Image Processing and Machine Learning for Fibril Porous Media

Zhenyu He, W. L. Gore; Jim Adriazola, NJIT; Kevin Cotton, CGU; Ellie Gurvich, UMBC; Qingxia Li, Fisk; Richard Moore, NJIT; Vrushaly Shinglot, UT-Dallas; Aprajita Singh, UT-Dallas; Yixuan Sun, NJIT; Thomas Tu, UCLA; Mingkai Yu, UMBC; Naren Vohra, Oregon State

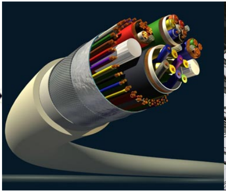
The 35th Annual Workshop on Mathematical Problems in Industry
New Jersey Institute of Technology
Newark, NJ

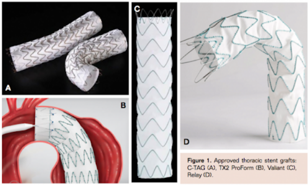
June 21, 2019

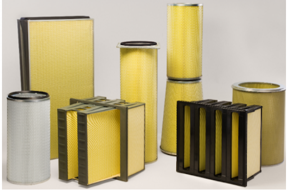
Overview

Background

- Founded in 1958 by Bill and Vieve Gore in Newark, DE with 10,000+ associates and more than \$3 billion in annual revenue
- Manufactures in U.S., Germany, Scotland, Japan, and China and has sales offices in many other countries
- Privately held and has 2,000+ patents worldwide

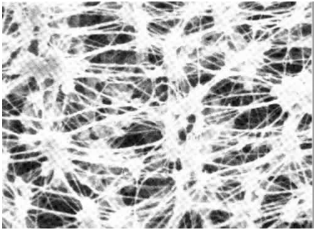
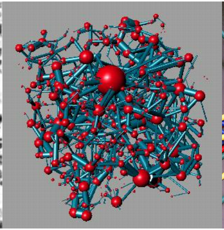
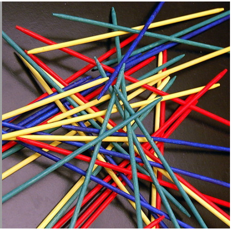






Characterization of Porous Media

- Network of cylindrical capillaries: Pore networks represented by graphs with connectivity (2D/3D lattices)
- Curves in space: Lines, circles, spheres, cylinders laid out in space to represent the solid portion of the porous medium
- Experimental Characterization: Gas/Liquid porometry and porosimetry and Filtration Efficiency for well defined particles/molecules

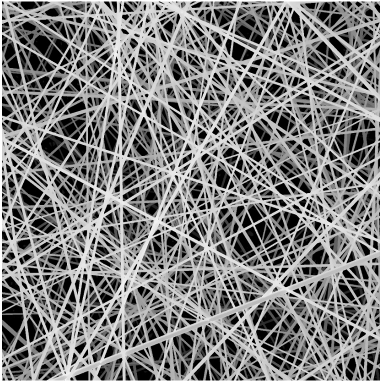




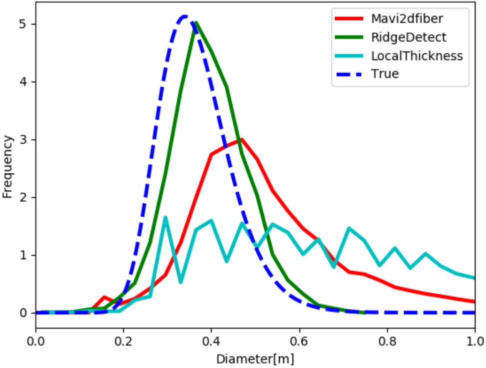
Setup

- The scanning electron microscope (SEM) in particular is a useful tool to analyze the structure of the pore networks
- Assume it generates negligible information depthwise and the distribution is homogeneous in depth
- We want a distribution of the diameters of the fibers in the network

Problem 1

Study the existing algorithms for Image Analysis





Existing Methods

- Local thickness:
  - Highly effective for 3D geometry (planes, spheres, cylinders).
  - Distance map is defined by the largest sphere that can fit in the object.
  - For 2D, it treats everything in an intersection as a single object.
- Pourdeyhimi : Compute diameter by computing the distance transform.
- Ridge detection:
  - Highly effective for detecting line segments
  - Highly dependent on 3 mandatory parameters
  - Creates lines from background noise
  - Mixes up fibers in the intersections

Problem 2

Develop models to study the diameter distribution of the fibres

- Transformation- Radon
- Neural Networks
- Probabilistic Approach

Radon Transform

- Definition:

$$R(f)(P) = \int_P f,$$

  where P is the hyperplane in $\mathbb{R}^n$, $f \in S(\mathbb{R}^n)$. (Fourier Analysis an introduction, Stein, E., Shakarchi, R.)

- In 2D, it coincides with X-ray transform:

$$X(\rho)(L) = \int_L \rho,$$

  where L is a line in $\mathbb{R}^2$, $\rho \in S(\mathbb{R}^2)$ is typically a density function.

- Hough transform (Hough, P. V. C. 1962 Patent No. 3,069,654).

$$\begin{cases} y = y_i x + x_i, & \text{Rosenfeld, 1969} \\ \rho = x_i \cos\theta + y_i \sin\theta, & \text{Duda and Hart, 1972} \end{cases}$$

Radon Transform 2D

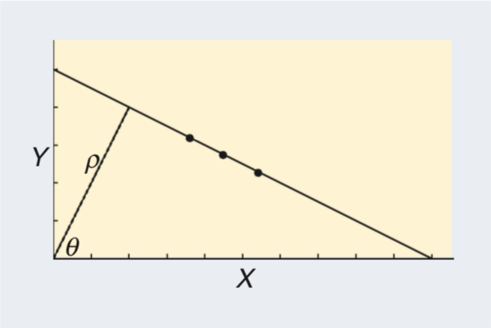


Figure: From Hart 2009, the normal parametrization of a straight line.

$$R(f)(\theta,\rho) = \int_{-\infty}^{\infty} f(z\sin\theta + \rho\cos\theta, -z\cos\theta + \rho\sin\theta)dz$$

Radon Transform for lines

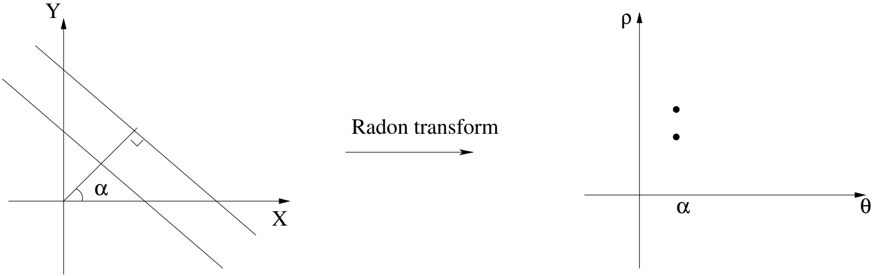


Figure: Radon transform for lines.

Image processing procedure

- change image to grey scale (rgb2gray)
- find the edge (edge)
- radon transform the image with just edges (radon)
- locate peaks (tricky part)
- peak location gives information about size and orientation (couple peaks)

Radon Transform for fibers

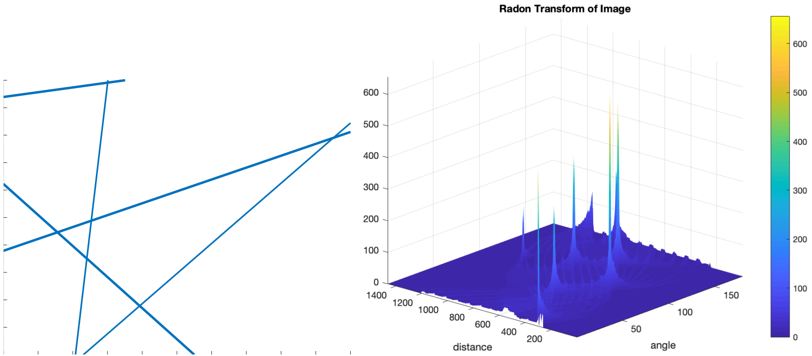


Figure: Fiber image processing using Radon transform.

Hough Transform for fibers

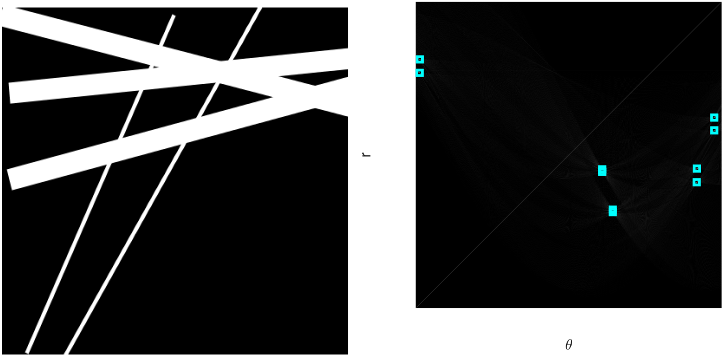


Figure: Fiber image processing using Hough transform. Code from Davin.

Edge Detection for real fibers

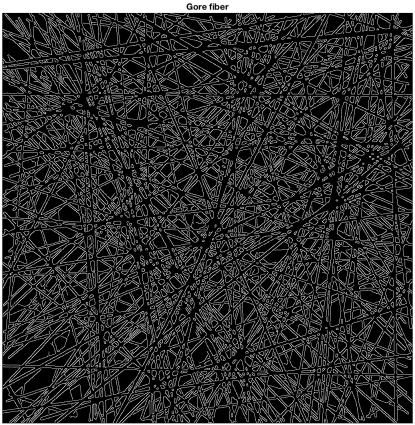


Figure: Actual fiber image after edge detection.

Radon Transform for real fibers

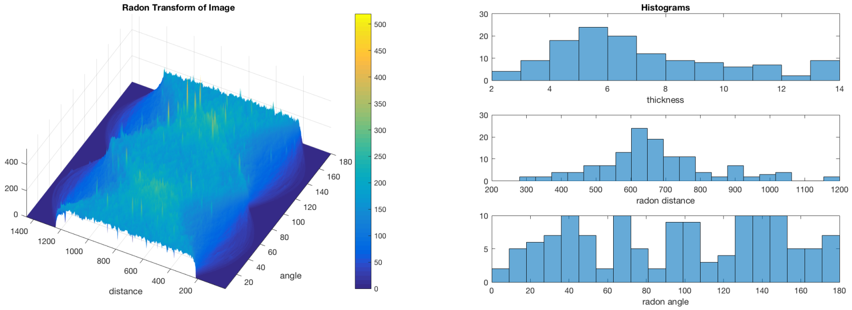


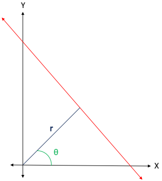
Figure: Actual fiber image processing using Radon transform. With distribution of fiber size.

Hough Transform

- A **feature extraction technique** used in image analysis.
- Helps to solve problem of missing points or pixels in detecting simple shapes like straight lines, circle or ellipses.
- Representing straight lines as $y = mx + b$, vertical lines pose a problem.
- In Hough transform, a straight lines in image is represented by its normal direction θ and its distance to the origin.

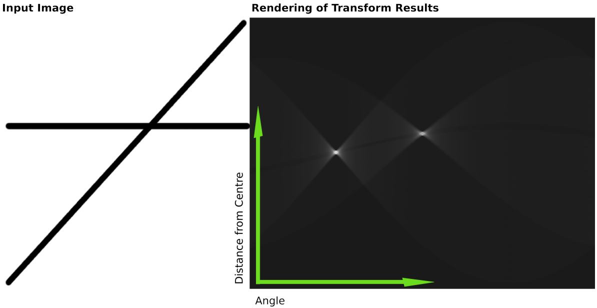
$$r = x\cos(\theta) + y\sin(\theta).$$

All the points on the straight line has a contribution to (θ, r) in Hough space.



Hough Transform

- Each straight line in image is a pair (r, θ) in Hough space.
- Straight lines passing through **a point** in image corresponds to a **sinusoidal curve** in Hough space.
- Detecting **straight line** is equivalent to finding the **bright spot** in Hough space contributed by **intersecting curves**.

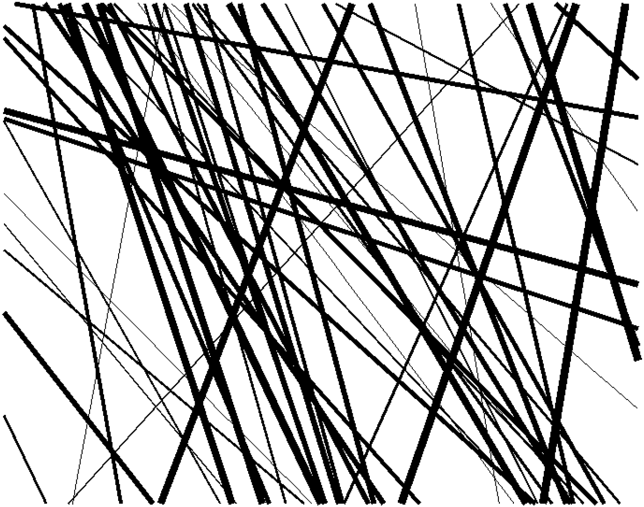


Data Synthesis:

- ML/NN models require training on large amounts of labeled data which we do not have. For this, we created synthetic images of fiber structure.
- Synthetic data is labeled based on number of fibers, angular distribution of fibers, and distribution of thickness.

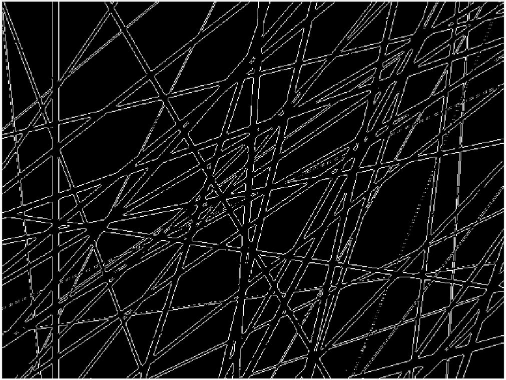
Machine Learning

- Here is a sample synthetic image with a uniform distribution of fiber widths and a normal (mod π) distribution of fiber angle with $\mu = 112^\circ$ and $\sigma = 29^\circ$



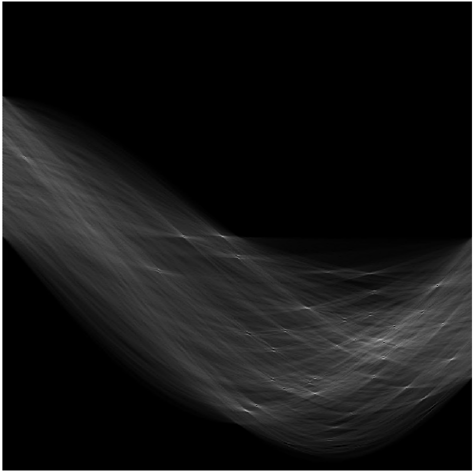
Pre-Processing:

- In order to train on the synthetic images, we first pre-process with edge detection, and then convert the image using the Hough Transform.
- Here we see a similar image after edge detection.



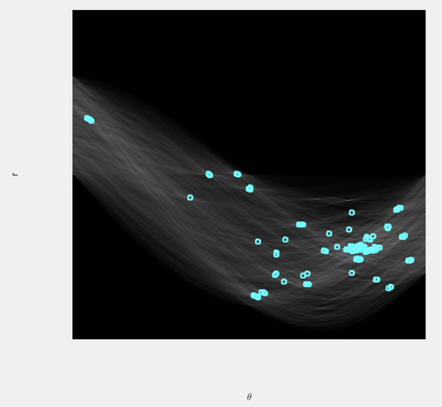
Pre-Processing:

- In order to extract non-localized features using a Convolutional Neural Network, we then take the Hough Transform



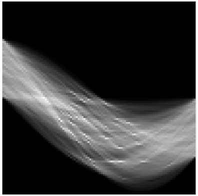
Pre-Processing:

- At this point, proper selection of Hough Transform peaks will lead to the identification of lines parameterized by θ and r. Taking the histogram of these theta values would give a sample of the distribution of fiber angles. Of course, you have to remap theta to your desired coordinate system. (i.e. $\frac{\pi}{2} - \theta$)



Pre-Processing:

- Instead, we opt for a Convolutional Neural Network to train on the Hough transformed image. We downsample the image, as training on smaller images is much faster. (Training on many large images can introduce GPU memory concerns)



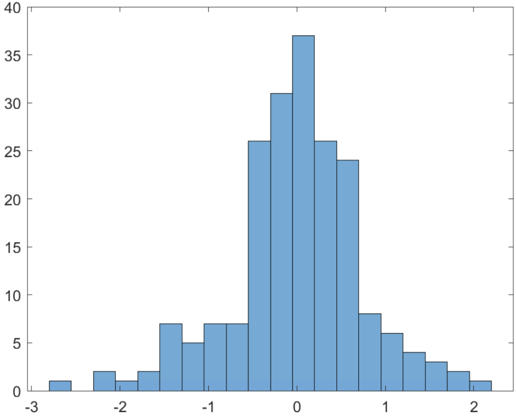
Model:

- We used a Convolutional Neural Network that was modified from one used to train on identifying anglular offset of MNIST data. Our network has a total of three convolution layers.

```
layers = [
    imageInputLayer([im_height im_width 1])

    convolution2dLayer(3,8,'Padding','same')
    batchNormalizationLayer
    reluLayer

    averagePooling2dLayer(2,'Stride',2)

    convolution2dLayer(3,16,'Padding','same')
    batchNormalizationLayer
    reluLayer

    averagePooling2dLayer(2,'Stride',2)

    convolution2dLayer(3,32,'Padding','same')
    batchNormalizationLayer
    reluLayer

    dropoutLayer(0.2)
    fullyConnectedLayer(1)
    regressionLayer];
```

Model:

- Setting the target variable as the mean of the distribution angles does not give bad results. We get an RMSE = 0.7391 on the test data (n=1000,80% training,20% testing).
- Here we see the histogram of errors on the Test data set in radians.



Convolutional Layer:

- A brief explination... A convolutional layer is an m x n kernel that is convolved over the image with weights that are are trained through back propigation. (Image taken from Skani, *Introduction to Deep Learning: From Logical Calculus to Artificial Intelligence*)

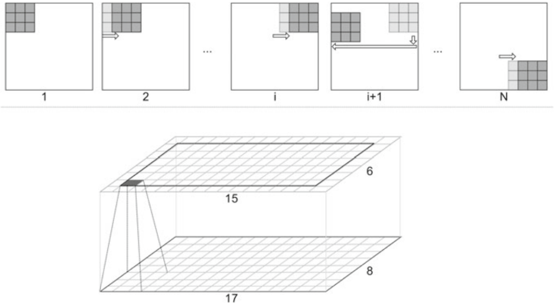


Fig. 6.2 2D Convolutional layer

Back Propogation:

- The regression output layer of the Neural Network uses the Mean Squared Error loss function defined by:

$$MSE = \sum_{i=1}^{R} \frac{(t_i - r_i)^2}{R}$$

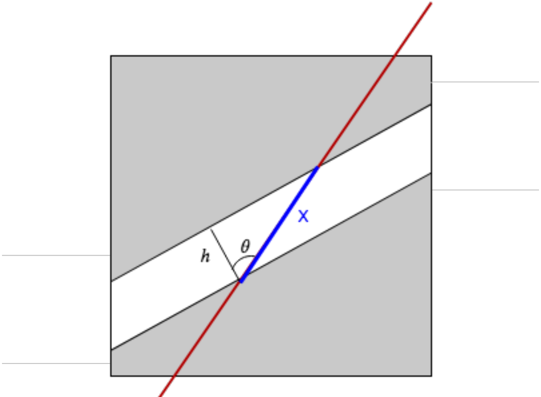
- This loss function does not take into account the periodicity of θ and could be improved by redefining in periodic space.
- In MATLAB, this could be done by modifying the source code for RegressionOutputLayer.

Room for Improvement:

- Further testing can be done on this problem by simulating more complex distributions of angle/thickness.
- More time would also permit exploring a neural network that outputs a vector of values that correspond to the the distribution of fiber angle. (Instead of just the mean statistic)
- Finally, more testing needs to be done on using a CNN over the Hough space to identify fiber thickness.

1D Slicing

- Consider measuring a one-dimensional slice of the image.
- The slice intersects a fiber at an angle θ, assumed to be normally distributed.
- Measure the width of the intersection X with the fiber by counting the number of consecutive light pixels.
- This can be used to find the distribution of h.



Expectation of Width of Intersection

For a fixed single fiber of width h,

$$X = \frac{h}{\cos(\theta)} = h\sec(\theta)$$
$$h = \frac{\mathbb{E}[X]}{\mathbb{E}[\sec(\theta)]}$$

Unfortunately, the expectation of $sec(\theta)$ is unbounded. However, the maximum measurement is approximately the length of fiber that is in the frame, L. Thus,

$$\mathbb{E}[X] = \mathbb{E}[\min(h\sec(\theta), L)]$$
$$\approx \frac{2h}{\pi}\left(1 + \log\left(\frac{2L}{h}\right)\right)$$

Multiplying with Width Distribution

- Consider instead that each measured intersection corresponds with a range of possible width values.
- By multiplying the empirical distribution with a distribution of possible widths, we arrive at a distribution much closer to the truth.
- Since the angle theta is distributed uniformly, and $X = h\cos(\theta)$, this width distribution is the derivative of the inverse cosine, $\frac{1}{\sqrt{1-x^2}}$

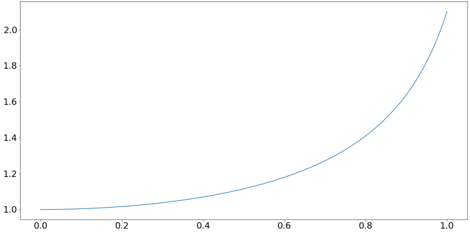


Figure: Plot of $\frac{1}{\sqrt{1-x^2}}$ from $x = 0$ to 1

Results

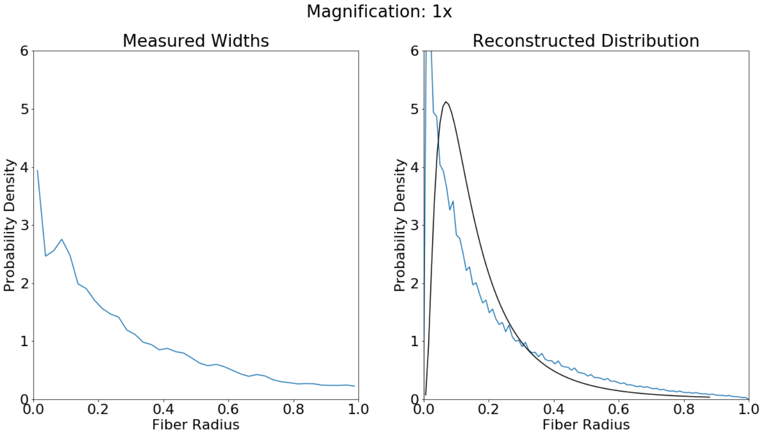


Figure: Distribution at 1x magnification. In black is shown the ground truth distribution the dataset was sampled from.

Truncating Small Fibers

Fibers of small diameter are often erroneously detected. If we truncate all intersections of width less than 2 pixels, we arrive at distributions much closer to the truth.

Results with Truncation

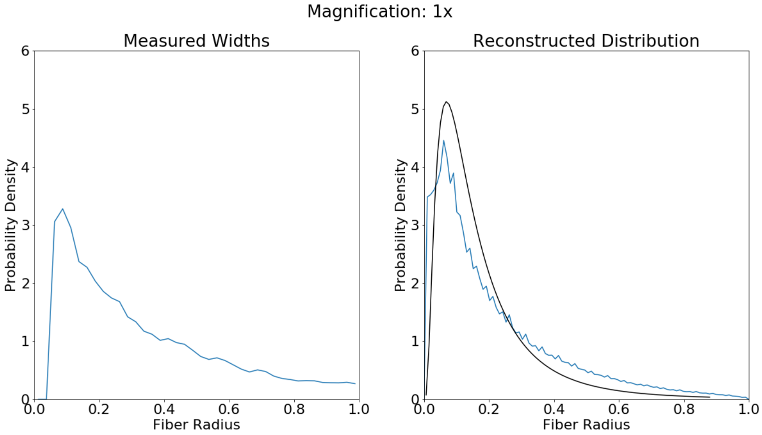


Figure: Distribution at 1x magnification. In black is shown the ground truth distribution the dataset was sampled from.

Results with Truncation

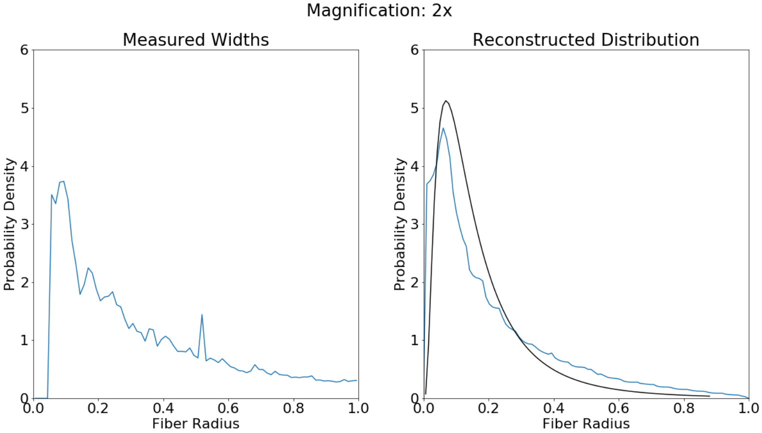


Figure: Distribution at 2x magnification. In black is shown the ground truth distribution the dataset was sampled from.

Results with Truncation

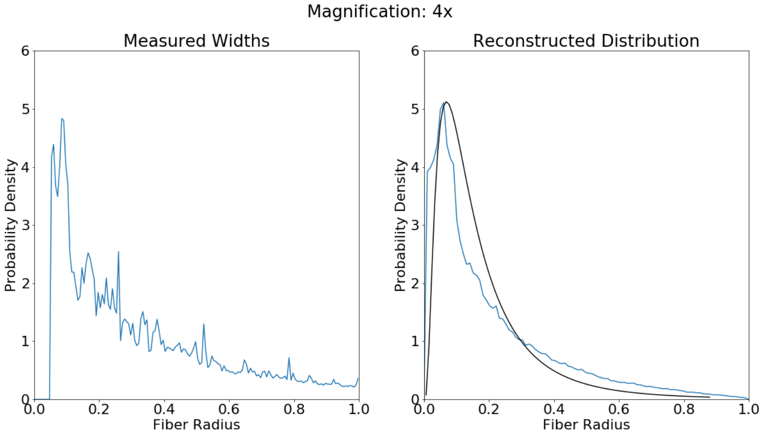


Figure: Distribution at 4x magnification. In black is shown the ground truth distribution the dataset was sampled from.

Results with Truncation

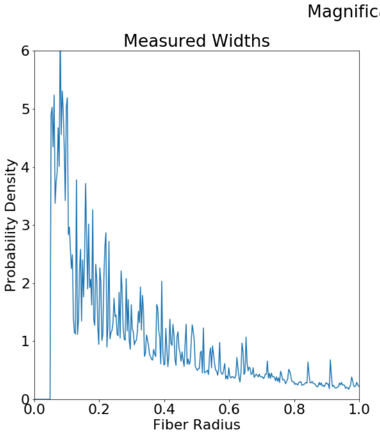


Figure: Distribution at 8x magnification. In black is shown the ground truth distribution the dataset was sampled from.

Observations about Magnification

- We observe that as the magnification increases, the distribution skews further to the right.
- However, it appears that at 8x magnification, the low diameter fibers are actually being over represented.
- Future work would be to quantify this effect and adjust for it when combining samples across different magnifications.

Computational Topology

- We also used tools from persistent homology to characterize different types of realistic fiber structures.
- Real life filters have different morphologies and these morphologies may be characterized by persistence diagrams.
- Specifically, we track Betti numbers, which characterize topological spaces, as the image is filtered in gray scale.
  - The 0 Betti number is the number of connected components
  - The 1 Betti number is the number of closed cycles.
- Code used was from the Perseus software project: http://people.maths.ox.ac.uk/nanda/perseus/index.html

We'll show movies after the slides.

Summary

Approaches:

- Neural Networks
- Radon and Hough Transform
- 1D Slicing
- Persistent Homology

References

B. Pourdeyhimi and R. Dent (1999)
Measuring Fiber Diameter Distribution in Nonwovens
Textile Res. 69(4), 233–236

Carsten Steger (1998)
An Unbiased Detector of Curvilinear Structures
IEEE Transactions on Pattern Analysis and Machine Intelligence 20(2), 113–125

Richard O. Duda, Peter E. Hart (1971)
Use of the Hough Transformation to Detect Lines and Curves in Pictures
Artificial Intelligence Center

D.H. Ballard (1981)
Generalizing the Hough Transform to Detect Arbitrary Shapes
Pattern Recognition, 13(2), 111–122

Elias M. Stein, Rami Shakarchi (2003)
Fourier Analysis: An Introduction
Princeton University Press

Acknowledgements

- We thank Zhenyu He, Marina Chugunova, Sunil Dhar, David Edwards, Richard Moore, and others for helpful discussions.
- We thank NSF, MPI, W.L. Gore and Associates, and NJIT for their support.

Questions?