# Fully Convolutional Mandible Segmentation on a valid Ground-Truth Dataset

Jan Egger, Birgit Pfarrkirchner, Christina Gsaxner, Lydia Lindner, Dieter Schmalstieg,
Jürgen Wallner

***Abstract*— This contribution presents the automatic segmentation of the lower jawbone (mandible) in humans' computed tomography (CT) images with the support of trained deep learning networks. CT acquisitions from the mandible frequently include radiological artifacts e.g. from metal dental restorations, ostheosynthesis materials or include trauma related free pieces of bones with missing bone contour anatomy. As a result, manual outlining these slices to generate the ground truth for evaluating segmentation algorithms lead to massive uncertainties and results in significant interphysician disagreement. Simply excluding these slices is also not the option of choice, regarding the treatment outcome. Hence, we defined strict inclusion and exclusion criteria for our datasets to avoid subjectivity or occurring bias in the ground-truth creation. Amongst others, datasets must display a complete physiological mandible without teeth. According to these data selection criteria such images are difficult to find since they originate from the clinical routine and therefore need a medical indication (such as trauma or pathologic lesions) to be provided as CT data. Furthermore, to prove the adequateness of our ground-truth, clinical experts segmented all cases twice manually, showing the great qualitative and quantitative agreement between them. Our dataset collection and the corresponding ground truth is an absolute novelty and the first serious evaluation of segmentation algorithms for the mandible.**

## I. Introduction

Deep learning [1] with neural networks is an increasingly important topic for research and economic purposes. Software giants use deep networks for the development of their latest technological gadgets. Daily examples are Facebook's face detection, Apple's speech recognition Siri or Google Translate, which all comprise deep learning algorithms [2].

The motivation of this contribution is to utilize deep learning networks for medical image processing and analysis, and create a more reliable ground-truth. In particular, the aim was to implement convolutional neural networks (CNNs) as well as to train and test them with computed tomography images from the clinical routine in order to enable an automatic segmentation of the lower jawbone. Automatic segmentations are helpful, since a manual slice-by-slice segmentation is tremendously time-consuming [3]-[10].

To train CNNs during this work, ten completely anonymized CT datasets of the human head-neck region originating from the clinical routine were provided by the Department of Oral and Maxillofacial Surgery of the Medical University of Graz in Austria. Furthermore, two clinical experts segmented manually the lower jaw of the present CT images in order to generate the ground truths, which were regarded as valid segmentations and control group [11]. The lower jawbones were segmented twice in order to enable an impartial comparison between the algorithmic segmentation and the inter-observer variability. Ten CT datasets is a small amount to train a network efficiently and to attain a good generalization ability. However, due to the data selection criteria (see A. Data Acquisition) only ten datasets could be identified from the clinical routine in the last years, but this strict inclusion and exclusion criteria allow a valid ground truth generation and further an algorithmic segmentation of the whole mandible. In contrast, all previous works we are aware of have here two significant shortcomings: either they skip "faulty" slices, e.g. showing artefacts, or they try their best to segment them, even if the meaningfulness of these (manual) segmentations are at least questionable.

Ibragimov et al. [12] presented in their work a tri-planar patch-based segmentation approach. Their goal was to segment automatically several organs of the head and neck region (inter alia the mandible) in CT datasets. For this purpose, they had 50 CT datasets at their disposal, which were manually segmented by clinical experts to produce the ground truth segmentations. Another approach to obtain an automatic segmentation is announced by Long et al. [13] in their contribution. Contrary to the previous presented work, they implemented a fully convolutional network (FCN), which produces an output of the same size as the input image. The output is, consequently, the direct pixel-wise prediction of the segmentation.

The deep learning implementations of this work comprise classification as well as segmentation networks. The idea is to mark out the images, which show parts of the lower jawbone, with a trained classification net and to provide those slices to the segmentation networks. The reason for this two-step implementation is that many CT slices occur, which don't display the anatomical region of interest. Hence, various classification and segmentation networks were implemented as well as trained and tested with the deep learning framework TensorFlow [14] and its higher level application programming interfaces (API). The results show that the automatic segmentation of the mandible works adequately for the available CT datasets.

J. Egger is with the Institute for Computer Graphics and Vision, Graz University of Technology, Inffeldgasse 16, 8010 Graz, Austria (phone: +43 316 873 5076; fax +43 316 873 5050; e-mail: egger@icg.tugraz.at).

J. Egger, B. Pfarrkirchner, C. Gsaxner, L. Lindner and J. Wallner are with the Computer Algorithms for Medicine Laboratory, Graz, Austria.

B. Pfarrkirchner, C. Gsaxner, L. Lindner and D. Schmalstieg are with the Institute for Computer Graphics and Vision, Graz University of Technology, Austria.

J. Wallner is with the Department of Oral and Maxillofacial Surgery, Medical University of Graz, Auenbruggerplatz 5/1, 8036 Graz, Austria (e-mail: j.wallner@medunigraz.at).

978-1-5386-3646-6/18/$31.00 ©2018 IEEE

## II. Methods

### A. Data Acquisition

For the segmentation process 45 CT-data sets were provided as DICOM files and collected during the clinical routine from a department of cranio-maxillofacial surgery in Austria. Only high resolution data sets (512x512) with slices not exceeding 1.0 mm with 0.25 mm pixel size and providing physiological, complete mandibular bone structures without teeth were included in the selection process. Further, no difference was made between atrophic and nonatrophic mandibular bones - both were included during the selection process. However, incomplete data sets consisting of mandibular structures altered by iatrogenic or pathological factors or fractured mandibles as well as data sets showing ostheosynthesis materials in the lower jaw were excluded in this trial. All data sets were acquired within a twelve month period (between 2013 and 2016). According to the inclusion criteria 20 CT-data sets were selected, 25 were excluded during the selection process in the clinical routine out of diagnosis and treatment reasons. From the 20 CT data sets, ten data sets (n=10, 6 male, 4 female) were further selected in a randomization process performed by a computer program (Randomizer©; https://www.randomizer.at; randomization for clinical and non-clinical trials), to form an experimental segmentation group. The control group consisted of objective created bone structure volumes of the lower jaw according to the selected ten data sets (ground truth). The Institution's Ethical Review Board approved all experimental procedures involving human subjects.

### B. Classification of the CT slices

The presented deep learning implementations of this section comprise a classification of the CT slices. The idea was to achieve a self-acting decision whether the mandible appears in an image or not, as many images occur in the CT datasets, that don't show the lower jawbone. These CT slices should be eliminated with a trained classification network and consequently, the segmentation network was just trained with images that show parts of the lower jaw. Ibragimov and Xing [12] removed in their work also CT slices, that don't include the mandible, from training and testing neural networks. They applied, however, geometrical methods for slice exclusion. The classification networks were implemented in a Python environment as well as the deep learning toolkit TensorFlow and its high-level API TFLearn [15] was utilized. The CNN, which offered the best performance, was selected for the further segmentation task.

Before a training of classification networks could be accomplished, it was necessary to know the label of every image. These labels indicate which class an image is belonging to. In the course of this contribution, two classes exist: There is the case that the lower jawbone appears in a CT slice and alternatively, that the mandible doesn't occur. The labels of the images were extracted from the ground truth segmentations. If a CT slice contains the mandible, its corresponding mask encompasses white pixels. Otherwise, the mask exhibits only pixel values of zero. Thus, it was possible to infer the classification label of a CT slice from its mask's pixel values.

During this work, classification networks with various net topologies were trained with four different sized datasets. Each dataset contained a diverse number of images, as there were different augmentation methods applied [16]. The first image set involved the initial CT images (1680 slices), the second one was enlarged with noisy images (6720 slices) and the third one with affine transformed ones (13440 slices). Dataset four covered both data augmentation types (18480 slices). The affine transformations were applied separately from each other (for each slice separately).

To produce meaningful results with a trained network, it was required to figure out optimal training and network parameters. Good classification accuracies were achieved with a max-pooling filter size of five, the learning rate was set to 0.00001 and the number of epochs per launched training was stated as 20, whereby each network was trained four times. Comparing all trained models according to their achieved loss values and accuracies, we marked out that the network with the topology of six convolutional and six maxpooling layers led to the best results. This network configuration was the deepest trained model. Moreover, this CNN comprises a fully connected layer with 1024 nodes and dropout was applied to this layer type with a rate of 0.8%. The number of generated feature maps was set to 64 for the second convolutional layer and to 32 for the remaining ones. The output layer exhibited two nodes, since each output unit delivers the class probability for an input image. Moreover, the convolutional filter size was set to seven and the images were down-sampled to a size of 50x50. Besides that, the best performing CNN was trained with the largest dataset four.

After testing slices of a dataset with the classification net, the minimum and the maximum slice displaying the mandibular were established. These two slices build the limitation of the images that are utilized for the succeeding segmentations.

### C. Segmentation of the CT slices

The implementation of the deep networks was conducted with TensorFlow and its high-level API TF-Slim [17]. Again, the Python interface of TensorFlow was utilized. The realized segmentation method follows the upsampling principle presented by Long et al. [13] in their Fully Convolutional Networks for Semantic Segmentation contribution as well as the contribution of Pakhomov et al. [18]. As already outlined, Long et al. [13] recommended a three-step training principle of a fully convolutional network. Figure 1 illustrates the workflow of the model implementations.

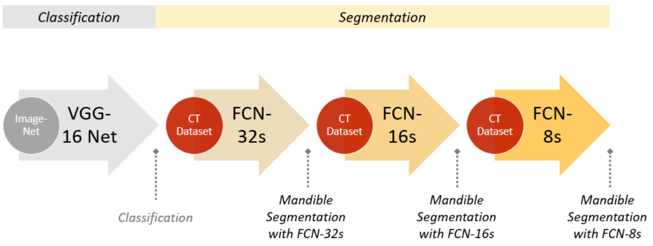


Figure 1. Workflow of the segmentation network implementations. The classification part was provided by the TF-Slim library, whereas the segmentation part was trained with the CT datasets during this work.

The first segmentation network is the FCN-32s, which is an adjustment of the VGG-16 net. The VGG-16 model [19] is originally trained for a classification with the ImageNet dataset of the ILSVRC 2014 challenge [20], whereby the weights for FCN-32s initialization were provided by Pakhomov et al. [18]. Moreover, the second segmentation net is the FCN-16s, which assumes weights of the trained FCN-32s model. The third and last one is the FCN-8s, which is again initialized with the weights of the previous network.

The first segmentation network, termed as FCN-32s, was implemented with a modification of the TF-Slim VGG-16 model definition. After the application of input images to the VGG-16 model, the matrices of the feature maps have a side length of 1/32 of the original input. As a result, to gain a segmentation with the initial image size, the downsampled feature maps have to be upsampled with a factor of 32. Therefore, the FCN-32s net topology adds one upsampling layer to the VGG-16 model, whereby this new layer is trained from scratch (Figure 1).

The FCN-16s network accomplishes upsampling with two additional layers, whereby the first upsampling is conducted with a factor of two. Contrary to the FCN-32s, the FCN-16s includes information from the down-sampling path of the VGG-16 architecture. For this purpose, the output of the max-pooling layer four is involved in the upsampling process by combining this information with the output of the first upsampling layer. The second and last upsampling step is conducted with a factor of 16 in order to attain a prediction with the original input size (see Figure 1).

Apart from that, the FCN-8s model achieves upsampling with three additional layers. The first upsampling is conducted in the same manner as it was achieved with the FCN-16s network. Hence, the output of the VGG-16 model is delivered as an input to the first upsampling layer, which executes a resize with a factor of two. Moreover, the involvement of the fourth max-pooling layer is achieved in the same way as it was done for the FCN-16s network. The resize factor of the second upsampling layer exhibits, however, a value of two. Furthermore, the information of maxpooling layer three of the VGG-16 architecture is combined with the output of the second upsampling layer. Finally, to gain the original matrix size for the final segmentation prediction, an upsampling factor of eight is essential for the third upsampling step (see Figure 1).

The presented segmentation architectures were trained with four different datasets. Two of the training sets (I and II) contain the original images, whereby the images of the first dataset are down-sampled to a size of 256x256. The other two datasets (III and IV) cover the original images and also artificially generated ones. Again, one dataset comprises the original sized images (IV), while the other one contains down-sampled CT slices (III). It has to be noticed that only slices, which show parts of the lower jawbone, were used to train the segmentation networks. Hence, the number of available training images reduces compared to the training data of the classification networks. The extraction of the slices, that don't comprise the mandible, was executed manually via the expert segmentations.

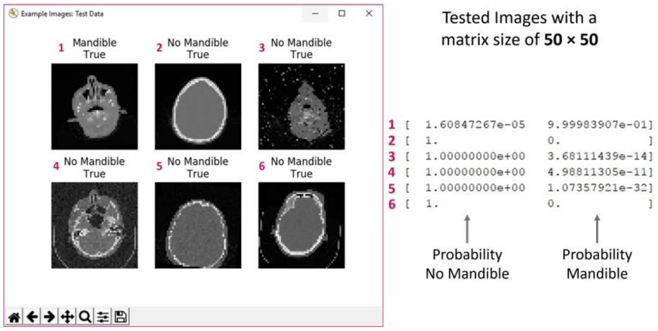
For training the segmentation networks, one image mask pair was used for the computations of the weight updates. On top of that, the number of epochs was set to ten for the three implemented networks, whereas the learning rate changed with the various topologies. The learning rate of the FCN-32s net had a value of 0.0001, while the rates of the FCN-16s and FCN-8s networks were set to 0.000001 and 0.0000001. As there was such a small amount of data available, it was also achievable to train the segmentation networks on a CPU. The consecutive training of the FCN-32s, the FCN-16s and the FCN-8s models took in total about one day and a half for the smaller sized datasets (I and II), while training with the datasets III and IV lasted about five days.

To evaluate the predicted segmentation results, the Dice scores were calculated for each patient dataset. Therefore, the image processing platform MeVisLab was used [21]-[25]. As expected, the segmentation network, which was trained with the largest dataset offered the best Dice coefficients. Moreover, the FCN-8s showed a better performance than the FCN-16s and the FCN-32s models.

## III. RESULTS

Figure 2 displays classified images (50x50) and their predicted probabilities. The class predictions were accomplished with the best performing classification model. This trained network delivered an accuracy of one for the training dataset, whilst the test accuracy had a value of 0.9877.

Apart from that, two clinical experts generated ground truth contours for supervised training, whereby the ground truths of clinical expert A were just utilized for training. Nevertheless, the inter-observer variability was calculated between those manual segmentations. Therefore, the Dice coefficients were computed for each patient's dataset. Averaging these values shows that the mean Dice score has a value of 0.9362 and the standard deviation is 0.0098. Additionally, the FCN-8s net, which was trained with the largest dataset, delivers a mean Dice coefficient of 0.9203 and a standard deviation of 0.0140 for the training dataset. Finally, the Dice scores were computed and averaged for test images. The mean Dice coefficient showed a value of 0.8964 and a standard deviation of 0.0169. Hence, the segmentation metrics of the training images are a bit worse than the inter-observer variability, whilst the test metrics decrease a bit more.



```
1 [  1.60847267e-05   9.99983907e-01]
2 [  1.               0.            ]
3 [  1.00000000e+00   3.68111439e-14]
4 [  1.00000000e+00   4.98811305e-11]
5 [  1.00000000e+00   1.07357921e-32]
6 [  1.               0.            ]
```

Figure 2. Test images (50x50) and their predicted classes. The network exhibited a topology of six convolutional and six max-pooling layers. Training was accomplished with the largest sized dataset.

Figure 3 illustrates a CT slice, its ground truth and predicted segmentations, which were generated with the FCN-32s, the FCN-16s and the FCN-8s topology. Beyond that, Figure 4 shows another CT slice, its ground truth and the predicted probabilities of the three network architectures. The networks used for those predictions were trained with the largest dataset and the original image sizes. Both figures indicate that the predictions improve with the involvement of information of the VGG-16 model. The segmentations of the FCN-32s architecture seem to be awkward, whilst the FCN-8s predictions are smoother.

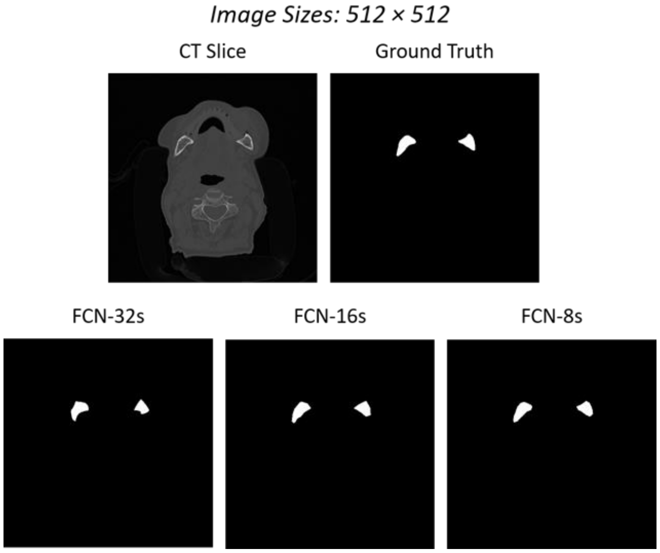


Figure 3. Comparison of a CT slice (512x512), its ground truth and the predicted segmentations. The segmentations were forecasted with the FCN-32s, the FCN-16s and the FCN-8s models, which were trained with dataset IV.

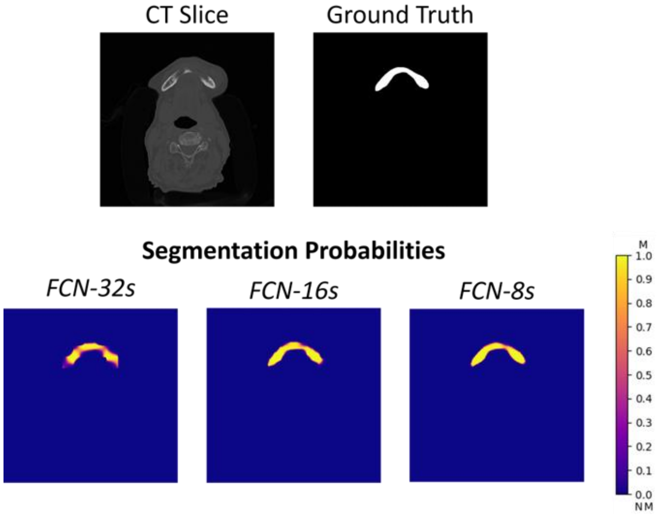


Figure 4. Depiction of a CT slice, its ground truth and the predicted probability maps. The maps were forecasted with the networks trained with dataset IV. The brighter the voxels, the more likely they are part of the mandible (M), whilst the blue color implies that there is probably no mandible (NM) appearing.

## IV. Conclusion

To bring up to mind, a MeVisLab [26]-[29] network and a macro module were generated to process and enlarge the head-neck CT datasets during this contribution. Moreover, the ultimate objective was to implement deep networks, which permit an automatic segmentation of the mandible. Therefore, classification networks were trained in order to distinguish whether a slice comprises the lower jawbone or not and consequently, segmentation networks computed the algorithmic demarcations within these slices. All networks were trained and tested with images exported by the MeVisLab [30]-[32] realizations.

On the whole, the most essential problem, which must be solved for additional deep learning implementations in medicine, is the lack of available images. If there are databases utilized, which comprise a huge amount of images, it must be kept in mind that the ground truths must be created manually by experts and must be proven for their validity for a supervised training. To overcome this problem the utilization of overlaid images may be an option (e.g. registration of nuclear medical images - like PET - on CT or MR images) [33]. For instance, cancerous tissue might be segmented in CT slices, whereby the nuclear medical information corresponds to the ground truths, as the tracers accumulate in tumors. If the problems of the lack of available data are resolved, more detailed investigations may be feasible in the field of network architectures or parallel GPU training. To conclude, the implemented networks of this contribution were an explanatory step for the application of deep models in the medical domain and for the first time on a data collection with a valid ground truth, but for a usage in clinical routine a training and also a testing with a large number of images is essential.

## Acknowledgment

The work received funding from BioTechMed-Graz in Austria, the 6th Call of the Initial Funding Program from the F&T-Haus at TU Graz, the TU Graz Lead Project (Mechanics, Modeling and Simulation of Aortic Dissection) and the Austrian Science Fund (FWF) KLI 678-B31: “enFaced: Virtual and Augmented Reality Training and Navigation Module for 3D-Printed Facial Defect Reconstructions” (Principal Investigators: Jan Egger & Jürgen Wallner).

## References

[1] Y. LeCun, Y. Bengio, and G. Hinton, “Deep learning,” Nature, 521(7553):436-444, May 2015.

[2] N. Jones, “Computer science: The learning machines,” Nature, 505(7482):146-148, Jan. 2014.

[3] P. F. Christ et al., “Automatic Liver and Lesion Segmentation in CT Using Cascaded Fully Convolutional Neural Networks and 3D Conditional Random Fields,” MICCAI, pp. 415-423, 2016.

[4] D. Zukic, et al., “Segmentation of Vertebral Bodies in MR Images,” Vision, Modeling, and Visualization, 135-142, 2012.

[5] D. Zukić, et al. “Robust Detection and Segmentation for Diagnosis of Vertebral Diseases using Routine MR Images,” Computer Graphics Forum, Volume 33(6), 190-204, 2014.

[6] R. Schwarzenberg, et al., “A Cube-Based Approach to Segment Vertebrae in MRI-Acquisitions,” BVM, Springer, 69-74, 2013.

[7] R. D. Renapurkar, et al., “Aortic volume as an indicator of disease progression in patients with untreated infrarenal abdominal aneurysm,” Eur J Radiol., 81(2):e87-93, 2011.

[8] J. Egger, “Refinement-Cut: User-Guided Segmentation Algorithm for Translational Science,” Sci. Rep., 4, 5164, 2014.

[9] J. Egger, et al., “Manual Refinement System for Graph-Based Segmentation Results in the Medical Domain” J Med Syst. 36, 2829-39, 2012.

[10] J. Egger, et al., “GBM Volumetry using the 3D Slicer Medical Image Computing Platform,” Sci Rep. 3, 1364, 2013.

[11] J. Egger, et al., “Algorithmic evaluation of lower jawbone segmentations,” SPIE Medical Imaging Conference, Paper 10137-11, 2017.

[12] B. Ibragimov and L. Xing, “Segmentation of organs-at-risks in head and neck CT images using convolutional neural networks,” Medical Physics, 44(2):547-557, 2017.

[13] J. Long, E. Shelhamer, and T. Darrell, “Fully Convolutional Networks for Semantic Segmentation,” CVPR, June 2015.

[14] M. Abadi et al., “TensorFlow: Large-Scale Machine Learning on Heterogeneous Distributed Systems,” CoRR, abs/1603.04467, 2016.

[15] TFlearn. TFLearn: Deep learning library featuring a higherlevel API for TensorFlow. Available: http://tflearn.org/, Aug. 2017. (Last access 29.09.2017).

[16] B. Pfarrkirchner et al., “Lower jawbone data generation for deep learning tools under MeVisLab,” SPIE Medical Imaging Conference, Paper 10578-96, 2018.

[17] S. Guadarrama and N. Silberman, “TensorFlow-Slim,” Available: https://github.com/tensorflow/tensorflow/tree/master/tensorflow/contrib/slim, 2017. (Last access 27.09.2017).

[18] D. Pakhomov et al., “Deep Residual Learning for Instrument Segmentation in Robotic Surgery,” arXiv preprint arXiv:1703.08580, 2017.

[19] K. Simonyan and A. Zisserman., “Very Deep Convolutional Networks for Large-Scale Image Recognition,” CoRR, abs/1409.1556, 2014.

[20] J. Deng et al., “ImageNet: A large-scale hierarchical image database,” CVPR, pages 248-255, June 2009.

[21] J. Egger et al., “Integration of the OpenIGTLink Network Protocol for Image-Guided Therapy with the Medical Platform MeVisLab,” Int J Med Robot. 8(3):282-90. 2012.

[22] S. Gunacker et al., “Multi-threaded integration of HTC-Vive and MeVisLab,” SPIE Medical Imaging Conference, Paper 10579-51, 2018.

[23] J. Egger, et al., “Fast Self-Collision Detection and Simulation of Bifurcated Stents to Treat Abdominal Aortic Aneurysms (AAA),” 29th Annual International Conference of the IEEE Engineering in Medicine and Biology Society, Lyon, France, 6231-6234, IEEE Press, 2007.

[24] J. Egger, et al., “Modeling and Visualization Techniques for Virtual Stenting of Aneurysms and Stenoses,” Comput Med Imaging Graph., 36(3):183-203, 2012.

[25] J. Egger et al., “HTC Vive MeVisLab integration via OpenVR for medical applications,” PLoS ONE 12(3): e0173972, 2017.

[26] M. H. A. Bauer, et al., “Boundary Estimation of Fiber Bundles derived from Diffusion Tensor Images,” Int J CARS, 6(1):1-11, 2011.

[27] J. Lu, et al., “Detection and Visualization of Endoleaks in CT Data for Monitoring of Thoracic and Abdominal Aortic Aneurysm Stents,” SPIE Med Img 6918, 69181F(1-7), 2008.

[28] K. Greiner, et al., “Segmentation of Aortic Aneurysms in CTA Images with the Statistic Approach of the Active Appearance Models,” Proceedings of Bildverarbeitung für die Medizin (BVM), Berlin, Germany, Springer Press, pp. 51-55, 2008.

[29] J. Egger, et al., “Preoperative Measurement of Aneurysms and Stenosis and Stent-Simulation for Endovascular Treatment,” ISBI, 392-5, 2007.

[30] D. Kuhnt, et al., “Fiber tractography based on diffusion tensor imaging (DTI) compared with High Angular Resolution Diffusion Imaging (HARDI) with compressed sensing (CS) – initial experience and clinical impact,” Neurosurgery, Volume 72, pp. A165-A175, 2013.

[31] J. Egger, et al., “Interactive Volumetry of Liver Ablation Zones,” Sci. Rep. 5, 15373, 2015.

[32] J. Egger, et al., “Simulation of Bifurcated Stent Grafts to Treat Abdominal Aortic Aneurysms (AAA),” SPIE, 6509, 1-6, 2007.

[33] C. Gsaxner et al., “Exploit 18F-FDG enhanced urinary bladder in PET data for deep learning ground truth generation in CT scans,” SPIE Medical Imaging Conference, Paper 10578-70, 2018.