abstract. We present a new numerical method for the solution of level set advection equation describing a motion in normal direction for which the speed is given by the sign function of the difference of two given functions. Taking one function as the initial condition, the solution evolves towards the second given function. One of possible applications is an optical flow estimation to find a deformation between two images in a video sequence. The new numerical method is based on a bilinear interpolation of discrete values as used for the representation of images. Under natural assumptions, it ensures a monotone decrease of the absolute difference between the numerical solution and the target function, and it handles properly the discontinuity in the speed due to the dependence on the sign function. To find the deformation between two functions (or images), the backward tracking of characteristics is used. Two numerical experiments are presented, one with an exact solution to show an experimental order of convergence and one based on two images of lungs to illustrate a possible application of the method for the optical flow estimation.

1. Introduction. One of the most common mathematical tools in level set methods is a solution of nonlinear advection equation that describes a motion of evolving function in a normal direction to its level sets. Very often, the level set method is used to track a position of a dynamic interface (e.g., a curve in 2D case) that is defined only implicitly as a zero level set of the evolving function. For a review of diverse applications of level set methods we refer to [13, 11, 7].

The main motivation of our study is a possible application of level set methods for so-called optical flow estimation between two greyscale images in a video sequence, when one searches a deformation of one image, a source, into another one, a target. We refer to [8, 1, 14, 2, 5, 9, 3] for more detailed descriptions of this problem used in image processing, we restrict here on a particular task of solving the nonlinear advection equation of which the solution evolves from the given source function towards the given target function by the level set motion in normal direction [1, 14].

We derive a new numerical scheme that handles appropriately a discontinuity of the speed in the model and that uses a form of numerical solution typical for the representation of images. In particular, the method shall approximate the solution using the bilinear interpolation, and it shall give under natural assumptions a monotone decrease of the absolute difference between the numerical solution and

2010 Mathematics Subject Classification. Primary: 65D18, 65M25; Secondary: 35L60.

Key words and phrases. Optical flow estimation, level set equation, motion in normal direction.

Work supported by grants VEGA 1/0728/15, APVV-15-0522 and APVV-16-0431. The authors are grateful for a support of company Tatramed in Bratislava, Slovakia.
the given target function, even when the speed of evolution allows a change of its sign.

For that purpose, we extend a so-called Rouy-Tourin scheme [13, 12] for the numerical solution of nonlinear advection equation using an idea of the so-called Corner Transport Upwind scheme that is described for linear advection equation in [4, 10, 6]. The Rouy-Tourin scheme is based on linear interpolation, while the proposed scheme uses the bilinear interpolation. By a careful definition of time steps, we ensure that the new scheme gives a monotone decrease in time of the absolute difference between the evolving function and the target function.

The proposed scheme is given not only in an Eulerian form to define the values of numerical solution in fixed points of rectangular grids, but also in a semi-Lagrangian form to obtain the numerical solution by backward tracking of characteristic curves. If the source and target functions differ significantly, one has to apply the scheme several times up to some stopping criteria are fulfilled. For such cases, we define a modification of the backward tracking method that does not require to store intermediate results of all time steps.

The paper is organized as follows. In Section 2, we describe the model of the motion in the normal direction for the level sets of evolving function towards the level sets of the target function. In Section 3, we propose the new numerical method to solve the model approximately, including the backward tracking of characteristic curves. In Section 4, we present two numerical experiments, one with a known exact solution and one with an illustrative application of the optical flow estimation using real biological images of lungs.

2. Level set motion from a source function towards a target function.

Let \( F = F(x) \) and \( G = G(x) \) be given functions defined on a domain \( \Omega \subset \mathbb{R}^2 \). The functions \( F \) and \( G \) are called the source function and the target function, respectively. Our aim is to evolve the source \( F \) by a motion of level sets in normal direction towards the target \( G \).

For that purpose we search for a function \( f = f(x,t) \) fulfilling the level set equation

\[
\partial_t f = \text{sgn}(G - f) |\nabla f|, \quad f(x, 0) = F(x), \quad x \in \Omega, \ t \geq 0,
\]

where \( \text{sgn} \) is the standard sign function. The equation (1) can be equivalently formulated in the form of nonlinear advection equation

\[
\partial_t f + \vec{u} \cdot \nabla f = 0, \quad f(x, 0) = F(x), \quad x \in \Omega, \ t \geq 0,
\]

with

\[
\vec{u} = \begin{cases} 
-\text{sgn}(G - f) \frac{\nabla f}{|\nabla f|}, & |\nabla f| \neq 0 \\
\vec{0}, & |\nabla f| = 0.
\end{cases}
\]

The vector field \( \vec{u} \), if nonzero, represents the normal vectors to the level sets of \( f \) with the orientation given by the sign of \( G - f \).

The equation (1), respectively (2), prescribes the movement of level sets of the function \( f \) in normal direction [13, 11]. The speed of such evolution can change abruptly from the values 1 or \(-1\) to 0 if \( f = G \) or \( \nabla f = \vec{0} \) that must be treated carefully in numerical methods.

From (1) one obtains that the difference \(|G(x) - f(x,t)|\) for a fixed \( x \) is monotonically decreasing in time if \( f(x,t) \neq G(x) \) and \( \nabla f(x,t) \neq \vec{0} \) as
Consequently, the function 

\[ f(x, t) = G(x) \]

satisfies the inequality

\[ |\nabla f(x, t)| \leq 0. \]

Furthermore, if for some \( T \geq 0 \) one has for all \( x \in \Omega \) that \( f(x, T) = G(x) \) or \( |\nabla f(x, T)| = 0 \) then \( f(x, t) \equiv f(x, T) \) for \( t \geq T \). In this sense, we say that the function \( f \) is evolving from the source function \( F \) towards the target function \( G \). In practical applications of image processing one may try to avoid \(|\nabla f| = 0\) in (1) by using some pre-processing techniques related to this term in (1), see, e.g., [14]. As we are here mainly interested in solving (1) numerically, we do not consider such techniques here.

Concerning boundary conditions for the equations (1) or (2), we set the outward normal component of \( \nabla f \) to zero at the boundary \( \partial \Omega \) of \( \Omega \).

In the section on numerical methods, we derive a new numerical scheme to solve (1), respectively (2), for which the obtained numerical solution will fulfill the property (4) in a discrete form. This scheme gives the approximate solution in fixed positions of grid points that corresponds to the Eulerian type of method [10]. Afterward, we consider the problem from a view of the Lagrangian type of methods.

Once the solution \( f \) of (2) is known, one can compute the characteristic curves \( X = X(t) \) generated by \( \bar{u} \) in (3) by solving backward in time the ordinary differential equations

\[ \dot{X} = \bar{u}(X, t), \quad X(\tilde{t}) = x, \quad (5) \]

for some \( x \in \Omega \) and \( \tilde{t} > 0 \), so we may write \( X = X(t) = X(x; \tilde{t}; t) \) and \( t \in [0, \tilde{t}] \).

The value \( X(x, t; t) \) denotes the position \( X \) at time \( t \) of the characteristic curve for which the position at time \( \tilde{t} \) is \( x \).

For the solution \( f(x, t) \) of (2) the time derivative of \( f(X(t), t) \) vanishes,

\[
\frac{d}{dt} f(X(t), t) = \partial_t f(X(t), t) + \dot{X}(t) \cdot \nabla f(X(t), t) \\
= \partial_t f(X(t), t) + \bar{u}(X(t), t) \cdot \nabla f(X(t), t) = 0. \quad (6)
\]

Consequently, the function \( f \) is constant along the characteristic curves, \( f(X(t), t) \equiv f(x, \tilde{t}), \quad t \in [0, \tilde{t}] \). Let us now suppose that the solution \( f \) of (1) is available for some interval \( [0, T] \) with large enough time \( T > 0 \). If one tracks backward the characteristic curves in (5) for \( x \in \Omega \) from \( t = \tilde{t} = T \) to \( t = 0 \) we obtain

\[ f(x, T) = f(X(x, T; T), T) = f(X(x, T; t), t) = f(X(x, T; 0), 0) = F(X(x, T; 0)). \]

Consequently, we can define the deformation

\[ \tilde{D} = \tilde{D}(x) = x - X(x, T; 0) \quad (7) \]

for which we can distinguish two situations.

Firstly, if \( f(x, T) = G(x) \) then

\[ G(x) = F(x - \tilde{D}(x)). \quad (8) \]

If this property is valid for all \( x \in \Omega \), the deformation \( \tilde{D} \) can be viewed as the optical flow estimation [8].

Secondly, if \( f(x, T) \neq G(x) \) then \( G(x) \neq F(x - \tilde{D}(x)) \). This can happen in practice either if \( T \) is not large enough (when we simply enlarge the value of \( T \)) or \( |\nabla f(x, t)| \equiv 0 \) for \( t \geq T \). The latter case means that the function \( f \) at \( x \) has a local extrema, and we can not further decrease the difference \( |G(x) - f(x, t)| \) in the
direction of $\nabla f(x,t)$ for $t \geq T$ as in (4). This can happen, e.g., when the range of $G$ is not a subset of the range of $F$. Nevertheless, analogously to (4), one has

$$\frac{1}{2} \frac{d}{dt} (G(x) - F(X(x,t;0)))^2 < 0, \quad t \in (0,T),$$

and so $|G(x) - F(x)| > |G(x) - F(X(x,T;0))|$ if $T > 0$. We are still interested to find the deformation $\tilde{D}$ in (7), although (8) is not fulfilled in this case for all $x \in \Omega$.

In the next section we propose an approximation of the deformation $\tilde{D}$ in (7) using a numerical method to approximate the characteristics $X(x,t;0)$.

3. Numerical methods. We suppose that the functions $F$ and $G$ in (1) are represented by their discrete values $F_{ij} = F(x_{ij})$ and $G_{ij} = G(x_{ij})$ in the points $x_{ij} := (ih,jh)$ such that $i = 0, \ldots, I$, $j = 0, \ldots, J$, and $h > 0$ is a given constant. The values $F(x)$ for $x \in ((i-1)h,ih) \times ((j-1)h,jh)$, $i > 0, j > 0$ are obtained by the standard bilinear interpolation of related four discrete values $F_{ij}, F_{i-1,j}, F_{i,j-1}, F_{i-1,j-1}$.

First, we consider a numerical approximation of the level set equation (1). Let $\tau$ be a chosen fixed time step and $t^n = n\tau$. We suppose that for some $n \geq 0$ the following values are available

$$f^n_{ij} \approx f(x_{ij}, t^n)$$

with the case $n = 0$ defined by $f^n_{ij} = F_{ij}$. Furthermore, we initialize

$$s_{ij} = \text{sgn}(G_{ij} - F_{ij}),$$

and we suppose that $\text{sgn}(G_{ij} - f^n_{ij}) = \text{sgn}(G_{ij} - F_{ij})$. Note that if $f^n_{ij} = G_{ij}$ we use $f^{n+k}_{ij} = f^n_{ij}$ for $k = 1, 2, \ldots$ that can be obtained by setting $s_{ij} = 0$.

First, we quote Rouy-Tourin numerical scheme \[12, 13\] to approximate (1),

$$n\tau f^{n+1}_{ij} = f^n_{ij} + \tau s_{ij} |\nabla f^n_{ij}|,$$  \hspace{1cm} (11)

where the approximation of $\nabla f^n_{ij} = (\partial_{x_1} f^n_{ij}, \partial_{x_2} f^n_{ij}) \approx \nabla f(x_{ij}, t^n)$ is given by

$$h\partial_{x_1} f^n_{ij} = \begin{cases} 
    f^n_{i+1,j} - f^n_{i-1,j} & f^n_{i-1,j} = \text{ext}\{f^n_{i-1,j}, f^n_{i,j}, f^n_{i+1,j}\} \\
    f^n_{i+1,j} - f^n_{i,j} & f^n_{i,j} = \text{ext}\{f^n_{i-1,j}, f^n_{i,j}, f^n_{i+1,j}\} \\
    0 & f^n_{i,j} = \text{ext}\{f^n_{i-1,j}, f^n_{i,j}, f^n_{i+1,j}\}
\end{cases}$$  \hspace{1cm} (12)

$$h\partial_{x_2} f^n_{ij} = \begin{cases} 
    f^n_{ij} - f^n_{i,j-1} & f^n_{i,j-1} = \text{ext}\{f^n_{ij-1}, f^n_{ij}, f^n_{ij+1}\} \\
    f^n_{ij+1} - f^n_{ij} & f^n_{ij} = \text{ext}\{f^n_{ij-1}, f^n_{ij}, f^n_{ij+1}\} \\
    0 & f^n_{ij} = \text{ext}\{f^n_{ij-1}, f^n_{ij}, f^n_{ij+1}\}
\end{cases}$$  \hspace{1cm} (13)

The notation $\text{ext}$ denotes the extreme values among the three discrete values of $f$ in (12) or (13) with the particular choice depending on the sign of $s_{ij}$, namely

$$\text{ext} = \begin{cases} 
    \min s_{ij} < 0 & \text{if } s_{ij} < 0 \\
    \max s_{ij} > 0 & \text{if } s_{ij} > 0
\end{cases}.$$

The definitions (12) and (13) are modified for boundary nodes with prescribed zero outward normal component of $\nabla f$ simply by skipping the non-existing discrete values $f^n_{ij}$ with $i = -1, i = I + 1$, $j = -1$, or $j = J + 1$ in the argument of $\text{ext}$.

Next, we rewrite the scheme (11) in an equivalent form that helps us to use it for our purpose. To do so, let $k, l \in \{-1, 1, 0\}$ be such that the value $f^n_{i+k,j}$ is the extreme value chosen in (12), and, analogously, the value $f^n_{i,j+l}$ is chosen in
Consequently one has that $h|\partial x_i f^n_{ij}| = |f^n_{i+kj} - f^n_{ij}| = s_{ij}(f^n_{i+kj} - f^n_{ij})$ and $h|\partial x_j f^n_{ij}| = |f^n_{ij+l} - f^n_{ij}| = s_{ij}(f^n_{ij+l} - f^n_{ij})$.

If $|\nabla f^n_{ij}| = 0$ then, clearly, $f^n_{ij} = f^n_{ij}$. In what follows we assume that $|\nabla f^n_{ij}| \neq 0$, and we define “directional Courant numbers” [10, 6] for the fixed time step $\tau$,

$$C^n_{ij} = \frac{\tau |\partial x_i f^n_{ij}|}{h|\nabla f^n_{ij}|}, \quad D^n_{ij} = \frac{\tau |\partial x_j f^n_{ij}|}{h|\nabla f^n_{ij}|},$$

(14) and the scheme (11) can be written in the form

$$\rho_T f^{n+1}_{ij} = (1 - C^n_{ij} - D^n_{ij}) f^n_{ij} + C^n_{ij} f^n_{i+kj} + D^n_{ij} f^n_{ij+l}.$$  

(15)

Clearly, the scheme (15) defines the value $\rho_T f^{n+1}_{ij}$ as a convex combination of three values if

$$C^n_{ij} + D^n_{ij} \leq 1.$$  

(16)

To obtain (16) for any numerical solution $f^n_{ij}$ in (9), it is necessary that $\tau \leq h/\sqrt{2}$ in (14).

In what follows, we modify the scheme (15) using an idea of so called Corner Transport Upwind scheme [4, 10, 6]. Opposite to the scheme (15) that uses the linear interpolation of three values, the following scheme will use the bilinear interpolation involving also the corner value $f^n_{i+kj+l}$, namely

$$\rho_{CTU} f^{n+1}_{ij} = (1 - C^n_{ij} - D^n_{ij}) f^n_{ij} + C^n_{ij} f^n_{i+kj} + D^n_{ij} f^n_{ij+l} +$$

$$+ C^n_{ij} D^n_{ij} (f^n_{ij} - f^n_{i+kj} - f^n_{ij+l} + f^n_{i+kj+l}).$$

(17)

The right-hand side of (17) is a convex combination of four values if

$$\max\{C^n_{ij}, D^n_{ij}\} \leq 1.$$  

(18)

that is less restrictive than (16). The necessary condition to obtain (18) for any numerical solution $f^n_{ij}$ in (9) is that $\tau \leq h$ in (14).

Next, we formulate the scheme (17) using a quadratic function of $\tau$ on the right-hand side, namely

$$\rho_{CTU} f^{n+1}_{ij} = \tilde{f}^{n+1}_{ij}(\tau), \quad \tilde{f}^{n+1}_{ij}(\tau) := f^n_{ij} + \tau s_{ij}|\nabla f^n_{ij}| + \tau^2 \alpha^n_{ij}$$

(19)

and

$$\alpha^n_{ij} := \frac{(f^n_{ij} - f^n_{i+kj} - f^n_{ij+l} + f^n_{i+kj+l}) |\partial x_i f^n_{ij} \partial x_j f^n_{ij}|}{h^2|\nabla f^n_{ij}|^2}.$$  

(20)

Note that if $f^n_{ij} \neq G_{ij}$ then analogously to (4) one has

$$\frac{d}{d\tau} \left. \frac{1}{2} (G_{ij} - \tilde{f}^{n+1}_{ij}(\tau))^2 \right|_{\tau=0} = - |G_{ij} - f^n_{ij}||\nabla f^n_{ij}| < 0.$$  

To finalize the numerical schemes, we have to suggest proper values of the time step $\tau$ to be used in (19). Our aim is to use the maximal value $\tau = h$ to speed up the evolution of the values $f^{n+1}_{ij}(\tau)$ towards the values $G_{ij}$. Nevertheless there are two cases when the scheme (19) has to be modified for such time step. We define the modifications using a variable choice of $\tau$ in (19), say $\tau = \tau^n_{ij}$, but we discuss also equivalent modifications of (19) for which $\tau = h$.

We distinguish two cases. In the case 1, we check if $\text{sgn}(G_{ij} - \tilde{f}^{n+1}_{ij}(h)) \neq \text{sgn}(G_{ij} - f^n_{ij}) = s_{ij}$. If this happens there exists $\tau < h$, say $\tau^n_{ij}$, such that $f^{n+1}_{ij}(\tau^n_{ij}) = ...$
\[ G_{ij} \text{ that is our desired property. Consequently, we solve a quadratic equation for } \tau_{ij}^n \text{ (if } \alpha_{ij}^n \neq 0) \]

\[ f_{ij}^n - G_{ij} + \tau_{ij}^n s_{ij} |\nabla f_{ij}^n| + (\tau_{ij}^n)^2 \alpha_{ij}^n = 0. \] (21)

The scheme (19), respectively (17), shall be considered only for \( 0 < \tau \leq \tau_{ij}^n \), and one shall consider \( c_{TU} f_{ij}^{n+1} = G_{ij} \) for \( \tau_{ij}^n \leq \tau \leq h \). We use simply (19) with the variable time step \( \tau = \tau_{ij}^n \).

In the case 2 we want to avoid that for some \( \tau < h \), say again \( \tau_{ij}^n \), the derivative \( \frac{d}{d\tau} f_{ij}^{n+1}(\tau^n) \) vanishes, so the difference \( |G_{ij} - f_{ij}^{n+1}(\tau)| \) becomes increasing for \( \tau > \tau_{ij}^n \). Clearly, this property is not desired, therefore we search for \( \tau_{ij}^n \) solving a linear equation (if \( \alpha_{ij}^n \neq 0 \))

\[ s_{ij} |\nabla f_{ij}^n| + 2 \tau_{ij}^n \alpha_{ij}^n = 0, \] (22)

and we again use (19) with the variable time step \( \tau = \tau_{ij}^n \).

Another approach is to modify the scheme (19) using an idea of “limiters” \([10]\) by “limiting” the value \( \alpha_{ij}^n \) towards zero when the scheme (19) is approaching the form of scheme (11). The limited version of (19) with the time step \( \tau = h \) shall give then the same value as the unlimited form (19) with \( \tau = \tau_{ij}^n \). Again for a simplicity, we prefer the second form with the variable time step.

We are now ready to define the time steps \( \tau_{ij}^n \) in (19) according to (21) and (22) in details. It is enough to define it only if \( |\nabla f_{ij}^n| \neq 0 \) and \( f_{ij}^n \neq G_{ij} \).

Firstly, if \( \alpha_{ij}^n = 0 \) then \( f_{ij}^{n+1}(\tau) \) becomes a linear function. Consequently, to control the case 1 we require

\[ \tau_{ij}^n = \min \{ h, \frac{|G_{ij} - f_{ij}^n|}{|\nabla f_{ij}^n|} \}. \] (23)

Let us now consider that \( \alpha_{ij}^n \neq 0 \). We denote the discriminant of quadratic equation (21) by

\[ D = |\nabla f_{ij}^n|^2 + 4 s_{ij} |G_{ij} - f_{ij}^n| \alpha_{ij}^n. \] (24)

If \( D < 0 \) then there exists no \( \tau \) such that \( \hat{f}_{ij}^{n+1}(\tau) = G_{ij} \), therefore we need to control only the case 2 by requiring

\[ \tau_{ij}^n = \min \{ h, \frac{s_{ij} |\nabla f_{ij}^n|}{2 \alpha_{ij}^n} \}. \] (25)

Note that if \( D < 0 \) then \( \text{sgn}(\alpha_{ij}^n) = -s_{ij} \), so one has that \( \tau_{ij}^n > 0 \) in (25).

If \( D > 0 \) then there exists at least one value of \( \tau \) such that \( \hat{f}_{ij}^{n+1}(\tau) = G_{ij} \). Consequently, we control the case 1 by

\[ \tau_{ij}^n = \min \{ h, \frac{s_{ij} \sqrt{D} - |\nabla f_{ij}^n|}{2 \alpha_{ij}^n} \}. \] (26)

Note that if \( s_{ij} \alpha_{ij}^n < 0 \) then \( \sqrt{D} < |\nabla f_{ij}^n| \) and if \( s_{ij} \alpha_{ij}^n > 0 \) then \( \sqrt{D} > |\nabla f_{ij}^n| \), so one has that \( \tau_{ij}^n > 0 \) in (26).

We can now summarize that depending on the value of \( \alpha_{ij}^n \) in (20) and \( D \) in (24) one can define the value of \( \tau_{ij}^n \) by (23), (25), or (26) that can be used in (19) to define the values \( \hat{f}_{ij}^{n+1}(\tau_{ij}^n) \).

Defining the directional Courant numbers with the variable time steps \( \tau_{ij}^n \)

\[ c_{ij}^n = \frac{\tau_{ij}^n |\partial_{x1} f_{ij}^n|}{h |\nabla f_{ij}^n|}, \quad d_{ij}^n = \frac{\tau_{ij}^n |\partial_{x2} f_{ij}^n|}{h |\nabla f_{ij}^n|}, \] (27)
the scheme (17) turns to our final form in the Eulerian form
\[ e_j^{n+1} = (1 - c^n_{ij} - d^n_{ij}) f^n_{ij} + c^n_{ij} f^n_{i+kj} + d^n_{ij} f^n_{ij+l} + c^n_{ij} f^n_{ij} \]
\[ c^n_{ij} = (f^n_{ij} - f^n_{i+kj} - f^n_{ij+l} + f^n_{i+kj+l}) / 4. \]
The values \( e_j^{n+1} \approx f(x_i, y_j, t^n) \) in (28) approximate the evolution of \( f \) from \( F \) towards \( G \) in the Eulerian framework. Such framework is used for a specific application of optical flow in [1].

Next, we introduce it also in the semi-Lagrangian framework by defining \( X_{ij}^{n+1} \approx X(x_{ij}, t^{n+1}; 0) \). Note that a standard approach to do it is to compute a numerical solution of ordinary differential equations (5) for which the approximations of \( \bar{u}(x, t^m) \) for all previous time steps \( m = 0, 1, \ldots, n \) must be available. We propose an alternative approach.

To do so, we suppose that some approximation \( X_{ij}^0 \approx X(x_{ij}, t^n; 0) \) is available. Clearly, \( X_{ij}^0 = x_{ij} \). To define \( X_{ij}^{n+1} \) we use the bilinear interpolation analogously to (28)
\[ X_{ij}^{n+1} = (1 - c^n_{ij} - d^n_{ij}) X_{ij}^n + c^n_{ij} X_{i+kj}^n + d^n_{ij} X_{ij+l}^n + c^n_{ij} a^n_{ij} (X_{ij}^n - X_{i+kj}^n - X_{ij+l}^n + X_{i+kj+l}^n). \]

Finally, to be consistent with our aim to determine an approximation of (8) we define
\[ \bar{f}_{ij}^{n+1} = F(X_{ij}^{n+1}). \]

Note that \( \bar{f}_{ij}^{n+1} = f_{ij}^n \) for \( n = 1 \), but this property is lost in general for \( n > 1 \).

The method (30) can be finished at some \( n \) when for all \( i = 0, 1, \ldots, I \) and \( j = 0, 1, \ldots, J \), one has either \( |G_{ij} - f^n_{ij}| < \epsilon_1 \) or \( |\nabla f^n_{ij}| < \epsilon_2 \) or \( n = N \). In such case, the approximation of deformation \( \bar{D} \) in (7) in the grid points can be defined by
\[ \bar{D}_{ij} = x_{ij} - X_{ij}^n. \]

We now summarize the method written in a pseudo code description:

- take input \( F_{ij}, G_{ij}, \epsilon_1, \epsilon_2, N \);
- initialize \( s_{ij} = \text{sgn}(G_{ij} - F_{ij}), X_{ij}^0 = x_{ij}, n = 0, \text{notfinished}=1; \)
- while (notfinished & \( n \leq N \)) do:
  - notfinished = 0;
  - for \( i = 0, 1, \ldots, I \) and \( j = 0, 1, \ldots, J \) do:
    - if \( s_{ij} = 0 \) then continue;
    - compute \( \nabla f^n_{ij} \) in (12) - (13); if \( |\nabla f^n_{ij}| < \epsilon_2 \) then continue;
    - notfinished=1;
    - compute \( \alpha^n_{ij} \) in (20);
    - if \( \alpha^n_{ij} = 0 \) then compute \( \tau^n_{ij} \) in (23);
    - if \( \alpha^n_{ij} \neq 0 \) then
      1. compute \( D \) in (24);
      2. if \( D < 0 \) then compute \( \tau^n_{ij} \) in (25) else compute \( \tau^n_{ij} \) in (26);
      - compute \( e_j^{n+1} \) in (28) using (27);
      - compute \( X_{ij}^{n+1} \) in (29) and \( \bar{f}_{ij}^{n+1} \) in (30);
      - if \( |G_{ij} - e_j^{n+1}| < \epsilon_1 \) or \( |G_{ij} - \bar{f}_{ij}^{n+1}| < \epsilon_1 \) then set \( s_{ij} = 0; \)
      - \( n = n + 1; \)
  - for \( i = 0, 1, \ldots, I \) and \( j = 0, 1, \ldots, J \) compute \( \bar{D}_{ij} \) in (31);
In the next section, we apply the method for two illustrative examples.

4. Numerical experiments. In this section, we present two examples where the first one shall illustrate the experimental order of convergence of the proposed method, and the second one illustrates a possible application of the method for an optical flow estimation of real biological images.

4.1. Example with exact solution. The domain $\Omega$ is a unit square. The functions $F$ and $G$ are defined as follows

\[ F(x) = |x - c|, \quad G(x) = \max\{0, F(x) - 0.1\}, \quad (32) \]

where $c = (0.5, 0.5)$ and $|\cdot|$ denotes the standard Euclidean distance. For a plot of both functions see Figure 1.

Firstly, we compare the origin Rouy-Tourin (RT) scheme (15) using the linear interpolation with the new Corner Transport Upwind (CTU) scheme (17) using the bilinear interpolation. We choose $\epsilon_1 = \epsilon_2 = 0$ in the pseudo code. The discretization is realized for $I = J = 10, 20, \ldots, 160$ giving the discretization steps $h = 0.1, 0.05, \ldots, 0.00625$. The time step is chosen maximal one to fulfill (18), i.e. $\tau = h$. The number of time steps is given by $N = I/10$. The results are summarized in Table 1, where the following discrete norm

\[ E_* = h^2 \sum_{i,j} |G_{ij} - f^N_{ij}|, \quad (33) \]

is computed with $* = RT$ and $* = CTU$. One can see that the new method can produce better and stable results.

Table 1. The error norm (35) and the corresponding $EOC$ for the example with exact solution.

| $I$ | $N$ | $E_{RT}$   | EOC | $E_{CTU}$    | EOC |
|-----|-----|------------|-----|--------------|-----|
| 10  | 1   | 0.00703    | -   | 0.00312      | -   |
| 20  | 2   | 0.00401    | 0.81| 0.00171      | 0.87|
| 40  | 4   | 0.00235    | 0.77| 0.000893     | 0.94|
| 80  | 8   | 0.00160    | 0.55| 0.000458     | 0.96|
| 160 | 16  | 0.00281    | -0.81| 0.000232   | 0.98|

Figure 1. The example with exact solution: the function $F$ (left), the function $G$ (middle), and the deformation $\vec{D}^x$ (right).
Secondly, we use the method (30) to compute $L f_{ij}^N$ for which also the deformation field $\vec{D}_{ij}$ in (29) is computed. The exact optical flow is described by the deformation

$$\vec{D}_{ex}(x) = \min\{1, \frac{0.1}{|x - c|}\} (x - c).$$  \hspace{1cm} (34)$$

The deformation is plotted in Figure 1 graphically as $-\vec{D}_{ex}$ to show that the value of $G$ at the position $x_{ij}$ (where an arrow starts) equals to the value of $F$ at the position $x_{ij} - \vec{D}_{ex}(x_{ij})$ (where the arrow ends), so $G(x_{ij}) = F(x_{ij} - \vec{D}_{ex}(x_{ij}))$.

The discretization parameters are identical to the first case. The results are presented graphically in Figure 2, where one can observe an improvement of the approximation $\vec{D}_{ij} \approx \vec{D}(x_{ij})$ with finer grids.

In Table 2 the discrete norm $E_*$ in (35) with $* = L$ is presented. Furthermore, the discrete norm $E_D$ for a difference between the exact first component of the
Table 2. The $L_1$-norms and the experimental rates of convergence for the example with exact solution.

| $I$ | $N$ | $E_L$  | EOC  | $E_D$  | EOC  |
|-----|-----|--------|------|--------|------|
| 10  | 1   | 0.003120 | -    | 0.004433 | -    |
| 20  | 2   | 0.001307 | 1.2556 | 0.002379 | 0.8985 |
| 40  | 4   | 0.000528 | 1.3075 | 0.001259 | 0.9175 |
| 80  | 8   | 0.000220 | 1.2667 | 0.000659 | 0.9339 |
| 160 | 16  | 0.000096 | 1.1947 | 0.000339 | 0.9590 |

Figure 3. The images of lungs scan: the source image $F$ (left), the target image $G$ (middle), the difference image $|G - F|$ (right).

Deformation $\vec{D}^{ex}$ and the numerical first component of $\vec{D}_{ij}$ is presented,

$$E_D = h^2 \sum_{i,j} |D^{ex}_1(x_{ij}) - D_{1,ij}|,$$

(35)

where $\vec{D}^{ex} = (D^{ex}_1, D^{ex}_2)$ and $\vec{D}_{ij} = (D_{1,ij}, D_{2,ij})$ (the norm in (35) for the second component of $\vec{D}^{ex} - \vec{D}$ is identical). One can observe in Table 2 that these discrete norms are giving the experimental order of convergence approximately 1 in both cases.

4.2. Example with images of lungs. In this example, the functions $F$ and $G$ are obtained by the bilinear interpolation of two pictures of the size $263 \times 190$ pixels with $h = 1$, see Figure 3, where also the difference picture $|G - F|$ is plotted. The range of values for $F$ and $G$ is between 0 and 192. To plot the difference image $|G - F|$, it is scaled to have the maximal value 256.

We choose a large enough $N$ and $\epsilon_1 = \epsilon_2 = 0$ to show clearly the behavior of the method. The normalized norm

$$e^n = \frac{1}{263 \cdot 190} \sum_{i,j} |G_{ij} - F(X^n_{ij})|$$

(36)

is presented in Figure 4 for $n \leq N = 10$. One can observe a fast decrease of this norm for first time steps.

The image obtained from the interpolation of values $F(x_{ij} - \vec{D}_{ij})$ is plotted in Figure 5 together with the difference image $|G(x) - F(x - \vec{D}(x))|$. The approximation of deformation $\vec{D}$ is plotted in Figure 6. To make a more clear presentation of this deformation, we show it also in Figure 7, where the arrows are plotted only in the points, where the values $|G_{ij} - F_{ij}|$ are larger than a chosen value $E_{crit}$, in this
5. Conclusions. We deal with the problem to find a function that evolves from a given source function to a given target function by the flow restricted only to the normal direction of level sets. The speed of evolution in the normal direction in the corresponding level set advection equation equals to the sign of the difference of the two given functions, and it changes discontinuously to zero when the evolving function takes the values of target function or when the gradient vanishes. One of a possible application of the method is the optical flow estimation that searches a deformation between two images, when the given images are represented by the functions defined by the bilinear interpolation of the greyscale values in pixels.

The new presented numerical method for such problems in the two-dimensional case has two major novelties. First, opposite to the standard Rouy-Tourin scheme that is based on the linear interpolation defined by three values of the numerical solution, we extend the scheme to use the bilinear interpolation based on four values of the numerical solution. The new method enables larger time steps than the Rouy-Tourin scheme. Second, the numerical method ensures a monotone evolution of the evolving numerical function under natural assumptions, and the evolution stops

particular case

\[ E_{\text{crit}} = \frac{1}{15} \max_{i,j} |G_{ij} - F_{ij}|. \]  

(37)
sharply when the value of the target function is reached locally. Finally, when several time steps shall be used to find the deformation by the backward tracking of characteristic curves, we propose its modification that does not require the storage of results for the intermediate steps.

In numerical experiments, we give an example with an exact solution, where the experimental order of convergence gives the expected first order accuracy. Furthermore, the example with two images of lungs illustrates a possible usage of the method for optical flow estimation for real biological data.

REFERENCES

[1] M. Bertalmío, G. Sapiro and G. Randall, Morphing active contours, *IEEE Trans. PAMI*, 22 (2000), 733–737.
[2] A. Bruhn, J. Weickert and C. Schnörr, Lucas/Kanade meets Horn/Schunck: Combining local and global optic flow methods, *Int. J. Comput. Vis.*, 61 (2005), 211–231.
[3] M. Burger, H. Dirks and C.-B. Schönlieb, A variational model for joint motion estimation and image reconstruction, *SIAM J. Imaging Sci.*, 11 (2018), 94–128.
[4] P. Colella, Multidimensional upwind methods for hyperbolic conservation laws, *J. Comput. Phys.*, 87 (1990), 171–200.
[5] V. Duay, N. Houhou and J.-P. Thiran, Atlas-based segmentation of medical images locally constrained by level sets, IEEE International Conference on Image Processing, Genova, Italy, 2005.
[6] P. Frolkovič and K. Mikula, Semi-implicit second order schemes for numerical solution of level set advection equation on Cartesian grids, *Appl. Math. Comput.*, 329 (2018), 129–142.
[7] F. Gibou, R. Fedkiw and S. Osher, A review of level-set methods and some recent applications, *J. Comput. Phys.*, 353 (2018), 82–109.
[8] B. Horn and B. Schunck, Determining optical flow, *Artificial Intelligence*, 17 (1981), 185–203.
Figure 7. The plot of deformation $\vec{D}$ for the example with the images of lungs. Only arrows in the points where $|G_{ij} - F_{ij}| > E_{crit}$ are plotted.

[9] C.-O. Lee, K. Jeon, Y. Ha and J. Hahn, A variational approach to blending based on warping for non-overlapped images, Comput. Vis. Image Und., 105 (2007), 112–120.
[10] R. J. LeVeque, Finite Volume Methods for Hyperbolic Problems, Cambridge Texts in Applied Mathematics, Cambridge University Press, Cambridge, 2002.
[11] S. Osher and R. Fedkiw, Level Set Methods and Dynamic Implicit Surfaces, Applied Mathematical Sciences, 153, Springer-Verlag, New York, 2003.
[12] E. Rouy and A. Tourin, A viscosity solutions approach to shape-from-shading, SIAM J. Numer. Anal., 29 (1992), 867–884.
[13] J. Sethian, Level Set Methods and Fast Marching Methods, Cambridge Monographs on Applied and Computational Mathematics, 3, Cambridge University Press, Cambridge, 1999.
[14] B. C. Vemuri, J. Ye, Y. Chen and C. M. Leonard, Image registration via level-set motion: Applications to atlas-based segmentation, Medical Image Anal., 7 (2003), 1–20.