

Analysis of Numerical Methods for Level Set Based Image Segmentation

Björn Scheuermann and Bodo Rosenhahn

Institut für Informationsverarbeitung
Leibniz Universität Hannover
{scheuerm,rosenhahn}@tnt.uni-hannover.de

Abstract. In this paper we analyze numerical optimization procedures in the context of level set based image segmentation. The Chan-Vese functional for image segmentation is a general and popular variational model. Given the corresponding Euler-Lagrange equation to the Chan-Vese functional the region based segmentation is usually done by solving a differential equation as an initial value problem. While most works use the standard explicit Euler method, we analyze and compare this method with two higher order methods (second and third order Runge-Kutta methods). The segmentation accuracy and the dependence of these methods on the involved parameters are analyzed by numerous experiments on synthetic images as well as on real images. Furthermore, the performance of the approaches is evaluated in a segmentation benchmark containing 1023 images. It turns out, that our proposed higher order methods perform more robustly, more accurately and faster compared to the commonly used Euler method.

1 Introduction

One popular problem in the field of computer vision is image segmentation. The problem has been formalized by Mumford and Shah as the minimization of a functional [1]. With the use of level set representations of active contours [2] one obtains a very efficient way to find the minimizers of such a functional. As shown by many seminal papers and textbooks on segmentation using these variational frameworks [2,3,4] there has been a lot of progress but it still faces several difficulties. The reason for these difficulties is in most cases a violation of model assumptions. For example the model usually assumes to have homogeneous [2] or smooth [1] object regions. Due to noise, occlusion, texture and shading this model is often not appropriate to delineate object regions. A successful remedy is the statistical modeling of regions [3] and the supplement of additional information such as texture [5] and motion [6], which increases the number of scenes where image segmentation can succeed.

To find a minimizer it is a common technique to numerically solve the corresponding Euler-Lagrange equation using the explicit Euler method for initial value problems [2]. The main contribution of this paper is the analysis of numerical methods like the explicit Euler method (EU) and higher order Runge-Kutta

methods (RK-2 and RK-3). We will compare the segmentation accuracy and further analyze the dependence for involved parameters like the timestep of the numerical methods or weighting parameters. The advantages of the higher order methods are demonstrated by several experiments on synthetic and real images.

Paper Organization. In Section 2 we continue with a short review of the variational approach for image segmentation, which is the basis for our segmentation framework. Section 3 introduces the various numerical methods and also describes how to find a minimizer for the functional described in Section 2. Experiments in Section 4 will demonstrate the advantages of the chosen higher order numerical methods over the standard method and other segmentation methods. The paper will finish with a short conclusion.

2 Image Segmentation Using a Variational Framework

The variational approach for image segmentation used in our framework is based on the works of [2,7,8,9]. Using the level set formulation for the general problem of image segmentation has several advantages. To allow a convenient and sound interaction between constraints that are imposed on the contour itself and constraints that act on the two regions separated by the contour, the 1-D curve is embedded into a 2-D, image-like structure. Another important advantage of the level set representation is the natural given possibility to handel topological changes of the 1-D curve. This is especially important if the object is particular occluded by another object or if the object consist of multiple parts.

In the case of a two-phase segmentation, the level set function $\varphi : \Omega \rightarrow \mathbb{R}$ splits the image domain Ω into the two regions $\Omega_1, \Omega_2 \subseteq \Omega$ with

$$\varphi(x) = \begin{cases} \geq 0, & \text{if } x \in \Omega_1 \\ < 0, & \text{if } x \in \Omega_2 \end{cases} . \tag{1}$$

The boundary between the object that is sought to be extracted and the background is represented by the zero-level line of the function φ . Like most of the works on level set segmentation do, we focus on this special segmentation case with two regions. The interested reader can find an extension to the presented method on multiple regions in [10,11]. Another successful remedy to extend the number of situations in which image segmentation can succeed is the use of additional constraints like the restriction to a certain object shape [4,12].

The following three constraints are imposed as an optimality criterion for contour extraction:

- (i) the data within each region should be similar
- (ii) the data between the object and the background should be dissimilar
- (iii) the contour dividing the region should be minimal

As shown in [2] these model assumptions can be expressed by the so called *Chan-Vese energy functional* that is:

$$E(\varphi) = - \int_{\Omega} \left(H(\varphi) \log p_1 + (1 - H(\varphi)) \log p_2 \right) d\Omega + \nu \int_{\Omega} |\nabla H(\varphi)| d\Omega \tag{2}$$

where $\nu \geq 0$ is a weighting parameter between the three given constraints, p_i are probability densities and $H(s)$ is a regularized Heaviside function with

$$(i) \lim_{s \rightarrow -\infty} H(s) = 0, \quad (ii) \lim_{s \rightarrow \infty} H(s) = 1, \quad (iii) H(0) = 0.5. \quad (3)$$

The regularized Heaviside function is needed to build the Euler-Lagrange equation and to make it possible to indicate at each iteration step to which region a pixel belongs. Minimizing the first term maximizes the total a-posteriori probability given the the two probability densities p_1 and p_2 of Ω_1 and Ω_2 , i.e., pixels are assigned to the most probable region according to the Bayes rule. The second term minimizes the length of the contour and act as a smoothing term.

Minimization of the Chan-Vese energy functional (2) can be easily performed by solving the corresponding Euler-Lagrange equation to φ

$$\frac{\partial \varphi}{\partial t} = \delta(\varphi) \left(\log \frac{p_1}{p_2} + \nu \operatorname{div} \left(\frac{\nabla \varphi}{|\nabla \varphi|} \right) \right), \quad (4)$$

where $\delta(s)$ is the derivative of $H(s)$ with respect to its argument. Starting with some initial contour φ^0 and given the probability densities p_1 and p_2 one has to solve the following initial value problem

$$\begin{cases} \varphi(x, 0) = \varphi^0 \text{ for } x \in \Omega \\ \frac{\partial \varphi}{\partial t} = \delta(\varphi) \left(\log \frac{p_1}{p_2} + \nu \operatorname{div} \left(\frac{\nabla \varphi}{|\nabla \varphi|} \right) \right) \end{cases}. \quad (5)$$

The way the two probability densities p_1 and p_2 are modeled is a very important factor for the quality of the segmentation process. In this paper we restrict to the very simple full Gaussian density using gray values [3]. This restriction is made because we only want to analyze several numerical methods and therefore it is not necessary to use different statistical models. Other possibilities for image cues to use for the density model are color and texture [5,13] or motion [6]. There are also various other possibilities to model the probability densities given these image cues, e.g., a Gaussian density with fixed standard derivation [2], a generalized Laplacian [14] or nonparametric Parzen estimates [5].

Let now μ_1 and μ_2 be the mean gray value in Ω_1 or rather Ω_2 and σ_1 and σ_2 the standard deviation of the two regions Ω_1 and Ω_2 . Then the probability of $u(x) \in \Omega$ to be in Ω_i is:

$$p_i(u(x)) = \frac{1}{\sqrt{2\pi}\sigma_i} e^{-\frac{(u(x)-\mu_i)^2}{2\sigma_i^2}} \quad \text{for } i \in \{1, 2\}, \quad (6)$$

where the probability densities p_1 and p_2 have to be updated after each iteration step. For our full Gaussian density model this comes down to updating

$$\begin{aligned} \mu_1 &= \frac{\int_{\Omega} u(x)H(\varphi) d\Omega}{\int_{\Omega} H(\varphi) d\Omega}; & \sigma_1 &= \left(\frac{\int_{\Omega} (u(x) - \mu_1)^2 H(\varphi) d\Omega}{\int_{\Omega} H(\varphi) d\Omega} \right)^{1/2} \\ \mu_2 &= \frac{\int_{\Omega} u(x)(1 - H(\varphi)) d\Omega}{\int_{\Omega} (1 - H(\varphi)) d\Omega}; & \sigma_2 &= \left(\frac{\int_{\Omega} (u(x) - \mu_2)^2 (1 - H(\varphi)) d\Omega}{\int_{\Omega} (1 - H(\varphi)) d\Omega} \right)^{1/2}, \end{aligned} \quad (7)$$

using the Heaviside function to indicate the two separated regions.

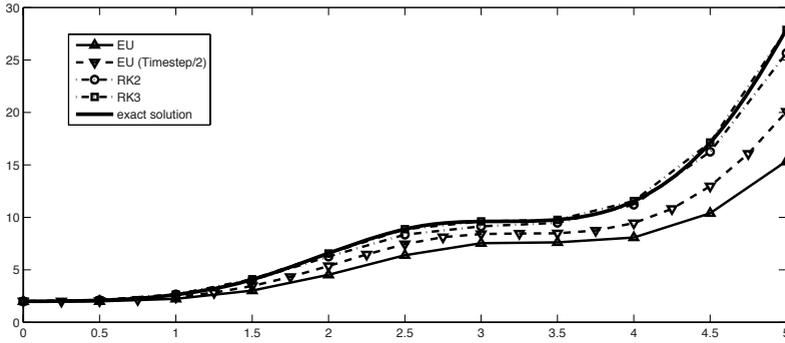


Fig. 1. Comparison of the different numerical methods for the initial value problem $y'(t) = \sin(t)^2 \cdot y(t)$ with $y(0) = 2$. The Euler method apparently fail to compute an accurate solution whereas both Runge-Kutta methods are almost exact.

3 Numerical Methods

3.1 Euler Method

Several methods exist to numerically compute the solution of an initial value problem of the type (5). The easiest method used by most previous works is the simple Euler method (EU) [2], which is an explicit first order numerical procedure for solving initial value problems. The idea of the Euler method is to assume that the state of change is constant for an interval Δt . For a given initial value problem $y'(t) = f(t, y(t))$, $y(t_0) = y_0$ the Euler method is defined by

$$y_{n+1} = y_n + \Delta t f(t_n, y_n) \text{ for } n \geq 0, \tag{8}$$

where Δt is the timestep and $t_{n+1} = t_n + \Delta t$. For the initial value problem given by Equation (5) this method leads to

$$\begin{cases} \varphi^0 & = \text{initial contour,} \\ \varphi^{n+1} & = \varphi^n + \Delta t L(\varphi^n) \text{ for } n \in \mathbb{N}. \end{cases} \tag{9}$$

Where $L(\varphi^n)$ is defined as the following operator:

$$L(\varphi^n) := \left[\delta(\varphi^n) \left(\log \frac{p_1}{p_2} + \nu \operatorname{div} \left(\frac{\nabla \varphi^n}{|\nabla \varphi^n|} \right) \right) \right] \text{ for } n \geq 0, \tag{10}$$

and Δt is the timestep. To increase the accuracy of the solution one has two possibilities. The first is to reduce Δt and the other is to choose a method with a higher order of convergence.

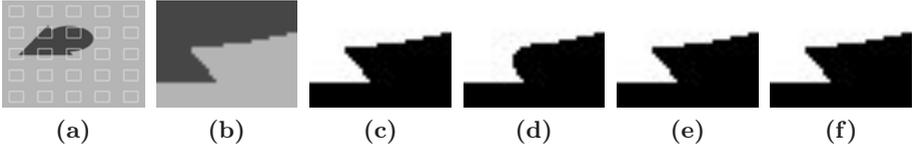


Fig. 2. (a) Synthetic image and level set initialization; (b) detail of the synthetic image; (c) detail of the final segmentation using EU and $\nu = 4$; (d) detail of the final segmentation using EU and $\nu = 10$; (e) detail of the final segmentation using RK-2 and $\nu = 4$; (f) detail of the final segmentation using RK-2 and $\nu = 10$

3.2 Runge-Kutta Methods

The Runge-Kutta schemes are well known methods with a higher order of convergence compared to EU. The Runge-Kutta methods used in this paper are explicit iterative methods for the approximation of solutions to initial value problems. Consider an initial value problem (8), an explicit second order Runge-Kutta method (RK-2), also called Heun’s method or modified Euler method, is given by:

$$\begin{aligned} \tilde{y}_{n+1} &= y_n + \Delta t f(t_n, y_n), \\ y_{n+1} &= y_n + \frac{\Delta t}{2} (f(t_n, y_n) + f(t_{n+1}, \tilde{y}_{n+1})) \text{ for } n \geq 0. \end{aligned} \tag{11}$$

For the initial value problem (5) RK-2 leads to

$$\begin{cases} \varphi^0 &= \text{initial contour}, \\ \tilde{\varphi}^{n+1} &= \varphi^n + \Delta t \cdot L(\varphi^n) \text{ for } n \in \mathbb{N}, \\ \varphi^{n+1} &= \varphi^n + \frac{\Delta t}{2} \cdot (L(\varphi^n) + L(\tilde{\varphi}^{n+1})) \text{ for } n \in \mathbb{N}. \end{cases} \tag{12}$$

A third order Runge-Kutta method (RK-3) for the initial value problem given by (5) can be defined analogue to Shu and Osher [15] by:

$$\begin{cases} \varphi^0 &= \text{initial contour}, \\ \tilde{\varphi}^{n+1} &= \varphi^n + \Delta t \cdot L(\varphi^n) \text{ for } n \in \mathbb{N}, \\ \tilde{\varphi}^{n+\frac{1}{2}} &= \varphi^n + \frac{\Delta t}{4} \cdot (L(\varphi^n) + L(\tilde{\varphi}^{n+1})) \text{ for } n \in \mathbb{N}, \\ \varphi^{n+1} &= \varphi^n + \frac{\Delta t}{6} \cdot (L(\varphi^n) + L(\tilde{\varphi}^{n+1}) + 2L(\tilde{\varphi}^{n+\frac{1}{2}})) \text{ for } n \in \mathbb{N}. \end{cases} \tag{13}$$

A simple 1-D example of these methods is shown by Figure 1. Obviously the accuracy of the solution increases with the choose of a smaller timestep and the choose of a higher order method.

Remark: In our algorithm the spatial discretization is done using finite differences.

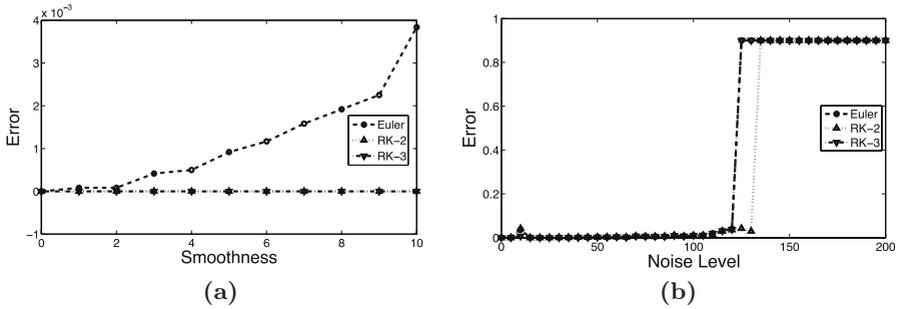


Fig. 3. (a) Segmentation error in dependence of the weighting parameter ν (the segmentation error of RK-2 and RK-3 is exactly the same); (b) segmentation error in dependence of the noise level (the segmentation error of EU and RK-3 is almost the same)

4 Experiments

In this Section we demonstrate the impact of the higher order methods RK-2 and RK-3 applied to level set based image segmentation. Experiments are performed on synthetic images, real images and on the segmentation-benchmark developed by Feng Ge and Song Wang [16].

4.1 Synthetic Images

For the analysis of the three presented numerical methods, we first use the synthetic image and the initialization of the level set function shown in Figure 2a. The dimension of this image was 400×300 pixels. We choose this image because the object consists of parts with high and small curvature. We define the region-based segmentation accuracy analogue to Ge et al. [16] by

$$P(R; G) = \frac{|R \cap G|}{|R \cup G|} = \frac{|R \cap G|}{|G| + |R| - |R \cap G|}, \quad (14)$$

where G is the ground-truth foreground object and the region R is the segment derived from the segmentation result using one of the numerical methods. The properties of this definition are discussed in [16]. Equation (14) leads to the following definition of the segmentation error

$$\varepsilon_{SE} := 1 - P(R; G). \quad (15)$$

Figure 2b shows a detail of the synthetic image, while Figures 2c and 2d demonstrate that the smoothness parameter ν has a large influence on the final segmentation if EU is used. Figures 2e and 2f apparently show that ν has little influence on using RK-2.

It can be seen from Figure 3a that for EU only choosing $\nu = 0$ leads to a segmentation error $\varepsilon_{SE} = 0$, but choosing $\nu = 20$ results in $\varepsilon_{SE} = 0,01$

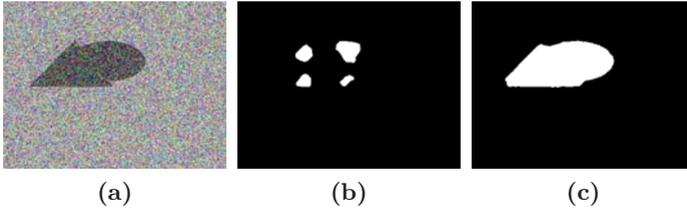


Fig. 4. (a) Synthetic image with noise; (b) final segmentation using EU; (c) final segmentation using RK-2

corresponding to 1200 wrong segmented pixels. Conversely, RK-2 and RK-3 lead to the segmentation error $\varepsilon_{SE} = 0$ for $\nu \in \{0, \dots, 20\}$. Obviously the higher order methods are more robust to choices of the smoothness parameter.

In Figure 3b we added Gaussian pixel noise to the synthetic image shown in Figure 2a. It can be seen from Figure 3b that the segmentation obtained by RK-2 is more robust to noise. Figure 4 shows the final segmentation results for a noisy synthetic image (4a) using EU (4b) and RK-2 (4c). Apparently the segmentation method using EU converges to a local minimum of the energy functional while RK-2 reaches the global minimum.

4.2 Real Images

To analyze the numerical methods on real images, we apply the segmentation methods on the image benchmark presented in [16]. To demonstrate the robustness of the level set based segmentation using RK-2 and RK-3 on the weighting parameter ν , we use the real image of the benchmark shown in Figure 5a. For RK-2 and RK-3 the segmentation error ε_{SM} is between 0.055 and 0.057 for $\nu \in \{0, \dots, 20\}$, which implies that the variance of the final segmentation is less than 0.5%. Using EU the error ε_{SM} is between 0.056 and 0.1 implying a variance bigger than 4% (see Figure 5d). Figures 5b and 5c show the final segmentation using EU and RK-2 with the weighting parameter $\nu = 10$. The dependence of all methods on the timestep Δt is shown in Figure 5e. Obviously, the final segmentation using RK-2 or RK-3 is almost identical for all $\Delta t \in \{1 \dots 30\}$, whereas for EU the segmentation error ε_{SE} increases with a bigger timestep Δt . We find that

Table 1. Comparison of the average performance of four image segmentation methods

Method	avg. perf.
Normalized-cut method	0.39
Euler method	0.50
2nd order RK method	0.52
3rd order RK method	0.52

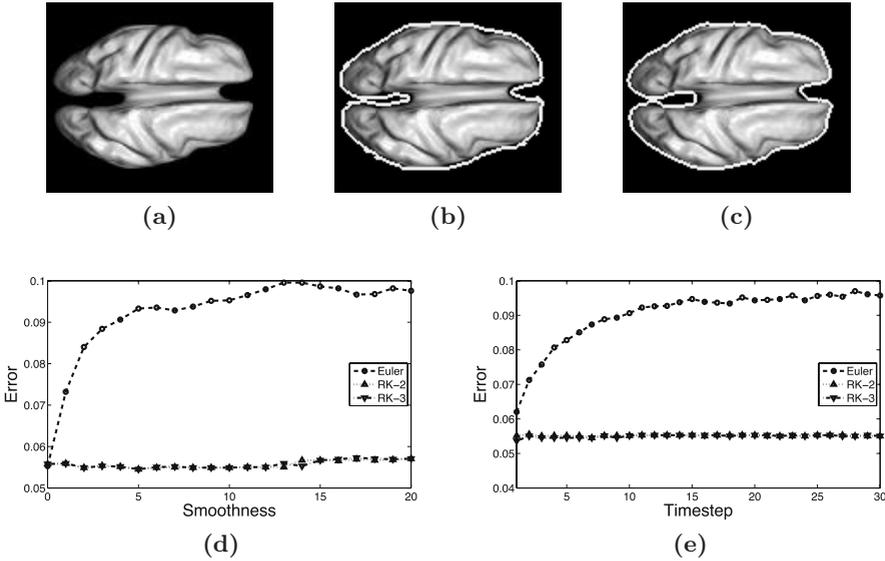


Fig. 5. (a) Real image from the segmentation benchmark presented in [16]; (b) final segmentation using EU and $\nu = 10$; (c) final segmentation using RK-2 and $\nu = 10$; (d) segmentation error in dependence of the weighting parameter ν (the segmentation error of RK-2 and RK-3 is almost identical); (e) segmentation error in dependence of the timestep Δt (the segmentation error of RK-2 and RK-3 is almost identical).

RK-2 converges more reliably than EU, even for large timesteps (cf. Figure 1). This is an important fact because a bigger timestep leads to a reduction of the total number of iterations and thereby to a reduction of the computational time needed to segment an object.

Figure 6 shows the total segmentation accuracy of the different methods. The curves indicate the performance distribution of the methods on all 1023 images of the segmentation benchmark. The horizontal axis denotes the proportion of images and the y -axis indicates the segmentation accuracy $p(x)$. A specific point $(x, p(x))$ on the curve indicates that $100 \cdot (1 - x)$ percent of images are segmented with an accuracy better than $p(x)$. The curve NC describes the performance of the Normalized-cut method (NC) implemented by Shi et al. [17], to compare our results to another segmentation strategy. We decided to compare our methods to the normalized-cut method, since it was the method with the best average performance on the segmentation benchmark [16].

Because the performance curves of EU, RK-2 and RK-3 are almost indistinguishable, Table 1 shows the average performance of our three segmentation methods in comparison to NC. Apparently the average performance of the level set based segmentation methods is saliently better than NC and RK-2 and RK-3 are slightly better than EU (the average performance increases by 4%).

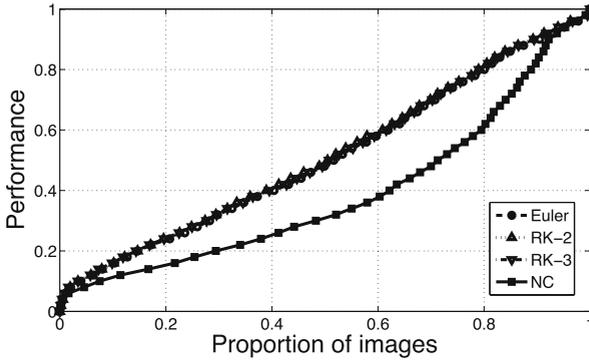


Fig. 6. Performance curve of the three numerical methods and the Normalized-cut method (NC) on the 1032 images of the segmentation benchmark

To compare the computational time of the methods we choose 100 Images of the segmentation benchmark where the segmentation accuracy was better than 90% for all numerical methods. Using the same timestep $\Delta t = 1$ it took 8.4 minutes to segment the images with the Euler method and 6.8 min using the described second order Runge-Kutta method. Choosing a bigger timestep for RK-2 and EU farther reduces the computational time. Table 2 shows the computational time needed to segment the 100 images for various methods and timesteps.

Table 2. Comparison of the computational time needed to segment 103 images from the segmentation benchmark

Method	comp. time	avg. perf.
Euler method, with $\Delta t = 1$	8.4 min	0,9440
Euler method, with $\Delta t = 2$	3.6 min	0,9415
Euler method, with $\Delta t = 4$	2.9 min	0,9413
2nd order RK method, with $\Delta t = 1$	6,6 min	0.9441
2nd order RK method, with $\Delta t = 2$	3,1 min	0.9415
2nd order RK method, with $\Delta t = 4$	2,5 min	0.9422

In Figure 7, we present more segmentation results on real images. We decided for these images to show, that RK-2 is able to find the global minimum in cases where EU converges to a local minimum (see Figures 7a - 7c). Figures 7d - 7g shows that, using the same timestep $\Delta t = 1$ and the same smoothing parameter $\nu = 4$, the segmentation accuracy increases using RK-2 instead of EU. Besides the total number of iterations is much smaller, which leads to a reduction of the computational time by the factor 2, even if the computational time for one iteration is bigger. These results clearly show that RK-2 more reliably achieves accurate segmentations.

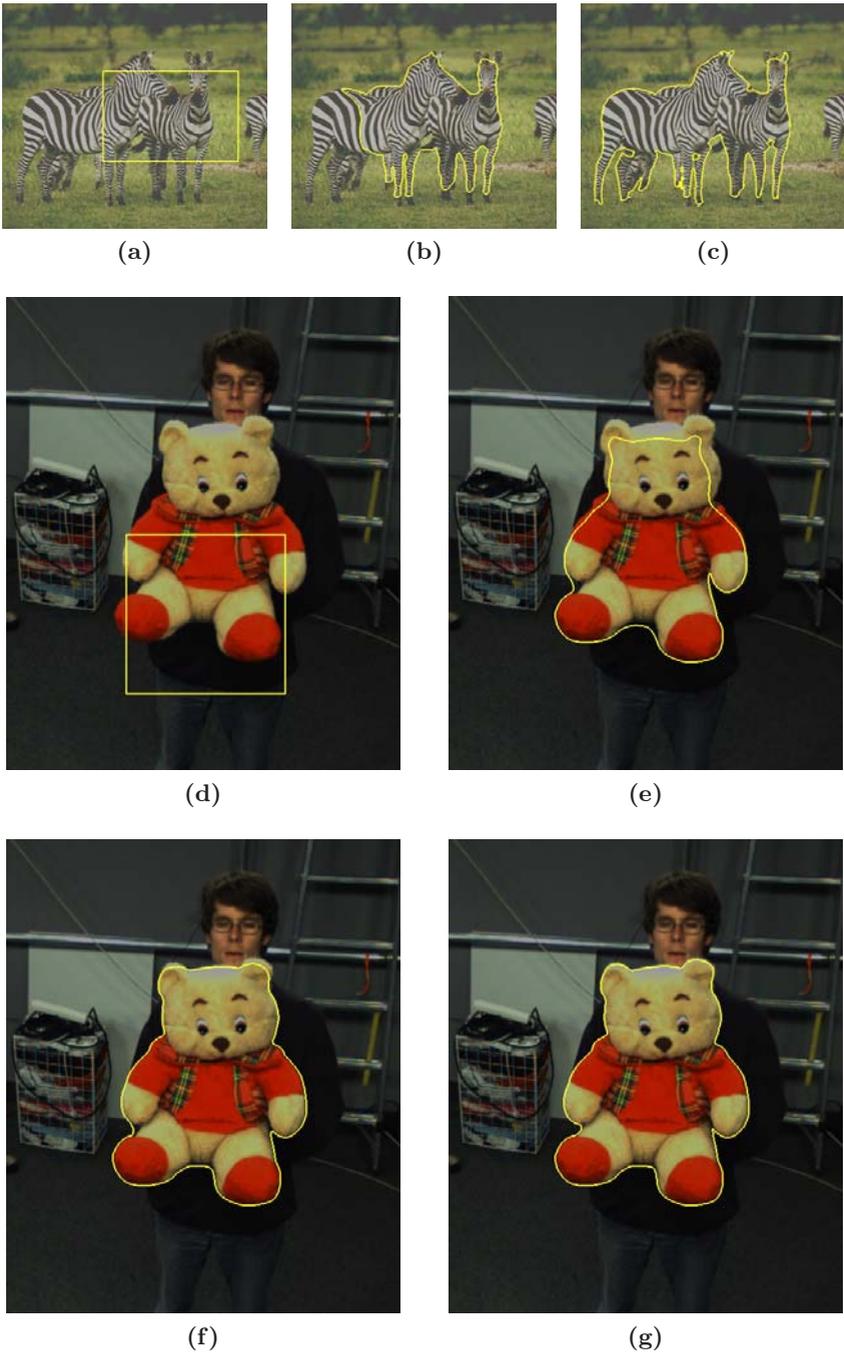


Fig. 7. (a) Level set initialization; (b) final segmentation using EU; (c) final segmentation using RK-2; (d) level set initialization; (e) segmentation result after 180 it. using EU; (f) final segmentation using EU (900 it.); (g) final segmentation using RK-2 (180 it.). The computational time is reduced by the factor 2.

5 Conclusion

In this paper we proposed to use higher order optimization schemes to solve the well known variational approach to image segmentation, and we compared our approach with the traditional method for this problem. By synthetic and real image experiments we showed that the use of higher order Runge-Kutta methods improves the average accuracy of the final segmentation and reduces the dependence on the timestep and smoothing, parameters which critically influence the performance of the Euler method. We showed that using the second order Runge-Kutta method more reliably achieves accurate segmentations. Using our proposed scheme increases the number of scenes in which image segmentation using the variational approach can succeed. Furthermore, the computational time decreases, in most cases.

References

1. Mumford, D., Shah, J.: Boundary detection by minimizing functionals. In: IEEE Computer Society Conference on Computer Vision and Pattern Recognition, San Francisco, CA, June 1985, pp. 22–26. IEEE Computer Society Press, Springer (1985)
2. Chan, T., Vese, L.: Active contours without edges. *IEEE Transactions on Image Processing* 10(2), 266–277 (2001)
3. Zhu, S.C., Yuille, A.: Region competition: unifying snakes, region growing, and bayes/mdl for multiband image segmentation. *IEEE Transaction on Pattern Analysis and Machine Intelligence* 18(9), 884–900 (1996)
4. Cremers, D., Tischhäuser, F., Weickert, J., Schnörr, C.: Diffusion snakes: introducing statistical shape knowledge into the mumford-shah functional. *International Journal of Computer Vision* 50(3), 295–313 (2002)
5. Rousson, M., Brox, T., Deriche, R.: Active unsupervised texture segmentation on a diffusion based feature space. In: IEEE Computer Society Conference on Computer Vision and Pattern Recognition, Madison, WI, pp. 699–704 (2003)
6. Cremers, D., Yuille, A.L.: A generative model based approach to motion segmentation. In: Michaelis, B., Krell, G. (eds.) DAGM 2003. LNCS, vol. 2781, pp. 313–320. Springer, Heidelberg (2003)
7. Malladi, R., Sethian, J., Vemuri, B.: Shape modelling with front propagation: A level set approach. *IEEE Transaction on Pattern Analysis and Machine Intelligence* 17(2), 158–174 (1995)
8. Paragios, N., Deriche, R.: Unifying boundary and region based information for geodesic active tracking. In: IEEE Computer Society Conference on Computer Vision and Pattern Recognition., Forth Collins, Colorado, vol. 2, pp. 300–305. IEEE Computer Society Press, Los Alamitos (1999)
9. Rosenhahn, B., Brox, T., Weickert, J.: Three-dimensional shape knowledge for joint image segmentation and pose tracking. *International Journal of Computer Vision* 73(3), 243–262 (2007)
10. Zhao, H.K., Chan, T., Merriman, B., Osher, S.: A variational level set approach to multiphase motion. *Journal of Computational Physics* 127, 179–195 (1996)
11. Brox, T., Weickert, J.: Level set based segmentation of multiple objects. In: Rasmussen, C.E., Bülthoff, H.H., Schölkopf, B., Giese, M.A. (eds.) DAGM 2004. LNCS, vol. 3175, pp. 415–423. Springer, Heidelberg (2004)

12. Rousson, M., Paragios, N.: Shape priors for level set representations. In: Heyden, A., Sparr, G., Nielsen, M., Johansen, P. (eds.) ECCV 2002. LNCS, vol. 2351, pp. 78–92. Springer, Heidelberg (2002)
13. Brox, T., Weickert, J.: A tv flow based local scale estimate and its application to texture discrimination. *Journal of Visual Communication and Image Representation* 17(5), 1053–1073 (2006)
14. Heiler, M., Schnörr, C.: Natural image statistics for natural image segmentation. *International Journal of Computer Vision* 63(1), 5–19 (2005)
15. Shu, C.W., Osher, S.: Efficient implementation of essentially non-oscillatory shock-capturing schemes. *Journal of Computational Physics* 77, 439–471 (1988)
16. Ge, F., Wang, S.: New benchmark for image segmentation evaluation. *Journal of Electronic Imaging* 16(3) (2007)
17. Cour, T., Yu, S., Shi, J.: Normalized cut image segmentation source code (2004), <http://www.cis.upenn.edu/~jshi/software/>