1. Reconstruction of capillaries in porous aluminium in 3D

The following procedure was used to identify pores in XRM recordings of partially infiltrated microporous aluminium and visualize the results.

1.1 Segmentation using cluster tracking

Data registration & pre-processing:

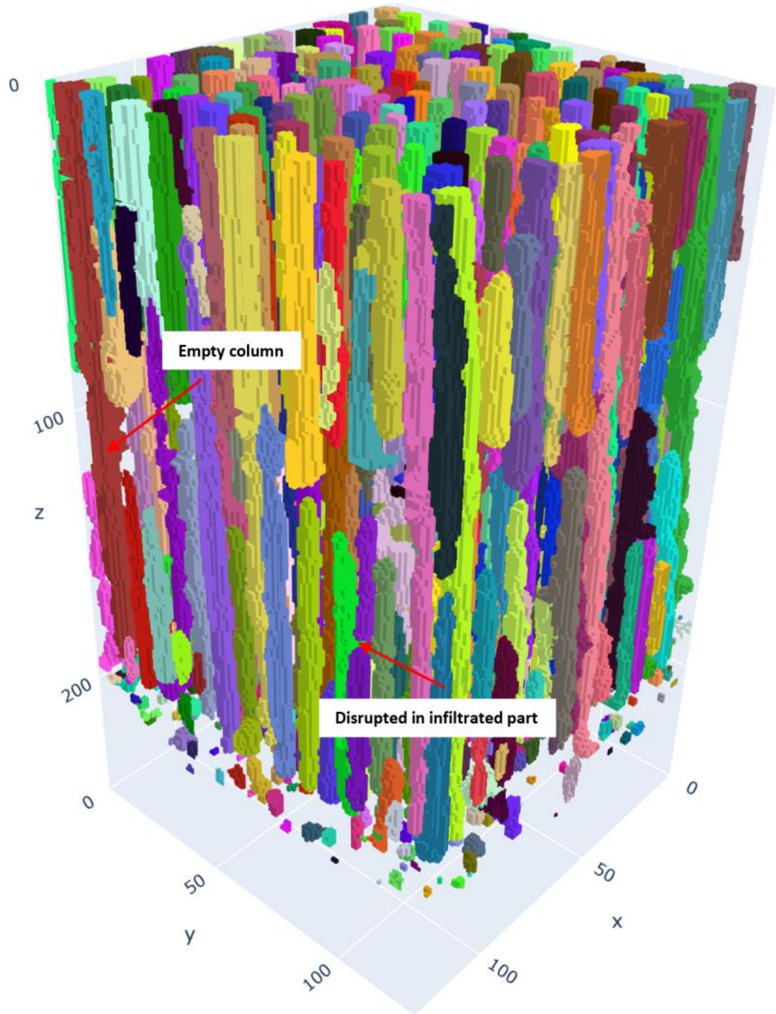
1. Rotation of the tensor such that columns are approximately of vertical orientation (parallel to Z axis)
2. Clipping of values (Assign maximal intensity to voxels intensity above threshold)
3. Gaussian smoothing along Z-axis (to ensure that potentially disconnected clusters of pixels are connected)

Segmentation & Cluster tracking (Pseudocode):

```
coords_t0 = fast_density_clustering(tensor[0]) # Detect cluster centers of dark pixels that represent pores

mappings = [ ] # A list of dictionaries that maps the label-values of slice[i] to the label-values of slice[i+1]

preliminary_label_tensor = [] # List of independantly labeled slices


For each 2D slice in tensor[1:] along z-axis:

        coords_t1, preliminary_labeled_slice = fast_density_clustering(slice)

        mappings.append(assign_closest(coords_t0, coords_t1))# Assign closest cluster center by checking within neighborhood

        preliminary_label_tensor.append(preliminary_labeled_slice)


For each position in mappings:

        output_tensor = empty_tensor(of same size as preliminary_label_tensor)

        current_index = 0

        Find matching cluster centers for subsequent slices, do recursively:

                If there is no subsequent layer or sufficiently close cluster: Stop

                Delete assigned label from available cluster centers along the way

                Mark voxels in output_tensor accoring to current index

                current_index ++

# The result is a labeled output_tensor where the label_value of the voxels is the same across slices for a column

# There are no biforcations / conjunctions but each slice of pore pixels has exactely one successor cluster in the next slice
```

1.2 Visualization in the form of an interactive 3D voxel plot

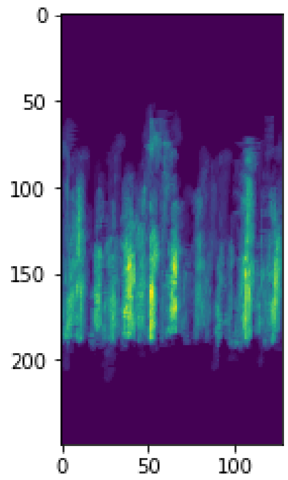
- I implemented a tool to convert voxel surfaces into 3d meshes
  - Surface voxels retrieved using binary operators
  - Distinction of top/bottom, back/front and left/right planes
  - Determine rectangles (low-poly) for each plane
- These 3d meshes can be plotted using Plotly for Python
- The plot is interactive (rotation, zoom etc.) and can be used with a Webbrowser
- Color-coding of different clusters is possible



Here the interface between the polymer and aluminium is on the bottom (Z>200). The infiltration front is around pixel position 100. Note that there are empty columns that range from the bottom to the top. Also, one can see that infiltrated columns are detected. In most cases they reflect the position of the imbibition front. However, sometimes they are disrupted within the infiltrated part.

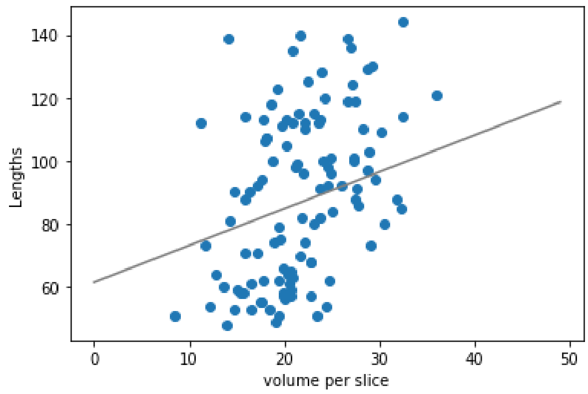
1.3 Correlation analysis of pore properties

- Having a labelled tensor of pores allows to analyse properties and put them into perspective
- Here I tried to find out if the pore diameter is a factor that potentially explains the roughness of the imbibition front (different penetrations depths for different columns)
- Fist I selected those pores with a lower end around the interface between polymer and aluminium (Z>180) and where the upper end is around the apparent imbibition front (Z>50 and Z<140). As empty pores reach up to the very top (Z=0) they are not in the selection.
- The following depiction shows the mean along dimension X for the selection:



- Next, I computed the volume per pore-slice and its length and correlated the values

Result:



For the given selection of pores there is a correlation between the lengths (pixels) and the average volume per pore-slice (r~.4).

Discussion:

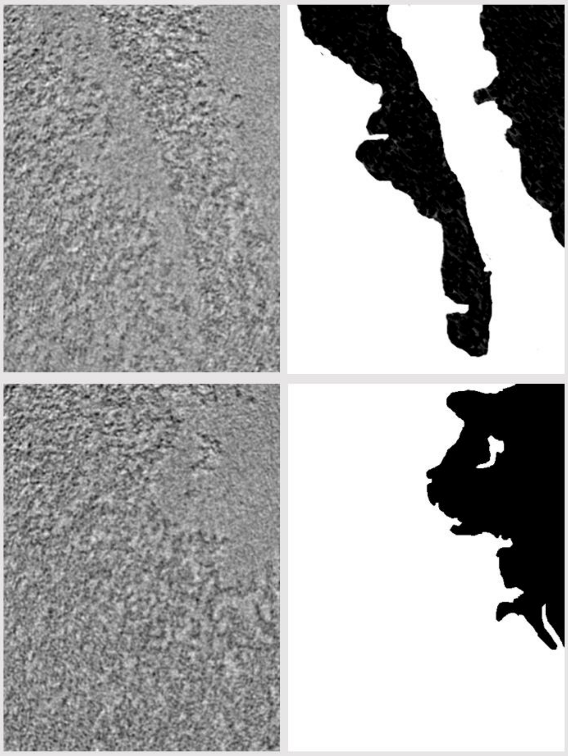
It is possible that the penetration depth is smaller for smaller diameter pores or the smaller diameter pores are physically disrupted. However, it could also be that smaller diameter pores are incorrectly detected as being discontinuous because they are harder to detect.

2. Towards texture-based segmentation of CPG

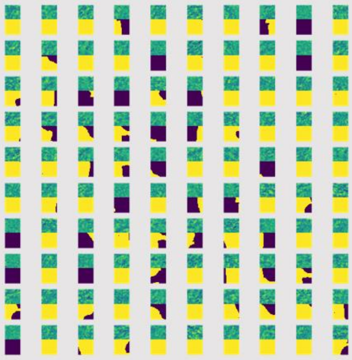
- CPG can be characterized by a spongy structure of pores
- As compared to capillary columns that rely mostly in parallel there is a network of 3D channels
- The interaction of the pore structure and the infiltrating substance is potentially more complex
- While material properties such as the tortuosity were measured using XRM recordings already in 2007 the characterization of infiltrated material is more challenging

Texture based segmentation of infiltrated and empty volumes using manually labelled training data?

1. Examples for raw data and labeled data



2. Example for sampled patches of textures



3. Implementation of machine learning classifiers

→ Feedforward CNN?
→ CNN-Autoencoder?

4. Training on test data and automatic segmentation

5. Detection of pores in empty/filled segments

Work in progress.