

# Interactive 2D/3D Image Denoising and Segmentation Tool for Medical Applications

Martin Urschler<sup>1,3</sup>, Gerd Leitinger<sup>2</sup>, and Thomas Pock<sup>3\*</sup>

<sup>1</sup> Ludwig Boltzmann Institute for Clinical Forensic Imaging, Graz, Austria

<sup>2</sup> Research Unit Electron Microscopic Techniques, Institute of Cell Biology, Histology, and Embryology, Medical University of Graz, Austria.

<sup>3</sup> Institute for Computer Graphics and Vision,  
BioTechMed, Graz University of Technology, Austria

**Abstract.** We present an interactive image and volume denoising and segmentation tool, that allows the use of state of the art algorithms based on the concept of total variation (TV) to be applied to 2D and 3D grey and color value data. Our aim is to provide a research tool for experimenting with TV related methods to the medical image analysis community. Our implementation supports the classic ROF and TVL1 denoising models, second order TV denoising variants as well as an interactive segmentation based on weighted TV. The numerical optimization procedures to solve the variational problems follow the primal-dual scheme of Chambolle and Pock. A highly efficient GPU implementation based on NVidia CUDA enables real-time feedback even for large volumetric data sets. We will provide a binary software tool to the public domain for interested researchers at our website and intend to perform a live demonstration on a notebook at the MICCAI workshop.

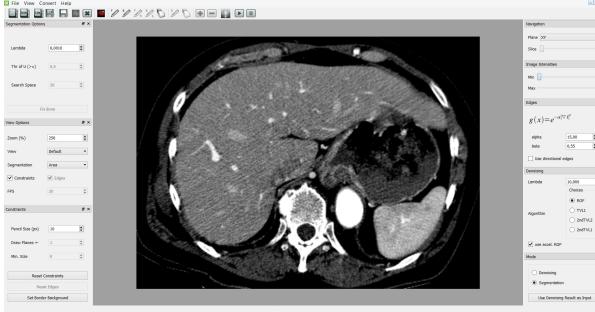
## 1 Introduction

While fully automatic image processing and segmentation methods are required in a variety of medical image analysis applications, especially in the presence of large amounts of data to process, recent years have seen an increasing interest in interactive image processing methods, which give immediate feedback about a solution and allow modifying results on the fly. The benefits of an expert user directly interacting with an image processing task have been widely acknowledged by researchers in the medical image analysis field, mostly for 2D images [1–4] but recently also for 3D volumetric applications [5–7]. A survey on interactive image segmentation techniques can be found in Zhao and Xie [8].

Successful interactive image segmentation tools require an easy-to-use graphical user interface, immediate, real-time feedback of processing tasks even for

---

\* This work was partly supported by the Austrian Science Fund (FWF) under the START project BIVISION, No. Y729. We are grateful to Philipp Aichinger from the Division of Phoniatrics-Logopedics at Medical University Vienna for providing the glottis video.



**Fig. 1.** Screenshot of the Qt-based user interface of our image processing tool. Shows a single slice of a liver CT data set.

large data sets, flexibility in terms of input modalities, which may be processed, and of course the use of state of the art image analysis algorithms. Another benefit for the research community is the publicly available access to such tools. We present a software tool, which adheres to the requirements stated above, and will make it available to the public domain in the form of a binary executable. The software makes extensive use of modern NVIDIA graphics adapters with the use of CUDA for implementing the core image processing parts. By using graphics adapters as numerical co-processors, immediate response becomes feasible, which facilitates efficient user interaction to solve a given image processing task. Further key features of our contributed tool are interactive edge-preserving denoising, support for 2D images, 3D videos or 3D volumes, as well as greyscale and RGB color inputs.

## 2 Method

Our image processing algorithms are presented in a Qt-based<sup>4</sup> graphical user interface, which can be seen in Fig. 1. The user interface enables loading and storing of 2D and 3D data sets from image and volume file formats, as well as image folders by specifying a directory name, which contains the individual images of a temporal time series of 2D images. We have implemented denoising and segmentation algorithms that make use of the framework of the calculus of variations to solve convex energy functionals, specifying the desired image processing effects. This is in contrast to the more often used graph based Markov Random Field setup, where image processing tasks are defined in a discrete manner using image pixels and their neighborhoods as nodes and edges in a graph, and which can very efficiently be solved using graph cuts [2], a prominent example of this paradigm being the GrabCut algorithm [3]. Paradigms like GrabCut are currently also implemented in commercial software like Adobe Photoshop®, but

---

<sup>4</sup> <http://qt-project.org/>

restricted to 2D segmentation applications. Our decision to choose the variational framework is mainly motivated by the less memory intensive and simpler parallelization of the partial differential equations that occur when optimizing the occurring energy functionals. In contrast, parallelization of graph cut based methods to directly implement them on a GPU is not straight-forward. We are able to efficiently implement a range of methods, like edge-preserving total variation (TV) based denoising, second-order TV denoising, as well as interactive segmentation based on a weighted TV model, using the same numerical optimization algorithm proposed by Chambolle and Pock [9].

## 2.1 Edge-preserving denoising

A typical image processing pipeline for solving segmentation problems consists of a denoising step followed by the interactive segmentation procedure. For denoising it is crucial that high-frequency information, which is identified as noise, is suppressed, while important edge information describing the structures of semantic interest remain unchanged. For this purpose, state of the art edge-preserving denoising models are formulated using a total variation (TV) regularization, as first presented in computer vision by the seminal paper of Rudin, Osher and Fatemi [10]. The model, which commonly is referred to as ROF model, formulates an energy functional that penalizes deviations of the sought solution  $u$  from the noisy input image  $f$  with an  $L2$ -norm, while the regularization penalizes jumps in  $u$  via the norm of the gradient of  $u$ . The corresponding energy minimization problem is

$$\min_u \left\{ \int_{\Omega} |\nabla u(x)| dx + \frac{\lambda}{2} \int_{\Omega} (u(x) - f(x))^2 dx \right\}. \quad (1)$$

It is well known that this edge-preserving denoising model is optimal under the assumption of Gaussian noise. To adapt to salt and pepper noise, a slight modification of 1 leads to the TVL1 model, where the data term also uses an  $L1$ -norm penalty to be more robust against outliers, thus the formulation reads

$$\min_u \left\{ \int_{\Omega} |\nabla u(x)| dx + \lambda \int_{\Omega} |u(x) - f(x)| dx \right\}. \quad (2)$$

The only free parameter in these two energy functionals is the parameter  $\lambda$ , that steers the trade-off between data fidelity and regularization (i.e., smoothing). Selecting an appropriate  $\lambda$  for a given problem is often a try-and-error procedure, which can benefit from immediate feedback to the user and the possibility of interactively changing the parameter. Further, the interpretation of  $\lambda$  for the ROF and TVL1 models is different. Therefore, in our graphical user interface, real-time computation and immediate feedback of the models stated above, is crucial to speed up the search for an appropriate  $\lambda$  in a certain application. Please note that our solvers based on the primal-dual scheme presented in Section 2.3 do not require the tuning of further parameters.



**Fig. 2.** Screenshot of the segmentation (in red) of the ROF denoised ( $\lambda = 10$ ) liver slice, which was produced by placing a foreground seed stroke (yellow) into the liver and removing some parts in the background (blue).

## 2.2 Interactive segmentation

Our interactive segmentation method is initialized by one of the denoising approaches stated in Section 2.1. Based on geodesic active contours (GAC) [11], it follows a convex weighted total variation (wTV) formulation according to the work of Bresson et al. [12] and its adaptation in [4]. Although the convex model is only an approximation to the GAC energy, since for the original GAC  $u \in \{0, 1\}$  has to hold, while the approximation relaxes this non-convex set to  $u \in [0, 1]$ , in practice the choice of a threshold for  $u$ , thus binarizing the result again, is not critical and may safely be assumed fixed to 0.5. The corresponding energy minimization functional is

$$\min_u \left\{ \int_{\Omega} g(x) |\nabla u(x)| dx + \lambda \int_{\Omega} u(x) f(x) dx \right\}. \quad (3)$$

One can immediately see the similarity to the denoising approaches in Section 2.1, and indeed the numerical optimization of the segmentation functional can be performed in the same primal-dual framework. The expression  $g(x)$  in 3 resembles the influence of the gradients of the input image, while  $f(x)$  describes user-specified seed values or a pre-computed classification into fore- and background pixels. Due to the energy minimization, 3 tries to assign the foreground value 1, where  $f$  is negative, and the background value 0, where  $f$  is positive. Thus, modeling hard fore- and background constraints with  $f = -\infty$  and  $f = +\infty$ , respectively, is beneficial. Figure 5 shows an example for a 2D liver slice segmentation task.

For the computation of  $g(x)$  we implemented a simple scheme, which can be efficiently performed on the fly during the numerical optimization. We define  $g(x) = \exp^{-\alpha|\nabla I(x)|^{\beta}}$ , where  $\alpha, \beta$  are tunable parameters steering the mapping of input intensity  $I$  gradients to the range between zero and one. Note that in regions with a strong edge,  $g(x)$  tends to zero to attract the final solution  $u$  to the edges. Combining constraints  $f$  and image gradients  $g$  in this way results in a

convex optimization scheme, where the user can interactively draw seed regions into an image and the solution will be located on the edges, while at the same time minimizing the total segmentation contour length (2D case) or surface area (3D case). In total for the wTV segmentation we have several parameters to tune ( $\lambda, \alpha, \beta$ ), thus an interactive user interface giving immediate feedback on the segmentation result with the possibility to remove unwanted parts using the background seed interaction is very valuable.

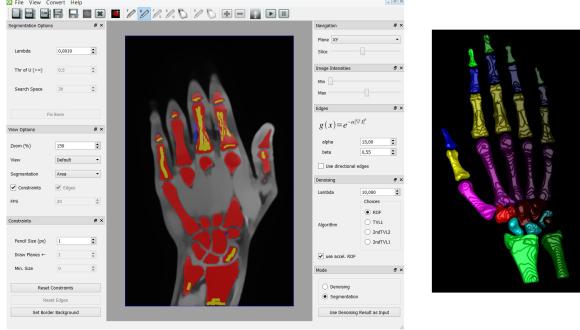
### 2.3 Implementation details

*Primal-dual optimization* To solve the energy functionals from Sections 2.1 and 2.2, the numerical primal-dual optimization scheme from Chambolle and Pock [9] is used. It describes a very general algorithm for solving convex problems with a saddle-point structure. With their proven convergence rate of  $O(1/N)$  and even  $O(1/N^2)$  for problems where either data or regularization term is uniformly convex (e.g., ROF), they are very well suited for a highly efficient parallelized implementation on a GPU, due to the simple convolution-like operations that occur after discretizing the partial differential equations (PDE) that resemble the optimal solution. In their algorithm discrete total variation is formulated as a linear operator  $K$  and a generic saddle point problem

$$\min_{u \in X} \max_{p \in Y} \langle Ku, p \rangle + G(u) - F^*(p) . \quad (4)$$

is solved. Functions  $G$  and  $F$  correspond to convex functions describing one of the denoising and segmentation models. We refer to  $u$  as the scalar-valued primal variable, and  $p$  as the vector-valued dual variable.  $X$  and  $Y$  are appropriate vector spaces. For more theoretical details please refer to [9].

From a practical point of view, we have to solve a gradient descent in the primal variable, and a gradient ascent in the dual variable. Primal and dual updates are tightly coupled, i.e., the update of  $u$  requires the divergence of  $p$  and the update of  $p$  requires the gradient of  $u$  in the ROF, TVL1 and wTV models. This can be seen in Algorithm 1 in [9]. Unfortunately, the algorithm is rather memory intensive, e.g., for the 3D ROF denoising model we require in addition to the input and output volumes three volumes for the dual variable  $p$  and a copy of the output volume  $u_0$  for the over-relaxation step. The primal-dual algorithm is implemented on a GPU to benefit from the massive parallelization possible by the NVidia CUDA programming model. Since our generic implementation needs to be flexible and also work for large volumetric data (possibly with multi-channel input like in the case of color videos), we have decided to implement primal-dual updates with a strategy of overlapping grid blocks and perform it in a red-black manner similar to implementations of the Gauss-Seidel method. By using the shared memory of the GPU we can get rid of the intermediate volume  $u_0$ , it is solely computed locally and stored in shared memory during iterations, thus only three intermediate volumes for the vectorial dual variable  $p$  are required.



**Fig. 3.** Bone segmentation using the interactive segmentation tool. After ROF denoising, fore- and background seeds specify the hand bones (left). After labeling the bones we show a 3D rendering of hand (right).

*2D,3D and color video support* Our software is generic and flexible in the sense that we support two- and three-dimensional data for all our algorithms as well as single channel intensity and multi-channel RGB inputs. Each channel may contain more than 8 bit of intensity resolution due to an internal representation of the channels as floating point values between zero and one. This is a limitation of the current prototype, since the floating point representation is not very memory efficient. We treat all input data as three-dimensional data, with 2D images being represented as a volume with a single slice. This way we can use the same algorithms for 2D and 3D data.

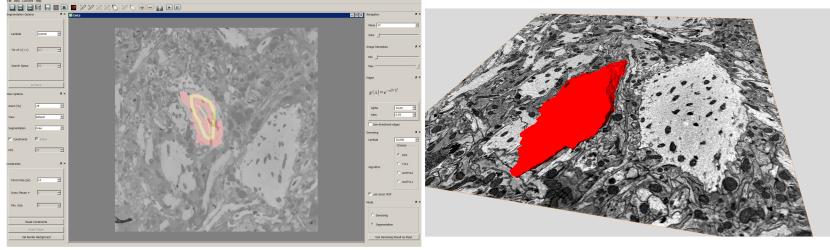
Color input is handled differently for denoising and segmentation algorithms. For edge-preserving color data denoising, we use the simplest possible vectorial TV model by splitting the three channels into separate single-channel denoising tasks, i.e., completely independently denoising the RGB channels. For segmentation, we compute image gradients from the color images, but later perform the minimization of 3 solely using a single-channel algorithm.

### 3 Experiments

We have performed several experiments to demonstrate the applicability of our software tool. We intend to demonstrate certain features of our tool during the MICCAI IMIC Workshop. Here we give an overview of three medical image segmentation tasks we have successfully used the tool for recently, which are volumetric MRI bone segmentation, neuron dendrite segmentation from microscopy image stacks, and glottis segmentation from laryngeal high speed color videos.

#### 3.1 MRI bone segmentation

To generate ground truth for a learning based bone segmentation approach, we have interactively segmented a training set of T1 weighted hand MRI volumes.



**Fig. 4.** Neuron dendrite segmentation using the interactive segmentation tool (left). 3D rendering of the segmented structure (right).

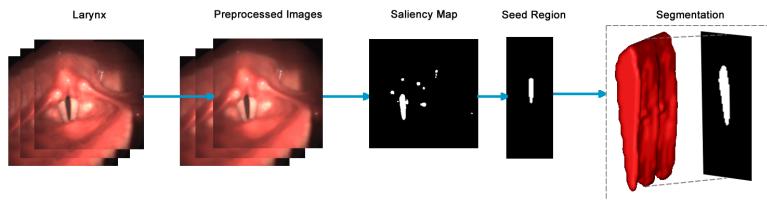
The interactive segmentation is performed by placing foreground seeds into the bones after a ROF denoising step with  $\lambda = 10$ . Small parts which grow together can easily be fixed with the background seed brush. See Fig. 3 for an illustration.

### 3.2 Neuron dendrite segmentation

Modeling the neuronal circuit is an important question in cell biology. In this application we are given a stack of SEM microscopy images of grasshopper brain samples and want to perform a segmentation of the 3D neuronal structures including their dendrites. Fig. 4 illustrates this task.

### 3.3 Segmentation of high speed glottis videos

In this application we are confronted with the need to segment the glottis from laryngeal high speed videos with the goal of detecting voice disorders from a frequency analysis of the temporal glottis vibration. Due to the large amount of frames in such high-speed videos, we have come up with an automatic method for glottis segmentation [13], that is illustrated in Fig. 4. We use the interactive segmentation tool to load this segmentation, inspect it for errors and correct them using the wTV model.



**Fig. 5.** Automatic glottis segmentation pipeline [13]. We perform an interactive segmentation on subsequent blocks of the spatiotemporal volume to correct segmentation errors.

## 4 Conclusion

In this work we have shown our software tool for performing interactive denoising and image/volume segmentation. It works on single and three-channel (RGB) input data, allowing the segmentation of images, volumes from tomographic or microscopic imaging devices, as well as videos, which are interpreted as a spatiotemporal volume. Key features of our tool are the highly efficient NVidia CUDA implementation, that allows to solve the PDEs from the energy functionals in parallel. This enables immediate user feedback, which is the key component in interactive segmentation. The proposed software will be published online as a binary Windows tool that requires a recent NVidia graphics adapter and a live demonstration will be given at the MICCAI IMIC workshop.

## References

1. Mortensen, E.N., Barrett, W.A.: Interactive Segmentation with Intelligent Scissors. *Graphical Models and Image Processing* **60**(5) (1998) 349–384
2. Boykov, Y.Y., Jolly, M.P.: Interactive Organ Segmentation using Graph Cuts. In: MICCAI. Volume LNCS 1935., Springer, Springer Verlag (2000) 276–286
3. Rother, C., Kolmogorov, V., Blake, A.: "GrabCut": Interactive foreground extraction using iterated graph cuts. *ACM Trans. Graph.* **23**(3) (2004) 309–314
4. Unger, M., Pock, T., Trobin, W., Cremers, D., Bischof, H.: TVSeg - Interactive Total Variation Based Image Segmentation. In: Proceedings British Machine Vision Conference 2008, Leeds, UK (September 2008)
5. Falcao, A.X., Bergo, F.P.G.: Interactive volume segmentation with differential image foresting transforms. *IEEE Transactions on Medical Imaging* **23**(9) (2004) 1100–1108
6. Armstrong, C.J., Price, B.L., Barrett, W.A.: Interactive segmentation of image volumes with Live Surface. *Computers & Graphics* **31** (2007) 212–229
7. Urschler, M., Bornik, A., Scheurer, E., Yen, K., Bischof, H., Schmalstieg, D.: Forensic-Case Analysis: From 3D Imaging to Interactive Visualization. *IEEE Computer Graphics and Applications* **32**(4) (2012) 79–87
8. Zhao, F., Xie, X.: An overview of interactive medical image segmentation. In: Annals of the BMVA. Volume 2013. (2013) 1–22
9. Chambolle, A., Pock, T.: A first-order primal-dual algorithm for convex problems with applications to imaging. *Journal Math Imaging and Vision* **40**(1) (2011) 120–145
10. Rudin, L., Osher, S., Fatemi, E.: Nonlinear total variation based noise removal algorithms. *Physica D: Nonlinear Phenomena* **60**(1) (1992) 259–268
11. Caselles, V., Kimmel, R., Sapiro, G.: Geodesic Active Contours. *International Journal of Computer Vision* **22**(1) (1997) 61–79
12. Bresson, X., Esedoglu, S., Vandergheynst, P., Thiran, J.P., Osher, S.: Fast global minimization of the active contour/snake model. *Journal of Mathematical Imaging and Vision* **28**(2) (2007) 151–167
13. Schenk, F., Urschler, M., Aigner, C., Roesner, I., Aichinger, P., Bischof, H.: Automatic glottis segmentation from laryngeal high-speed videos using 3d active contours. In: Proc Medical Image Understanding and Analysis. (2014) 111–116