Enhancing a diffusion algorithm for 4D image segmentation using local information

Philipp Lösel, Vincent Heuveline

Philipp Lösel, Vincent Heuveline, "Enhancing a diffusion algorithm for 4D image segmentation using local information," Proc. SPIE 9784, Medical Imaging 2016: Image Processing, 97842L (21 March 2016); doi: 10.1117/12.2216202

SPIE.

Event: SPIE Medical Imaging, 2016, San Diego, California, United States

Enhancing a Diffusion Algorithm for 4D Image Segmentation Using Local Information

Philipp Lösel and Vincent Heuveline

Engineering Mathematics and Computing Lab (EMCL),
Interdisciplinary Center for Scientific Computing (IWR),
Heidelberg University, Heidelberg, Germany

ABSTRACT

Inspired by the diffusion of a particle, we present a novel approach for performing a semiautomatic segmentation of tomographic images in 3D, 4D or higher dimensions to meet the requirements of high-throughput measurements in a synchrotron X-ray microtomograph. Given a small number of 2D-slices with at least two manually labeled segments, one can either analytically determine the probability that an intelligently weighted random walk starting at one labeled pixel will be at a certain time at a specific position in the dataset or determine the probability approximately by performing several random walks. While the weights of a random walk take into account local information at the starting point, the random walk itself can be in any dimension. Starting a great number of random walks in each labeled pixel, a voxel in the dataset will be hit by several random walks over time. Hence, the image can be segmented by assigning each voxel to the label where the random walks most likely started from. Due to the high scalability of random walks, this approach is suitable for high-throughput measurements. Additionally, we describe an interactively adjusted active contours slice by slice method considering local information, where we start with one manually labeled slice and move forward in any direction. This approach is superior with respect to accuracy towards the diffusion algorithm but inferior in the amount of tedious manual processing steps. The methods were applied on 3D and 4D datasets and evaluated by means of manually labeled images obtained in a realistic scenario with biologists.

Keywords: Image segmentation, random walks, diffusion, interactive segmentation, semi-automatic segmentation, active contours, level set method, high performance computing, Multi-GPU, Chan-Vese algorithm.

1. INTRODUCTION

X-ray imaging and tomography provide the opportunity to visualize internal structures of optically dense materials in 3D and 4D. Specifically, the invention of synchrotron X-ray microtomography (SRμCT) was the onset of a new era of morphological research on microscopically small animals (e.g. micro-arthropods), since it enables unrivaled opportunities of high-throughput measurements of 3D and 4D tomographic imaging of dynamic systems and living organisms in sub-micron spatial and sub-second time resolution.[1–4] However, it results in such large amounts of data that analyzing such data is time consuming and technically challenging. To analyze structures and objects, it is necessary to model them by means of a three-dimensional segmentation, but SRμCT images are characterized by low contrast and similar segments which are separated by diffuse edges (Fig. 3). Furthermore, several hundreds of significantly varying samples are scanned and no a priori knowledge of their topology exists. Therefore, it is not possible to use techniques like thresholding or knowledge-based methods. Due to this fact, the common method for segmentation is manually processing which sometimes lasts weeks or even months.

Further author information: (Send correspondence to Philipp Lösel)
E-mail: p.loesel@uni-heidelberg.de, Telephone: +49 (0) 6221 545278.

Medical Imaging 2016: Image Processing, edited by Martin A. Styner, Elsa D. Angelini, Proc. of SPIE Vol. 9784, 97842L · © 2016 SPIE
CCC code: 1605-7422/16/$18 · doi: 10.1117/12.2216202

Most segmentation algorithms are based on an interactive process with its user, namely, initialization, parameter setting and post-processing.[5,6] Still, this interaction ensures accuracy but it is also time consuming. The algorithms presented in this paper tie with this tradition, however, to meet the requirements of analyzing several (hundreds) of large datasets in 3D or even in 4D, it is necessary to put the manual process to a minimum while first, preserving accuracy and second, reaching a sufficiently high computational speed to allow an interactive process.

Therefore, we propose a diffusion model where it is only necessary to define seed points and which does not depend on a manually chosen parameter. In addition, it bases on the application of new computer architectures like Multi-Core and Multi-GPU to challenge the tasks of large datasets. Furthermore, it takes the complete volume into account, since considering only 2D slices in a 3D volume meant a huge loss of information. Specifically, once the user has manually defined some seed points for an arbitrary number of segments (but at least for two), several intelligently weighted random walks start at each labeled pixel. The seed points should be spread over the whole volume, but as we will show, only a few slices need to be labeled. Due to the definition of the weights, it is first, not necessary to define a parameter and second, the random walks tend to stay in an area which is similar to their starting point. Thus, based on the number of times a voxel was hit, the volume can be segmented. The segmentation succeeds even if gradients are weak and the volume is highly noisy or inhomogeneous (Fig. 2).

In addition to it, we present an active contours slice by slice method, which can be used to support and accelerate a manual segmentation. Further, it can be extended to an active shapes volume by volume method to segment 4D datasets. Active contours and active shapes have become very popular in recent years, since they have the ability to adapt to shapes of unknown topology.[5,7–10] The idea is to evolve a curve or surface by solving a partial differential equation (PDE) until the curve or surface stops on the region of interest. Moving from one level to another ensures an initialization close to the respective solution and thus reduces the risk of undesirable disturbances.

2. METHODS

In Sec. 2.1 we describe a GPU-based diffusion algorithm. In Sec. 2.2 we consider an advanced active shapes model to refine the result of the diffusion algorithm. Finally, in Sec. 2.3 we discuss an active shapes volume by volume model to handle 4D datasets.

2.1 GPU-based diffusion without edges

A graph is a couple (V, E) where V is a set of vertices in $\mathbb{R}^n$ and E is a set of edges, that is, E consists of some couples (x, y) where $x, y \in V$. We write $x \sim y$ if x is adjacent to y, i.e. $(x, y) \in E$. A weighted graph is a couple $((V, E), P)$ where P is a non-negative function on $V \times V$ and represents the weight of the edge between two nodes x and y. Let $I(x)$ be the image data, i.e. I maps x to a gray-value (e. g. for 8-bit images $I\colon \mathbb{R}^n \to \{0, \ldots, 255\}$). Further, let x_0 the starting point of a random walk and σ_{x_0} the standard deviation from x_0 in a local area of x_0 (Fig. 1), then we set

$$\mu_{x_0}(y) = \frac{1}{\sqrt{2\pi\sigma_{x_0}^2}} \exp\left(-\frac{(I(x_0) - I(y))^2}{2\sigma_{x_0}^2}\right) \quad \text{for all } y \in V,$$

and

$$\mu(x) = \sum_{y \in V, y \sim x} \mu_{x_0}(y) \quad \text{for all } x \in V.$$

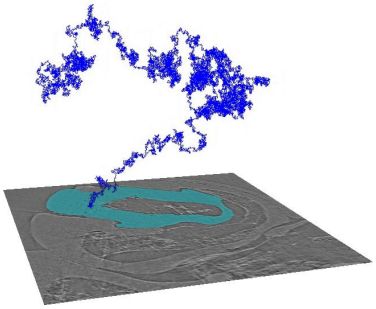


Figure 1: Weighted random walk starting at an initial seed point x_0.

The function μ induces a Markov kernel

$$P_{x_0}(y,x) = \begin{cases} \frac{\mu_{x_0}(y)}{\mu(x)} & \text{if } y \sim x \\ 0 & \text{else,} \end{cases} \tag{1}$$

where $P_{x_0}(y,x)$ is the conditional probability of a random walk moving from x to y given its position x. Due to the definition of P_{x_0}, the random walk tends to stay in an area which is similar to its starting position. Although P_{x_0} is defined for gray-valued images, it can be easily extended to multichannel images or any other image features. In common random walk algorithms which were introduced in Ref. 6, one has to solve the combinatorial dirichlet problem

$$\begin{aligned} \nabla \cdot (D(x)\nabla u(x)) &= 0 \\ u(x) \mid_{\partial\Omega_1} &= 1 \\ u(x) \mid_{\partial\Omega_2} &= 0, \end{aligned} \tag{2}$$

to segment an image into two regions Ω_1 and Ω_2. The solution u of Eq. (2) represents the probability that a random walk starting at an unlabeled position x, first hits $\partial\Omega_1$ before it hits $\partial\Omega_2$, which are the boundaries of the manually labeled seeds. In the discrete representation of Eq. (2), the diffusion coefficient D can be defined by means of the weights

$$\omega_{xy} = \exp(-\beta(I(x) - (I(y))^2).$$

Thus, the weights are gradient-based, and the user needs to define parameter β with respect to the strengths of the desired edges. But first, it is difficult to find the best value for β (Fig. 2) and second, the strengths may vary significantly between regions or even within one region. As opposed to the gradient-based approach, the diffusion model in this paper is parameter-free, which is an essential condition to handle several (hundreds) of different datasets, specially if a large number of objects have to be segmented (Fig. 9). The proposed model is not based on gradients, instead the weights are calculated with respect to the inner area of the seeded points. Hence, we call it a diffusion without edges (DWE) to distinguish it from standard diffusion methods (SDM).

Remark: This method can be extended to several regions $\Omega_1, \ldots \Omega_n$, where $n-1$ partial differential equations have to be solved. To calculate for each unlabeled x the probability that a random walk starting at position x first hits $\partial\Omega_i$ before it hits the rest, the boundary conditions have to be chosen

$$u = 0 \text{ on } \partial\Omega_1, \ldots, u = 0 \text{ on } \partial\Omega_{i-1}, u = 1 \text{ on } \partial\Omega_i, u = 0 \text{ on } \partial\Omega_{i+1}, \ldots, u = 0 \text{ on } \partial\Omega_n$$

respectively. Subsequently, the volume can be segmented by assigning each unlabeled x to the most likely boundary. It is only necessary to solve $n-1$ equations, since the sum of the probabilities have to equal 1.

The weights P_{x_0} in Eq. (1) depend on the starting point x_0. In addition, P_{x_0} is not symmetric, i.e. $P_{x_0}(x,y) \neq P_{x_0}(y,x)$. Therefore, we do not solve a PDE like Eq. (2), instead, we perform several random walks on GPUs,

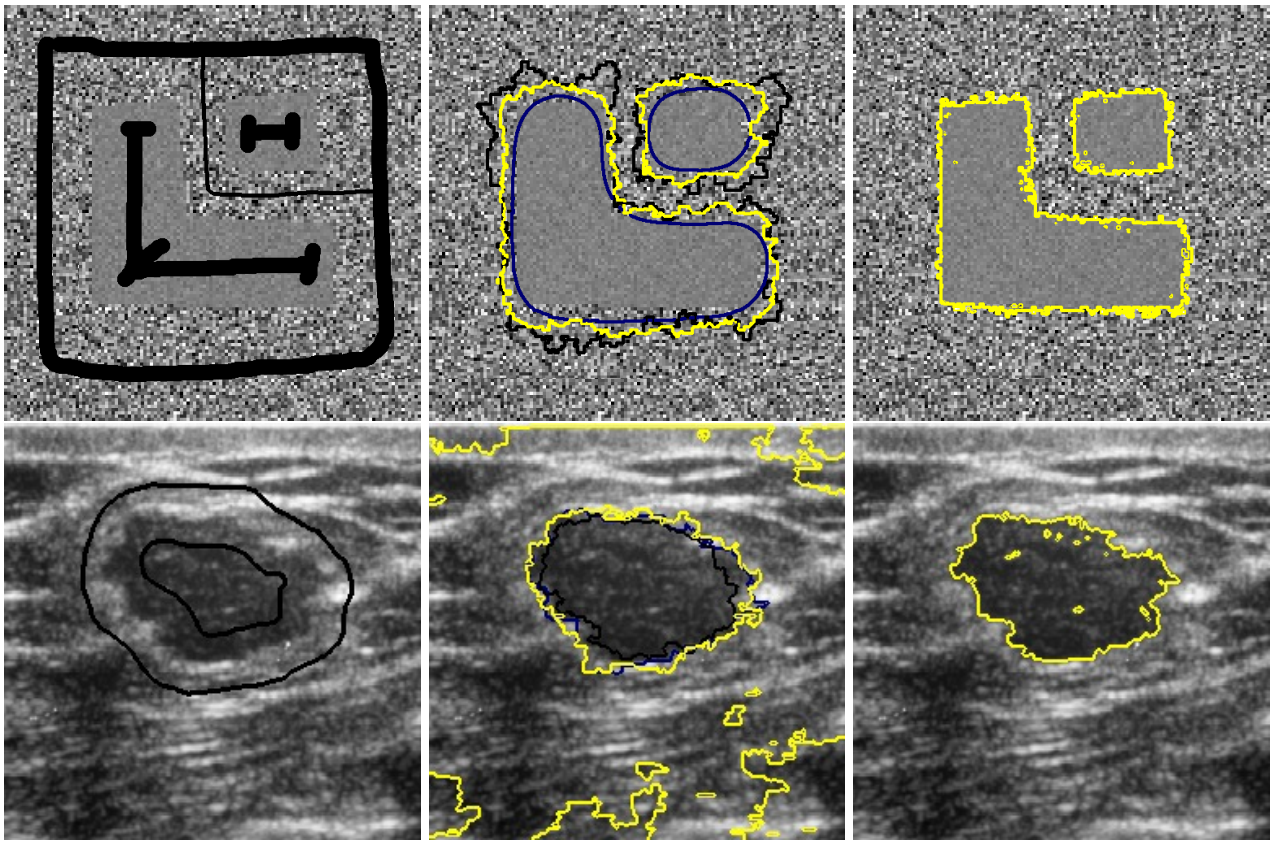


Figure 2: *Left*: Manually labeled seeds (inside and outside). *Center*: Edge-based diffusion for $\beta = 10$ (blue), $\beta = 100$ (yellow), $\beta = 1000$ (black) where we used the *scikit-image* library. *Right*: diffusion without edges (without any parameters). *Top*: noisy picture with same mean inside and outside but varying variances. *Bottom*: noisy ultrasound image with diffuse edges.[7]

which in contrast to SDM start at each labeled pixel. As a result, an unlabeled voxel in $\mathbb{R}^n$ will be hit by several random walks over time. Some of them might hit the voxel twice or more often. The number of hits by random walks, which started in the same manually labeled segment, leads to the probability that the voxel can be assigned to this segment. Thus, the volume can be segmented by assigning the voxels to the most likely segment. If a voxel is never hit, it will be assigned to the background. Due to its inherent high scalability, the algorithm can be implemented using any number of GPUs. In addition to it, one can easily revise the results by adding some new seeds to critical areas. Then, only the random walks for these new starting points have to be performed and added to the former results (Fig. 3). Whereas in SDM, it was necessary to solve again $n-1$ systems of PDEs, since the solution depends on the boundary conditions which changed in each system.

2.2 Active contours and shapes

Depending on the image quality and the manually defined seeds, the result obtained by the diffusion algorithm in Sec. 2.1 might be refined by a post-processing. Since the result is already close to the final solution, we recommend a local active shape model to fit the shape to the desired boundary. By the combination of the diffusion without edges and active shapes, we benefit from the features of the latter ones like a smooth solution, while we overcome their most common problem, namely, sticking in local minima.

The active shapes method bases on the Chan-Vese algorithm.[5] By minimizing the energy functional F in

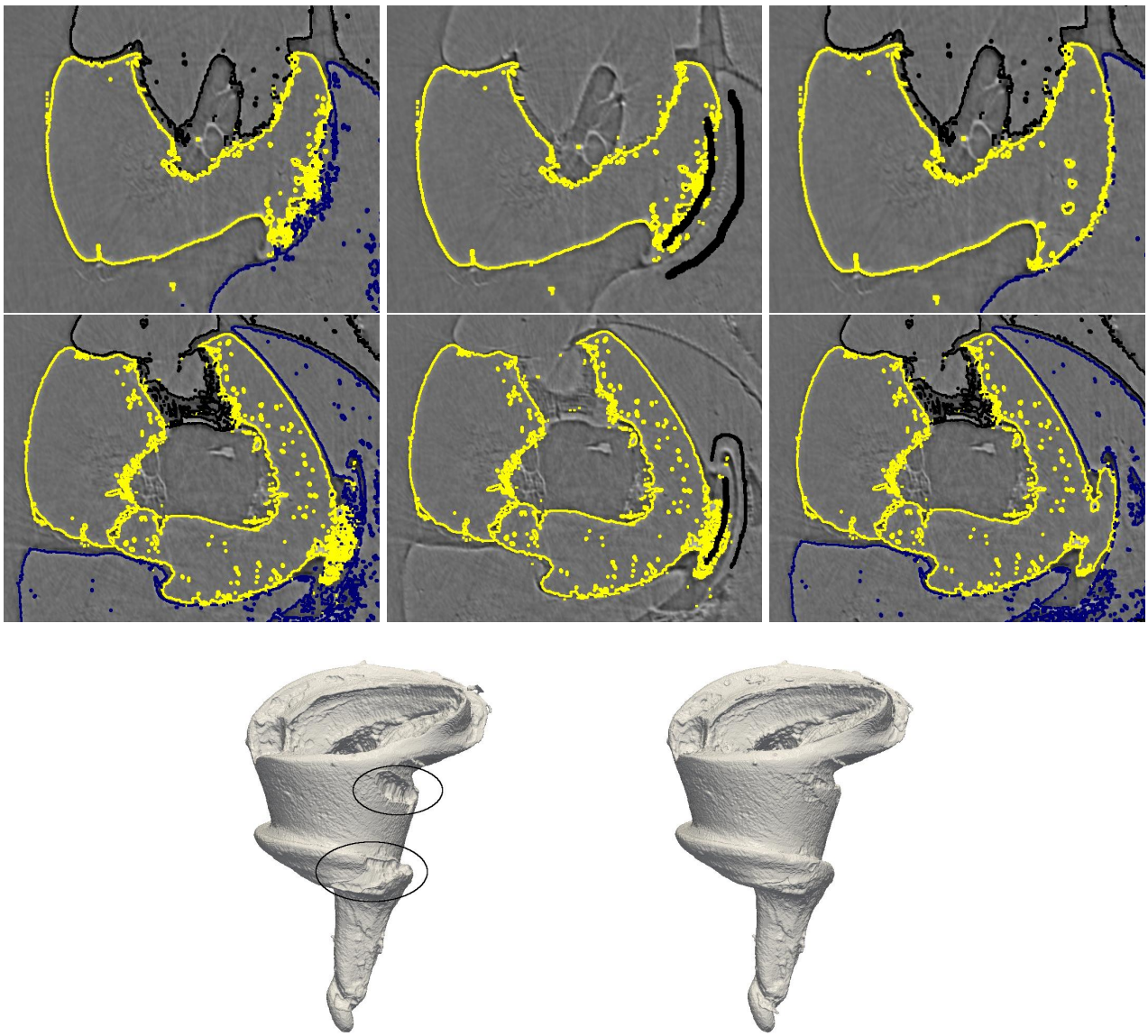
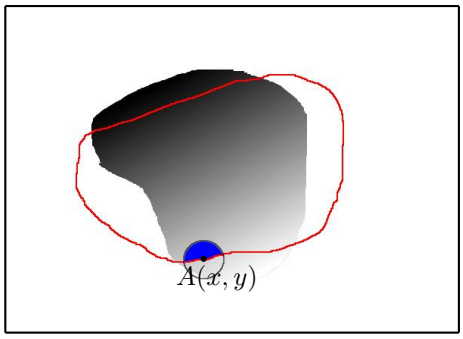
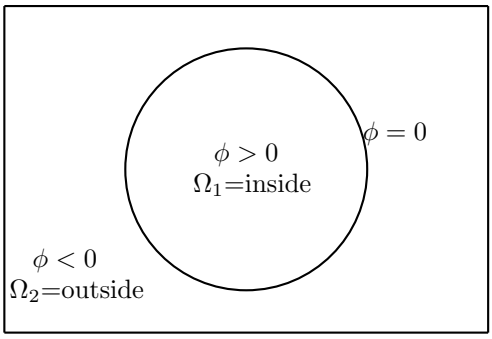


Figure 3: 3D SRμCT dataset of size ($687 \times 707 \times 650$). Screw joint of a beetle leg.[1,2] *Top left*: Incorrect segmentation. *Top center*: Additional seeds (for each case one slice). *Top right*: Updated result. *Bottom left*: Incomplete volume. *Bottom right*: Fixed volume.



(a) Local area A for a point x lying on the curve. Local mean and variance are calculated within the labeled area.



(b) 2D scenario: The curve is given by the zero level set of the function ϕ.

Figure 4: Active contours for two phases.

Eq. (3) a surface S will evolve until it stops due to a balance of internal and external forces. Most active contour methods[5,7,10] depend on global models, hence these methods have difficulties with inhomogeneous images. Local models were already proposed in Ref. 8,9,11. Since local models tend to intensify the problem to stick in local minimums, we consider global means and variances of each region

$$p_i(I(x)) = \frac{1}{\sqrt{2\pi\sigma_i^2}} \exp\left(-\frac{(\mu_i - I(x))^2}{2\sigma_i^2}\right),$$

as well as local means and variances by assuming them as spatially varying parameters

$$p_{i,x}(I(y)) = \frac{1}{\sqrt{2\pi\sigma_i(x)^2}} \exp\left(-\frac{(\mu_i(x) - I(y))^2}{2\sigma_i(x)^2}\right).$$

A combined model was also proposed in Ref. 12 but without considering variances. The surface S divides the volume into two regions $\Omega_1 = inside(S)$ and $\Omega_2 = outside(S)$ (Fig. 4b). Let

$$A(x,y) = \begin{cases} 1 & \text{if } \|x-y\| < r \\ 0 & \text{else,} \end{cases}$$

be a local area of x with radius r (Fig. 4a),

$$F_i = -\log p_i(I(x)),$$
$$F_i^{loc} = -\log p_{i,x}(I(y)),$$

then we refine the surface S by minimizing the functional

$$F(S,c) = \alpha \cdot \text{Area}(S) + \int_{\Omega_1} (1-k)F_1 + k\left[\int_{\Omega_1 \cap A} F_1^{loc}\, dy\right] dx + \int_{\Omega_2} (1-k)F_2 + k\left[\int_{\Omega_2 \cap A} F_2^{loc}\, dy\right] dx, \quad (3)$$

with respect to S and $c = (\mu_1, \mu_2, \sigma_1^2, \sigma_2^2, \mu_1(x), \mu_2(x), \sigma_1(x)^2, \sigma_2(x)^2)$, where $k \in [0,1]$. Additionally, one might optimize the size of $A(x,y)$ by maximizing $\|\mu_1(x) - \mu_2(x)\|$. Using the Level Set Method, where S is represented by the zero level set of a Lipschitz function $\phi : \Omega \to \mathbb{R}$, i.e. $S = \{x \in \mathbb{R}^3 | \phi(x) = 0\}$ (Fig. 4b), and the Heaviside function

$$H(z) = \begin{cases} 1 & \text{if } z \geq 0 \\ 0 & \text{if } z < 0, \end{cases}$$

the energy functional F in Eq. (3) can be expressed as

$$F(\phi, c) = \int \alpha|\nabla H(\phi(x))| + (1-k)\left[F_1 H(\phi(x)) + F_2(1 - H(\phi(x)))\right] dx \quad (4)$$
$$+ k \int\int A(x,y) F_1^{loc} H(\phi(y)) + A(x,y) F_2^{loc}(1 - H(\phi(y)))\, dy\, dx.$$

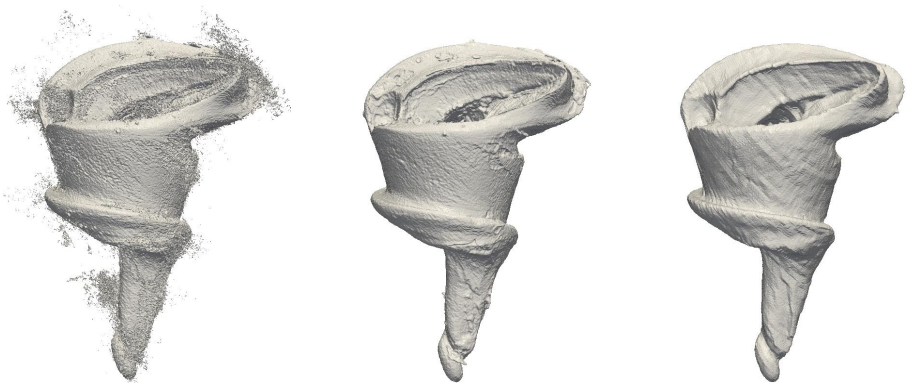


Figure 5: 3D SRμCT dataset of size ($687 \times 707 \times 650$). *Left*: Solution of DWE, firstly, initialized in 8 slices, then seeds were added to 2 slices for a correction (Fig. 3). *Center*: Final solution, after refining the DWE result by means of active shapes for $k = 0.5$. *Right*: Manually labeled volume by an expert. Accuracy: DWE 94%, DWE+active shapes 97%. Ratio detected area / ground truth: DWE 0.97, DWE+active shapes 0.99.

By calculus of variations, it can be shown that the functions $\mu_1(x)$, $\mu_2(x), \sigma_1(x)^2, \sigma_2(x)^2$ and the constants $\mu_1, \mu_2, \sigma_1^2, \sigma_2^2$ that minimize the energy functional F in Eq. (4) for a fixed ϕ satisfy the Euler-Lagrange equations:

$$\begin{aligned}
\mu_1 &= \frac{\int I(x)H(\phi)\mathrm{d}x}{\int H(\phi)\mathrm{d}x} & \sigma_1^2 &= \frac{\int (\mu_1 - I(x))^2 H(\phi)\mathrm{d}x}{\int H(\phi)\mathrm{d}x} \\
\mu_2 &= \frac{\int I(x)(1-H(\phi))\mathrm{d}x}{\int (1-H(\phi))\mathrm{d}x} & \sigma_2^2 &= \frac{\int (\mu_2 - I(x))^2 (1-H(\phi))\mathrm{d}x}{\int (1-H(\phi))\mathrm{d}x} \\
\mu_1(x) &= \frac{\int A(x,y)I(y)H(\phi)\mathrm{d}y}{\int A(x,y)H(\phi)\mathrm{d}y} & \sigma_1(x)^2 &= \frac{\int A(x,y)(\mu_1(x) - I(y))^2 H(\phi)\mathrm{d}y}{\int A(x,y)H(\phi)\mathrm{d}y} \\
\mu_2(x) &= \frac{\int A(x,y)I(y)(1-H(\phi))\mathrm{d}y}{\int A(x,y)(1-H(\phi))\mathrm{d}y} & \sigma_2(x)^2 &= \frac{\int A(x,y)(\mu_2(x) - I(y))^2 (1-H(\phi))\mathrm{d}y}{\int A(x,y)(1-H(\phi))\mathrm{d}y}.
\end{aligned}$$

Thus, μ_i and σ_i are global intensity means and standard deviations of the regions Ω_i respectively and $\mu_i(x)$ and $\sigma_i(x)$ are the corresponding local terms of $\Omega_i \cap A(x,y)$. By using the gradient descent method and by minimizing F with respect to ϕ the evolution of the surface S can be described by the partial differential equation

$$\frac{\partial \phi}{\partial t} = \delta_\epsilon(\phi)\left[\alpha \ \mathrm{div}\left(\frac{\nabla\phi}{|\nabla\phi|}\right) + (1-k)\left[F_2 - F_1\right] + k\int A(x,y)\left[F_2^{loc} - F_1^{loc}\right]\ \mathrm{d}y\right],$$

where for numerical reasons $\delta_\epsilon(z)$ is an approximation of $\delta_0(z) = H'(z)$, see Ref. 5 for details.

2.3 Active contours slice by slice

The active contours slice by slice method, or active shapes volume by volume, is an advanced method of the active contours presented in section 2.2, where we consider the final solution of one slice or volume as the initialization for the next one (Fig. 6). Due to high resolution images in SRμCT, changes in the image are small. Therefore, in each step, the algorithm converges fast to the respective solution. We particularly mention that considering only 2D slices in a 3D volume means a hough loss of information. Hence, the active contours slice by slice method often needs a manual post-processing depending on the identifiability of the boundaries. Due to this interaction, this method might increase the accuracy but also the amount of tedious manual processing steps.

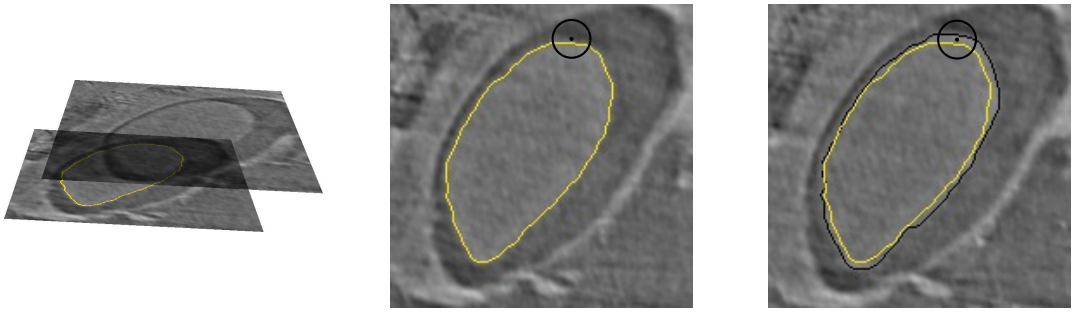


Figure 6: Active contours slice by slice. *Left*: Use prior segmentation as initialization and move forward in z-direction. *Center*: Slice n with related solution. *Right*: Slice $n+1$ with initialization (yellow) and final solution (black). The circle represents a local region $A(x, y)$ at poistion x. In step n, x lies outside, in step $n+1$, x lies inside the contour.

3. NUMERICAL EXPERIMENTS AND RESULTS

The algorithms were implemented in Python 2.7. We used PyCUDA 2015 with CUDA 7.5 for parallelization of the random walks and performed them on up to 8 GPUs [*]. In all scenarios we started in each seeded point 20 random walks with 4000 steps respectively. Depending on the amount of labeld seeds, the calculation lasted for one manually labeled slice of size (707×687) between 10 minutes (for a small number of seeds) and up to 30 minutes in case of a completely labeled slice. In Fig. 5 we defined seed points in 8 slices. Since we used 8 GPUs, the computational time for the beetle leg with the diffusion without edges method was around 30 minutes (regardless of the correction). Besides the CUDA implementation, computationally expansive parts were written in C and embedded with the *scipy.weave* library. Whenever active shapes were applied, we used a multiphase level set approach, introduced by Chan-Vese in Ref. 10, to avoid the problems of vacuum and overlapping. In addition, by using the multiphase level set, we only need to consider n level set functions for m regions, where m satisfies $m = 2^n$. That means, in case of 4 regions, we had to evolve only 2 level set functions.
In Fig. 2, we show how our diffusion method works on noisy images with diffuse edges and compare the results with a standard diffusion method. We show, that in contrast to the standard method, it is not necessary to define a parameter for a successful segmentation.
In Fig. 3, we show how results of DWE can easily be corrected. As mentioned, we set seed points in 8 slices, but due to difficulty areas, we had to add seeds to two additional slices. Only the random walks for these additional seeds had to be computed and added to the previous. Thus, it was not necessary to calculate all random walks again.
In Fig. 5, we show how results of DWE can be refined by using active shapes and we compare a refinement by means of a manually labeled segment from a biologist.
In Fig. 9, we show a 3D dataset with 33 different objects, including hard tissues like skeleton and soft tissues like muscles. 21 slices were labeled by a biologist. These slices were used as seeds for the diffusion without edges. In this case, the result was not refined with active shapes but slightly smoothed with Amira.
In Fig. 7, we show one time step of a 4D chest CT scan and in Fig. 8 five time steps of a moving beetle. The initialization was given in both cases by DWE. Then, the surfaces were evolved by the active shapes volume by volume method.

The diffusion without edges presented in section 2.1 is independent of the persistence of strong edges, it is suitable for low contrast images and in contrast to common gradient-based approaches, it does not require a manually chosen parameter. Particularly, considering the unrestrained scalability of random walks and the small amount of manual processing, we expect a substantial impact on the analysis of large amounts of datasets generated in SRμCT systems.

[*]2 Nvidia Tesla K40, 4 Nvidia Tesla K20, 1 Nvidia Grid K2, 1 Nvidia GeForce GTX 770.

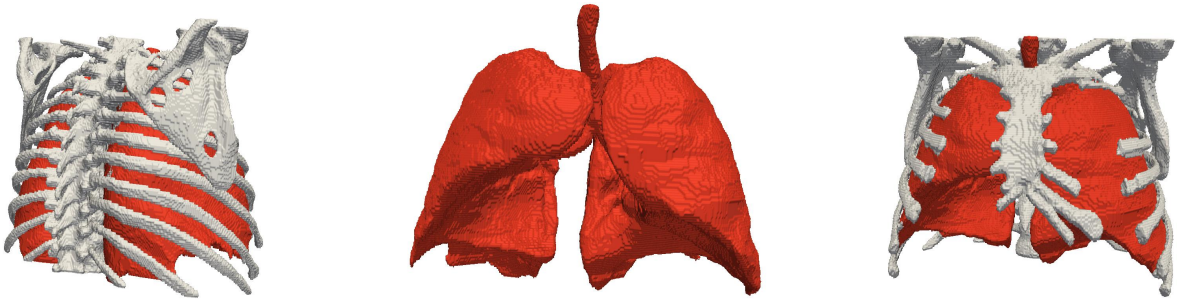


Figure 7: 4D chest CT scan of size ($512 \times 512 \times 141 \times 10$), i.e. the dataset consists of 10 time steps with each a size of ($512 \times 512 \times 141$). Time step 5 was segmented with the diffusion without edges method. For the time series we used active shapes volume by volume for $k = 0.25$. The images show time step 3. *From left to right*: Back view of lungs and skeleton, front view of lungs, front view of lungs and skeleton. The data were obtained from the Lon Brard Cancer Center & CREATIS lab, Lyon, France.[13]

Figure 8: 4D SRμCT dataset of size ($384 \times 384 \times 192 \times 343$), i.e. 343 time steps. Sequence of a moving beetle using diffusion without edges for the first volume and active shapes volume by volume for the time steps, $k = 0.25$.

The manual labeling of the dataset in Fig. 5 took 1-2 weeks for an expert, which was interrupted by breaks, to be sure, but the expert assumes, that still one week should be necessary for someone who is used to it and significantly longer for someone who is not. In the latter case, it could last up to one month. The definition of seeds for the diffusion without edges algorithm was done by non-experts. We seeded roughly 8 slices which took approximately 5-10 minutes for each slice. As already mentioned, the diffusion without edges on 8 GPUs lasted around 30 minutes. In addition to it, the method allows an easy correction of its solution. Users with a strong background of the objects of interest, should be able to define the seeds from the beginning with respect to difficult areas.

Though the diffusion method can easily be used for 4D datasets as well, we recommend the active shapes volume by volume method for 4D image segmentation, since in a high resolution image only few steps are necessary to fit the shape to its new boundaries. The active contours slice by slice method can be suggested to support a tedious manual segmentation, but the loss of information by considering 2D slices only causes more segmentation faults than in volume-based approaches, and thus, it needs more interaction with its user.
All presented algorithms are meant to be interactive methods and can be easily combined with any segmentation toolkit like Amira. Therefore, a user needs to define seed points within the software of its choice, save them in an adequate format, such as multif, run the methods of this paper as an external functionality and reload the result to the toolkit to add seed points for a correction or to do any other post-processing step. The collaboration center Astor, where the presented methods were developed, aims to deliver a software to extend any segmentation tool with these methods.

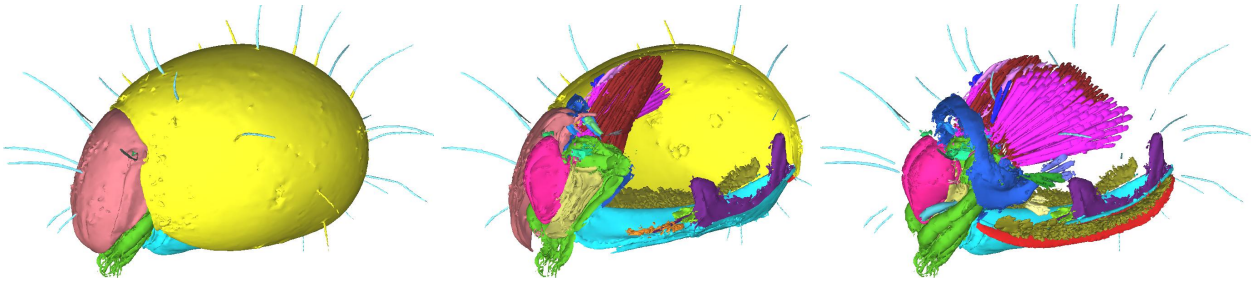


Figure 9: *Euphthiracarus reticulatus*: 3D SRμCT dataset of size ($695 \times 766 \times 1088$). Containing 33 objects including soft and hard tissues. *From left to right*: Complete volume, sliced along x-axes, without shell.

4. CONCLUSIONS

In this paper, we developed a diffusion model which satisfies the requirements of analyzing high-throughput measurements in a synchrotron X-ray microtomograph if no a priori knowledge of the objects of interest exists, namely, high accuracy as well as an easy and fast correction of the results. Based on probability theory, we derived a chaotic system for a weighted diffusion of particles in an arbitrary number of dimensions and performed the associated random walks on GPUs. To refine the result of the diffusion method, we combined it with an active shapes model considering local and global image information. Furthermore, we presented an active shapes volume by volume method for 4D image segmentation, where we suggest to use the result of the diffusion without edges as an initialization.

ACKNOWLEDGMENTS

This work was carried out with the support of the Federal Ministry of Education and Research (BMBF), Germany, within the collaboration center (Verbundprojekt 05K2013) Astor (Arthropod Structure revealed by ultra-fast Tomography and Online Reconstruction). We would also like to thank Thomas van de Kamp from KIT, Germany, Michael Heethoff and Sebastian Schmelzle, both from Technische Universität Darmstadt, Germany, for providing the SRμCT data.

REFERENCES

[1] van de Kamp, T., V. P. B. T. and Riedel, A., "A biological screw in a beetles leg," *Science* **333**, 52 (2011).

[2] van de Kamp, T., d. S. R. T. V. P. B. T. and Riedel, A., "Three-dimensional reconstructions come to life interactive 3d pdf animations in functional morphology," *PLoS ONE* **9(7): e102355** (2014).

[3] Schmelzle, S., H. L. N. R. and Heethoff, M., "The ptychoid defensive mechanism in euphthiracaroidea (acari: Oribatida): A comparison of muscular elements with functional considerations," *Arthropod Structure and Development* **38(6)**, 461–472 (2009).

[4] Schmelzle, S., N. R. and Heethoff, M., "Mechanics of the ptychoid defense mechanism in ptyctima (acari, oribatida): one problem, two solutions," *Zoologischer Anzeiger - A Journal of Comparative Zoology* **254(27)**, 27–40 (2015).

[5] Chan, T. and Vese, A., "Active contours without edges," *IEEE Transactions on Image Processing* **10**, 266–277 (2001).

[6] Grady, L., "Random walks for image segmenation," *IEEE Transactions on Pattern Analysis and Machine Intelligence* **28**, 1768–1783 (2006).

[7] Marquez-Neila, P., B. L. and Alvarez, L., "A morphological approach to curvature-based evolution of curves and surfaces," *IEEE Transactions on Pattern Analysis and Machine Intelligence* **36**, 2–17 (2014).

[8] Wang, L., H. L. M. A. L. C., "Active contours driven by local gaussian distribution fitting energy," *Signal Processing* **89**, 2435–2447 (2009).

[9] Lankton, S. and Tannenbaum, A., "Localizing region-based active contours," *IEEE Transactions on Image Processing* **17**, 2029–2039 (2008).

[10] Vese, L. and Chan, T., "A multiphase level set framework for image segmentation using the mumford and shah model," *International Journal of Computer Vision* **50(3)**, 271–293 (2002).

[11] Rousson, M. and Deriche, R., "A variational framework for active and adaptive segmentation of vector valued images," *HAL archives-ouvertes* **RR-4515**, inria–00072073 (2002).

[12] Wang, L., L. C. S. Q. X. D. K. C., "Active contours driven by local and global intensity fitting energy with application to brain mr image segmentation," *Computerized Medical Imaging and Graphics* **33**, 520–531 (2009).

[13] Vandemeulebroucke, J., R. S. K. J. C. P. and Sarrut, D., "Spatiotemporal motion estimation for respiratory-correlated imaging of the lungs," *Med Phys* **38(1)**, 166–178 (2011).