2018 IEEE 15th International Symposium on Biomedical Imaging (ISBI 2018)
April 4-7, 2018, Washington, D.C., USA

Learning-based Long-range Axon Tracing in Dense Scenes

Mark Hernandez[1], Adam Brewster[1], Larry Thul[1], Brian A. Telfer[1], Arjun Majumdar[1], Heejin Choi[2], Taeyun Ku[2], Kwanghun Chung[2], and Laura J. Brattain[1]

[1]MIT Lincoln Laboratory, Lexington, MA USA
[2]Institute of Medical Engineering and Science, Massachusetts Institute of Technology, Cambridge, MA USA

***Abstract*—Recent advances in intact brain imaging, such as CLARITY and MAP (Magnified Analysis of the Proteome), enable volumetric imaging of brain tissue at subcellular resolution. Currently there is no effective tool for automated axon tracing at single fiber resolution in densely populated imagery. To address this need, we developed a machine learning-based high performance computing pipeline: 1) a convolutional neural network detects axon fiber voxels, 2) morphological operations extract fiber centerlines, 3) tracking logic connects fiber segments across low-intensity gaps and unresolved fiber crossings. The pipeline was implemented on a CPU cluster and tested on a 250 gigabyte volume of SMI-312 densely labeled axon fibers, imaged from parts of the hippocampus and cortex of a MAP-processed mouse brain. The pipeline automatically traced 221,298 fibers across gray and white matter in 10 hours. Of the traced fibers, 104 exceeded 1 mm, with the longest being 2.16 mm. An accuracy of 84% was reported based on manual evaluation of the 200 longest fibers. While there is room to improve accuracy, this pipeline offers a significant speed-up and increased efficiency over tracing one neuron at a time or manual fiber tracing. As the pipeline is scaled up to evaluate larger regions of the brain, connectivity patterns derived from these automatically traced long-range axons can potentially provide insights into brain function.**

Index terms — Brain connectivity, Convolutional neural network, Neuron segmentation, Neuron tracing

I. Introduction

Microscopy-based intact brain imaging such as CLARITY [1] and Magnified Analysis of the Proteome (MAP) [2] enable the resolution of subcellular structures. Despite imaging advances, dense mapping and analysis of large-scale, long-range connectivity has been hindered by the lack of automated image processing capabilities [3]. Existing neuron tracing methods primarily focus on sparsely labeled data [4]–[8] or tractography [9], and are insufficient for tracing dense axons at high resolution [10].

DISTRIBUTION STATEMENT A. Approved for public release: distribution unlimited. This material is based upon work supported by the Assistant Secretary of Defense for Research and Engineering under Air Force Contract No. FA8721-05-C-0002 and/or FA8702-15-D-0001. Any opinions, findings, conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the Assistant Secretary of Defense for Research and Engineering.

In order to evaluate these dense scenes more efficiently, we have developed a processing pipeline for semi-automated tracing and connectivity mapping at a single fiber resolution. The pipeline aims to trace high-confidence fibers automatically and leave ambiguous cases for human intervention, thus greatly reducing the burden on human labor. Major improvements over our previous work [11] include: 1) increased segmentation accuracy with a Convolutional Neural Network (CNN), 2) automatic resolution of fiber gaps and unresolved crossings, 3) a scaled-up pipeline that can process gigabytes (GB) or more of data using high performance computing (HPC). Section 2 describes our methods, followed by experimental results and performance analysis in Section 3. Section 4 concludes with discussions on future work.

II. Methods

To enable parallel processing, the original microscopy data are divided into subvolumes of 400 x 400 x 800 voxels. A high-performance computer cluster executes the steps shown in Fig. 1: axon voxels are segmented using the CNN, the axon centerlines are extracted and transformed into a graph, and artifacts such as gaps and unresolved junctions are automatically corrected. Fibers in a given volume are segmented and traced in parallel. This approach differs from commonly used tracing algorithms where tracing is done one fiber at a time.

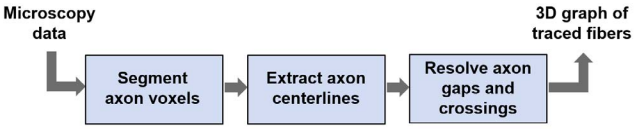

Fig. 1: Dense axon tracing pipeline.

A. Axon Voxel Segmentation

The first step detects and segments voxels containing axons. A simple threshold is not suitable because the signal and background levels vary significantly throughout the image. Instead, a CNN [12] is applied, with an output less than 0.5 indicating background, and larger values

indicating axons. Because the training data are selected from regions with diverse imaging conditions, the output is largely invariant to those variations, and a constant threshold can be used to binarize the CNN output. The network layers are described in Table I. In each of the convolution layers, stride is 1, no padding, and ReLU is used as the activation function, except in the last layer where sigmoid is used.

TABLE I: Four-layer CNN configuration.

Layer Type	Size	Filters
Input	$19 \times 19 \times 19$	1
$5 \times 5 \times 5$ Valid Convolution	$15 \times 15 \times 15$	6
$3 \times 3 \times 3$ Max Pooling	$5 \times 5 \times 5$	6
$3 \times 3 \times 3$ Valid Convolution	$3 \times 3 \times 3$	12
$3 \times 3 \times 3$ Valid Convolution	$1 \times 1 \times 1$	49
$1 \times 1 \times 1$ Valid Convolution	$1 \times 1 \times 1$	1

Training data for the network consist of $19 \times 19 \times 19$ voxel volumes which are labeled as either background (0) or axons (1). Following an active learning approach [13], [14], the model was trained in the background while the user selected misclassified voxels, which were added to the training set. After training the network on the manually collected examples (about 100 for each class), the full volume was processed with these initial weights, and fibers were identified. We then found the longest detected fibers, performed manual reviews, and built a much larger training set (approx. 10,000 for each class) by choosing points on the centerline of a long fiber as positive cases and points that were near but not on the centerline as negative cases. After learning a new set of parameters based on this expanded training set, we reprocessed the image data, and repeated the neuron detection process. The number of voxels in the training set was less than 10^{-8} of the total voxels in the data set.

B. Centerline Extraction

A homotopic thinning algorithm is applied to the binary CNN output of each subvolume to produce a 3D medial axis skeleton that estimates the locations of the axon fiber centerlines [15]. To avoid boundary issues caused by partitioning, the subvolumes are padded in all directions by 32 voxels. Following [16], a graph is produced from the skeletonization, encoding both the spatial orientation and connectivity of the axon fibers. The edges of the graph represent fiber segment centerlines, the end nodes represent the termination of a single centerline, and the junction nodes represent locations where two or more centerlines are perceived to be connected.

C. Resolving Artifacts

There are three types of artifacts in the graphs: 1) intra-subvolume gaps that fragment a fiber within a subvolume, 2) inter-subvolume gaps in fibers traversing subvolume boundaries, 3) false junctions where two or more crossing fibers are erroneously connected. Intra-subvolume gaps are due primarily to signal intensity dropouts and varying fiber diameters. The inter-subvolume gaps are a byproduct of subvolume partitioning. False junctions arise when fibers closely cross in the raw image and become indistinguishable at the voxel level due to insufficient imaging resolution.

It is impractical to correct all artifacts manually due to the sheer volume of cases. Instead, learning algorithms are used to correct a large number of artifacts with high confidence, leaving only a fraction of ambiguous cases for human inspection. Features are extracted from the spatial orientation and intensity profile of the fiber segments. One classifier is trained to resolve gaps, and a second classifier is trained to resolve junctions. Valid artifact corrections confirmed by the classifiers are updated in the graph data. The algorithm flow is illustrated in Fig. 2.

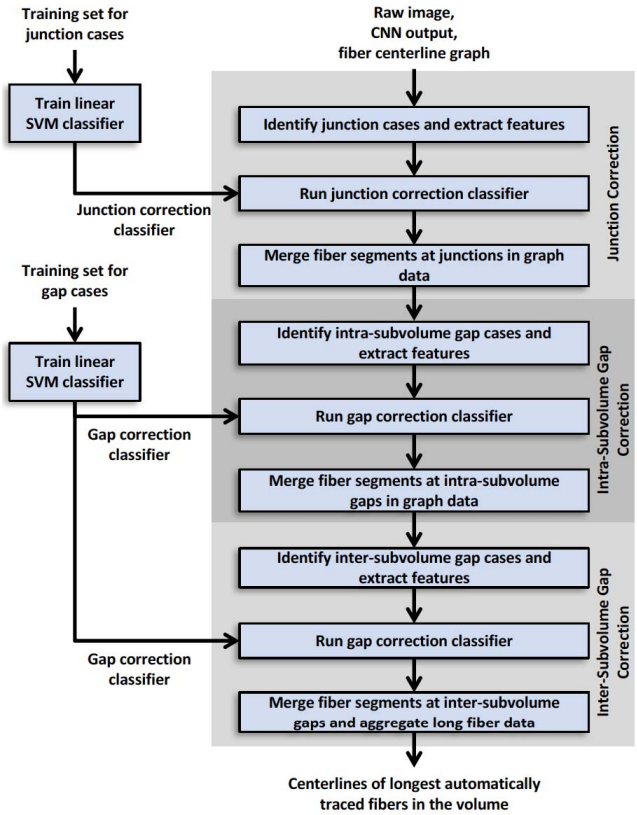

Fig. 2: Artifact classification and correction flow diagram.

1) Identification of Potential Artifacts: To identify junction correction candidates, vectors of varying lengths (l_v) are fit to the fiber segments leading to a particular junction. Candidate pairings for fiber segments are chosen according to greatest co-linearity. To identify intra-subvolume gap candidates, pairs of neighboring end nodes within 20 voxels of each other are located. If an end node has more than one neighbor, the most likely fiber segment pair to merge is chosen based on greatest co-linearity. To identify inter-subvolume gap candidates, only end nodes within 10 voxels of a subvolume boundary are considered.

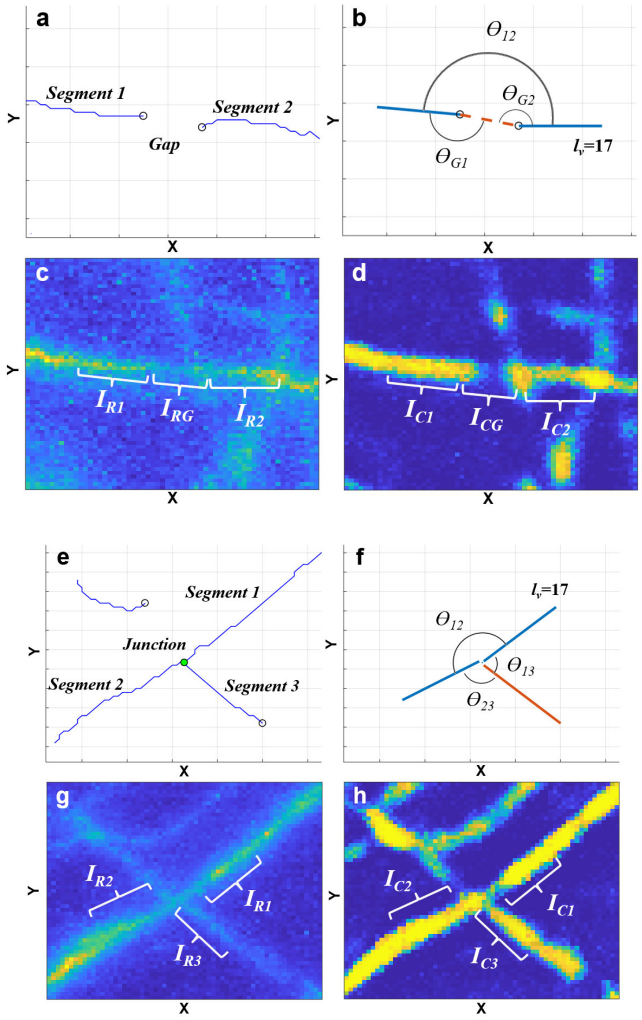

Fig. 3: Feature extraction for artifact correction. (a) Example 3D graph of a typical gap between two fiber segments. (b) Directionality vector plot between the gap and fiber segments. (c)-(d) Raw image and CNN output of (a). (e) Example 3D graph of a typical fiber junction. (f) Directionality vector plot between the three fiber segments. (g)-(h) Raw image and CNN output of (e).

End nodes from the adjacent subvolume's boundary are found that are within 20 voxels of the reference end node. Candidate segments to merge are chosen based on greatest co-linearity.

2) Feature Extraction: For each potential artifact, features are derived from the graphs, raw images, and CNN outputs of the two fiber segments being considered for merging (Fig. 3). These features measure how well the directionality and intensity profile are preserved if the segments are connected. Fig. 4 summarizes the main features used for gap and junction classifications. In addition to evaluating the continuity between the two fiber segments,

Feature Name	Gap or Junction Classifier?	Feature Description
Segment Co-linearity Angle Deviation*	Both	$\lvert\theta_{12} - \pi\rvert$
Gap Co-linearity Angle Deviation*	Gap	$\mu(\lvert\theta_{G1} - \pi\rvert, \lvert\theta_{G2} - \pi\rvert)$
Other Perpendicular Angle Deviation*	Junction	$\mu(\lvert\theta_{23} - \frac{\pi}{2}\rvert, \lvert\theta_{13} - \frac{\pi}{2}\rvert)$
Segment Intensity Difference (Raw & CNN)	Both	$\Delta\mu_{IR} = \lvert\mu(I_{R1}) - \mu(I_{R2})\rvert$ $\Delta\sigma_{IR} = \lvert\sigma(I_{R1}) - \sigma(I_{R2})\rvert$ $\Delta\mu_{IC} = \lvert\mu(I_{C1}) - \mu(I_{C2})\rvert$ $\Delta\sigma_{IC} = \lvert\sigma(I_{C1}) - \sigma(I_{C2})\rvert$
Gap Intensity Difference (Raw & CNN)	Gap	$\Delta\mu_{IRG} = \lvert\mu(I_{RG}) - \mu([I_{R1}, I_{R2}])\rvert$ $\Delta\sigma_{IRG} = \lvert\sigma(I_{RG}) - \sigma([I_{R1}, I_{R2}])\rvert$ $\Delta\mu_{ICG} = \lvert\mu(I_{CG}) - \mu([I_{C1}, I_{C2}])\rvert$ $\Delta\sigma_{ICG} = \lvert\sigma(I_{CG}) - \sigma([I_{C1}, I_{C2}])\rvert$
Other Intensity Difference (Raw & CNN)	Junction	$\Delta\mu_{IRO} = \lvert\mu(I_{R3}) - \mu([I_{R1}, I_{R2}])\rvert$ $\Delta\sigma_{IRO} = \lvert\sigma(I_{R3}) - \sigma([I_{R1}, I_{R2}])\rvert$ $\Delta\mu_{ICO} = \lvert\mu(I_{C3}) - \mu([I_{C1}, I_{C2}])\rvert$ $\Delta\sigma_{ICO} = \lvert\sigma(I_{C3}) - \sigma([I_{C1}, I_{C2}])\rvert$
Node Distance	Gap	Euclidean distances between end nodes
Number Nearby Links	Both	Junction: number of other links in a junction Gap: number of links with neighboring end nodes

* Angle features are average over all vector lengths (l_v)

Fig. 4: Summary of features used for gap and junction classification. Equations refer to terms in Fig. 3.

we generate features specifically for potential gap artifacts. These features measure how the gap characteristics differ from those of the two fiber segments that it is connecting. Features that are specific to potential junction artifacts are also generated to assess the characteristics of the other fiber segments approaching a junction node relative to the fiber segments that were chosen for correction. If multiple fiber segment pairs at a junction share similar characteristics, the junction is perceived as more ambiguous and the correction becomes less likely.

3) Classifiers: Two binary classifiers are used to identify which potential artifacts should be corrected. The first classifier is trained using 200 junction candidates with labels indicating the valid junction corrections. The second is trained with a mix of 290 intra- and inter-subvolume gap candidates labeled with valid gap corrections. Several different classification methods, including linear discriminant analysis, linear and Gaussian support vector machines (SVMs), k-nearest neighbors (k-NN), and random forest decision trees, were evaluated. Based on the test results, a linear SVM was chosen as the classifier for both cases. Since our goal is to correct the more obvious artifacts with high confidence and prioritize low false alarm rates, the training included a cost function of 10:1 which weights false alarms more heavily than detection. The number of gaps and junctions in the training sets for each classifier was less than 10^{-4} and 4×10^{-3} of the total respective numbers in the data set, thus nearly all of the artifact cases in the data set are independent test samples.

III. Results

The pipeline was tested on a 19200 x 19600 x 800 voxel, 250 GB volume of SMI-312 densely labeled axons,

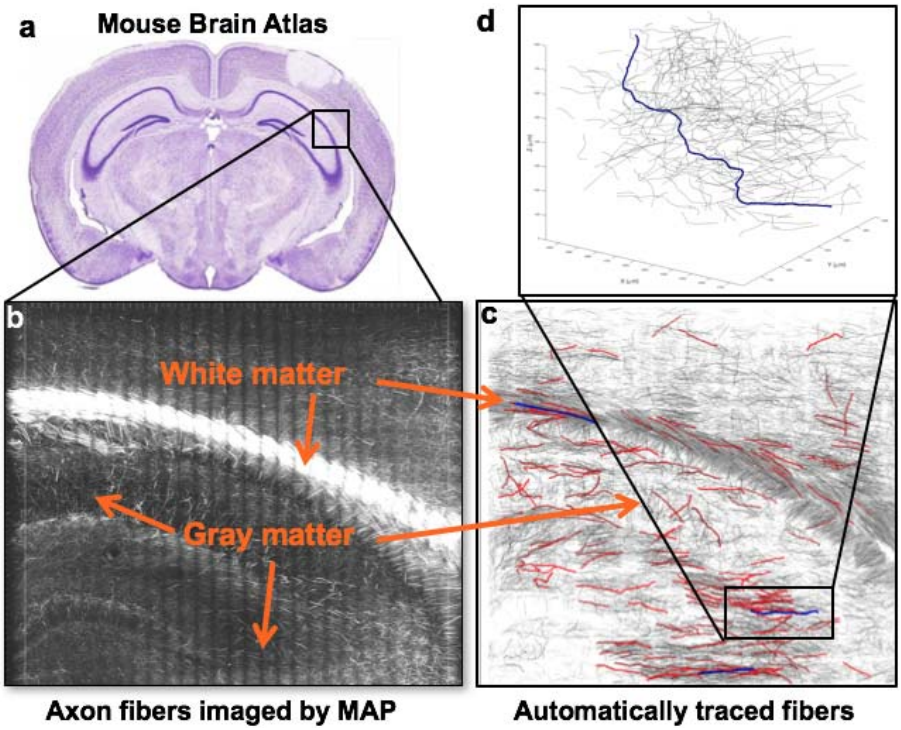

Fig. 5: Dense axon fiber tracing results. (a) Example mouse brain atlas. (b) Axon fibers imaged using MAP. (c) Automatically traced fibers. Gray: axons longer than 300 μm. Red: the 200 longest fibers. Blue: the top 3 longest fibers. (d) Zoomed-in view showing one of the longest fibers traversing through neighboring axons.

imaged from parts of the hippocampus and cortex of a MAP-processed mouse brain (resolution: 0.325 μm in x, y and 1 μm in z). Original tissue size is 1.6 mm x 1.6 mm x 170 μm, followed by 4-fold tissue expansion. The process took 10 hours, including 3 hours for the CNN on 256 compute nodes and 7 hours for centerline extraction, artifact correction, and subvolume stitching on 50 compute nodes.

Table II details the length distribution of traced axon fibers. There were 104 fibers that exceeded the length of 1 mm while the longest traced fiber was 2.16 mm. Fig. 5 shows the traced fibers longer than 300 mm (gray) overlaid with the 200 longest validated fibers (red) and the top 3 longest fibers (blue). The traced fibers in Fig. 5(c) match the patterns in Fig. 5(b), including the white matter region, where axons are particularly dense. A zoomed-in view of the volume highlights an example of one of the longest fibers traversing through neighboring axons.

To validate the accuracy of these fiber tracings, the 200 longest fibers were manually inspected for any tracing errors. A tracing error was defined as any point along the tracing where two distinct fibers were incorrectly connected. If one or more errors appeared along the fiber, it was considered an incorrect tracing. Among the longest fibers, 167 were correct tracings (84% accuracy). Of the 33 incorrect tracings, only 13 were caused by erroneous artifact classifications. The rest were due to severe intensity dropouts in the original image causing errors in the CNN detection and graph transform.

TABLE II: Fiber length distribution

Fiber length	Number of fibers in range
1 mm or longer	104
800 μm - 1 mm	288
600 μm - 800 μm	1379
600 μm or shorter	219527

IV. Summary and Discussion

Our learning-based HPC pipeline can be a powerful tool for identifying and understanding the subcellular connections in the brain. For our 250 GB sample volume, the algorithm traced 221,298 axons, while achieving a significant processing time speed-up compared to single neuron tracing or manual tracing. The sheer number of automatically traced fibers offers a basis for high-level analysis to assess large-scale distribution statistics of long-range fiber connectivity, orientation, length, and diameter etc. The pipeline allows for new training data generated from the validated fiber tracings to be fed back into the CNN model for continuous improvement of system performance. As the pipeline is scaled up to trace axon connections in larger regions of the brain, the connectivity patterns can potentially provide insight into the underlying mechanisms involved with various brain disorders.

References

[1] K. Chung and K. Deisseroth, "CLARITY for mapping the nervous system," *Nature Methods*, vol. 10, no. 6, pp. 508–513, May 2013. [Online]. Available: http://www.nature.com/doifinder/10.1038/nmeth.2481

[2] T. Ku, J. Swaney, J.-Y. Park, A. Albanese, E. Murray, J. H. Cho, Y.-G. Park, V. Mangena, J. Chen, and K. Chung, "Multiplexed and scalable super-resolution imaging of three-dimensional protein localization in size-adjustable tissues," *Nat Biotech*, vol. 34, no. 9, pp. 973–981, Sep. 2016. [Online]. Available: http://dx.doi.org/10.1038/nbt.3641

[3] J. W. Lichtman, H. Pfister, and N. Shavit, "The big data challenges of connectomics," *Nature Neuroscience*, vol. 17, no. 11, pp. 1448–1454, Oct. 2014. [Online]. Available: http://www.nature.com/doifinder/10.1038/nn.3837

[4] P. Chothani, V. Mehta, and A. Stepanyants, "Automated tracing of neurites from light microscopy stacks of images," *Neuroinformatics*, vol. 9, no. 2-3, pp. 263–278, 2011.

[5] Y. Wang, A. Narayanaswamy, C.-L. Tsai, and B. Roysam, "A broadly applicable 3-d neuron tracing method based on open-curve snake," *Neuroinformatics*, vol. 9, no. 2-3, pp. 193–217, 2011.

[6] H. Peng, A. Bria, Z. Zhou, G. Iannello, and F. Long, "Extensible visualization and analysis for multidimensional images using Vaa3d," *Nature Protocols*, vol. 9, no. 1, pp. 193–208, Jan. 2014. [Online]. Available: http://www.nature.com/doifinder/10.1038/nprot.2014.011

[7] H. Peng, Z. Zhou, E. Meijering, T. Zhao, G. Ascoli, and M. Hawrylycz, "Automatic Tracing of Ultra-Volume of Neuronal Images," *bioRxiv*, p. 087726, Nov. 2016. [Online]. Available: https://www.biorxiv.org/content/early/2016/11/14/087726

[8] C.-W. Wang, Y.-C. Lee, H. Pradana, Z. Zhou, and H. Peng, "Ensemble Neuron Tracer for 3d Neuron Reconstruction," *Neuroinformatics*, vol. 15, no. 2, pp. 185–198, Apr. 2017. [Online]. Available: https://link.springer.com/article/10.1007/s12021-017-9325-1

[9] L. Ye, W. E. Allen, K. R. Thompson, Q. Tian, B. Hsueh, C. Ramakrishnan, A.-C. Wang, J. H. Jennings, A. Adhikari, C. H. Halpern *et al.*, "Wiring and molecular features of prefrontal ensembles representing distinct experiences," *Cell*, vol. 165, no. 7, pp. 1776–1788, 2016.

[10] R. Tomer, L. Ye, B. Hsueh, and K. Deisseroth, "Advanced CLARITY for rapid and high-resolution imaging of intact tissues," *Nature protocols*, vol. 9, no. 7, p. 1682, 2014.

[11] L. J. Brattain, B. A. Telfer, S. Samsi, T. Ku, H. Choi, and K. Chung, "Automated dense neuronal fiber tracing and connectivity mapping at cellular level," in *Biomedical Imaging (ISBI 2017), 2017 IEEE 14th International Symposium on*. IEEE, 2017, pp. 332–336.

[12] Y. LeCun, K. Kavukcuoglu, and C. Farabet, *Convolutional networks and applications in vision*, 2010, pp. 253–256.

[13] F. Ricci, L. Rokach, and B. Shapira, "Introduction to recommender systems handbook," in *Recommender systems handbook*. Springer, 2011, pp. 1–35.

[14] N. Rubens, M. Elahi, M. Sugiyama, and D. Kaplan, "Active learning in recommender systems," in *Recommender systems handbook*. Springer, 2015, pp. 809–846.

[15] M. Sofka and C. V. Stewart, "Retinal vessel centerline extraction using multiscale matched filters, confidence and edge measures," *IEEE transactions on medical imaging*, vol. 25, no. 12, pp. 1531–1546, 2006.

[16] M. Kerschnitzki, P. Kollmannsberger, M. Burghammer, G. N. Duda, R. Weinkamer, W. Wagermaier, and P. Fratzl, "Architecture of the osteocyte network correlates with bone material quality," *Journal of Bone and Mineral Research*, vol. 28, no. 8, pp. 1837–1845, Aug. 2013. [Online]. Available: http://doi.wiley.com/10.1002/jbmr.1927