Image Sequence Interpolation Based on Optical Flow, Segmentation, and Optimal Control

Kanglin Chen and Dirk A. Lorenz

***Abstract*—When using motion fields to interpolate between two consecutive images in an image sequence, a major problem is to handle occlusions and disclusions properly. However, in most cases, one of both images contains the information that is either discluded or occluded; if the first image contains the information (i.e., the region will be occluded), forward interpolation shall be employed, while for information that is contained in the second image (i.e., the region will be discluded), one should use backward interpolation. Hence, we propose to improve an existing approach for image sequence interpolation by incorporating an automatic segmentation in the process, which decides in which region of the image forward or backward interpolation shall be used. Our approach is a combination of the optimal transport approach to image sequence interpolation and the segmentation by the Chan–Vese approach. We propose to solve the resulting optimality condition by a segregation loop, combined with a level set approach. We provide examples that illustrate the performance both in the interpolation error and in the human perception.**

***Index Terms*—Active contours, image sequence interpolation, optimal control, optical flow, segmentation, transport equation.**

I. INTRODUCTION

IMAGE sequence interpolation is the generation of intermediate images between two given consecutive images, i.e., a process that is, for example, relevant if the image acquisition is slowly or expensive and has broad applications in the fields of video compression, medical imaging, etc. In video compression, the knowledge of motions helps in removing the nonmoving parts of images and in compressing video sequences with high compression rates. For example, in the Motion Pictures Expert Group format, motion estimation is the most computationally expensive portion of the video encoder and is normally solved by mesh-based matching techniques [1]. While decompressing a video, intermediate images are generated by warping the image sequence with motion vectors. In the field of medical imaging, image sequence interpolation is also desired. For example, the diagnostic requires a point-by-point correspondence between the same tissue from the image sequence taken at difference times [2]. Moreover, image sequence interpolation is also able to improve the quality of historic movies by increasing the frame rate to the modern standard. Similarly, in disease diagnostics, an image of a patient's tissue may need to compare with a healthy tissue [3]. This is an example of how image sequence interpolation in some cases can be used to solve the application normally classified as image registration. However, in this paper, we focus on movielike image sequences; these sequence are notably different from registration problems in which we may have different objects that move in different directions resulting in disclusions and occlusions.

Considering the problem of image sequence interpolation, the optical flow (the measurable 2-D motion field between two images) plays a decisive role. Since Horn and Schunck proposed the variational approach to estimate optical flow in their celebrated work [4], this field has been widely developed. To preserve the flow edges, nonlinear isotropic constraint was applied instead of the linear constraint in the Horn and Schunck method [5], [6], an anisotropic diffusion constraint improved the preservation of edges by an oriented smoothness constraint in which smoothness is not imposed across edges [7], [8], and the TV-L^1 method is not only able to preserve the flow edges but also able to robustly work against the outliers [9].

There are several existing variational methods based on optical flow to interpolate missing intermediate images. In [10], the variational method penalized by the elastic regularization is considered, i.e.,

$$J_{\text{rigid}}(u,b) = \int\limits_{[0,T]\times\Omega} (u_t + b\cdot\nabla u)^2 + \lambda|\nabla b' + \nabla b|^2 dxdt$$

where b denotes the optical flow and $\nabla b'$ denotes the transpose of the Jacobi matrix of b. Hence, they do not exactly enforce the brightness constancy constraint $u_t + b\cdot\nabla u = 0$ but penalize its violation as in the classical Horn and Schunck approach. Minimizing this functional gives the interpolated images with maximal rigidity and has applications in the field of medical image registration, e.g., registration of magnetic resonance images. In [11], the authors keep the assumption of brightness constancy without differentiating it and update the flow field with the help of robust estimators. There, the authors also incorporated object-based motion segmentation. In [12], the authors also keep the assumption of brightness constancy without differentiating it and apply the time-dependent Horn and Schunck functional, i.e.,

$$J_{\text{cons}}(b) = \frac{\lambda}{2}\int\limits_0^T \|u(t) - u_T\|^2_{L^2(\Omega)}\, dt + \frac{1}{2}\int\limits_0^T\int\limits_\Omega |\nabla b|^2 dxdt$$

where $u(0) = u_0$ and u_T are the given two images. After calculating the time-dependent optical flow, one can warp the initial

Manuscript received April 28, 2011; revised August 31, 2011 and November 09, 2011; accepted December 01, 2011. Date of publication December 09, 2011; date of current version February 17, 2012. The associate editor coordinating the review of this manuscript and approving it for publication was Prof. Kenneth K. M. Lam.

K. Chen is with the Center for Industrial Mathematics (ZeTeM), Department of Mathematics, University of Bremen, 28359 Bremen, Germany (e-mail: kanglin@math.uni-bremen.de).

D. A. Lorenz is with the Department of Mathematics, Technische Universität Braunschweig, 38092 Braunschweig, Germany (e-mail: d.lorenz@tu-braunschweig.de).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TIP.2011.2179305

image u_0 to a certain time. In [13], the authors do enforce the brightness constancy constraint again and minimize a functional with equation $u_t + b \cdot \nabla u = 0$ as a constraint.

Different from the global variational methods are the so-called pixelwise methods. In [14], the path-based interpolation sequence method is considered. There, one searches where every pixel comes from and traces out the path of every pixel from the given two images. To stabilize the interpolation and to handle occlusion, postprocessing is used by means of verification of the displacement flow. In [15] and [16], another pixelwise method is introduced, namely, the perception-based interpolation. They simulate human visual perception in the following way: To begin with, they detect the edges and homogeneous regions, and then, they estimate the translets by matching edges; finally, they use the forward warping and feather the interpolated images.

Aside from the aforementioned image sequence interpolation methods, the image warping technique was introduced in [17] to generate the intermediate image based on *a priori* known optical-flow field, e.g., estimated by the Horn and Schunck method. However, this kind of optical flow may not be suitable for image sequence interpolation (see [12] and [15]). In [18], we introduced a more natural way to utilize the optical flow into image sequence interpolation under the framework of optimal control similar to [13]. This method can be applied to the cases that image sequence obeys rigid and nonrigid movements, and also robustly works against noise.

In this paper, we aim to eliminate a common drawback of all flow-based methods for image sequence interpolation. While using forward interpolation, it is impossible to obtain good results for regions that are disclosed, since any method has to guess the appearing pixels. Similarly, backward interpolation will fail in regions that are occluded. To solve this problem, we propose an extension of our method proposed in [18], which incorporates a segmentation process for the image domain to automatically detect regions in which forward or backward interpolation shall be employed.

This paper is organized as follows: In Section II, we review the segmentation model in [19], while in Section III, we recall the basics of our proposed optimal control approach to image sequence interpolation. Section IV presents the combination of both approaches, and Section V presents details on the numerics.

II. SEGMENTATION WITH ACTIVE CONTOURS

The classical active contours models or snakes [20], [21] are widely used in image segmentation. However, in these models, an edge detector related to the image gradient is needed to stop the evolving curve on the boundaries of objects. In [19], Chan and Vese introduced a model based on active contours and the Mumford–Shah segmentation [22], which does not require an edge detector. Consequently, this model can detect contours both with and without gradient, e.g., for objects with very smooth boundaries or even with discontinuous boundaries. We review the model of active contours without edges for the sake of completeness.

Let us define curve C as the boundary of an open subset ω of a bounded domain $\Omega \subset \mathbb{R}^2$. Assume that C segments Ω into ω and $\Omega \setminus \overline{\omega}$, and constants c_1 and c_2 depending on C are the average of image u inside and outside C, respectively. Denoting with $|C|$ the length of C and with $|\omega|$ the area of ω, the segmentation will be achieved by minimizing the following energy:

$$F(c_1, c_2, C) = \lambda_1 \int\limits_{\omega} |u - c_1|^2 dx + \lambda_2 \int\limits_{\Omega\setminus\overline{\omega}} |u - c_2|^2 dx + \mu|C| + \nu|\omega| \quad (1)$$

where $\mu \geq 0$, $\nu \geq 0$, λ_1, and $\lambda_2 > 0$ are the regularization parameters. To minimize (1), one uses a level set formulation. Suppose C is represented by the zero level set of a Lipschitz function $\phi : \Omega \to \mathbb{R}$, such that

$$\begin{cases} C = \partial\omega = \{x \in \Omega : \phi(x) = 0\} \\ \omega = \{x \in \Omega : \phi(x) > 0\} \\ \Omega \setminus \overline{\omega} = \{x \in \Omega : \phi(x) < 0\}. \end{cases}$$

Using the Heaviside function H and the 1-D Dirac measure δ_0 defined as

$$H(z) = \begin{cases} 1 & \text{if } z \geq 0 \\ 0 & \text{if } z < 0 \end{cases} \qquad \delta_0(z) = \frac{d}{dz}H(z)$$

one can reformulate (1) in the following way:

$$\begin{aligned} F(c_1, c_2, \phi) = {} & \lambda_1 \int\limits_{\Omega} |u - c_1|^2 H(\phi(x))\, dx \\ & + \lambda_2 \int\limits_{\Omega} |u - c_2|^2 (1 - H(\phi(x)))\, dx \\ & + \mu \int\limits_{\Omega} \delta_0(\phi(x)) |\nabla\phi(x)|\, dx \\ & + \nu \int\limits_{\Omega} H(\phi(x))\, dx. \end{aligned}$$

In order to compute the associated Euler–Lagrange equations with respect to ϕ, one chooses a smooth approximation H_s and $\delta_s = H_s'$, e.g.,

$$\begin{aligned} H_s(z) &= \frac{1}{2}\left(1 + \frac{2}{\pi}\arctan\left(\frac{z}{s}\right)\right) \\ \delta_s(z) &= \frac{1}{s\pi}\cos^2\left(\arctan\left(\frac{z}{s}\right)\right) \end{aligned}$$

which converge to H (pointwise almost everywhere) and δ (in the sense of distributions) as $s \to 0$. Let us define for $s, \varepsilon > 0$, functional $F_{s,\varepsilon}$ by

$$\begin{aligned} F_{s,\varepsilon}(c_1, c_2, \phi) = {} & \lambda_1 \int\limits_{\Omega} |u - c_1|^2 H_s(\phi(x))\, dx \\ & + \lambda_2 \int\limits_{\Omega} |u - c_2|^2 (1 - H_s(\phi(x)))\, dx \\ & + \mu \int\limits_{\Omega} \delta_s(\phi(x)) |\nabla\phi(x)|_\varepsilon\, dx \\ & + \nu \int\limits_{\Omega} H_s(\phi(x))\, dx \end{aligned}$$

where $|\cdot|_\varepsilon$ denotes the ε-smoothed total variation functional defined by

$$|\nabla\phi|_\varepsilon = \sqrt{|\nabla\phi|^2 + \varepsilon}.$$

To minimize $F_{s,\varepsilon}$ with respect to ϕ, one deduces the associated Euler–Lagrange equations for ϕ and parameterizes the descent direction by an artificial time $t \geq 0$. The equation in $\phi(t,x)$ with the initial contour $\phi(0,x) = \phi_0(x)$ is

$$\begin{cases} \frac{\partial \phi}{\partial t} = \delta_s(\phi)\left(\mu\nabla\cdot\left(\frac{\nabla\phi}{|\nabla\phi|_\varepsilon}\right) - \nu - \lambda_1(u-c_1)^2\right. \\ \qquad\qquad \left. +\lambda_2(u-c_2)^2\right) \text{ in } (0,\infty)\times\Omega \\ \phi(0) = \phi_0 \text{ in } \Omega \\ \frac{\delta_s(\phi)}{|\nabla\phi|_\varepsilon}\frac{\partial\phi}{\partial n} = 0 \text{ on } \partial\Omega \end{cases} \tag{2}$$

where $\partial\phi/\partial n$ denotes the normal derivative of ϕ on the boundary.

III. OPTICAL-FLOW-BASED OPTIMAL CONTROL FOR IMAGE SEQUENCE INTERPOLATION

Given two consecutive images u_0 and u_T, we desire to find a flow field such that the field drives the transport equation with the initial value u_0 to fit u_T at time T as well as possible. This process has been accomplished in [18] under the framework of optimal control, and we briefly review this method. Consider the Cauchy problem for the transport equation in $[0,T]\times\Omega$, $\Omega\subset\mathbb{R}^2$, i.e.,

$$\begin{cases} \partial_t u(t,x) + b(t,x)\cdot\nabla u(t,x) = 0 \text{ in }]0,T]\times\Omega \\ u(0,x) = u_0(x) \text{ in } \Omega \\ u_n(t,x) = 0 \text{ in }]0,T]\times\partial\Omega. \end{cases} \tag{3}$$

Here, the (time-dependent) flow field is denoted by $b : [0,T]\times\Omega \longrightarrow \mathbb{R}^2$, the image function depending on t and x is denoted by u, and u_n denotes its normal derivative. The Neumann boundary condition $u_n = 0$ is not essential in this case since we assume that b vanishes on $\partial\Omega$ for almost everywhere $t \in]0,T]$ in the following context.

Our intention is to find a flow field b such that the "transported" image $u(T)$ at time T matches image u_T as well as possible. This motivates us to minimize functional $(1/2)\|u(T) - u_T\|^2_{L^2(\Omega)}$. However, this problem is ill posed, and hence, we add an additional regularization term in the cost functional. In addition, we add the divergence-free constraint of b and obtain an optimal control problem as follows for a given $\lambda > 0$: Minimize

$$J(b) = \frac{1}{2}\|u(T)-u_T\|^2_{L^2(\Omega)} + \frac{\lambda}{2}\int_0^T \|\nabla b(t,\cdot)\|^2_{L^2(\Omega)^4}\,dt.$$

$$\text{subject to} \quad \text{div} b = 0 \text{ and } (3).$$

The associated Karush–Kuhn–Tucker system for the optimal control problem uses a dual variable p for constraint (3) and a dual variable q for the divergence-free constraint, and is given by

$$\begin{cases} u_t + b\cdot\nabla u = 0 \text{ in }]0,T]\times\Omega \text{ with } u(0) = u_0 \text{ in } \Omega \\ p_t + b\cdot\nabla p = 0 \text{ in } [0,T[\times\Omega \\ \qquad\qquad \text{with } p(T) = -\left(u(T)-u_T\right) \text{ in } \Omega \\ \lambda\Delta b + \nabla q = p\nabla u \text{ in } [0,T]\times\Omega \\ \text{div} b = 0 \text{ in } [0,T]\times\Omega \end{cases} \text{with } b = 0 \text{ on } \partial\Omega \tag{4}$$

According to the conservation law [23] and the divergence theorem [24], the divergence-free constraint of b makes the flow volume conserving, smooth, and varying not too much inside a moving object. At least, the last two properties are desirable for the computation of the optical flow. Moreover, the divergence-free constraint is a somehow technical assumption as it implies that the equation for the dual variable p of u is also a transport equation and hence simplifies the numerical implementation.

To solve (4) numerically, we apply a modified segregation loop. We suppose $n = 1,\ldots,N_{\text{loop}}$ and N_{loop} is the iteration number. Given u_0, u_T, $b^{n-1}(t)$, and λ, the iteration process at iteration n proceeds.

1) Compute $u^{n-1}(t)$, $\nabla u^{n-1}(t)$, and $u^{n-1}(T)$ by the forward transport equation using u_0 and $b^{n-1}(t)$.
2) Compute $p^{n-1}(t)$ by the backward transport equation using $-(u^{n-1}(T) - u_T)$ and $b^{n-1}(t)$.
3) Compute the solution of the Stokes equations with right-hand side $p^{n-1}(t)\nabla u^{n-1}(t)$ and λ. Then, denote it by $\delta b^{n-1}(t)$.
4) Update $b^n(t) = b^{n-1}(t) + \delta b^{n-1}(t)$.

Although the segregation loop does not solve (4) directly, it is shown in [18] that the modification with update δb^{n-1} actually solves the necessary conditions of another optimization problem, namely, minimize

$$\frac{1}{2}\|u(T)-u_T\|^2_{L^2(\Omega)}$$

subject to

$$\begin{cases} u_t + b\cdot\nabla u = 0 \text{ in }]0,T]\times\Omega, \text{ with } u(0) = u_0 \text{ in } \Omega \\ \text{div} b = 0 \text{ in } [0,T]\times\Omega, \text{ with } b = 0 \text{ on } \partial\Omega. \end{cases}$$

From the point of view of the regularization theory, one may see this segregation loop as a kind of a Landweber method for minimizing $\|u(T)-u_T\|^2_{L^2(\Omega)}$, which is inspired by a Tikhonov functional.

IV. OPTICAL-FLOW- AND SEGMENTATION-BASED OPTIMAL CONTROL FOR IMAGE SEQUENCE INTERPOLATION

A. *Modeling*

Observing the movement of objects in an image sequence, we may divide the domain into the "covered" and "disclosed" domains. The "covered" domain refers to the regions in which the characteristics of two different pixels starting at time 0 end up at time T in a same place. Obviously, the " covered" domain is suitable for the forward interpolation from 0 to T. In contrast, the " disclosed" domain refers to the regions in which no characteristic of a pixel starting at time 0 ends up at time T in a place. Since our interpolation method under the framework of optimal control will produce a continuous optical flow, the " disclosed" domain will be filled in with the neighbors, and hence, we get a dense optical flow. However, using the filled-in optical flow is still impossible to recover the objects in the " disclosed" domain, if we only take information from u_0. To overcome this drawback, which is inherent in all flow-based methods, we can apply a backward interpolation from T to 0 in the "disclosed" domain, i.e., the "disclosed" domain is turned to the "covered" domain in this case. An illustrative example of this phenomenon is the data set MiniCooper,[1] which is shown in Fig. 2. In the zoomed-in subimages, one easily observes that, in the upper part of the head region and the rear part of the car, some new objects (pixels) appear.

---

[1] http://vision.middlebury.edu/flow/data/

Motivated by this explanation, we propose to apply active contours to achieve an automatic selection process of the regions for forward or backward interpolation. To that end, we incorporate the Chan–Vese segmentation process described in Section II into our optimal control framework from Section III. Moreover, we modify our optimal control framework in which we use a smoothed TV penalty for the flow field b to obtain sharper edges for flow. We assume that b vanishes on $\partial\Omega$ and model the evolving curve C in Ω as the boundary of an open subset ω of Ω. The forward interpolation, denoted by $\hat{u}$, shall take place in set ω, and the backward interpolation $\tilde{u}$ shall be used in $\Omega \setminus \omega$. Hence, our cost functional is defined as

$$\begin{aligned} L(b,C,\omega) &= \frac{1}{2}\|\hat{u}(T)-u_T\|^2_{L^2(\omega)} + \frac{1}{2}\|\tilde{u}(0)-u_0\|^2_{L^2(\Omega\setminus\overline{\omega})} \\ &\quad +\lambda\int_0^T\int_\Omega \sqrt{|\nabla b|^2+\varepsilon}\,dx\,dt + \mu|C| + \nu|\omega| \end{aligned} \tag{5}$$

governed by the forward transport equation, i.e.,

$$\begin{cases} \hat{u}_t + b\cdot\nabla\hat{u} = 0 \text{ in }]0,T]\times\Omega \\ \hat{u}(0) = u_0 \text{ in } \Omega \end{cases}$$

the backward transport equation, i.e.,

$$\begin{cases} \tilde{u}_t + b\cdot\nabla\tilde{u} = 0 \text{ in } [0,T[\times\Omega \\ \tilde{u}(T) = u_T \text{ in } \Omega \end{cases}$$

and the divergence-free equation, i.e.,

$$\text{div} b = 0 \text{ in } [0,T]\times\Omega.$$

The desired interpolation u at time t is estimated by

$$u(t,x) = \begin{cases} \hat{u}(t,x) & x\in\omega \\ \tilde{u}(t,x) & x\in\Omega\setminus\overline{\omega}. \end{cases} \tag{6}$$

Minimizing (5), we obtain the optical flow and the active contours for interpolation. Although we do not compute $u(t)$ directly from (5), in Section V, we shall see that it is necessary to compute $\hat{u}(t)$ and $\tilde{u}(t)$ by computing the optical flow and the active contours. Thus, interpolating $u(t)$ from (6) requires almost no additional computation.

To turn the cost functional (5) into a functional that is computationally feasible, we follow the lines of Chan and Vese described in Section II: We assume that ϕ is the zero level set of C introduced in Section II and use a smoothed Heaviside function to reformulate (5) in terms of level set as

$$\begin{aligned} J_{s,\varepsilon}(b,\phi) &= \frac{1}{2}\int_\Omega |\hat{u}(T)-u_T|^2 H_s(\phi)\,dx \\ &\quad + \frac{1}{2}\int_\Omega |\tilde{u}(0)-u_0|^2 (1-H_s(\phi))\,dx \\ &\quad + \lambda\int_0^T\int_\Omega \sqrt{|\nabla b|^2+\varepsilon}\,dx\,dt \\ &\quad + \mu\int_\Omega \delta_s(\phi)|\nabla\phi|_\varepsilon dx + \nu\int_\Omega H_s(\phi)\,dx. \end{aligned} \tag{7}$$

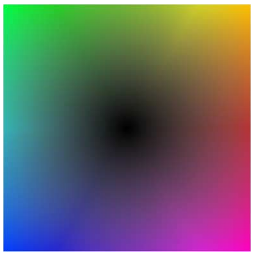


Fig. 1. Color coding map of the optical flow.

B. First-Order Necessary Optimality Conditions

We obtain the first-order necessary optimality conditions by defining the Lagrangian (with Lagrange multipliers $\hat{p}$, $\tilde{p}$ and q) as

$$\begin{aligned} &L(\hat{u},\tilde{u},b,\phi,\hat{p},\tilde{p},q) \\ &\quad = \frac{1}{2}\int_\Omega |\hat{u}(T)-u_T|^2 H_s(\phi)\,dx \\ &\qquad + \frac{1}{2}\int_\Omega |\tilde{u}(0)-u_0|^2 (1-H_s(\phi))\,dx \\ &\qquad + \lambda\int_0^T\int_\Omega \sqrt{|\nabla b|^2+\varepsilon}\,dx\,dt + \int_0^T\int_\Omega (\hat{u}_t + b\cdot\nabla\hat{u})\hat{p}\,dx\,dt \\ &\qquad + \int_0^T\int_\Omega (\tilde{u}_t + b\cdot\nabla\tilde{u})\tilde{p}\,dx\,dt + \int_0^T\int_\Omega q\,\text{div} b\,dx\,dt \\ &\qquad + \mu\int_\Omega \delta_s(\phi)|\nabla\phi|_\varepsilon dx + \nu\int_\Omega H_s(\phi)\,dx. \end{aligned}$$

Finally, the necessary optimality conditions system consists of four equations:

1) the forward transport equation and its adjoint equation, i.e.,

$$\begin{cases} \hat{u}_t + b\cdot\nabla\hat{u} = 0 \text{ in }]0,T]\times\Omega \\ \hat{u}(0) = u_0 \text{ in } \Omega \\ \hat{p}_t + b\cdot\nabla\hat{p} = 0 \text{ in } [0,T[\times\Omega \\ \hat{p}(T) = -\left(\hat{u}(T)-u_T\right)H_s(\phi) \text{ in } \Omega \end{cases} \tag{8}$$

2) the backward transport equation and its adjoint equation, i.e.,

$$\begin{cases} \tilde{u}_t + b\cdot\nabla\tilde{u} = 0 \text{ in } [0,T[\times\Omega \\ \tilde{u}(T) = u_T \text{ in } \Omega \\ \tilde{p}_t + b\cdot\nabla\tilde{p} = 0 \text{ in }]0,T]\times\Omega \\ \tilde{p}(0) = \left(\tilde{u}(0)-u_0\right)\left(1-H_s(\phi)\right) \text{ in } \Omega \end{cases} \tag{9}$$

3) the TV-Stokes equation (cf. [25]), i.e.,

$$\begin{cases} \lambda\nabla\cdot\left(\frac{\nabla b}{|\nabla b|_\varepsilon}\right) + \nabla q = \hat{p}\nabla\hat{u} + \tilde{p}\nabla\tilde{u} \text{ in } [0,T]\times\Omega \\ \text{div} b = 0 \text{ in } [0,T]\times\Omega \\ b = 0 \text{ on } \partial\Omega \end{cases} \tag{10}$$

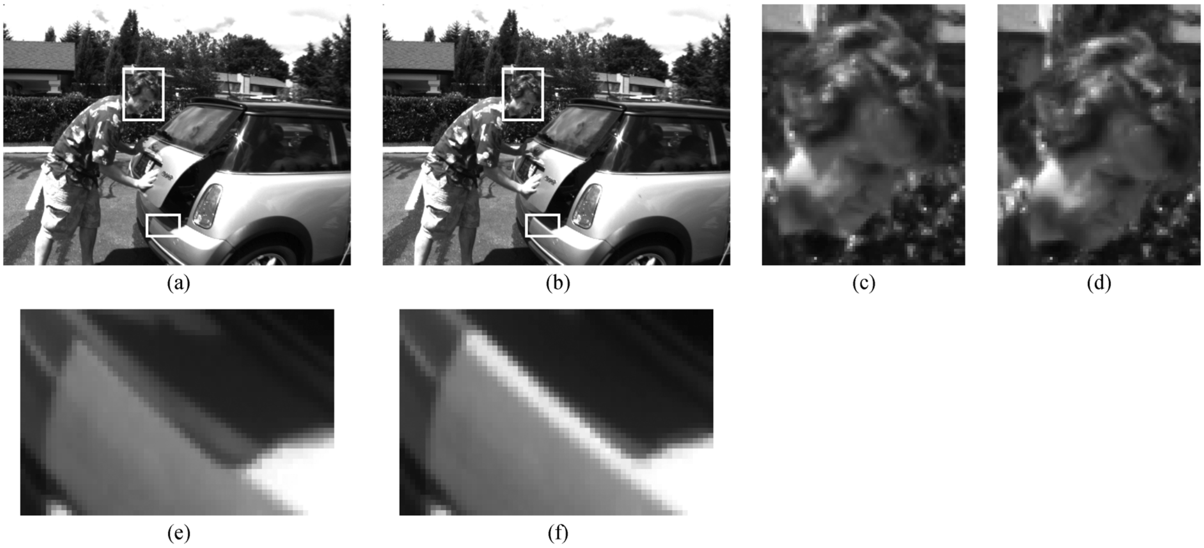


Fig. 2. Frames (a) 10 and (b) 11. (c) Zoomed-in region of the head area in (a). (d) Zoomed-in region of the head area in (b). (e) Zoomed-in region of the rear of the car in (a). (f) Zoomed-in region of the rear of the car in (b).

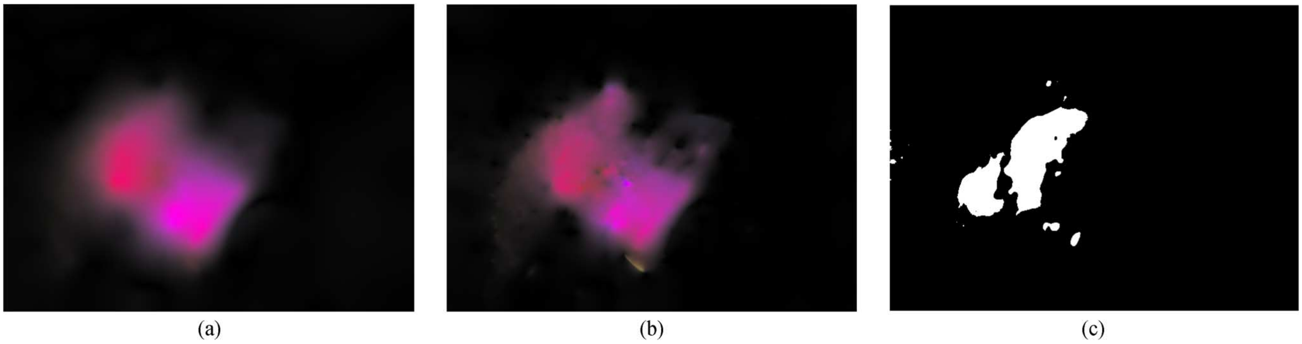


Fig. 3. Experiment on frames 10 and 11 of Fig. 2. (a) Optical flow calculated by the smooth method. (b) Optical flow calculated by the TV$_\varepsilon$-segment method. (c) Active contours calculated by the TV$_\varepsilon$-segment method. (Black) Backward and (white) forward interpolation regions.

4) the equation for segmentation

$$\begin{cases} \delta_s(\phi)\left(\mu\nabla\cdot\left(\frac{\nabla\phi}{|\nabla\phi|_\varepsilon}\right)-\nu-\frac{1}{2}\left|\hat{u}(T)-u_T\right|^2\right. \\ \qquad\qquad\left.+\frac{1}{2}\left|\tilde{u}(0)-u_0\right|^2\right)=0 \text{ in } \Omega \\ \frac{\delta_s(\phi)}{|\nabla\phi|_\varepsilon}\frac{\partial\phi}{\partial n}=0 \text{ on } \partial\Omega. \end{cases} \tag{11}$$

V. NUMERICAL ASPECTS

To solve the forward- and backward-transport equations (8) and (9), we utilize the method of characteristics by solving the associated ODE using Runge–Kutta fourth order [26]. To solve the TV-Stokes equation (10) at time t, we apply the following iterative procedure to update b and q with time step Δt, i.e.,

$$\begin{aligned} b^{n+1}(t) &= b^n(t)+\Delta t\left(\nabla\cdot\left(\frac{\nabla b^n(t)}{|\nabla b^n(t)|_\varepsilon}\right)+\frac{1}{\lambda}\nabla q^n(t)\right. \\ &\qquad\left.-\frac{1}{\lambda}\hat{p}(t)\nabla\hat{u}(t)-\frac{1}{\lambda}\tilde{p}(t)\nabla\tilde{u}(t)\right) \\ q^{n+1}(t) &= q^n(t)+\Delta t\nabla\cdot b^n(t). \end{aligned} \tag{12}$$

In [27], it is shown that this explicit (forward Euler) time-marching scheme is conditionally stable, i.e., the time step Δt should be selected in a manner that gives sufficient decrease in the functional. However, the forward scheme has rather undesirable asymptotic convergence properties, which can make it inefficient. Hence, Vogel and Oman introduced the lagged diffusivity fixed point iteration, denoted by FP-iteration, in [27]. The FP-iteration linearizes the nonlinear diffusion part in (12) at iteration $n+1$, i.e., we apply the following diffusion operator:

$$DF(b^n)v=\nabla\cdot\left(\frac{\nabla v}{|\nabla b^n|_\varepsilon}\right)$$

at the active iteration $n+1$. Hence, we can formulate it into an implicit scheme, i.e.,

$$(1-\Delta t DF(b^n))\,b^{n+1}=z$$

where z denotes the rest of the terms not involving b^{n+1}. In [28], it was shown that this algorithm is robust and globally linearly


(a) (b)

Fig. 4. Experiment on frames 10 and 11 of Fig. 2. (a) Interpolated frame by the smooth method at time $T/2$. (b) Interpolated frame by the TV_ε-segment method at time $T/2$.

convergent. The details of underlying scheme according to v read as follows (using notation $b = (v, w)$):

$$\begin{aligned}\partial_x\left(\frac{v_x^{n+1}}{|\nabla b^n|_\varepsilon}\right) &= \partial_x\left(|\nabla b^n|_\varepsilon^{-1}\right)v_x^{n+1} + \frac{v_{xx}^{n+1}}{|\nabla b^n|_\varepsilon}\\ &= -|\nabla b^n|_\varepsilon^{-3}\left(v_x^n v_{xx}^n + v_y^n v_{xy}^n + w_x^n w_{xx}^n\right.\\ &\qquad\left.+w_y^n w_{xy}^n\right)v_x^{n+1} + \frac{v_{xx}^{n+1}}{|\nabla b^n|_\varepsilon}\\ \partial_y\left(\frac{v_y^{n+1}}{|\nabla b^n|_\varepsilon}\right) &= \partial_y\left(|\nabla b^n|_\varepsilon^{-1}\right)v_y^{n+1} + \frac{v_{yy}^{n+1}}{|\nabla b^n|_\varepsilon}\\ &= -|\nabla b^n|_\varepsilon^{-3}\left(v_x^n v_{xy}^n + v_y^n v_{yy}^n + w_x^n w_{xy}^n\right.\\ &\qquad\left.+w_y^n w_{yy}^n\right)v_y^{n+1} + \frac{v_{yy}^{n+1}}{|\nabla b^n|_\varepsilon}.\end{aligned}$$

Altogether, the discretization of (12) w.r.t. v yields

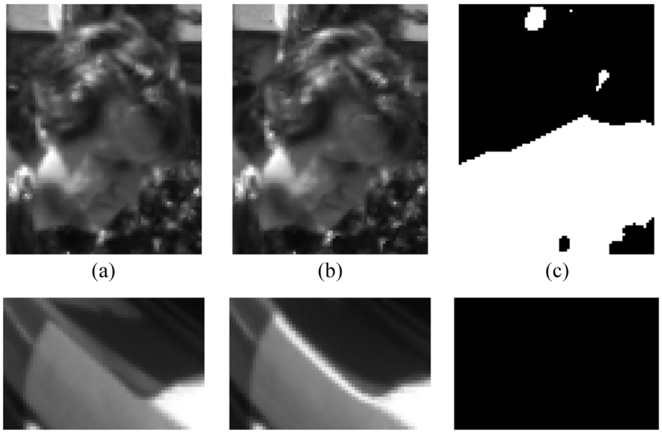
$$\begin{aligned}v^{n+1}&+\Delta t|\nabla b^n|_\varepsilon^{-3}\left(v_x^n v_{xx}^n+v_y^n v_{xy}^n+w_x^n w_{xx}^n+w_y^n w_{xy}^n\right)v_x^{n+1}\\ &-\Delta t\frac{v_{xx}^{n+1}}{|\nabla b^n|_\varepsilon}+\Delta t|\nabla b^n|_\varepsilon^{-3}\\ &\times\left(v_x^n v_{xy}^n+v_y^n v_{yy}^n+w_x^n w_{xy}^n+w_y^n w_{yy}^n\right)v_y^{n+1}-\Delta t\frac{v_{yy}^{n+1}}{|\nabla b^n|_\varepsilon}\\ &\quad=v^n+\frac{\Delta t}{\lambda}q_x^n-\frac{\Delta t}{\lambda}(\hat{p}\hat{u}_x+\tilde{p}\tilde{u}_x).\end{aligned}$$

Similarly, solving (11), we also use a time-marching scheme and apply the FP-iteration.

A. Segregation Loop

As explained in Section III, we apply a modified segregation loop to solve the equation system (8)–(11). We suppose $n = 1, \ldots, N_{\text{loop}}$ and N_{loop} is the iteration number. Given u_0, u_T, $b^{n-1}(t)$, ϕ^{n-1}, λ, μ, and ν, the iteration process at iteration n proceeds.

1) Compute $\hat{u}^{n-1}(t), \nabla\hat{u}^{n-1}(t)$, and $\hat{u}^{n-1}(T)$ using u_0 and $b^{n-1}(t)$.
2) Compute $\hat{p}^{n-1}(t)$ using $\hat{u}^{n-1}(T)$, u_T, and $H_s(\phi^{n-1})$.
3) Compute $\tilde{u}^{n-1}(t)$, $\nabla\tilde{u}^{n-1}(t)$, and $\tilde{u}^{n-1}(0)$ using u_T and $b^{n-1}(t)$.
4) Compute $\tilde{p}^{n-1}(t)$ using $\tilde{u}^{n-1}(0)$, u_0, and $H_s(\phi^{n-1})$.


(a) (b) (c)

(d) (e) (f)

Fig. 5. (a) Zoomed-in region of the head area in (a) of Fig. 4. (b) Zoomed-in region of the head area in (b) of Fig. 4. (c) Zoomed-in region of the contours of the head area generated by the TV_ε-segment method. (d) Zoomed-in region of the rear of the car in (a) of Fig. 4. (e) Zoomed-in region of the rear area of the car in (b) of Fig. 4. (f) Zoomed-in region of the contours of the rear of the car generated by the TV_ε-segment method.

5) Compute the solution of the TV-Stokes equation with the right-hand side $\hat{p}^{n-1}(t)\nabla\hat{u}^{n-1}(t) + \tilde{p}^{n-1}(t)\nabla\tilde{u}^{n-1}(t)$. Then, denote it by $\delta b^{n-1}(t)$.
6) Compute solution ϕ^n of (11) using $\hat{u}(T), u_T, \tilde{u}(0), u_0$, and ϕ^{n-1} as the initial value of the time-marching scheme.
7) Update $b^n(t) = b^{n-1}(t) + \delta b^{n-1}(t)$.

Similar to the segregation loop in Section III, this segregation loop does not solve the original problem but actually approximates a solution of the necessary conditions of another minimizing problem, namely, minimize

$$\frac{1}{2}\left\|\hat{u}(T) - u_T\right\|^2_{L^2(\omega)} + \frac{1}{2}\left\|\tilde{u}(0) - u_0\right\|^2_{L^2(\Omega\setminus\overline{\omega})}$$

subject to

$$\begin{cases}\hat{u}_t + b\cdot\nabla\hat{u} = 0 \text{ in }]0,T]\times\Omega \text{ with } \hat{u}(0) = u_0 \text{ in } \Omega\\ \tilde{u}_t + b\cdot\nabla\tilde{u} = 0 \text{ in } [0,T[\times\Omega \text{ with } \tilde{u}(T) = u_T \text{ in } \Omega\\ \mathrm{div} b = 0 \text{ in } [0,T]\times\Omega \text{ with } b = 0 \text{ on } \partial\Omega.\end{cases}$$

Again, we may see the segregation loop as a kind of a Landweber method for minimizing $(1/2)\|\hat{u}(T) - u_T\|^2_{L^2(\omega)} +$

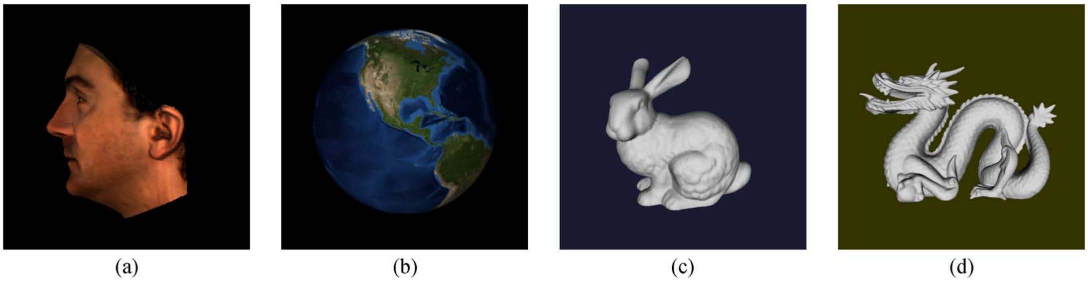

(a) (b) (c) (d)

Fig. 6. Data sets of Stich. (a) Face. (b) Earth. (c) Bunny. (d) Dragon.

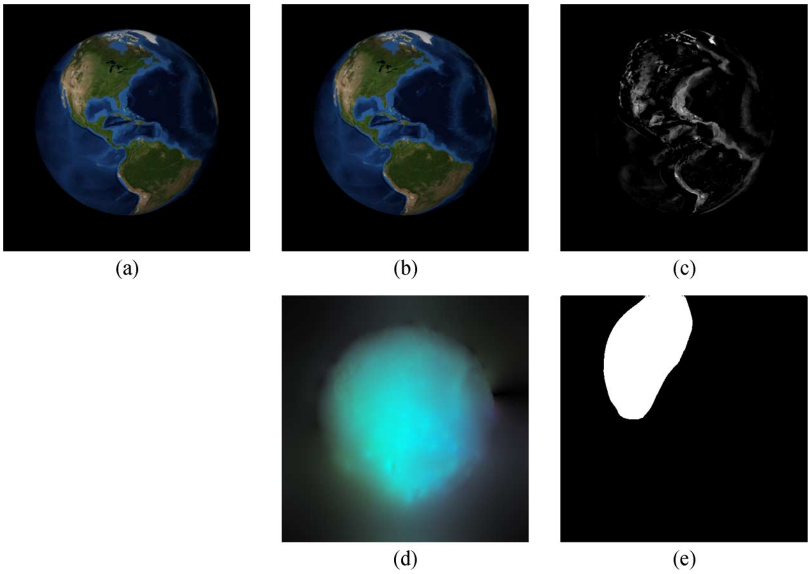

(a) (b) (c) (d) (e)

Fig. 7. (a) Frame 9 of Earth. (b) Frame 12 of Earth. (c) Absolute difference of (a) and (b). (d) Optical flow calculated by the TV$_\varepsilon$-segment method. (e) Active contours of segmentation calculated by the TV$_\varepsilon$-segment method.

$(1/2)\|\tilde{u}(0) - u_0\|^2_{L^2(\Omega\backslash\varpi)}$, which is inspired by a special Tikhonov functional.

B. Implementation

In optical-flow estimation, it is common to use the hierarchical processing (cf. [29] and [30]) to handle large displacements, and we apply this technique to get a start value b^0 for the optimality system. We execute a procedure.

1) Downsample the images into level l.
2) Carry out the segregation loop in level l, and get b^l.
3) Upsample the optical flow into level $l-1$, and get b^{l-1}.

The estimated optical flow b^{l-1} is a start value of the hierarchical method in level $l-1$, and we repeat it until level 0. In the coarsest level, we assume the start value to be zero.

The essential parameters of the quality of image sequence interpolation are the regularization parameters λ, μ, and ν. Parameter λ strongly depends on the intensities of the optical flow (velocities). For larger velocities, we have to penalize it with larger λ. In the praxis, if the velocities are smaller than 25 pixels between two image, then we can set $\lambda \in [10^{4.4}, 10^5]$. Parameters μ and ν, which involves the segmentation, were set both equal to 1. Regarding the time step in the time-marching scheme of the parabolic equations for optical flow and segmentation, we set $\Delta t = 10^3$, since the implicit method is stable with the arbitrary time step. Then, in few steps, we can achieve the convergence. The smoothing parameter ε was set equal to 1 and s equal to 10^{-2}.

All data sets that we are considering are 8-bit RGB color images. Dealing with the RGB color images, we convert them into 8-bit grayscale first, calculate the optical flow and active contours, and, at the end, warp every color channel with that flow field.

C. Experiments and Evaluation

To visualize the flow field both in angles and intensities, we utilize the color coding map in Fig. 1 (cf. [17]). The direction of the flow is coded by hue, and the intensity is coded by saturation, i.e., the brighter the color, the larger the velocity.

First, we compare the new introduced method, denoted by the TV$_\varepsilon$-segment method, with the smooth method introduced in [18] (see Fig. 2). To illustrate the ability to handle disclusions and occlusions, we apply them on the data set MiniCooper. First, Fig. 3 shows that the TV$_\varepsilon$-segment method preserves the flow edges better than the smooth method. Consequently, the interpolation by the TV$_\varepsilon$ segment keeps the boundary of objects (shapes) better than the smooth method. Additionally, the associated active contours for segmentation are shown in Fig. 3. In Fig. 4, we present the interpolated image of the data set in Fig. 2 at time $T/2$ by the smooth method and the TV$_\varepsilon$-segment method. The associated zoomed-in subimages in Fig. 5 show

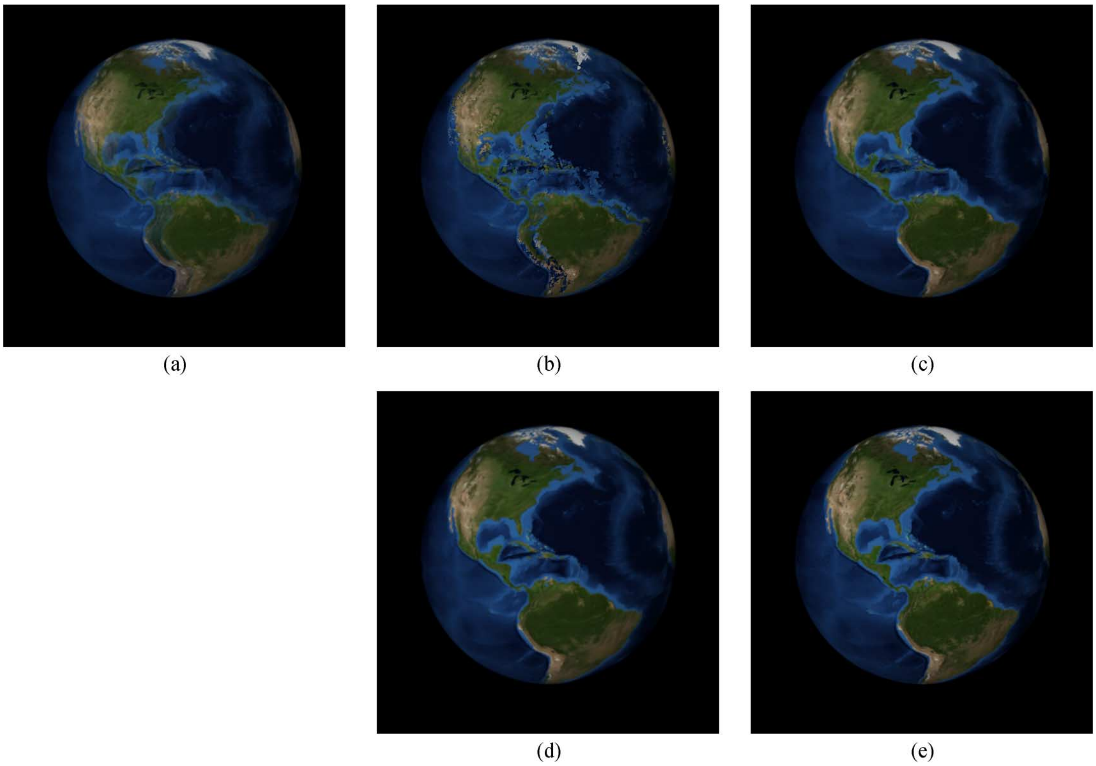

(a) (b) (c) (d) (e)

Fig. 8. Frame 11 calculated by the (a) blend, (b) optical-flow, (c) full, (d) multiscale, and (e) TV_ε-segment methods.

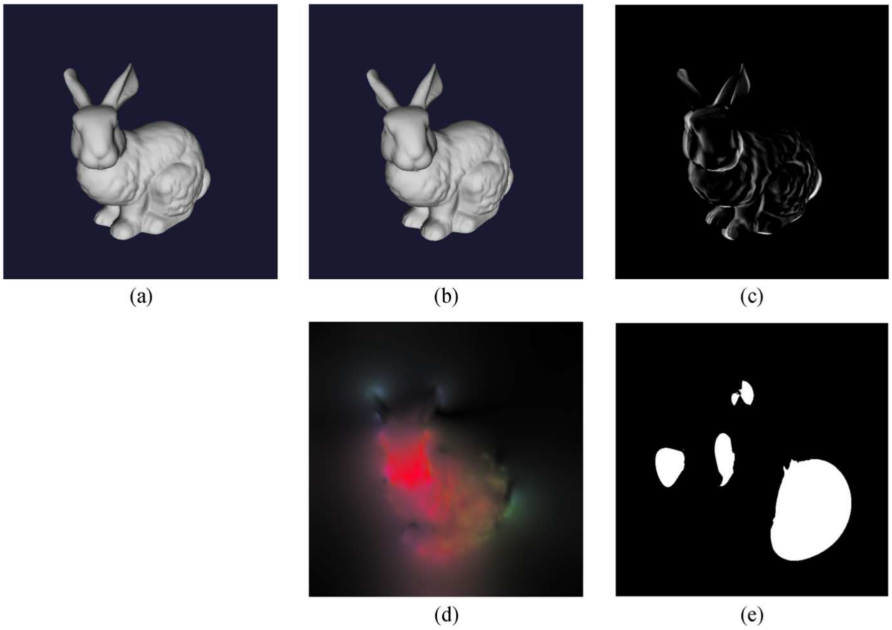

(a) (b) (c) (d) (e)

Fig. 9. (a) Frame 15 of Bunny. (b) Frame 18 of Bunny. (c) Absolute difference of (a) and (b). (d) Optical flow calculated by the TV_ε-segment method. (e) Active contours of segmentation calculated by the TV_ε-segment method.

that the TV_ε-segment method interpolated the disclosed regions (the upper part of head and the rear of the car) better than the smooth method. Comparing the zoomed-in subcontours, we can easily establish that the segmentation process has successfully identified the "covered" regions (the bottom part of the head) and the "disclosed" regions (the top part of the head and the rear of the car).

To evaluate our image sequence interpolation method, we performed an experiment based on human visual perception. We choose four data sets, i.e., Face, Earth, Bunny, and Dragon, from ftp://graphics.tu-bs.de/pub/public/people/lipski/stimuli/ (see Fig. 6). The data set Face is composed of 60 images, and the other data sets are composed of around 120 images each. We compare our methods with other seven existing methods from [31] and [15] for which interpolation results have been available, and we designed the experiment in the following way: Among every four images, two middle images are taken out, and using the first and last ones, the missing images will be

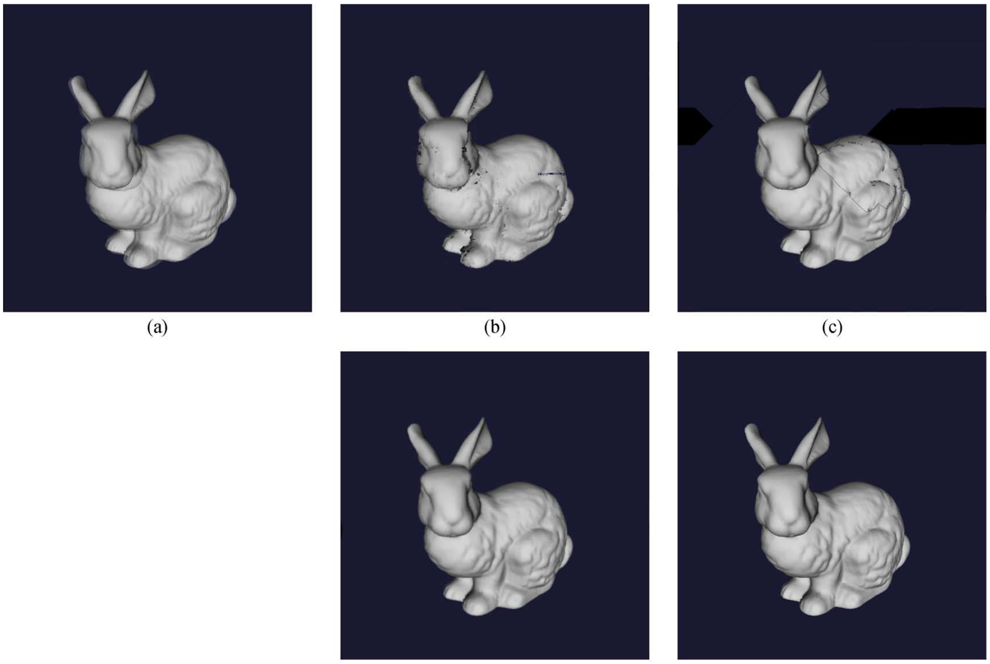


Fig. 10. Frame 16 calculated by the (a) blend, (b) optical-flow, (c) full, (d) multiscale, and (e) TV$_\varepsilon$-segment methods.

TABLE I
NORMALIZED SCORE OF EVERY ALGORITHM APPLIED ON EVERY DATA SET AND AVERAGE OF ALL SCORES OF EVERY ALGORITHM

	Face	Earth	Bunny	Dragon	**average**
original	0.90	0.86	0.88	0.96	**0.90**
blend	0.24	0.14	0.32	0.26	**0.24**
opticalflow	0.01	0.02	0.02	0.02	**0.02**
nofeathering	0.49	0.50	0.35	0.50	**0.46**
nooptim	0.26	0.32	0.39	0.21	**0.29**
full	0.50	0.53	0.42	0.51	**0.49**
multiscale	0.82	0.82	0.84	0.82	**0.82**
TV$_\varepsilon$-segment	0.77	0.82	0.78	0.73	**0.78**

TABLE II
INTERPOLATION ERROR OF EVERY ALGORITHM APPLIED ON EVERY DATA SET AND AVERAGE OF ALL INTERPOLATION ERRORS OF EVERY ALGORITHM

	Face	Earth	Bunny	Dragon	**average**
blend	3.73	4.18	2.41	3.49	**3.45**
opticalflow	3.11	4.25	2.36	3.38	**3.28**
nofeathering	2.29	2.15	1.95	2.58	**2.24**
nooptim	1.72	1.52	1.40	2.02	**1.67**
full	1.72	1.52	1.40	2.02	**1.67**
multiscale	1.31	0.75	1.16	1.97	**1.31**
TV$_\varepsilon$-segment	2.08	1.91	1.65	2.40	**1.99**

interpolated (i.e., we performed a subsampling with respect to time by the factor of 3). Now, each participant in our experiment is shown two interpolation results side by side and is asked which one looks better. For each of the scenes, we compare all seven interpolation methods and the ground truth against each other (only comparing methods A and B, excluding self-comparison and the reverse comparison method B to method A). This yields a total of $4 \cdot (8 \cdot 7/2) = 112$ trials per participant. We denote by N the number of participants, and in each trial, the perceptually better image gets one point. After all trials, we divide the score of every algorithm of each data set by $7N$ and all data sets by $35N$ to get the normalized score for every data set and all data sets. The range of the score is in [0, 1], and a higher score stands for better human visual perception.

In total, 17 participants took part in this experiment, and in Table I, we can observe that the so-called "multiscale" and TV$_\varepsilon$-segment methods visual perceptually perform better than the other methods.

In Table II, we also evaluate the interpolation error (IE) measure [17], i.e., the root-mean-square difference between the ground-truth image u_{GT} and the interpolated image u, i.e.,

$$\mathrm{IE} = \left(\frac{1}{MN} \sum_{i=1}^{N} \sum_{j=1}^{M} \left(u(x_i, y_j) - u_{\mathrm{GT}}(x_i, y_j)\right)^2 \right)^{\frac{1}{2}}$$

where $M \times N$ is the image size. Observe that the TV$_\varepsilon$-segment method does not outperform the "nooptim" and "full" methods with respect to the IE, which is in contrast with the results from the visual perception. The IE measure does not reveal human visual perception due to two reasons. First, the human eyes are sensitive for the shocks that are the common drawbacks of the optical-flow, no-feathering, and full methods (see, e.g., Fig. 10), and also sensitive for the ghosting effects, which are characterized by the blend method [cf. Figs. 8(a) and 10(a)]. Second, there are indeed many ways to interpolate, but this does not mean that all results are possible ground-truth data. In addition to the experimental results, we present examples of interpolated images in Figs. 7–10. Again, our method correctly identifies the regions (the white in the associated contours images) in which occlusions occur (in data set Earth/Bunny on the left-/right-hand side of the objects where parts of the objects "disappear," respectively).

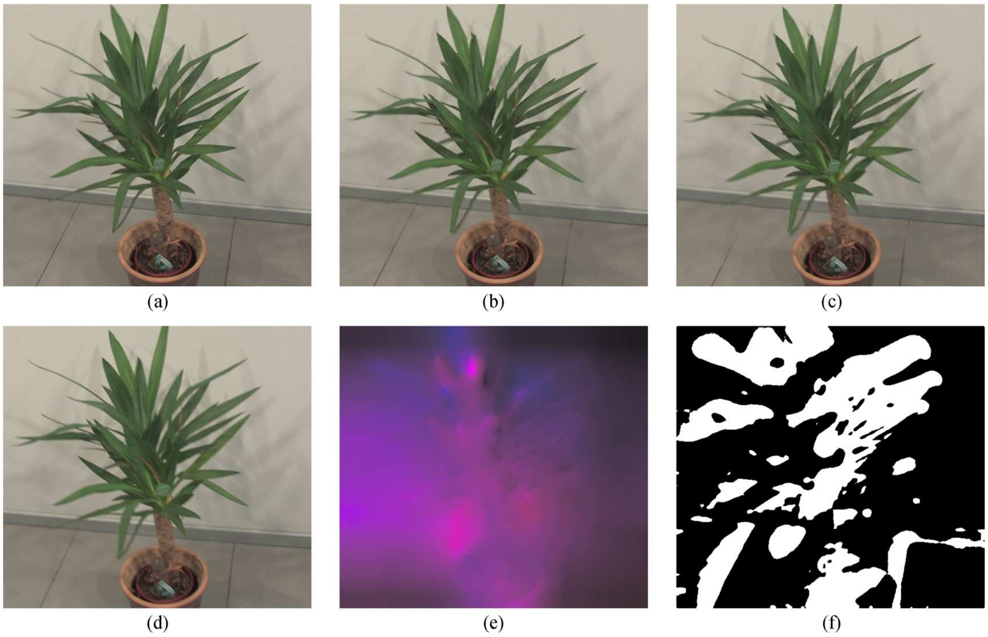


Fig. 11. (a) Frame 54. (b) Frame 55 calculated by the TV_ε-segment method. (c) Frame 56 calculated by the TV_ε-segment method. (d) Frame 57. (e) Optical flow calculated by the TV_ε-segment method. (f) Active contours calculated by the TV_ε-segment method.

In the last example, we consider the real video sequence "Plant"[2] (see Fig. 11) consisting of 124 images. Again, we performed a temporal downsampling by a factor of 3 and interpolated the missing frames. Comparing our interpolated images and the given images, we can conclude that our interpolation method also works well with real video sequences.

All these routines in the segregation loop were implemented in MATLAB on a Windows 7 with Intel Core i7 Q720 central processing unit. The computational time is strongly related to the image size and the iteration number N_{loop}, e.g., using a 641 × 480 image in the finest resolution level of 0 applying five iterations, the elapsed time is 780 s, and in one level coarser resolution with the same iteration number, the elapsed time is always approximately four times less. In all experiments, we used the iteration numbers [20, 20, 20, 20, 5] from the coarsest level to the finest level in the hierarchical processing.

VI. CONCLUSION AND OUTLOOK

The approach to image sequence interpolation based on the optical flow in the framework of optimal control avoids shocks and ghosting effects. The improvement by TV_ε flow and segmentation has shown that it is able to produce more natural interpolation for human visual perception. However, as already explained in [18], this method has a limited application if the illumination of the object varies in time, since we have only considered the transport equation with the right-hand side of 0. This means that the external illumination variation, e.g., light or flash, does not come to consideration.

In the further work, it might be interesting to introduce another control f in the right-hand side of the transport equation to simulate the external illumination variation. However, since the movement of an object can be also generated as a drastic change "illumination," it might be difficult to obtain meaningful results for both b and f.

ACKNOWLEDGMENT

The authors would like to thank all the participants who joined the visual perceptual experiment.

REFERENCES

[1] J. Watkinson, *The MPEG Handbook*, 2nd ed. Waltham, MA: Focal Press, 2004.

[2] D. Rueckert, L. I. Sonoda, C. Hayes, D. L. G. Hill, M. O. Leach, and D. J. Hawkes, "Nonrigid registration using free-form deformations: Application to breast MR images," *IEEE Trans. Med. Imag.*, vol. 18, no. 8, pp. 712–721, Aug. 1999.

[3] J. M. Fitzpatrick and M. Sonka, *Handbook of Medical Imaging, Volume 2 Medical Image Processing and Analysis*, 1st ed. Bellingham, WA: SPIE, 2000.

[4] B. K. P. Horn and B. G. Schunck, "Determining optical flow," *Artif. Intell.*, vol. 17, pp. 185–203, Aug. 1981.

[5] G. Aubert and P. Kornprobst, "A mathematical study of the relaxed optical flow problem in the space $BV(\Omega)^*$," *SIAM J. Math. Anal.*, vol. 30, no. 6, pp. 1282–1308, Oct. 1999.

[6] A. Bruhn, J. Weickert, and C. Schnörr, "Lucas/Kanade meets Horn/Schunck: Combining local and global optic flow methods," *Int. J. Comput. Vis.*, vol. 61, no. 3, pp. 211–231, Feb./Mar. 2005.

[7] H. H. Nagel, "Constraints for the estimation of displacement vector fields from image sequences," in *Proc. Int. Joint Conf. Artif. Intell.*, 1983, pp. 945–951.

[8] W. Enkelmann, "Investigation of multigrid algorithms for the estimation of optical flow fields in image sequences," *Comput. Vis., Graph., Image Process.*, vol. 43, no. 2, pp. 150–177, Aug. 1988.

[9] A. Wedel, T. Pock, C. Zach, H. Bischof, and D. Cremers, "An improved algorithm for TV-L1 optical flow," in *Statistical and Geometrical Approaches to Visual Motion Analysis*. Berlin, Germany: Springer-Verlag, 2009.

[10] S. L. Keeling and W. Ring, "Medical image registration and interpolation by optical flow with maximal rigidity," *J. Math. Imag. Vis.*, vol. 23, no. 1, pp. 47–65, 2005.

[11] E. Mémin and P. Pérez, "Dense estimation and object-based segmentation of the optical flow with robust techniques," *IEEE Trans. Image Process.*, vol. 7, no. 5, pp. 703–719, May 1998.

[12] W. Hinterberger and O. Scherzer, "Models for image interpolation based on the optical flow," *Computing*, vol. 66, no. 3, pp. 231–247, 2001.

[13] A. Borzıacute;, K. Ito, and K. Kunisch, "Optimal control formulation for determining optical flow," *SIAM J. Sci. Comput.*, vol. 24, no. 3, pp. 818–847, 2002.

---

[2] Also available at ftp://graphics.tu-bs.de/pub/public/people/lipski/stimuli/.

[14] D. Mahajan, F. Huang, W. Matusik, R. Ramamoorthi, and P. Belhumeur, "Moving gradients: A path-based method for plausible image interpolation," *ACM Trans. Graph.*, vol. 28, no. 3, p. 42, Aug. 2009.

[15] T. Stich, C. Linz, C. Wallraven, D. Cunningham, and M. Magnor, "Perception-motivated interpolation of image sequences," in *Proc. ACM Symp. APGV*, 2008, pp. 97–106.

[16] T. Stich, C. Linz, C. Wallraven, D. Cunningham, and M. Magnor, "Time and view interpolation in image space," *Comput. Graph. Forum (Proc. Pacific Graph.)*, vol. 27, no. 7, pp. 1781–1787, 2008.

[17] S. Baker, D. Scharstein, J. P. Lewis, S. Roth, M. J. Black, and R. Szeliski, "A database and evaluation methodology for optical flow," *Int. J. Comput. Vis.*, vol. 92, no. 1, pp. 1–31, Mar. 2011.

[18] K. Chen and D. Lorenz, "Image sequence interpolation using optimal control," *J. Math. Imag. Vis.*, vol. 41, no. 3, pp. 222–238, Nov. 2011.

[19] T. F. Chan and L. A. Vese, "Active contours without edges," *IEEE Trans. Image Process.*, vol. 10, no. 2, pp. 266–277, Feb. 2001.

[20] V. Caselles, F. Catté, T. Coll, and F. Dibos, "A geometric model for active contours in image processing," *Numer. Math.*, vol. 66, no. 1, pp. 1–31, Dec. 1993.

[21] M. Kass, A. Witkin, and D. Terzopoulos, "Snakes: Active contour models," *Int. J. Comput. Vis.*, vol. 1, no. 4, pp. 321–331, Jan. 1988.

[22] D. Mumford and J. Shah, "Optimal approximations by piecewise smooth functions and associated variational problems," *Commun. Pure Appl. Math.*, vol. 42, no. 5, pp. 577–685, Jul. 1989.

[23] C. Hirsch, *Numerical Computation of Internal & External Flows*. Amsterdam, The Netherlands: Elsevier, 2007.

[24] K. F. Riley, M. P. Hobson, and S. J. Bence, *Mathematical Methods for Physics and Engineering*. Cambridge, U.K.: Cambridge Univ. Press, 2006.

[25] X. C. Tai, S. Osher, and I. Holm, "Image inpainting using a TV-Stokes equation," in *Image Processing based on partial differential equations*. Heidelberg, Germany: Springer-Verlag, 2006.

[26] W. H. Press, S. A. Teukolsky, W. T. Vetterling, and B. P. Flannery, *Numerical Recipes: The Art of Scientific Computing*, 3rd ed. Cambridge, U.K.: Cambridge Univ. Press, 2007.

[27] C. R. Vogel and M. E. Oman, "Iterative methods for total variation denoising," *SIAM J. Sci. Comput.*, vol. 17, no. 1, pp. 227–238, Jan. 1996.

[28] T. F. Chan and P. Mulet, "On the convergence of the lagged diffusivity fixed point method in total variation image restoration," *SIAM J. Numer. Anal.*, vol. 36, no. 2, pp. 354–367, 1999.

[29] J. L. Barron and M. Khurana, "Determining optical flow for large motions using parametric models in a hierarchical framework," in *Proc. Vis. Interface*, 1994, pp. 47–56.

[30] T. Brox, A. Bruhn, N. Papenberg, and J. Weickert, "High accuracy optical flow estimation based on a theory for warping," in *Proc. Comput. Vis. ECCV*, 2004, pp. 25–36.

[31] T. Stich, C. Linz, C. Wallraven, D. Cunningham, and M. Magnor, "Perception-motivated interpolation of image sequences," *ACM Trans. Appl. Percept.*, vol. 8, no. 2, p. 11, Jan. 2011.

Kanglin Chen received the Diplom degree and the Ph.D. degree in mathematics from the University of Bremen, Bremen, Germany, in 2008 and 2011, respectively.

Since 2008, he has been attending the Ph.D. program, i.e., Scientific Computing in Engineering, with the University of Bremen.

Dirk A. Lorenz received the Diplom degree and the Ph.D. degree in mathematics from the University of Bremen, Bremen, Germany, in 2002 and 2005, respectively.

Since 2009, he is a "Junior Professor" (equivalent to Assistant Professor) for applied analysis with Technische Universität Braunschweig, Braunschweig, Germany. His research interests include mathematical image processing, inverse problems, and optimization.