

SwifTree: Interactive Extraction of 3D Trees Supporting Gaming and Crowdsourcing

Mian Huang^(✉) and Ghassan Hamarneh

Medical Image Analysis Lab, School of Computing Science,
Simon Fraser University, Burnaby, BC, Canada
{mianh,hamarneh}@sfu.ca

Abstract. Analysis of vascular and airway trees of circulatory and respiratory systems is important for many clinical applications. Automatic segmentation of these tree-like structures from 3D data remains an open problem due to their complex branching patterns, geometrical diversity, and pathology. On the other hand, it is challenging to design intuitive interactive methods that are practical to use in 3D for trees with tens or hundreds of branches. We propose SwifTree, an interactive software for tree extraction that supports crowdsourcing and gamification. Our experiments demonstrate that: (i) aggregating the results of multiple SwifTree crowdsourced sessions achieves more accurate segmentation; (ii) using the proposed game-mode reduces time needed to achieve a pre-set tree segmentation accuracy; and (iii) SwifTree outperforms automatic segmentation methods especially with respect to noise robustness.

1 Introduction

Analysis of anatomical branching trees in the human body (i.e. vascular and airway trees of circulatory and respiratory systems) is important for a wide range of application (e.g., [22,24]). There are numerous methods for segmenting tree-like structures from 2D and 3D images, which may be generally classified into automatic (e.g., [5,15]) and interactive (e.g., [2,8,12,20,21,26]). Fully automatic tree segmentation methods are not yet completely accurate and reliable as they are often sensitive to parameters setting, are prone to leaking into nearby structures or to missing true bifurcating branches [15]. On the other hand, among interactive methods, optimal path techniques are commonly employed, which require the definition of start and end points (seeds) for each target branch (e.g., vessel) [8,26]. Other works proposed manual correction techniques to be applied after automatic segmentation [20,27]. Generally, interactive methods are hard to design and utilizing them for complex branching 3D trees with tens or hundreds of branches, which is not uncommon, is impractical.

There is a growing need for large numbers of segmented 3D imaging datasets for training machine learning systems and for validating newly proposed methods, however, there is a scarcity of segmented complex 3D trees. This work, which leverages gamification and crowdsourcing, is a first step towards enabling the collection of large numbers of segmented anatomical trees.

The objective of gamification is to transform a mundane task into an immersive and engaging experience. Gamification has been leveraged in many ways, e.g., improving work productivity, patient rehabilitation, education and enhancing cognitive skills, etc. Crowdsourcing, on the other hand, provides a possible source of labelled (so called ground truth) data by leveraging humans’ cognitive abilities and intelligence. Crowdsourcing is increasing in popularity and target applications, e.g., missing person search, disaster management, astronomy, and rehabilitation.

Table 1. Comparison of closest works. The meanings of the column headings are as follows. Crowd: method leverages crowdsourcing; Game: offers a “game” mode; MIA: designed for medical image analysis; 3D: handles 3D data; View: provides a view within the 3D volume; Control: controls the viewing position and angle; Tree: supports extracting branching tree-like structures; Skeleton: extracts centerline; Hierarchy: generates abstract representation of tree hierarchy.

Work	Crowd	Game	MIA	3D	View	Control	Tree	Skeleton	Hierarchy
Donath et al. [9]	✓								
Albarqouni et al. [3]	✓		✓						
Maier-Hein et al. [19]	✓		✓						
Chavez-Aragon et al. [6]	✓		✓						
Maier et al. [18]	✓		✓						
Luengo et al. [17]	✓	✓	✓						
Albarqouni et al. [4]	✓	✓	✓						
Hennersperger et al. [13]	✓	✓	✓						
Sommer et al. [23]			✓	✓			✓		
Poon et al. [21]			✓				✓	✓	
Vickerman et al. [26]			✓				✓	✓	✓
Abeyasinghe et al. [2]			✓	✓			✓	✓	
Yu et al. [27]			✓	✓			✓	✓	✓
Marks et al. [20]			✓	✓			✓	✓	✓
Straka et al. [25]			✓	✓	✓		✓		
Abdoulaev et al. [1]			✓	✓	✓		✓		
Edmond et al. [10]			✓	✓	✓		✓	✓	
Coburn et al. [7]	✓	✓	✓	✓	✓	✓			
Heng et al. [12]			✓	✓	✓		✓	✓	✓
Diepenbrock et al. [8]			✓	✓	✓		✓	✓	✓
Proposed SwifTree	✓	✓	✓	✓	✓	✓	✓	✓	✓

Table 1 contrasts our proposed work with some of the most related literature. Although there has been several works that deployed gamification and/or crowdsourcing for medical image analysis, to the best of our knowledge, this is the first work to utilize gamification and crowdsourcing for vascular/airway tree extraction from 3D images. We argue that without the user confirming the segmentation everywhere along all branches of the tree, there is significant possibility of erroneously segmented regions. Therefore, we set out to develop SwifTree, a tool

that allows the user to quickly and intuitively traverse and extract the anatomical tree in its entirety in a 3D volume, while supporting and leveraging gamification and crowdsourcing. Briefly, using SwifTree, the operator steers their way down along the bifurcating tree branches using intuitive controls. To address the mundane and time-consuming nature of delineating many branches, SwifTree employ gamification concepts. Finally, leveraging crowdsourcing, SwifTree allows multiple users to cooperate and generate multiple results that are then aggregated to produce the final extracted tree.

2 Method

Overview: After a 3D image is loaded into SwifTree, the image is processed to extract image features for controlling the properties of glyphs placed in a 3D scene to provide helpful cues to the user. In order to provide the user with multiple alternative views of the 3D scene, multiple virtual cameras at suitable vantage points are used. Each user is provided with controls (e.g. keyboard shortcuts) to facilitate navigating through the tree within the 3D image. In the crowdsourcing setup, the users travel virtually through the tree branches to construct trees in, both, a 3D spatial layout and in an abstract graph tree representation (an example is shown in Fig. 1). The results are aggregated to yield the final extracted tree and graph. The details follow.

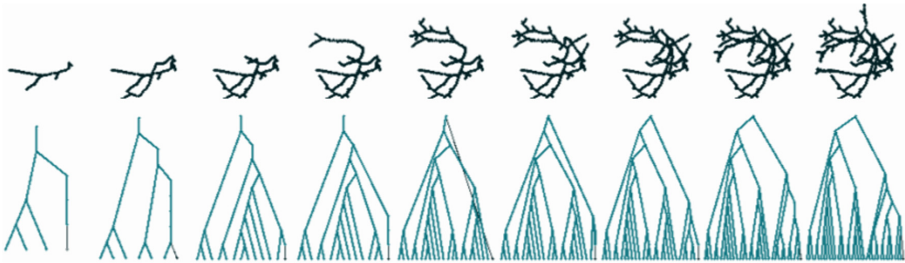


Fig. 1. Illustration of the sequence of steps which SwifTree uses to extract a 3D tree. Top: 3D spatial domain; bottom: corresponding abstract tree graph.

Image processing and glyph visualization: Figure 2 shows a schematic of the components that comprise a SwifTree 3D scene. The user interrogates different locations within the volume via a 3D polyhedral cursor. In a first attempt to visualize the image data for the user, we found that surface rendering (via marching cubes) and volume rendering (e.g. via ray casting) of the image data to overcrowd the scene. Instead we used slices and glyphs as described next. A grayscale oblique slice, cutting through the 3D volume, is rendered facing the user's viewing direction so that the slice would depict the cross-section of a branch as a single bright disk. As the user moves towards a bifurcation, the disk gradually splits into two, one for each child branch. We also render gradient glyphs

based on the 3D image intensity gradient to highlight an estimate of the surface boundary surrounding the tree branches. To highlight the voxels in the interior to tree branches, we use tree-core glyphs calculated using the Frangi filter [11]. We experimented with different glyph densities (i.e. at every voxel or not), opacity values, sizes and shapes, and found the following settings to provide useful cues with minimal clutter: the size of each glyph was close to the size of a single voxel; the glyphs were rendered only at voxels with a strong response (i.e. gradient magnitude and tubularness surpassed an empirically-set threshold); and the opacity of a glyph was set proportional to the response magnitude. 3D glyphs were used for the tree-core glyphs but, for the gradient glyphs, flat 2D polygons with their normals pointing along the gradient direction were used in order to visually capture the local edges. Additionally, two virtual cameras are added to the scene: one camera provides a first-person local view whereas the other displays a more global bird's-eye view.

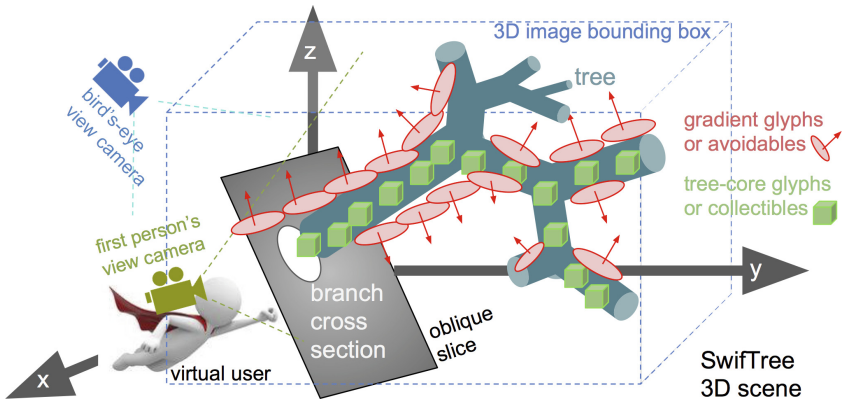


Fig. 2. Elements of SwifTree 3D scene (see text).

Navigation and movement: The aforementioned 3D cursor can be moved and rotated interactively by the user (move-forward, rotate-left, etc.). Additionally, once the user encounters a bifurcation (by observing the branch cross-section splitting), they press a key to *push* the current state parameters (i.e. location and camera viewpoints) into a *bifurcation stack*. After the user traverses one of the child branches (and optionally the grandchild branches), they *pop* the state parameters, to move the cursor and cameras back up the tree hierarchy to a previously-identified bifurcation location, so that the other child branches can be explored. Note that a trail of glyphs is left along the path explored by the user in order to ensure that the user does not explore the same branch twice.

Interactive and game mode: In SwifTree’s game-mode, the cursor is an avatar that possesses a velocity controlled by the player. The player navigates the 3D volume by ‘flying’ through branches and identifying bifurcation locations using

game-like controls (e.g., speed up, slow down, turn left). Also in game mode, the tree-core glyphs are set to be *collectibles*, i.e., as the user’s cursor passes over these glyphs, they are collected and hidden with an accompanying sound effect and a score increment. The gradient glyphs, on the other hand, are *avoidables* that reduce the score, since they represent branch boundaries that should not be crossed. In SwifTree’s non-game interactive mode, the user’s cursor can be seen as an inertia-less paintbrush manipulated by the user.

Crowdsourcing and aggregation: We recruit multiple users or players to carry out a tree (or part of the tree) extraction session. The collected tree branches for the same image across all sessions are first unioned together and then a 3D spherical kernel is used to perform morphological closing. Then a medial axis transform is applied to extract the tree skeleton and network analysis is performed to create the abstract graph tree representation [14].

Implementation details: We used MATLAB (R2015b) to test several visualization and interaction mechanisms. Then we ported SwifTree to: (i) the cross-platform game engine Unity3D (unity3d.com) and (ii) an online cross-browser version using JavaScript (v6.0) and the WebGL-based 3D graphics library Three.js (r83) (threejs.org), with PHP and MySQL to automatically collect the tree segmentation data generated by the users.

3 Results

Data: In-silico phantoms, physical phantom, and real images were used in our experiments. Refer to Fig. 3 for details.



Fig. 3. Datasets: (a–c) In-silico phantoms: Y-Junc ($60 \times 60 \times 60$ voxels; 1 mm isotropic voxel), Helix ($50 \times 50 \times 100$; 1 mm isotropic), and VasuSynth ($101 \times 101 \times 101$; 1 mm isotropic); (d) Physical phantom ($168 \times 168 \times 159$; 1 mm isotropic); (e) Renal MRA ($576 \times 448 \times 72$; $0.625 \times 0.625 \times 1.4 \text{ mm}^3$); (f) Brain CTA ($352 \times 448 \times 176$; $0.5134 \times 0.5134 \times 0.8 \text{ mm}^3$); (g) Airways in CT ($512 \times 512 \times 587$; $0.5859 \times 0.5859 \times 0.6 \text{ mm}^3$).

Supplementary material: The reader is referred to a simplified web-based version of SwifTree at <http://swiftree-org.stackstaging.com> and to the supplementary video <https://youtu.be/AReIFQc47H4>.

Evaluation criteria: We adopt the following criteria as described by Lo et al. [16]: branch count (BC); branches detected (BD); tree length (TL); tree length detected (TLD); leakage count (LC); and false positive rate (FPR).

Table 2. Accuracy of tree extraction by ITK-Snap, Gorgon and SwiftTree. Highest accuracy in bold.

Data	Y-Junc			Helix			VascuSynth			Phantom			Kidney			Brain			Airway		
Tool	ITK-Snap	Gorgon	SwiftTree	ITK-Snap	Gorgon	SwiftTree	ITK-Snap	Gorgon	SwiftTree	ITK-Snap	Gorgon	SwiftTree	ITK-Snap	Gorgon	SwiftTree	ITK-Snap	Gorgon	SwiftTree	ITK-Snap	Gorgon	SwiftTree
BC	3	1	3	3	1	3	58	27	87	47	28	52	13	5	21	30	†	82	57	†	151
BD(%)	100	33	100	100	33	100	52	24	79	72	43	80	56	21	91	24	†	65	19	†	51
TLD(cm)	4	1	5	15	1	22	75	40	99	90	56	84	40	19	47	34	†	64	28	†	91
TLD(%)	86	5	90	51	1	75	53	28	70	77	48	72	55	27	66	30	†	56	17	†	55
LC	1	2	1	35	1	5	273	85	152	45	98	27	57	9	1	82	†	144	81	†	284
FPR(%)	2	85	1	37	97	1	59	72	14	9	43	4	42	79	5	19	†	12	11	†	19

†: software froze and could not handle the complex tree.

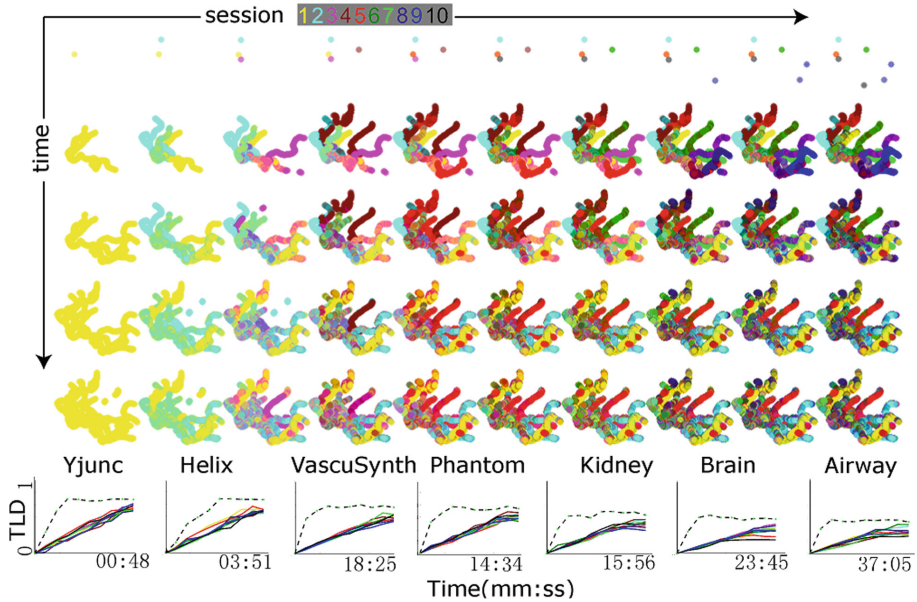


Fig. 4. Benefits of crowdsourcing. Top: The temporal progress of each of 10 sessions running SwiftTree on the Brain dataset. As time advances and more sessions are included, the aggregated tree becomes more accurate and complete. Bottom: Plots of TLD vs time, for all data sets. Each solid colored curve corresponds to one tree extraction session. The black dashed curve, with better tree detection (i.e. higher than other curves), corresponds to the aggregated tree from all 10 sessions.

Tree extraction accuracy: Table 2 compares SwiftTree to the ITK-Snap (itksnap.org) and Gorgon (gorgon.wustl.edu) tools. In ITK-Snap the user had to visit different slices to annotate pixels as tree branches, whereas in Gorgon, the user selected the end points of branches. We see that SwiftTree gives the highest BD accuracy for all datasets, the highest TLD for all datasets except Phantom, and the lowest FPR for all datasets except Airway.

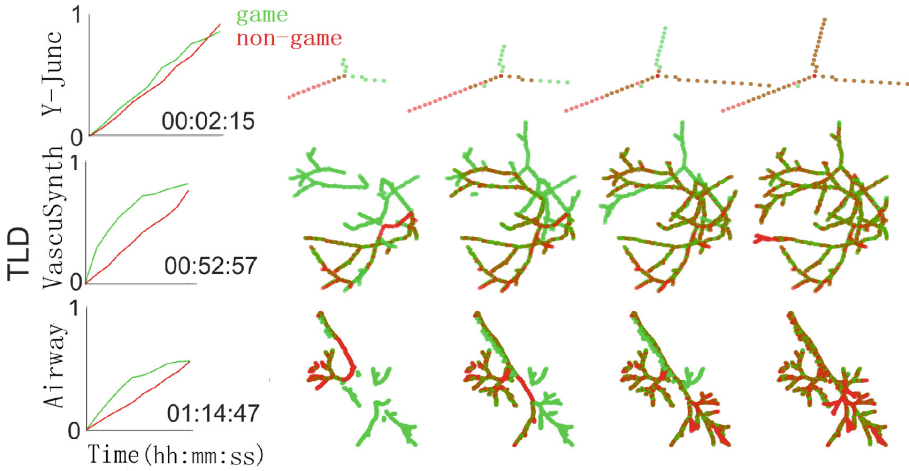


Fig. 5. Benefit of gamification. Results on 3 dataset: Y-Junc (top row); VascuSynth (middle); and Airway (bottom). Left: TLD vs time for game-mode (green) and interactive (non-game) mode (red). Right: Progress of tree extraction shown at 4 instants. Game-mode sessions extract more branches quicker than non-game mode. (Color figure online)

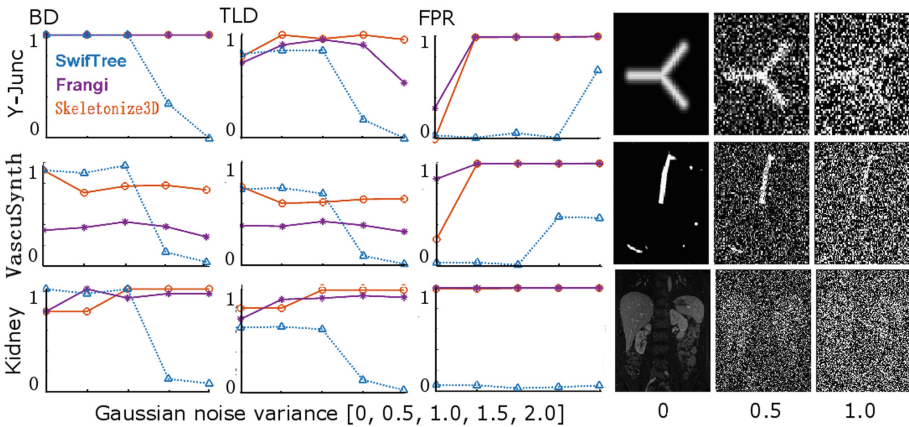


Fig. 6. Robustness to noise. Left: Comparison of Frangi filter, ImageJ Skeletonize3D and SwifTree in terms of robustness to noise. BD, TLD, and FPR are reported for the 3 methods across 3 datasets: Y-Junc (top), VascuSynth (middle) and Kidney (bottom). Right: Sample slices from each dataset at selected noise levels for illustration.

Benefit of crowdsourcing: We collected the results from 10 tree extraction sessions for each dataset using SwifTree (i.e., 70 sessions). The results are aggregated to obtain a single tree per dataset. As can be seen in Fig. 5, the tree aggregated from all participating sessions gives a more complete tree than any of the trees from the individual sessions. Also, the aggregated tree has the

highest tree length detected with the highest initial slope (i.e. fastest increase). A small dip can be seen in the TLD of the aggregated tree due to false positive branches from some sessions.

Benefit of gamification: Figure 4 shows that enabling SwifTree game-mode features (i.e. velocity, sound effects, score, collectibles, and avoidables) reduces the time needed to reconstruct a pre-set tree compared to the non-game mode.

Robustness to noise: In Fig. 6, we compare SwifTree’s results to those obtained by Frangi filter and ImageJ Skeletonize3D plug-in under different levels of Gaussian noise. We see that Frangi filter and Skeletonize3D report high detection rates of branches and trees (top and middle rows). However, they suffer from a high number of false positives (bottom row). SwifTree’s false positive rate is much lower.

4 Conclusion

We proposed SwifTree, a novel tool for extracting tree-like structures from 3D images. We showed that by leveraging gamification and crowdsourcing, SwifTree can achieve more accurate results faster and is more robust to noise than traditional segmentation tools. The next phase of our work involves releasing SwifTree publicly as a “Human Intelligence Task” (HIT) on the established crowdsourcing platform Amazon Mechanical Turk, then analyzing the results collected from a large scale study involving hundreds of workers or “Turkers”. There are several directions to explore that can improve the performance of the tool, such as more elaborate game design (e.g. improved visualization, sound, scoring system, and game-levels); an aggregation approach that gives higher weights to more expert users; detecting branch thickness; as well as performing large-scale user studies.

References

1. Abdoulaev, G., Cadeddu, S., Delussu, G., Donizelli, M., Formaggia, L., Giachetti, A., Gobetti, E., Leone, A., Manzi, C., Pili, P., et al.: ViVa: the virtual vascular project. *IEEE Trans. Inform. Technol. Biomed.* **2**(4), 268–274 (1998)
2. Abeyasinghe, S.S., Ju, T.: Interactive skeletonization of intensity volumes. *Vis. Comput.* **25**(5–7), 627–635 (2009)
3. Albarqouni, S., Baur, C., Achilles, F., Belagiannis, V., Demirci, S., Navab, N.: Aggnet: deep learning from crowds for mitosis detection in breast cancer histology images. *IEEE Trans. Med. Imaging* **35**(5), 1313–1321 (2016)
4. Albarqouni, S., Matl, S., Baust, M., Navab, N., Demirci, S.: Playsourcing: a novel concept for knowledge creation in biomedical research. In: Carneiro, G., et al. (eds.) LABELS/DLMIA -2016. LNCS, vol. 10008, pp. 269–277. Springer, Cham (2016). doi:[10.1007/978-3-319-46976-8_28](https://doi.org/10.1007/978-3-319-46976-8_28)
5. Cetin, S., Demir, A., Yezzi, A., Degertekin, M., Unal, G.: Vessel tractography using an intensity based tensor model with branch detection. *TMI* **32**(2), 348–363 (2013)
6. Chávez-Aragón, A., Lee, W.-S., Vyas, A.: A crowdsourcing web platform-hip joint segmentation by non-expert contributors. In: MeMeA, pp. 350–354. IEEE (2013)

7. Coburn, C.: Play to cure: genes in space. *Lancet Oncol.* **15**(7), 688 (2014)
8. Diepenbrock, S., Ropinski, T.: From imprecise user input to precise vessel segmentations. In: VCBM. EG, pp. 65–72 (2012)
9. Donath, A., Kondermann, D.: Is crowdsourcing for optical flow ground truth generation feasible? In: Chen, M., Leibe, B., Neumann, B. (eds.) ICVS 2013. LNCS, vol. 7963, pp. 193–202. Springer, Heidelberg (2013). doi:[10.1007/978-3-642-39402-7_20](https://doi.org/10.1007/978-3-642-39402-7_20)
10. Edmond, E.C., Sim, S.X.-L., Li, H.-H., Tan, E.-K., Chan, L.-L.: Vascular tortuosity in relationship with hypertension and posterior fossa volume in hemifacial spasm. *BMC Neurol.* **16**, 120 (2016)
11. Frangi, A.F., Niessen, W.J., Vincken, K.L., Viergever, M.A.: Multiscale vessel enhancement filtering. In: Wells, W.M., Colchester, A., Delp, S. (eds.) MICCAI 1998. LNCS, vol. 1496, pp. 130–137. Springer, Heidelberg (1998). doi:[10.1007/BFb0056195](https://doi.org/10.1007/BFb0056195)
12. Heng, P.-A., Sun, H., Chen, K.-W., Wong, T.-T.: Interactive navigation of virtual vessel tracking with 3D intelligent scissors. *IJIG* **1**(02), 273–285 (2001)
13. Hennersperger, C., Baust, M.: Play for me: image segmentation via seamless playsourcing. *Comput. Games J.* **6**(1–2), 1–16 (2017)
14. Kerschnitzki, M., Kollmannsberger, P., Burghammer, M., Duda, G.N., Weinkamer, R., Wagermaier, W., Fratzl, P.: Architecture of the osteocyte network correlates with bone material quality. *JBMR* **28**(8), 1837–1845 (2013)
15. Lesage, D., Angelini, E.D., Bloch, I., Funka-Lea, G.: A review of 3D vessel lumen segmentation techniques: models, features and extraction schemes. *MIA* **13**(6), 819–845 (2009)
16. Lo, P., Van Ginneken, B., JosephMReinhardt, T.Y., De Jong, P.A., Irving, B., Fetita, C., Ortner, M., Pinho, R., Sijbers, J., et al.: Extraction of airways from CT (EXACT’09). *TMI* **31**(11), 2093–2107 (2012)
17. Arranz, A., Frean, J.: Crowdsourcing malaria parasite quantification: an online game for analyzing images of infected thick blood smears. *J. Med. Internet Res.* **14**(6), e167 (2012)
18. Maier-Hein, L., et al.: Can masses of non-experts train highly accurate image classifiers? In: Golland, P., Hata, N., Barillot, C., Hornegger, J., Howe, R. (eds.) MICCAI 2014. LNCS, vol. 8674, pp. 438–445. Springer, Cham (2014). doi:[10.1007/978-3-319-10470-6_55](https://doi.org/10.1007/978-3-319-10470-6_55)
19. Maier-Hein, L., et al.: Crowdsourcing for reference correspondence generation in endoscopic images. In: Golland, P., Hata, N., Barillot, C., Hornegger, J., Howe, R. (eds.) MICCAI 2014. LNCS, vol. 8674, pp. 349–356. Springer, Cham (2014). doi:[10.1007/978-3-319-10470-6_44](https://doi.org/10.1007/978-3-319-10470-6_44)
20. Marks, P.C., Preda, M., Henderson, T., Liaw, L., Lindner, V., Friesel, R.E., Pinz, I.M.: Interactive 3D analysis of blood vessel trees and collateral vessel volumes in magnetic resonance angiograms in the mouse ischemic hindlimb model. *OJMI* **7**, 19 (2013)
21. Poon, K., Hamarneh, G., Abugharbieh, R.: Live-vessel: extending livewire for simultaneous extraction of optimal medial and boundary paths in vascular images. In: Ayache, N., Ourselin, S., Maeder, A. (eds.) MICCAI 2007. LNCS, vol. 4792, pp. 444–451. Springer, Heidelberg (2007). doi:[10.1007/978-3-540-75759-7_54](https://doi.org/10.1007/978-3-540-75759-7_54)
22. Sankaran, S., Grady, L., Taylor, C.A.: Fast computation of hemodynamic sensitivity to lumen segmentation uncertainty. *TMI* **34**(12), 2562–2571 (2015)
23. Sommer, C., Straehle, C., Koethe, U., Hamprecht, F.A.: Ilastik: Interactive learning and segmentation toolkit. In: ISBI, pp. 230–233. IEEE (2011)

24. Sotelo, J., Urbina, J., Valverde, I., Tejos, C., Irarrázaval, P., Andia, M.E., Uribe, S., Hurtado, D.E.: 3D quantification of wall shear stress and oscillatory shear index using a finite-element method in 3D CINE PC-MRI data of the thoracic aorta. *TMI* **35**(6), 1475–1487 (2016)
25. Straka, M., Cervenansky, M., La Cruz, A., Kochl, A., Sramek, M., Groller, E., Fleischmann, D.: Focus & context visualization in CT-angiography. *The Vessel-Glyph*. IEEE (2004)
26. Vickerman, M.B., Keith, P.A., McKay, T.L., Gedeon, D.J., MichikoWatanabe, M.M., Karunamuni, G., Kaiser, P.K., Sears, J.E., Ebrahim, Q., et al.: VESGEN 2D: automated, user-interactive software for quantification and mapping of angiogenic and lymphangiogenic trees and networks. *Anat. Rec.* **292**(3), 320–332 (2009)
27. Yu, K.-C., Ritman, E.L., Higgins, W.E.: Graphical tools for improved definition of 3D arterial trees. In: *Medical Imaging 2004*. SPIE, pp. 485–495 (2004)