

Dear Dr. Thomas Poulet,

Today, I have uploaded a revised version of our manuscript gmd-2018-335 title “CobWeb 1.0: Machine Learning Tool Box for Tomographic Imaging”. As mentioned in our replies to the respective referees, we have implemented all the changes. We think these updates to the manuscripts have considerably improved it quality.

In a rough overview the updates are as follows (more details can be found in the referee-replies and respective changes in the manuscript):

- 1) Both referees, emphasized the structure of the manuscript was not suitable and referee 2 suggested a scheme to structure the manuscript and further pointed an absence of discussion section. This suggestion was taken into consideration and the structure has been changed accordingly.
- 2) Referee 2 found, that the REV section was out of the scope of the study. We agree to his comment and have removed the section. Both the referees enquired about the selection procedure of the REV and segmentation scheme. In the result and discussion section, subsection 5.1. (data selection) elaborated on the REV selection and subsection 5.2 (data processing) discusses the 2D slice by slice segmentation.
- 3) Referee 2 suggested that the role of ML to improve multiphase segmentation needs to be addressed. This is included in section result and discussion subsection 5.3 (multiphase image segmentation), where suggestions are given to preserve the intermediate phase, which gets misclassified. As enquired by referee 1, the role of placing centroid centers in improving the performance (accuracy and speed) of ML techniques has been addressed in subsect 5.3. (multiphase image segmentation).
- 4) As pointed by referee 2, Figure 1 has been updated; CobWeb user manual page 55 section 6.2 (Validation), following line “[The validation analysis is not a direct validation of segmentation but is rather to verify the performance of the machine learning algorithms](#)” has been included for clarification. License file has been included. The Introduction section, includes relevance of CobWeb for the XCT community (reply to referee 1 comments RC1.P3).

In the manuscript, the updated [passages](#) are colored in [blue](#) and the deleted lines are strikethrough in [red](#).

Best Regards

Swarup Chauhan

We thank the referee for the time and effort to review our manuscript. Below, we give point-by-point response to your concern.

Sincerely,

Swarup Chauhan, Kathleen Sell, Frieder Enzmann, Wolfram Rühak, Thorsten Wille, Ingo Sass, Michael Kersten.

Anonymous Referee #1

The reviewer comments are formatted in italics and the authors response to the comments are formatted in bold.

Notation *RC1.P#* represents ReviewersComment.ParagraphNumber

RC1.P1 Manuscript GMD-2018-335 presents a new collection of image processing and analyses tools for tomographic 3-D images. As the title suggests, the implementation of machine learning tools for image segmentation is in the focus of the manuscript. It is also claimed that the presented imaging software would be particularly suited for identifying representative elementary volumes.

Yes. Apart from that, the manuscript also highlights a new procedure termed as, dual filtering and dual segmentation to remove edge enhancement artefact in synchrotron based images using machine learning approach. To our knowledge this hasn't been published before.

The scientific community will benefit from the novelty of this approach. The code has been made available. We can certainly elaborate this in the introduction of the manuscript.

Not revised

RC1.P2 The connection of this paper to models is only weak, in form of the REV detection. However, the manuscript does not elaborate on how REVs are identified. It simply states that "voxel sizes around 480^3 suited best for..." , without explaining how this conclusion was reached.

OK. In the revised manuscript we will extend the section 3.3 and elaborate on the identification of REV for all the three samples.

Basically, it was a combination of visual inspection and consecutively segmenting and plotting trends in relative porosity, pore size distribution and volume fraction. This was done by loading the complete stack in the CobWeb software, during the loading process a 2D movie of the tomogram is displayed in the display window and saved in the root folder. Carefully monitoring the movie gives an objective evaluation of the heterogeneity of the respective XCT sample. Thereafter, based on this subjective information different ROIs are selected, cropped, segmented and their respective geometrical parameter are intercompared.

In the case of Berea sandstone, four different ROIs were investigated, whereas Grosmount carbonate rock seven different ROIs where need to identify the best REVs. Through our previous scientific studies on the GH sediments (Sell et al., 2016; Sell et al., 2018) we were aware of best-suited REVs. The identification of best REV for Grosmount was relatively tedious compared to Berea sandstone and GH sediment; due to the low resolution and microporosity present in the Grosmount tomograms.

Revised page 18 | line 13 - 23

The intention of showing particularly only two REV trends of relative porosity in Figure 7. is due to a very good agreement in porosity values to the benchmark publication of Andrä et al., (2013a, 2013b).

RC1.P3. The presented software appears to be a promising piece of work, but as the authors write themselves, it still has limited capabilities. The maybe most innovative part is implementation of machine learning routines for segmentation, albeit this in itself is not a scientific novelty.

The software is built on scientific studies which have been peer-reviewed and accepted in the scientific community Chauhan et al., 2016a,b. The spinoff for these studies was not the lack of accuracy provided by manual segmentation schemes, but the subjective assessment and non-comparability caused by the individual human assessments. Therefore, the automated segmentation schemes offer speed, accuracy and possibility to intercompare results, enhancing traceability and reproducibility in the evaluation process. To our knowledge none of the XCT software used in rock science community relies on machine learning for segmentation explicitly, which makes the software unique if not novel.

Despite many review articles and scientific publication highlight potential of machine learning and deep learning (Iassonov et al., 2009; Cnudde and Boone, 2013; Schlüter Steffen et al., 2014), software libraries or toolbox are seldom made available. Thus, with CobWeb we started for the first time to fill this gap, and despite its limited volume rendering capabilities— it is a useful tool and current version of the software can be applied in scientific and industrial studies. Certainly a conscious decision need to be taken on our side if to dedicate CobWeb as a segmentation tool or expand it towards simulation software like MATH2MARKET, GeoDict or Volume Graphics. On the other side CobWeb provides an appropriate test platform, where new segmentation and filtration schemes can be tested and used as a complementary tool to the simulation software GeoDict and Volume Graphics. The simulation softwares (GeoDict and Volume Graphics) have benchmarked solvers for performing flow, diffusion, dispersion, advection type simulation, but their accuracy relies heavily on the finely segmented datasets.

Revised page 2 | line 11-26

RC1.P3 The manuscript is moreover vague when it comes to describing how the different machine learning options are implemented (with exception of the K-means clustering). It is neither explained how the cross-validation option function.

We acknowledge reviewers concern. But, in section 2.4.2 (page 5) we have cited Chauhan et al., (2016a); Chauhan et al., (2016b) which covers the details about the algorithms and cross-validation schemes. Also, Figure 3 gives a visual overview how the ML techniques fit into the framework. Thereafter, the user manual published on the zendo repository (<https://dx.doi.org/10.5281/zenodo.2390943>) explains the implementation and how to use the segmentation algorithms. Within the scope of the manuscript we find the description sufficient.

But if required, we will expand further on the implementation of the ML techniques and the cross-validation options.

Revised:

2. Image processing subsection page 3 | line 11-17; page 5 | line 1-23

Section 2.2 Image Segmentation page 4 | line 24 to – to Page 9 | line 19

Section 2.3 Performance page 7 | line 19 – to page 9 | line 1

RC1.P4 The manuscript is written in adequate English. Its structure could be improved (e.g. the description of the workings of the filters do not belong in the materials and methods).

Ok. The description and working of filters is written under the sub section 3.1 image processing not under materials and methods.

Revised :

2. Image processing subsection page 3 | line 11-17; page 5 | line 1-23

RC1.P4 At several points I found the manuscript rather inconcise. It is e.g. not explained whether the filters and segmentation approaches work in 3-D or only in 2-D.

We thank the reviewer for pointing it out. The CobWeb 1.0 uses a slice-by-slice 2D approach. It was observed that the ML techniques tend to underestimate porosity values compared to manually segmented analysis at an REV scale size $> 500^3$. This substantial degree of uncertainty is caused due to 2D slice-by-slice processing rather than the ML techniques. The 2D slice-by-slice approach, passes only, the spatial information (X, Y coordinate direction) to the ML algorithms, which ends up sorting the intensity variation in the spatial domain (local maxima). Therefore, the lack of temporal information (Z coordinate direction) restricts the degree of freedom to find at a global spatial-temporal optimum. In other words, as the temporal changes arise, due to bedding (sedimentary rock) or micro porosity (carbonate rocks) in the rock texture, they are represented as sudden spike or dip in porosity values; which to an inexperienced eye appear as artefact or anomalies– and often-then-not discarded.

This correction will be implemented in the next software version; in the current workflow it has not been accounted for (CobWeb 1.0). Since, it requires refactoring the loop-based scalar-oriented framework to matrix and vector operation approach called *vectorization*. The 2D slice-by-slice processing scheme is much faster compared to the 3D approach. So, the choice of 2D processing for this research study was made to make it affordable to compute on desktop, laptop for near real-time and onsite evaluation.

Revised: page 16 | line 19-31

RC1.P4 Or like at p9L19: what choice of the cluster centers influence the performance of the K-means algorithm? Its initial location? The number of clusters?

Thanks again, for raising the question, we can certainly elaborate on this in the discussion section. In general, performance in terms of accuracy and speed is directly proportional to starting point (initial location) in the segmentation process. Meaning, the closer the starting point (initial location) is to the

global minima— faster will the algorithm converge and even so better is the performance (accuracy & speed).

However, in unsupervised technique by default the choice of the starting point is through random seed unless explicitly specified. So, in the case of the dual segmentation approach, the intuition was to capture all the material phases, including the edge enhancement artefact, speck and noise etc. in the first step and thereafter in the second step to rescale them to the plausible phases.

Hence, in the first step the 20 clusters were initialized using random seed. Since, the priority was to capture all phases in GH tomograms not the performance. And, after the rescaling processes, we were aware of the initial locations which we used as starting point (initial location) to assist the algorithm to move towards identifying correct phases.

Revised: Page 17 | line 21 - 33

RC1.P5 In summary, this manuscript presents a promising software tool for tomographic image analyses. We thank the reviewer for the acknowledgement.

RC1.P5 But I do not think the manuscript fits within the scope of GMD, nor do I think that the manuscript is developed enough to reward revisions with another round of reviews.

We disagree with the reviewer on the above comment. The uniqueness of this journal is that, it gives the possibility of accepting six different types of manuscripts, and we were careful in placing the work in the *model description papers* category as it fulfils most of its norms if not all. This has been clearly highlighted in the cover letter to the topical editor.

References

- Andrä, H., Combaret, N., Dvorkin, J., Glatt, E., Han, J., Kabel, M., Keehm, Y., Krzikalla, F., Lee, M., Madonna, C., Marsh, M., Mukerji, T., Saenger, E. H., Sain, R., Saxena, N., Ricker, S., Wiegmann, A., and Zhan, X.: Digital rock physics benchmarks—Part I: Imaging and segmentation, Benchmark problems, datasets and methodologies for the computational geosciences, 50, 25–32, doi:10.1016/j.cageo.2012.09.005, 2013a.
- Andrä, H., Combaret, N., Dvorkin, J., Glatt, E., Han, J., Kabel, M., Keehm, Y., Krzikalla, F., Lee, M., Madonna, C., Marsh, M., Mukerji, T., Saenger, E. H., Sain, R., Saxena, N., Ricker, S., Wiegmann, A., and Zhan, X.: Digital rock physics benchmarks—part II: Computing effective properties, Benchmark problems, datasets and methodologies for the computational geosciences, 50, 33–43, doi:10.1016/j.cageo.2012.09.008, 2013b.
- Chauhan, S., Rühaak, W., Anbergen, H., Kabdenov, A., Freise, M., Wille, T., and Sass, I.: Phase segmentation of X-ray computer tomography rock images using machine learning techniques: an accuracy and performance study, *Solid Earth*, 7, 1125–1139, doi:10.5194/se-7-1125-2016, 2016a.
- Chauhan, S., Rühaak, W., Khan, F., Enzmann, F., Mielke, P., Kersten, M., and Sass, I.: Processing of rock core microtomography images: Using seven different machine learning algorithms, *Computers & Geosciences*, 86, 120–128, doi:10.1016/j.cageo.2015.10.013, 2016b.

- Cnudde, V. and Boone, M. N.: High-resolution X-ray computed tomography in geosciences: A review of the current technology and applications, *Earth-Science Reviews*, 123, 1–17, doi:10.1016/j.earscirev.2013.04.003, 2013.
- Iassonov, P., Gebrenegus, T., and Tuller, M.: Segmentation of X-ray computed tomography images of porous materials: A crucial step for characterization and quantitative analysis of pore structures, *Water Resour. Res.*, 45, doi:10.1029/2009WR008087, 2009.
- Schlüter Steffen, Sheppard Adrian, Brown Kendra, and Wildenschild Dorthe: Image processing of multiphase images obtained via X-ray microtomography: A review, *Water Resour. Res.*, 50, 3615–3639, doi:10.1002/2014WR015256, 2014.

Dear Dr Kirill Gerke,

Thanks for your time, praise for our work and useful comments. Your comments and suggestions gave us some fresh ideas and a new perspective on the topic and helped us to improve our manuscript. Please find below a point-by-point response to your concerns.

The reviewer comments are formatted in italics and the author's response to the comments are formatted in bold.

Notation *RC2.P#* represents Reviewers Comment. Paragraph Number

RC1.P1: The paper of Chauhan et al. presents a tool developed by the leading author through a series of papers. This particular manuscript, unlike its predecessors, focuses on presenting the toolbox for already existing methods. I am sorry to post this review late, i guess this qualifies me as being a bad referee, but it took me quite a while to absorb everything, plus I struggled with trying to run the toolbox. On the plus side I got two reviews already posted and, thus, can rely on them to put in my vision here. I believe that topic of the paper - development of a free software to process XCT tomography images of porous media, is a very relevant topic. We do not have a solution, and those available as commercial solutions are expensive and not doing their job even closely. Scientific quality of paper is good, it is in general robust except for some particular parts

Thanks for the trust and the praise.

RC1.P1: Scientific quality of paper is good, it is in general robust except for some particular parts picked up by Reviewer1 and now here by Reviewer2. Scientific reproducibility was a problem for me. First, i had to borrow my wife's laptop, as mine is running under Linux. She has Matlab 2016b installed - the package is not working with it. Downloading Matlab runtime took time, as it is 1.8 Gbs of an archive. After reading the license i got puzzled - totally agree with David Ham at this point. Moreover, it was not clear how much disk space on the small SSD it will occupy - guessed it will be much more than 1.8 Gbs and i gave up. Presentation quality and writing are fine with me, i did not have any problem with these (except structure, see below) while reading the manuscript.

A possible solution is that we can send you the source code with the make file, so you may generate the GUI and/or test the source code in the Matlab environment. As mentioned in the code availability section (page 13 line 10) the GUI requires Matlab runtime compiler R2017b (9.3).

As addressed in the short comments from Dr David Ham we will include an appropriate licence file in the zendo repository.

Revised: Updated license file

Major comments:

1) REV In Section 3.3 you discuss classical REV idea of property convergence with increasing subsample volume. Yet, you do not explain how do you determine REV's for all your samples. Are these REV's were chosen in terms of porosity, pore sizes, etc.? If so, what was the threshold you used to stop increasing the volume?

As explained in the reply to the comments from reviewer 1 (RC1.P2): Basically, it was a combination of visual inspection and consecutively segmenting and plotting trends in relative porosity, pore size

distribution and volume fraction. We gradually increase the sub-sample volume at a different location (X, Y) and depth (Z) inside the XCT tomograms. And, for each subsample volume, the geometrical parameters are intercompared.

The main indicator, however, was the porosity trend; i.e. when regression coefficient R2 was close to zero, it was an indicator that the sub-volume has accumulated the heterogeneity along the z-axis of the sample. Therefore, based on the trend analysis approach, the sub-volume dimension where R2 value was close to zero was chosen as the suitable REV.

Revised: page 15 | line 13 – 30.

Moreover, here you write: "In particular, while performing permeability tensor simulation using XCT data, the size of minimum REV should be assessed not only based on porosity but also on geometrical parameters such as pore size distribution, void ratio, and coordinate number (Al-Raoush and Papadopoulos, 2010; Costanza & Robinson M.S. et al., 2011)." I did not re-read these papers to check if these authors stated exactly this, but for performing permeability tensor simulation using XCT data, the size of minimum REV should be assessed based on permeability tensor values! More generally, REVs do not necessarily exist, finding REV for one property has nothing to do with finding it for the other. The topic is very deep and is well beyond the scope of this manuscript. The possible ways to fix the REV problem here is to either remove it altogether, or explain for which property is it analyzed and calculated.

Thanks for the suggestion, The section 3.3 and the respective figure 6 has been removed P11 |line 4-21.

Removed P11 | line 4-21.

By the way, if the REV analysis of porosity is based on Fig.6 and 7, i.e. by calculating porosity within separate 2D slices, it is not appropriate, it should be done within increasing volumes.

In, the current version of CobWeb the porosity is calculated for 2D slices. The 2D slice-by-slice approach incorporated in CobWeb may lead to some uncertainties as the porosity variation is only accounted for along the Z-axis. However, by comparing different subsamples and accurately isolating (segmenting) the pore phase and calculating the mean porosity of the complete stack these fluctuations can be normalized.

As a consequence of your suggestion, now the porosity is calculated for a complete 3D stack. By indexing the pore phase pixels for the 3D stack and calculating the ratio between the pore phase pixel and matrix phase.

Not revised

2) Structure of the methods/results Again, i totally agree with Reviewer1 and believe that current structure is hard to follow. I would suggest putting the methods description first, next describe the toolbox and functionality, then the objects and specifics of their processing. So, the structure of the paper could be something like this: - Intro as it is - Methods: 1. Image processing algorithms (here you describe all filters and segmentation techniques in better detail with references) 2. Toolbox and its functionality (your current Section 2.4) 3. Objects and specs (here you describe a sample+its processing steps, next sample, ...) - Results: 1. you current section 3.1, as well as all descriptions of filters, equations and such, belongs more to methods, not results! 2. just describe what you got for each sample 3. you do not have discussion

We thank the reviewer for this suggestions. We have implemented it accordingly in the revised manuscript.

Revised:

2. Image processing subsection page 3 | line 11-17; page 5 | line 1-23

Section 2.2 Image Segmentation page 4 | line 24 to – to Page 9 | line 19

Section 2.3 Performance page 7 | line 19 – to page 9 | line 1

3) *Code/toolbox* The way you deliver your toolbox is not suitable for many researchers. Indeed, the code would be more useful for me. But i understand that Authors wanted a way to performs computations +deliver a GUI. And doing this with Matlab is much easier than using other tools. Simply giving away the code is not that useful for many users, because the code itself is much harder to use - you need to write a script to run something. GUI provide a possibility to use a couple of buttons and slider to do the job. Making a good GUI is very challenging. This is the reason we never released our in-house C++ image processing code with Qt GUI into public - we can use it, but for anybody else it will be buggy and not user-friendly. So, both releasing code and executables have their downsides, but in this case the need for Matlab runtime is a big obstacle for potential users. It should be relatively easy to port Matlab code into Python or Julia, which would avoid Matlab - slow and expensive dev-environment. In short, i understand why Authors decided to redistribute executables and not the code, but still see more value in free environments (python and julia) - this does not imply that Authors have to port the code as a part of the revision.

We thank the reviewer for his support and understanding. Despite the easy semantics provided by Matlab, development of the GUI in Matlab was not easy. We used tips and tricks from undocumented Matlab by (<https://undocumentedmatlab.com/>). The idea was to make it available for many researchers of the science communities, therefore GUI was a feasible option.

4) You write: "The NLM filter was implemented in 3D mode to attain desired spatial and temporal accuracy and was processed on a CPU device." But Manual states that "Non-local means is only implemented as a 2D filter because filtration is done slice-by-slice".

Yes. The NLM filter is hard-coded as 2D in the CobWeb standalone version (GUI). But, by tweaking or modifying the source code we could initially pre-processed the XCT images using NLM 3D filtration and thereafter subjected it to segmentation.

The information given above will be added to the discussion section and will also be highlighted in the conclusions

Revised page 19 | line 20-23

5) Multiphase segmentations

Any segmentation method can be made multiphase, be it indicator kriging, converging active contours or region growing. But machine learning algorithms used in this paper do not solve the major problem for all multiphase segmentations - the phase having intermediate grey scale values gets sandwiched between the other two phases, e.g., see numerous fugues in the paper and in the manual. I would like you to

elaborate on this and why do you think clustering and k-means approaches are better than aforementioned techniques?

Thanks for your suggestion, certainly we have elaborate on this in the manuscript in the discussion section.

In a practical sense, machine learning tries to separate grayscale values in to disjoint sets. The creation of these disjoint sets can be created in two ways.

1) By binning the greyscale values to the nearest representative values which is iteratively updated using an optimization function. This optimization function can be a simple regression or distance function (Jain et al. 1999), commonly used in the unsupervised techniques.

2) Or by regularizing pre-trained models which store certain pattern information of the datasets such as topology features, contour intensities, pixel value etc. (Hopfield 1982; Haykin 1995; Suykens and Vandewalle 1999). Or by using a voting system in a bootstrap ensemble of linear models (Breiman 1996).

So during this process, the intermediate greyscale values which have low sample size are merged in to larger sets of a high sample size to create disjoint boundaries.

One way to overcome this problem is by using supervised techniques such as LSSVM or Ensemble classifiers. When constructing a training dataset (feature vector selection), careful selection of intermediate phases as a sufficiently large sample size compared to the predominate phases will preserve the intermediate phases. And, the likelihood that the trained model will identify them and cluster them separately is higher (Chauhan et al. 2016).

In this study, in particular, we made tests using supervised techniques (LSSVM, Ensemble classifiers) and unsupervised technique (FCM) but the results weren't superior compared to K-means. Therefore we choose K-means as it was faster compared to other ML techniques.

Revised: page 17 | line 1-22

6) Verification In the Section 6.2 in the Manual your provide validation metrics to prove your segmentation results. I do not see how any of these are actually do the job! They are simply the metrics for performance of the machine learning algorithms, not the segmentation results. I strongly suggest remove or re-write this part.

We agree to the reviewers views. In the revised version of the manual it is rewritten that these are simply the metrics for performance of the machine learning algorithms not validation.

Revised: CobWeb manual page 55

We technically do not have verification data - the ideal segmentations, which can be produced only using synthetic tomography. You could discuss these things a little bit, so that comments 5-6) will give material for small discussion subsection you lack at the moment.

We thank the reviewers for his suggestion. We agree and it will incooperate it in the discussion section.

Revised: CobWeb manual page 55

Minor comments:

1) In introduction you write: "Our hypothesis is that 3D tomographic REV analysis of Berea SandstoneTM (BS), Grosmont carbonate rock (GCR), and gas hydrate (GH)-bearing sediment datasets, would benefit of this new approach." I find this statement scientifically awkward (even without considering the REV problems in this paper), moreover this hypothesis is nowhere to be tested.

As per reviewers suggestion, Page 1 line 15- 17 "Our hypothesis is that 3D tomographic REV analysis of Berea SandstoneTM (BS), Grosmont carbonate rock (GCR), and gas hydrate (GH)-bearing sediment datasets, would benefit of this new approach." has been removed from the manuscript

2) Section 3.1.1.2, you write: "In this study, the NLM filter was set to a search window of 21, local neighbourhood of 6 and a similarity value of 0.71." As i was unable to test the toolbox, it seems after reading the Manual that too many things are fixed. How useful is it going to be if these NLM and other parameters are not changeable within the toolbox?

Data preprocessing is a very important step before segmentation. Depending on the resolution of the dataset, the fixed parameters of NLM and other filters should do a fairly good job. In case, their still exists noise and artefacts it is recommended to use supervised techniques. The residual noise or artefact pixel values can be captured through proper feature vector selection, and their after training the appropriate model and performing classification. Thereby, the existing noise and artefact can be isolated and segmented as separate labels.

Another alternative option could be to pre-process the data with desired filters data and imported the data into CobWeb for segmentation and analysis.

Revised: page 16 | line 9-17

3) As far as i understood, technically you do not compute pore-size distribution (e.g., extract the pore-network model), but distance-map transform and analyse its distribution. If so, please, change the description of the methodology accordingly. Moreover, i do not understand the legends on the graphs, be it fig.8 in the Manual or fig.8 in the paper (what are these numerous pore-size distributions for the same sample?).

Yes, It will be revised in the manuscript.

As mentioned in (Rabbani et al. 2014), the aim is to breakdown the monolithic void structure of rock into specific pores and throats connecting to each other. Rabbani et al. (2014) used image processing algorithm (median filter) to perform pre-processing. In our case, the tomograms have already been pre-processed and segmented through ML techniques and thereafter subjected it to the watershed algorithm.

Page 18 | line 3 - 10

4) You mention a number of free software solutions. I strongly suggest adding our recent effort FDMSS software to simulate single-phase flow within 3D pore geometries into your list of tools that could be used after segmentation of 3D images using CobWeb.

Thanks for pointing, we will update the figure and cite the reference in the revised manuscript.

Revised

Moreover, you could compute permeabilities for your segmentations and compare them against available data, e.g., "Laboratory measurements of porosity and permeability reported by (Andrä et al., 2013b) are around 21 % ($\phi = 0.21$) and $\kappa = 150 \text{ mD} \hat{\sim} T \hat{\sim} A 470 \text{ mD}$, respectively".

Thanks for your suggestion, we will implement this in future studies.

Not revised

References

- Breiman, L., 1996. Bagging predictors. *Machine Learning* 24, 123–140. doi:10.1007/BF00058655.
- Chauhan, S., Rühaak, W., Anbergen, H., Kabdenov, A., Freise, M., Wille, T., Sass, I., 2016. Phase segmentation of X-ray computer tomography rock images using machine learning techniques: an accuracy and performance study. *Solid Earth* 7, 1125–1139. doi:10.5194/se-7-1125-2016.
- Haykin, S.S., 1995. *Neural networks: A comprehensive foundation*, Macmillan, New York, NY, 696pp.
- Hopfield, J.J., 1982. Neural networks and physical systems with emergent collective computational abilities. *Proc Natl Acad Sci USA* 79, 2554. doi:10.1073/pnas.79.8.2554.
- Jain, A.K., Murty, M.N., Flynn, P.J., 1999. Data Clustering: A Review. *ACM Comput. Surv.* 31, 264–323. doi:10.1145/331499.331504.
- Rabbani, A., Jamshidi, S., Salehi, S., 2014. An automated simple algorithm for realistic pore network extraction from micro-tomography images. *Neural network applications to reservoirs: Physics-based models and data models* 123, 164–171. doi:10.1016/j.petrol.2014.08.020.
- Suykens, J.A.K., Vandewalle, J., 1999. Least Squares Support Vector Machine Classifiers. *Neural Processing Letters* 9, 293–300. doi:10.1023/A:1018628609742.

[CobWeb 1.0](#) License statement and permissions for [CobWeb 1.0](#) package.
Copyright (C) 1993--2019

Citation

This software is distributed free of charge. If you use it in research resulting in a scientific presentation or publication, the software should be acknowledged and cited as:

Chauhan, Swarup, Sell, Kathleen, Enzmann, Frieder, Rühaak, Wolfram, Wille, Thorsten, Sass, Ingo, Kersten, Michael.
CobWeb 1.0: machine learning tool box for tomographic imaging
Zenodo : OpenAIRE. Vol. 2018, Issue December. Genève, : CERN, 2018.

License

[CobWeb 1.0](#) is distributed under the "Creative Commons License Version 4.0", available at <http://creativecommons.org/licenses/by-nc-sa/4.0/>

You are not allowed:

To reuse and redistribute the APS GmbH and Technische Universität Darmstadt Logos

You are free:

To Share -- copy and redistribute the material in any format
To Adapt -- remix, transform, and build upon the material

Under the following conditions:

Attribution -- You must give appropriate credit, provide a link to the license, and indicate if changes were made. You may do so in any reasonable manner, but not in any way that suggests the licensor endorses you or your use.

Non-commercial -- You may not use the material for commercial purposes.

Share Alike -- If you remix, transform, or build upon the material, you must distribute your contributions under the same license as the original.

See the above link for the full text of the license.

Disclaimer

This software is provided 'as-is', without any express or implied warranty. In no event will the author(s) be held liable for any damages arising from the use of this software.

CobWeb 1.0: Machine Learning Tool Box for Tomographic Imaging

Swarup Chauhan^{1,2}, Kathleen Sell^{1,3,*}, Frieder Enzmann¹, Wolfram Rühaak^{4,**}, Thorsten Wille⁵, Ingo Sass², Michael Kersten¹

5 ¹Institute for Geosciences, Johannes Gutenberg-University, Mainz 55099, Germany

²Institute of Applied Geosciences, University of Technology, Darmstadt 64287, Germany

³igem – Institute for Geothermal Resource Management, Berlinstr. 107a, Bingen 55411, Germany

⁴Federal Institute for Geosciences and Natural Resources (BGR), Hannover 30655, Germany

⁵APS Antriebs-, Prüf- und Steuertechnik GmbH, Götzenbreite 12, Göttingen-Rosdorf 37124, Germany

10

* now at: [Ministry of Economic Affairs Rhineland Palatine, Stiftsstrasse 9, Mainz 55116, Germany](#)

** now at: [Bundesgesellschaft für Endlagerung mbH \(BGE\), Peine 38226, Germany](#)

*Correspondence to: Swarup Chauhan (ssschauhan@gmail.com)

15 **Abstract.**

Despite the availability of both commercial and open source software, an ideal tool for digital rock physics analysis for accurate automatic image analysis at ambient computational performance is difficult to pin point. More often image segmentation is driven manually where the performance remains limited to two phases. Discrepancies due to artefacts causes inaccuracies in image analysis. To overcome these problems, we have developed CobWeb 1.0 which is automated and explicitly tailored for accurate grayscale (multi-phase) image segmentation using unsupervised and unsupervised machine learning techniques. The simple and intuitive layout of the graphical user interface enables easy access to perform Image enhancement, Image segmentation and further to obtain the accuracy of different segmented classes. The graphical user interface enables not only processing of a full 3D digital rock dataset but also provides a quick and easy region-of-interest selection, where a representative elementary volume can be extracted and processed. The CobWeb software package covers image processing and machine learning libraries of MATLAB® used for image enhancement and image segmentation operations, which are compiled into series of windows executable binaries. Segmentation can be performed using unsupervised, supervised and ensemble classification tools. Additionally, based on the segmented phases, geometrical parameters such as pore size distribution, relative porosity trends and volume fraction can be calculated and visualized. The CobWeb software allows the export of data to various formats such as ParaView (.vtk), DSI Studio (.fib) for visualization and animation and Microsoft® Excel and MATLAB® for numerical calculation and simulations. The capability of this new software is verified using high-resolution synchrotron tomography datasets, as well as lab-based (cone-beam) X-ray micro-tomography datasets. Albeit the high spatial resolution (sub-micrometer), the synchrotron dataset contained edge enhancement artefacts which were eliminated using a novel dual filtering and dual segmentation procedure.

20
25
30

1. Introduction

Currently a vast number of available commercial and open source software packages for pore-scale analysis and modelling exists (compiled in Figure 1), but dedicated approaches to verify the accuracy of the segmented phases are lacking. To the best of our knowledge, the current practice among researchers is to alternate between different available software tools and to synthesize the different datasets using individually aligned workflows. Porosity and in particular, permeability can vary dramatically with small changes in segmentation, as significant features on the pore-scale get lost when thresholding greyscale tomography images to binary images, even if using the most advanced data acquiring techniques like synchrotron tomography (Leu et al., 2014). Our new CobWeb 1.0 visualization and image analysis toolkit addresses some of the challenges of selecting representative elementary volume (REV) for X-ray computed tomography (XCT) datasets reported earlier by several researchers (Zhang D et al., 2000; Gitman et al., 2006; Razavi et al., 2007; Al-Raoush and Papadopoulos, 2010; Costanza-Robinson et al., 2011; Leu et al., 2014). ~~The software is built on scientific studies which have been peer-reviewed and accepted in the scientific community (Chauhan et al., 2016b; Chauhan et al., 2016a). The spinoff for these studies was not the lack of accuracy provided by manual segmentation schemes, but the subjective assessment and non-comparability caused by the individual human assessor. Therefore, an automated segmentation schemes offer speed, accuracy and possibility to intercompare results, enhancing traceability and reproducibility in the evaluation process. To our knowledge none of the XCT software used in rock science community relies explicitly on machine learning to perform segmentation, which makes the software unique.~~

Despite many review articles and scientific publication highlight potential of machine learning and deep learning (Iassonov et al., 2009; Cnudde and Boone, 2013; Schlüter et al., 2014), software libraries or toolbox are seldom made available. Thus, with CobWeb we started for the first time to fill this gap, and despite its limited volume rendering capabilities— it is a useful tool and current version of the software can be applied in scientific and industrial studies. CobWeb provides an appropriate test platform, where new segmentation and filtration schemes can be tested and used as a complementary tool to the simulation software GeoDict and Volume Graphics. The simulation softwares (GeoDict and Volume Graphics) have benchmarked solvers for performing flow, diffusion, dispersion, advection type simulation, but their accuracy relies heavily on the finely segmented datasets. ~~CobWeb 1.0 was customized to fill the gap by performing image analysis and accurate greyscale phase segmentation of reconstructed high-resolution XCT and synchrotron tomographic datasets.~~ This software is based on a machine learning approach with great potential for segmentation analysis as introduced previously (Chauhan et al., 2016b; Chauhan et al., 2016a). Further, this software tool package was developed on a MATLAB® workbench and can be used as Windows stand-alone executable (.exe) files or as a MATLAB® plugin. ~~Our hypothesis is that 3D tomographic REV analysis of Berea Sandstone™ (BS), Grosmont carbonate rock (GCR), and gas hydrate (GH) bearing sediment datasets, would benefit of this new approach. Although~~ The dataset for the gas hydrate sediment (GH) geomaterials was acquired using monochromatic synchrotron X-ray, unhampered by beam hardening; (Sell et al., 2016) highlighted problems with edge enhancement (ED)

artefact and recommended image morphological strategies to tackle with this challenge. In this paper, we describe therefore also a strategy to eliminate ED artefacts using the same dataset but applying the new machine learning approach.

2. Image Processing

Image pre-processing is one of the essential and precautionary steps before image segmentation (Iassonov et al., 2009; Schlüter et al., 2014). Image enhancement filtering techniques help to reduce artefacts such as blur, background intensity and contrast variation. Whereas, denoise filter such as median filter, non-local means filter, and anisotropic diffusion filter can assist in lowering the phase misclassification and improving the convergence rate of automatic segmentation schemes. CobWeb 1.0 is equipped with image enhancement and denoise filters, namely, *fspecial*, *imsharpen*, *non-local-means*, *anisotropic diffusion* and *fspecial*, which are commonly used in the XCT image analysis community.

2.1.1. Imsharpen Image Enhancement

Despite at the instrument level different measures can be taken to improve resolution of the X-ray volumetric data. The contrast, in the XCT images depends particularly on the composition and corresponding densities (optical depth) of the test sample, therefore it is somewhat difficult to enhance contrast at the experimental setup or at the x-rays system design control stage. Therefore, the contrast needs to be enhanced or adjusted after the volumetric image has been generated. For this purpose, image sharpening can be used. Image sharpening is a sort of contrast enhancements. The contrast enhancements generally takes places at the contours, where high and low greyscale pixel values intensities meet (Parker, 2010).

2.1.2. Anisotropic Diffusion Image Filtering

For intuition purposes, Anisotropic Diffusion filter (AD) can be thought as (Gaussian) blur filter. AD blurs the image, where it carefully smooths the textures in the image by preserving its edges (Kaestner et al., 2008; Porter Mark L. et al., 2010; Schlüter et al., 2014). To achieve the smoothing along edge preservation, the AD filter performs an iteration to solve non-linear partial differential equations (PDE) of diffusion:

$$\frac{\partial I}{\partial t} = c(x, y, t)\Delta I + \nabla c \cdot \nabla I \quad (1)$$

Where,

I is the image, t is the time of evolution and c is the flux which controls the rate of diffusion at any point in the image. (P. Perona and J. Malik, 1990) introduce a flux function c to follow an image gradient and stop or restrain the diffusion when it reaches the region boundaries (edges preservation).

Given by

$$c(\|\nabla I\|) = e^{-(\|\nabla I\|/\kappa)^2} \quad (2)$$

$$\|\nabla I\| = \frac{1}{1+(\frac{\|\nabla I\|}{\kappa})^2} \quad (3)$$

Here, the parameter κ is a tuning parameter that determines if the given edge to be considered as a boundary or not. Large value of κ lead to an isotropic solution and the edges are removed. For our investigations the **parameter** κ (threshold stop) was **fixed** to the value 22 968, which is the edge preservation **limit** between quartz grain and hydrate phase. The desired denoising (blurring/smoothing) was achieved within five iteration steps.

5 2.1.3. Nonlocal Means **Image Filtering**

Nonlocal means (NLM) filter is based on the assumption that the image contain an extensive amount of self-similarity (A. Buades et al., 2005; Shreyamsha Kumar, 2013). Based on this assumption (A. Buades et al., 2005) extended the linear neighborhood SUSAN filter (Smith and Brady, 1997) with nonlocal class. Thus, through the nonlocal class, the spatial search for similar pixel values is not restricted to a constrained neighborhood pixel but the whole image is part of the search for similar
10 pixel values. Given by the equation

$$NL(i) = \sum_{j \in I} w(i, j)v(j) \quad (4)$$

Where,

$NL(i)$ is the estimated nonlocal intensity of the pixel i

I is the image

15 $w(i, j)$ is the weight (or average value) applied to noisy image $v(j)$ to obtain and restore the pixel i .

However, for practical and computational reason, the search is performed within a search window or neighborhood patches, and $w(i, j)$ evaluates similarity in pixel intensities between of local neighborhood patches. Where the weight $w(i, j)$ is calculated as

$$w(i, j) = \frac{1}{Z(i)} e^{-\frac{\|v(N_i) - v(N_j)\|_{2, \sigma}^2}{h^2}} \quad (5)$$

20 Where. $Z(i)$ is a normalization constant

$$Z(i) = \sum_j e^{-\frac{\|v(N_i) - v(N_j)\|_{2, \sigma}^2}{h^2}} \quad (6)$$

$v(N_i)$, $v(N_j)$ are the local neighborhood patches.

The similarity is fulfilled as the Euclidean distance between the local neighborhoods patches exponentially decreases.

$\sigma > 0$ is the standard deviation.

25 In eq. (5) and (6), the distance function $\|v(N_i) - v(N_j)\|^2$ is pointwise multiplied (convolved) with σ , to ensure fair contribution of pixel values to the weighted function.

2.1.4. fspecial Image Filtering

fspecial helps in creating 2D high pass and low pass filters. High pass filters are used for sharpening and edge detection, whereas low pass filters are used for smoothing the image quality. Frequently used high pass filters are Laplacian and Sobel masks (kernel), and most often used low pass filter is the Gaussian smoothing mask (mask).

5 Laplacian mask can be mathematically formulated as

$$L(x, y) = \frac{\partial^2 I}{\partial x^2} + \frac{\partial^2 I}{\partial y^2} \quad (7)$$

Where $L(x, y)$ is the Laplacian of pixels with intensity $I(x, y)$

The first and second derivative implies that the masking is done twice, adding the first mask to the original image enhance thick edges in the image, and further adding the mask second time to the modified images sharpens fine details of the image. Thereby, sharpening edge contours of different intensities which were previously blurred in the original image. The Sobel filter eases edge detection in horizontal and vertical direction in regions of high pixel gradient (edges).

Whereas, Gaussian 2D smoothing operator is used to smooth images in order to remove details and noise. This is done by adding a mask (kernel) to the original image. This transformation stretches the pixel values of the original image to an extent that they appear smoothed. The mask is a 2D Gaussian (circular symmetric) operator mathematically formulated as

$$g(x, y) = \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{d^2}{2\sigma^2}} \quad (8)$$

$$d = \sqrt{(x - x_c)^2 + (y - y_c)^2} \quad (9)$$

20

Where (x, y) are the pixels, σ is the standard deviation in the intensity of the pixels, d is the distance calculated between the neighborhood pixels to the center of the pixel (x_c, y_c) . So, the higher the σ standard deviation of the pixel intensities the smoother is the transformed image.

2.2. Image Segmentation

25 Digital image comprises of pixels of color or greyscale intensities. Image segmentation is partitioning or classification the pixel intensities into disjoint regions that are homogenous with respect to some characteristics (Bishop, 2006). There is continuous research efforts done in various international groups to improve and develop image segmentation approaches (Mjolsness and DeCoste, 2001). In particular, the most popular and relevant image segmentation approaches for analyzing X-ray tomographic rock images is presented in the review studies done by (Iassonov et al., 2009) and (Schlüter et al., 2014). We use machine learning techniques for image segmentation and have implemented algorithms such as K-means, Fuzzy C-means (unsupervised), least square support vector machine (LSSVM) (supervised), bagging and boosting (ensemble classifiers) for

automatic segmentation Chauhan et al., (2016a), (Chauhan et al., 2016b) and references within. The performance of these machine learning technique can be assessed by matrices such as, entropy, receiver operational characteristics (ROC), 10-fold crossvalidation (Chauhan et al., 2016b). Below all the above mentioned algorithms are discribed in brief.

2.2.1. Unsupervised Machine Learning Techniques

- 5 K-means is one of the simplest, yet, robust unsupervised machine learning (ML) algorithms commonly used in partitioning data (MacQueen, 1967; Jain, 2010; Chauhan et al., 2016b). Through an iterative approach the K-means algorithm computes the Euclidean distance between the data points (pixel value) to its nearest centroid (cluster). The iteration converges when the objective function, i.e. the mean square root error of Euclidean distance, reaches the minimum. This is, when each of the pixel in the dataset is assigned to its nearest centroid (cluster). However, the K-means algorithm has the tendency to converge at
- 10 local minima without reaching the global minimum of the objective function. Therefore, it is recommended to repeatedly run the algorithm to increase the likelihood that the global minimum of the objective function will be identified. The performance of the K-means algorithm is influenced predominantly by the choice of the cluster centers (Chauhan et al., 2016b).

The Fuzzy C-means (FCM) clustering procedure involves minimizing the objective function (Dunn, 1973)

15
$$J_{fcm}(Z; U; V) = \sum_{j=1}^n \sum_{i=1}^k (\mu_{ij})^m \|x_i^{(t)} - c_k\|^2 \quad (10)$$

where $c_k = \sum_{j=1}^n u_{ij} x_i$

c_k is the k^{th} fuzzy cluster centre, m is the fuzziness parameter, m . u_{ij} is the membership function.

- Unlike k-means, FCM performs a sort of soft clustering, in the FCM iterative scheme each data point can be a member of multiple clusters (Jain, 2010 and Jain et al., 1999). So, for example, a tight sandstone rock sample with a mean diameter and
- 20 height of 4-5 cm respectively, with high density and relative porosity less then 6% will result in XCT tomograms of voxel resolution of $\sim 13 \mu\text{m}$ with very low contrast. The low contrast, makes it difficult to identify pixel gradients. Since an intermediate phase pixel in between the gradient exists. These intermediate phases pixels in physical sense can represent connectivity in between the pores. If Kmeans is used, it performs a conservative segmentation and closes such connectivity by classifying them with absolute low intensity pixels (pores). Therefore, it can be beneficial to used FCM algorithm. FCM has a
- 25 membership function and by selecting different membership values, the distance function can be regularized “loosely” or “tightly” and the intermediate pixel (connectivity) can be preserved. However, it is essential to test different combination of membership values with several centroid centres (segmentation classes) to obtain reliable results.

2.2.2. Supervised Machine Learning Technique

- Similar to unsupervised techniques, the objective of supervised machine learning technique is to separate data. The advantage
- 30 supervised technique offers compared to unsupervised technique is that, it is effective in separating non-linear separable data

(Haykin, 1995; Bishop, 2006). In supervised technique the prediction is made by a model. The model is a mathematical function, which fit a line or a plane in between linearly or non-linearly separable data to classify them into different categories. The models ability or intuition, where to place the line or plane in between the dataset to clearly separate (classify) them– is based on its (model) apriori knowledge of the dataset– this apriori knowledge is called the training dataset. Therefore, unlike
5 unsupervised technique, the supervised model needs be trained on a subset of the dataset. The training dataset, is the only ‘window’ through which, the model knows some pattern about the linear or non-linear separable dataset. How good, the model has acquired the knowledge of the training dataset, determines its success in prediction. If it has learned the training data accurately, it picks up noise along with the pattern and loose it ability of generalization, thus fails when introduce to (unknow) separable-dataset. Alternatively, if the model has ‘vaguely’ learnt the training dataset; could be due very less training dataset,
10 or if the model is to simple or complex to learn. This could also lead to the failure in prediction. Therefore, to manage a good tradeoff, cross-validation techniques are used to monitor the learning rates of the model (Haykin, 1995).

Support vector machine (SVM) (Haykin, 1995) and its modified version least square support vector machine (LSSVM) are one such category of supervised ML technique (Suykens and Vandewalle, 1999) and use the principles mentioned above. The
15 plane separating the data is termed as hyperplane. The hyperplane has a boundary around it, which is called the *margin* and the data points that lie closest or on the margin are called the *support vectors*. The width of the margin governs the tradeoff, i.e. if the model is overfitted or underfitted to the training dataset; and can be verified through cross-validation techniques. If the width of the margin is two narrow (high learning rate), the model is overfitted (high variance) to the training dataset and will loose it generalization capability and may not separate the linear or non-linear separable (unknow) data accurately. If the
20 width of the margin is too wide (very low learning rate), the model is underfitted (high bias) to the training dataset and will fail. Optimal learn model has a just the appropriate width, to maintain the generalization and also learn the patterns in the dataset.

Now, if the training dataset is non-linear and inseparable in 2D coordinate system, it is useful to project the dataset in 3D coordinate system– thus, by doing so, the added dimension (3D) helps to visualize the data and find a place fit an hyperplane
25 to separate them (T. M. Cover, 1965). So, SVM and LSSVM use the principle of cover theorem (T. M. Cover, 1965) to project the data in to higher dimension to make them linearly separable and transform them back the original coordinate system (Suykens and Vandewalle, 1999). Hence what type of projection is to be performed by the SVM or LSSVM is done by choosing appropriate kernel function (van Gestel et al., 2004). This gives them the capability to attain the knowledge of the data and also preserve the generalization behaviour of the model or the classifier. In the original or the 2D coordinate system, the
30 hyperplane, is no longer a line but a convex shaped curved which has clearly, separated the data and suitable margins to the support vectors.

2.2.3. Ensemble Classifiers Technique

As the name implies ensemble classifiers is an approach, where the decision of several simple models is considered to improve the prediction performance. The idea behind using ensemble methods emulates from a typical human approach of exploring several options before making a decision. The ensemble technique is faster compared to supervised techniques. Basically, the evaluation of the decisions predicted by the simple models can be either done sequentially (Bragging or Boosting) or in parallel (Random Forest). Our toolbox used the sequential approach with a variation Bragging and Boosting for classification. These Bragging and Boosting used tree learners (Seiffert et al., 2008; Breiman, 1996), inherited from the MATLAB® libraries.

The main difference between Bragging and Boosting are as follows. Bragging generates a set of simple models, first trains these models with the random sample and evaluates the classification performance of each model using the test subset of data.

In the second step only those models whose classification performance was low are retrained. The final predictive performance rate of the Bragging classifier– is an average of individual model performances. This approach minimizes the variance in the prediction–, meaning if several Bragging classifiers are generated from same sample of data–, their prediction capability when exposed to unknow dataset will not differ much. The main different of Boosting to Bragging is that; Bragging retrain selected models (high misclassification rate) with the complete training dataset until their respective accuracy increases. Whereas, in Boosting the size of the data which has been misclassified in increased in ratio to the data which has been accurately classified– and thereafter *all* the models are retrained sequentially. The predictive performance is calculated same as in Bragging by averaging the predictive performance rate of the individual models. This approach of Boosting minimizes the bias in the prediction.

2.3. Performance

It is necessary to monitor the performance of an ML model. This ensures that the trained modelled does not overfit or underfit with the training dataset. The main reason for overfitting and underfitting of the model with the training dataset is directly proportional to the complexity of the ML models. However, the consequence is that an overfitted trained ML model will capture noise along with the information pattern from the training dataset and will lose its ability of generalizability. Hence leading to inaccurate classification when exposed to unknow dataset; as it has high variance toward the training dataset. On the opposite side, when the ML model underfit with the training dataset, it is unable to learn or capture the essence of the training dataset; this can happen either due to a choice of a simple type model (E,g linear instead of quadratic) or very less amount of data to build a reliable model. As a consequence the ML fails to predict as it has low variance towards the training dataset (Dietterich, 1998). So, the performance of the ML model; (low variance and low bias) is an indication how accurate it can predict. The above explanation is valid for supervised ML techniques. For unsupervised clustering techniques where there isn't any model available to train. The quality of the classification is judge from the classified result. One such commonly used metrics is the Entropy (Stehl 2002; Meila 2003 and Amigó et al., 2009). In CobWeb, the performance of the ML models and the quality of the classification can be evaluated using 10-fold Cross Validation, Entropy, Receiver Operational Characteristics

(ROC). The explanation of these methods are briefly described in the subsection below. For detailed information the readers are referred to Stehl (2002) (Dietterich, 1998; Bradley, 1997) and references within.

2.3.1. 10-K fold Cross-Validation

The idea for K-fold cross-validation was first recommended by Larson (1931). K-fold cross validation is a performance evaluation technique which check the overfitting and underfitting of the ML model. In the K-fold technique, the training data is divided into k partitions. Thereafter, the ML model is trained with k-1 partition of data, and tested on a withheld kth subset of data that has not been use for training. This process is repeated k-times, through this each datapoint in the training dataset get to be tested atleast once, and is used, for training k-1 times. As it can be seen, this approach should significantly reduce the overfitting (low variance) as most of the data is used for testing and underfitting (low bias) as almost all the data is used for training. As from empirical evidence in k = 10 is preferred.

2.3.2. Entropy

The Entropy of a class reflects how the members of the k pixels are distributed within each class; the global quality measure is by averaging the entropy of all classes.

$$\text{Entropy} = - \sum_j \frac{n_j}{n} \sum_i P(i, j) \times \log_2 P(i, j) \quad (11)$$

Where $P(i, j)$ is the probability of find an item from the category i in the class j , where n_j is the number of items in class j and n the total number of items in the distribution.

2.3.3. Receiver Operational Characteristics

Receiver Operational Characteristics (ROC) curves is one of the popular methods to cross-validation of ML model performance (probability of models correct response $P(C)$ to the predicted result) (Bradley, 1997). It has three variables:

$$\text{Accuracy (1 - Error)} = \frac{T_p + T_n}{C_p + C_n} = P(C) \quad (12)$$

$$\text{Sensitivity (1 - } \beta) = \frac{T_p}{C_p} = P(T_p) \quad (13)$$

$$\text{Specificity (1 - } \alpha) = \frac{T_n}{C_n} = P(n) \quad (14)$$

Where, T_p and T_n are the true positive and true negative examples and C_p and C_n are total number of true positive and true negative examples obtained from the training dataset.

Probability of false positive is $P(F_p) = \alpha$

Probability of true positive is $P(T_p) = (1 - \beta)$

The accuracy is determined by calculation area under the curve (AUC), and the simplest was to do that is by using trapezoidal approximation.

$$AUC = \sum_i \left\{ (1 - \beta_i \cdot \Delta\alpha) + \frac{1}{2} (\Delta(1 - \beta) \cdot \Delta\alpha) \right\} \quad (15)$$

3. Toolbox and functionalities – CobWeb Key Features

The first version of CobWeb offers the possibility to read and to process reconstructed XCT files in both .tiff and .raw formats. The graphical user interface (GUI) is embedded with visual inspection tools to zoom in/out, cropping, color, and scale, to assist in the visualization and interpretation of 2D and 3D stack data. Noise filters such as non-local means, anisotropic diffusion, median and contrast adjustments are implemented to increase the signal-to-noise ratio. The user has a choice of five different segmentation algorithms, namely K-means, Fuzzy C-means (unsupervised), least square support vector machine (LSSVM) (supervised), bragging and boosting (ensemble classifiers) for accurate automatic segmentation and cross-validation. Relevant material properties like relative porosities, pore size distribution trends, volume fraction (3D pore, matrix, mineral phases) can be quantified and visualized as graphical output. The data can be exported to different file formats such as Microsoft® Excel (.xlsx), MATLAB® (.mat), ParaView (.vkt) and DSI studio (.fib). The current version is supported for Microsoft® Windows PC operating systems (Windows 7 and 10).

The main GUI window panel divides into three main parts (Figure 2), the tool menu strip, the inspector panel, and the visualization panel. The tool strip contains menus for zoom in and out, pan, rotate, point selection, color bar, legend bar, and measurement scale functionalities. The inspector panel is divided into subpanels where the user can configure the initial process settings such as segmentation schemes (supervised, unsupervised, ensemble classifiers), filters (contrast, *non-local means*, *anisotropic filter*, *fspecial*), and distance functions (link distance, Manhattan distance, box distance) to assist segmentation and geometrical parameter selection for image analysis (REV, porosity, pore size distribution (PSD), volume fraction). The display subpanel *records*, displays the 2D video of the XCT stack and the respective histogram. History subpanel is a *uilibox* that displays errors, processing time/status, processing instruction, files generated/exported and executed callbacks. Control subpanel is an assemblage of *uibuttons* to initialize the XCT data analysis process and the progress bar. Visualisation panel is where the results are displayed in several resized windows, which can be moved, saved or deleted. The pan-windows displayed inside the visualization module are embedded with *uimenu* and *submenu* to export, plot and calculate different variables like porosity, PSD, volume fraction, entropy, or receiver operational characteristics. To get the desired user functionalities, MATLAB® internal user-interface libraries were inadequate. Therefore, numerous specific adaptations are adopted from Yair Altman's undocumented MATLAB® website and the MATLAB® File Exchange community. Specifically, the GUI Layout Toolbox of David Sampson is used to configure the CobWeb GUI layout; the preprocessing *uitable*, uses the MATLAB® java-component; it was designed using *uitable* customization report provided by (Altman, 2014).

As a stand-alone, the CobWeb GUI can be executed on different PC and HPC clusters without any license issues. The framework of CobWeb 1.0 is schematically illustrated in Figure 3 and the direction for the arrow (left to right) represents the

series in which the various functions are executed. The backend architecture can be broadly classified in to three different categories, namely:

- Control module
- Analysis module
- 5 • Visualization module

3.1.1. Control module

Initially, the main figure panel is generated, followed by the tool strip dividing the main figure into different panels and subpanels as shown in Figure 2. After that, the control buttons *Load*, *Start*, *Stop*, *Volume Rendering* and *Clear* are created, initialized and the relevant information is appended in a main structure. Ideally, at this point, any button can be triggered or
10 activated. However, on doing so, an exception will be displayed in the history subpanel, indicating the next arbitrary steps. That is, to first load the data by pressing the *Load* button, where the *Load* function checks the file properties, loads the data in .tiff and .raw format, creates and displays 2D video of the selected stack, save the video file in the current folder, and updates the respective variables to the main structure. The *Stop* button (*Stop* function) ends the execution. However, when the processing is inside a loop, the *Stop* function can break the loop only after the *i*-th iteration. The *Clear* button (*Clear* function),
15 deletes the data and clears all the variables in the main structure, resetting the graphical window.

3.1.2. Analysis module

The next step is data processing; triggered by pressing the *Start* button, which activates the *Start* function. The *Start* function concatenates the entire analysis procedure and is shown as *Start* () in Figure 3. It is a densely nested function, where the bullet points and the sub-bullet points shown in Figure 3, symbolizes the outer and the inner nested loops. Initially, the data is gathered
20 and a sanity check is performed to evaluate, if the user, selected the relevant checkboxes and respective suboptions in the preprocessing *uitable*. If, the checkboxes are not selected, an exception alert is displayed in the *History* panel, highlighting the error and suggesting the next possible action. The next loop is the image modification loop, where the user inputs are required. These inputs are desired classes for segmentation, the image resolution, and the representative slice number. Thereafter, the representative slice is displayed on a resizable pan-window inside the visualization panel shown in Figure 2. Further, an option
25 to select a region of interest (ROI) is proposed, which can be accepted or rejected. If accepted, a REV is cropped from the 3D image stack based on user defined ROI dimensions. On rejection, the complete 3D stack is prepared for processing.

The next step is the segmentation process; an unsupervised or supervised algorithm is initialized based on the selection made by the user in the preprocessing *uitable*. Hereafter, the programming logic implemented at the back-end for unsupervised and supervised segmentation schemes is briefly explained. It is an easy, one-step process in the case of unsupervised techniques.
30 Based on the options selected in the preprocessing *uitable*, the image is filtered and subsequently, segmented. But, for unsupervised segmentation technique, Fuzzy C-means (FCM), an addition user input is required. A positive decimal number *x*, where *x* is equal to $1 \leq x \leq 2$ to set the membership criteria; when pixel values of different phases (ex. Rotligend sandstone,

Gorsmont carbonate rock etc.) are in close vicinity or subsets of each other, FCM uses, the membership criteria to constrain the segmentation ‘loosely’ or ‘tightly’ with the purpose to segregate different phases (Chauhan et al., 2016b).

In the case of supervised segmentation schemes (LSSVM, Bragging and Boosting) apriori information, also known as feature vector dataset or training dataset, is required to train the model(s) (Chauhan et al., 2016a; Chauhan et al., 2016b), and

5 consequently, the trained model is ready to classify the rest of the dataset. The following five steps accomplish this procedure:

- First, the visualization panel displays a single 2D slice of the REV or 3D image stack in a resizable pan-window. The embedded *uimenu* in the pan-window offers to use the *subuimenu* options to feature vector selection, training and testing.
- Second, by pressing the *subuimenu* option *Pixel Selection*, the feature vector selection (FV) performs. The *Pixel Selection* callback function initializes the subroutine *uPixelSel()*, which sequentially displays a *uitable* in a resizeable pan-window. The *uitable* contains columns *Features*, *X-Coordinate*, and *Y-Coordinate*, which is for example the pixel coordinates of pore, matrix, minerals, noise/speaks. This is a mandatory step, to build the training dataset. The user enters this information in the respective columns of the *uitable*.
- In the third step, the user has to identify features, such as, pores, minerals, matrix, noise/specks, in the 2D image, using zoom in and out tools available in the toolbar. The X-coordinates and Y-coordinates of the identified features need to be extracted using the data cursor tool, also available in the toolbar. If satisfied, the user can enter, the features and the corresponding X,Y coordinates in the *Pixel Selection uitable*.
- In the fourth step, the data is gathered and exported for training. This is done by pressing the export button placed on the *uitable* pan-window; which initiates the subroutine *uExportTable()*. The export subroutine collects a total of 36 (6 x 6) pixel values in the perimeter of the user specified X,Y coordinates in the *uitable*.
- In the fifth step, the model is trained. This is done by using the *subuimenu* in the 2D pan-window. As and when the *training* is finished, a notification appears on the *History* panel. Thereafter by pressing the *testing* option in the *subuimenu* the complete REV or 3D stack can be segmented.

25 A progress bar offers to monitor the state of process. Further, the *History* window displays information related to processing time, implemented image filters and the segmentation scheme. Finally, all relevant information and the segmented data is append to the main structure.

3.1.3. Visualisation Module

Once the processing is finished, the segmented data can be visualized in 2D format using *Plot* button or in 3D rendered stack using *VolRender* button. Figure 3 depicts the nested loop structure of the *Plot()* and *VolRender()* callback functions. On initialization, the *Plot()* callback accesses the main structure, and plots the segmented 2D image of the segmented slice consecutive, in a resizable pan-window in the visualization panel. The displayed pan-window is embedded with a *uimenu* and corresponding *subuimenu*. The *uimenu* items and the *subuimenu* options are

- Geometrical Parameters → Porosity, Pore Size Distribution, Volume Fraction
- [Performance](#) → Entropy, Receiver Operational Characteristics (ROC), 10-fold Cross Validation
- Export Stack → ParaView, Raw

5

The methods used to calculate geometrical parameters and validation schemes are benchmarked in (Chauhan et al., 2016a) (Chauhan et al., 2016b). Therefore, the selection of desired options initialize respective subroutines (uPoreSzVol, uCalVal, uExport) and plot the results as shown in Figure 2. If required, the export of these parameters (Porosity, PSD, Volume Fraction, Entropy, ROC; 10-fold Cross Validation) is possible to Excel, ASCII or MATLAB® for further statistical analysis. Using the
 10 *Export Stack* item, the export of the 3D segmented volume to ParaView (.vtk files) or as .raw format files is feasible for the purpose of visualization or DRP analysis. The volume rendering functionalities of CobWeb 1.0 is simple in comparison to ParaView or DSI studio. The *VolRender* () function renders the 3D data set using orthogonal plane 2-D texture mapping technique (Heckbert, 1986) and is best suited for OpenGL hardware. The user has the option to render the 3D stack in the original resolution or at lower resolution; the lower resolution enhances the plotting speed but degrades the image quality by
 15 10-folds. Due to this, we recommend to export the 3D stack to ParaView or DSI studio for visualization.

4. Materials and Methods

4.1. Gas-Hydrate bearing Sediment

The in-situ synchrotron-based tomography experiment and post-processing of synchrotron data conducted to resolve the microstructure of gas hydrate-bearing (GH) sediments is given in detail by Chaouachi et al. (2015), Falenty et al. (2015), and
 20 Sell et al. (2016). In brief, the tomographic scans were acquired with a monochromatic X-ray beam energy of 21.9 KeV at Swiss Light Source (SLS) synchrotron facility (Paul-Scherrer-Institute, Villigen, Switzerland) using the TOMCAT beamline (Tomographic Microscope and Coherent Radiology Experiment; Stampanoni et al. 2006). Each tomogram was reconstructed from sinograms by using the gridded Fourier transformation algorithm (Marone und Stampanoni 2012). Later, a 3D stack of 2560 x 2560 x 2160 voxels (volume pixels) was generated resulting in a voxel resolution of 0.74 μm and 0.38 μm at 10-fold
 25 and 20-fold optical magnification.

4.1.1. Dual Filtering of Gas-Hydrate bearing Sediment

The ED artefact, is the high and low image contrast seen, between the edges, of the void, quartz and GH phases, in the GH tomograms. It certainly, aids in clear visual distinction, of these phases, but, becomes a nuisance during segmentation process. Several approaches to reduce ED artefact in GH tomograms and its effect on segmentation and numerical simulation have
 30 been discussed in (Sell et al., 2016). Based on our experience, a combination of the non-local means (NLM) filter and the

anisotropic diffusion filter (AD), implemented using Avizo (ThermoFisher Scientific), works best in removing ED artefacts for our GH data. In short, AD was used for edge preservation and NLM for denoising. In this study, the NLM filter was set to a search window of 21, local neighbourhood of 6 and a similarity value of 0.71. The NLM filter was implemented in 3D mode to attain desired spatial and temporal accuracy and was processed on an CPU device.

5 4.1.2. Gas Hydrate (GH) bearing Sediment Dual Clustering

The edge enhancement (ED) effect was significant in all the reconstructed slices of the GH dataset. The ED effect was noticeable around the quartz grains, with high and low pixel intensities adjacent to each other. The high intensity pixel values (EDH) were very close to GH pixel values, while the low intensity pixel values (EDL) showed a variance between noise and void phase pixel values. Therefore, immediate segmentation performed on the pre-filtered GH datasets using CobWeb 1.0 resulted in misclassification. Further parameterizing and tuning the unsupervised (K-means) and supervised (LSSVM) modules of CobWeb 1.0 specifically, distance function (i.e., functions euclidean distance *squeclidean*, sum of absolute differences *cityblock*, and *mandist*) and different permutation and combination between of kernel type, bandwidth and cross-validation parameters, showed significant improvement, but the segmentation was still not optimal. The aim was to eliminate the ED features completely without altering the phase distribution between GH and the void. This prompted to develop a GH-specific workflow as explained below. The appendix provides the MATLAB[®] script for this workflow comprised of 6 steps:

Step 1: Filtering and REV selection

Four REV's of size 4×700^3 were cropped from the raw (16 bit) data stack. These REV's were dual-filtered using AD and NLM filters (see section 4.1.1). Figure 5 depicts a 2D dual-filtered image from REV1. In this study, the NLM filter was set to a search window of 21, local neighbourhood of 6 and a similarity value of 0.71. The NLM filter was implemented in 3D mode to attain desired spatial and temporal accuracy and was processed on an CPU device.

Step 2: K-means clustering

After, dual-filtration (Step 1), it was essential to segregate the noise, edge enhancement effects and different phases into labels of various classes. This was accomplished by K-means segmentation. In order to capture all the phases accurately along with noise and ED effects a segmentation process with up to twenty class labels was needed and performed. As a result class seven captured all the desired phases (noise, edge enhancement low intensities (EDL), void, quartz, edge enhancement high intensities (EDH), GH).

Step 3: Indexing

In the next step the purpose was to retrieve pixel values of various phases from the dual-filtered REV stacks. The indexing scheme is the following:

- First, through visual inspection of the segmented image (step 2) different phases and their corresponding labels were identified, shown in Table 1.
- Thereafter, pixel indices of these phases, were extracted from the segmented image based on their labels.
- Further, these indices were used as a reference mask to retrieve pixel values of the phases from the 16-bit raw REV stacks.

The obtained pixel values represent noise, void (liquid), EDL, quartz, EDH, and GH phases in the raw images. Then, histogram distribution of the pixel values in each phase was plotted. The skewness of the histograms was investigated where the max, min, mean and standard deviation for each of the histogram was calculated. Thereafter, max and min of the histograms were compared, and the indexing limits were adjusted, for as-long-as there was no overlap found amidst the histogram boundaries.

Step 4: Rescaling raw REV

Now, in this step, the raw pixel values of the respective phases, i.e void, quartz, and GH, were replaced by their mean values, with an exception for EDH pixel values. The latter (EDH pixels) were replaced with the mean value of quartz. These assignments lead to optimal segregation of the phase boundaries in the raw dataset and finally to the elimination of the ED effect.

Step 5: K-means clustering

Finally, the re-scaled raw REV was segmented into three class labels using K-means segmentation to obtain the final result.

4.2. Grosmont Carbonate Rock

The digital rock images of the Grosmont carbonate rock were obtained from the FTP server GitHub (<http://github.com/cageo/Krzikalla-2012>) used in the benchmark study published by (Andrä et al., 2013a, 2013b). The Grosmont carbonate rock was acquired from Grosmont formation Alberta, Canada. The Grosmont formation was deposited during upper Devonian and is divided into four facies members, LG UG-1, UG-2, and UG-3 (bottom to top). The sample was taken from UG-2 facies and is mostly composed of dolomite and karst breccia (Machel and Hunter, 1994; Buschkuehle et al., 2007). Laboratory measurements of porosity and permeability reported by (Andrä et al., 2013b) are around 21 % ($\phi = 0.21$) and $\kappa = 150 \text{ mD} - 470 \text{ mD}$, respectively. The Grosmont carbonate dataset was measured at the high-resolution X-ray computer tomographic facility of the University of Texas with an Xradia MicroXCT-400 instrument (ZEISS, Jena, Germany). The measurement was performed using 4x objective lenses, 70 kV polychromatic X-ray beam energy, and a 25 mm CCD detector. The tomographic images were reconstructed from the sinograms using proprietary software and corrected for the beam hardening effect, which is typical for lab-based polychromatic cone-beam X-ray instruments (Jovanović et al., 2013). The retrieved image volume was cropped to a dimension of 10243 with voxel size of 2.02 μm .

4.3. Berea Sandstone Rock

The Berea sandstone digital rock images were part of a benchmark project published by Andrä et al. (2013a, 2013b) and obtained from the *GitHub* FTP server. The Berea sandstone sample plug was acquired from Berea Sandstone Petroleum Cores™ (Ohio USA). The porosity value of 20 % ($\phi = 0.20$) was obtained using a Helium pycnometer
5 AccuPyc™ 1330 (Micromeritics Instrument Corp., Germany) and a Pascal-Mercury porosimeter (Thermo Scientific™) as described in Giesche (2006). The permeability ranges between $\kappa = 200$ mD and $\kappa = 500$ mD as reported by Andrä et al. (2013b). Machel und Hunter (1994) identified minerals using a polarized optical microscope and a scanning electron microscope, and reported a mineral composition of Ankerite, Zircon, K-feldspar, Quartz, and Clay in the Berea sandstone sample. The synchrotron tomographic scans of Berea sandstone were also obtained at the SLS TOMCAT beamline. The beam
10 energy was monochromatized to 26 keV for an optimal contrast with an exposure time of 500 ms. This resulted in a 3D tomographic stack with a dimension of 1024^3 voxels with a voxel size of $0.74 \mu\text{m}$.

5. Result and Discussions

5.1. Data Selection

The represented elementary volume (REV) selection basically, was a combination of visual inspection and consecutively
15 segmenting and plotting trends in relative porosity, pore size distribution and volume fraction. This was done by loading the complete stack in the CobWeb software, during the loading process a 2D movie of the tomogram is displayed in the display window and saved in the root folder. Carefully monitoring the movie gives an objective evaluation of the heterogeneity of the respective XCT sample. We observed, several sub-sample volumes at various location (X, Y) and depth (Z) inside the XCT tomograms. Thereafter, based on a subjective visual consensus different ROIs were selected, cropped, segmented and their
20 respective geometrical parameter were intercompared. The main indicator, however, was the porosity trend; i.e. when regression coefficient R2 value was close to zero, it was an indicator that its sub-volume has accumulated the heterogeneity along the z-axis of the sample. Therefore, based on the trend analysis approach, the sub-volume dimension where R2 value was close to zero was chosen as the suitable REV.

25 In the case of Berea sandstone, four different ROIs were investigated; whereas Grosmount carbonate rock seven different ROIs were needed to identify the best REV. Cubical stack size in between 300^3 to 700^3 slices were tested and later established that stack size around 480^3 suited the best. Through our previous scientific studies on the GH sediments (Sell et al., 2016; Sell et al., 2018) we were aware of the best-suited REV and established that stack size of 700^3 was an appropriate stack size. The identification of best REV for Grosmount was relatively tedious compared to Berea sandstone and GH sediment; due to the low
30 resolution and microporosity present in the Grosmount tomograms. Figure 4 shows the chosen ROIs of Berea, Grosmount and GH dataset and Figure 6 and Figure 8 show the surface plot for respective REV.

5.2. Data Processing

~~Figure 4 shows the ROIs selected for the Berea, Grosmont and gas hydrate (GH) samples.~~ In the case of Berea sandstone, the 3D reconstructed raw images (1024^3) had sufficient high resolution and contrast, thus did not show any noticeable change to the ~~above mentioned~~ filtration. Whereas, the XCT images (1024^3) of the Grosmont carbonate rock needed a non-local means filtering which yielded in better visualization and performance results compared to those enhanced with anisotropic diffusion filter. However for GH synchrotron dataset, the CobWeb 1.0 filters were insufficient to normalize the edge enhancement artifact (ED), therefore a unique dual filtering scheme was implemented, ~~which is explained in the following section 4.1.1.~~ explained in section 4.1.2. It is to be noted that, the NLM filter is hard-coded as 2D in the CobWeb standalone version (GUI).
10 But, by tweaking or modifying the source code we could initially pre-processed the XCT images using NLM 3D filtration and thereafter subjected it to segmentation. In general, our observation is that, depending on the resolution of the dataset, the fixed parameters of NLM and other filters should do a fairly good job. In case, their still exists noise and artefacts we recommended that the supervised techniques be used. The supervised techniques offer the possibility to select the residual noise or artefact pixel values before or after the filtration (pre-processing) through proper feature vector selection, and further training the
15 appropriate model and performing classification. Through which, the existing noise and artefact can be isolated and segmented as separate labels. Another alternative option could be to pre-process the data with desired filters data and imported the data into CobWeb for segmentation and analysis.

Another issue which needs to be discussed is the implementation of image segmentation. CobWeb 1.0 uses a slice-by-slice 2D
20 approach. It was observed that the ML techniques tend to underestimate porosity values compared to manually segmented analysis at an REV scale size $> 500^3$. This substantial degree of uncertainty is caused due to 2D slice-by-slice processing rather than the ML techniques. The 2D slice-by-slice approach, passes only, the spatial information (X, Y coordinate direction) to the ML algorithms, and the ML algorithm ends up sorting the intensity variation in the spatial domain (local optimum). Therefore, the lack of temporal information (Z coordinate direction) restricts the degree of freedom to find, a global spatial-
25 temporal optimum. In other words, as the temporal changes arise, due to bedding (sedimentary rock) or micro porosity (carbonate rocks) in the rock texture, they are represented as sudden spike or dip in porosity values; which ~~to an inexperienced eye~~ appear as artefact or anomalies— and more often-then-not discarded. We acknowledge this issue and correction will be implemented in the next software version; in the current workflow it has not been accounted for (CobWeb 1.0). ~~Since, it requires refactoring the loop based scalar oriented framework into ways to process the 3D stack.~~ The 2D slice-by-slice
30 processing scheme is much faster compared to the 3D approach. So, the choice of 2D processing for this research study was made to make it affordable to compute on desktop, laptop for near real-time and onsite evaluation. And, the inaccuracies in porosities is compensated by calculation mean porosity of the complete stack.

5.3. Multiphase Image Segmentation

The major problem for all multiphase segmentations is that phase having intermediate greyscale values gets sandwiched between two different phases. These intermediate phases sometimes represent some of the vital material property such as connectivity. Therefore, it is vital to emphasize, how ML can assist in issues related to multiphase segmentation. In a practical sense, machine learning tries to separate grayscale values in to disjoint sets. The creation of these disjoint sets is commonly done in two ways

- 1) By binning the greyscale values to the nearest representative values which is iteratively updated using an optimization function. This optimization function can be a simple regression or distance function (Jain et al., 1999), commonly used in the unsupervised techniques.
- 2) Second, by regularizing pre-trained models which store certain pattern information of the datasets such as topology features, contour intensities, pixel value etc. (Hopfield, 1982; Haykin, 1995; Suykens and Vandewalle, 1999). Or by using a voting system in a bootstrap ensemble of linear models (Breiman, 1996).

So, in this processes, the intermediate greyscale values which commonly have low sample size and exhibit Poisson distribution are merged in to larger sets of a high sample size to create disjoint boundaries. Through which the intermediate phase information misclassified and destroyed. One way to overcome this problem is by using, supervised techniques such as LSSVM or Ensemble classifiers. When constructing a training dataset (feature vector selection), careful selection of intermediate phases as a sufficiently large sample size compared to the predominate phases will preserve the intermediate phases. And, the likelihood that the trained model will identify them and cluster them separately is higher (Chauhan et al., 2016a). In this study, in particular, we made tests using supervised techniques (LSSVM, Ensemble classifiers) and unsupervised technique (FCM) but the results weren't superior compared to K-means. Therefore, we choose K-means as it was faster compared to other ML techniques. Since we have used K-means for segmentation, it is necessary that we justify performance of K-means in terms of accuracy and speed. In the current research work, since we have used unsupervised technique, it safe to say that, accuracy and speed is directly proportional to starting point (initial location) in the segmentation process. Meaning, the closer the starting point (initial location) is to the global minima— faster will the algorithm converge and even so better is the performance (accuracy & speed). But, in unsupervised technique by default the choice of the starting point is through random seed unless explicitly specified. So, in the case of the dual segmentation approach used for segmentation, the intuition was to capture all the material phases, including the edge enhancement artefact, speck and noise etc. in the first step and thereafter in the second step to rescale them to the plausible phases. Hence, in the first step the 20 clusters were initialized using random seed. And, after the rescaling processes, we were aware of the initial locations which we used as starting point (initial location) to assist the algorithm to move towards identifying correct phases. Therefore, we could increase both the speed and accuracy of K-means.

5.4. Estimation of Relative Porosity and Pore Size Distribution

~~In the case of the The Grosmont carbonate and Berea sandstone, the respective REV's were segmented using K-means, as well as the PSD module based on the morphological scheme suggested in Rabbani et al. (2014) for the analysis.~~ The pore size distribution (PSD) of the respective REV's were calculated using the CobWeb PSD module. The PSD module is based on an image processing morphological scheme (watershed transformation) suggested by Rabbani et al. (2014). As stated by Rabanni, in (Rabbani et al., 2014), the aim is to breakdown the monolithic void structure of rock into specific pores and throats connecting each other. (Rabbani et al., 2014) used unsegmented images and performed image filtration and thereafter segmented using watershed transformation. In our case, the tomograms were already pre-processed and segmented using ML techniques and thereafter subjected to the watershed algorithm. Thereafter, the mean relative porosity value obtained for Berea sandstone is $\phi = 17.3 \pm 2.6 \%$, whereas for Grosmont carbonates the mean porosity value is lower ($\phi = 10.5 \pm 2.3 \%$) shown in Figure 6. The regression coefficient value of $R^2 = 0.092$ for the Berea sandstone porosity trend indicates that porosity remains constant throughout the REV sizes chosen, and therefore consolidated for scale-independent heterogeneities. In the case of Grosmont carbonate rock, the chosen REV size was the best out of five obtained, which consolidate again for scale-independent heterogeneities. The average pore size distribution thus obtained was $6.70 \mu\text{m} \pm 0.68 \mu\text{m}$ and $14.21 \mu\text{m} \pm 0.66 \mu\text{m}$ for Berea and Grosmont plug samples, respectively.

Similarly, the porosity and PSD of the four GH REV's were analyzed using CobWeb 1.0 ~~except but for image filtration and image segmentation, was performed using a different workflow as discussed in section 4.1.1 and 4.2.2. and Figure 8 shows the comparison of the porosity trends of different GH REV's~~ and is shown in Figure 7. The low R^2 values of the porosity trends justify that, these GH REV's are scale-independent, and are an accurate representation of a large-scale system and are best suited for digital rock analysis. However, there is high variance compared with the mean PSD values. The exact reason is unknown, but may be due to the drastic increase and decrease of the quartz grains ~~as can be noticed~~ which can be seen in Figure 5. The first and last 2D slices of ROI 1 in Figure 5 show either non-isotropic or isotropic distribution of quartz grains, which might have contributed to the respective high and low standard deviation, seen in the porosity distribution. Figure 8 shows the surface and volume rendered plots of REV 1 and REV 2, due to the high accuracy of segmentation the quartz grain, brine and GH boundaries are clearly segregated and ED effect completely eliminated.

6. Conclusions and Outlook

This paper introduces with CobWeb 1.0 a new visualization and image analysis toolkit dedicated to representative elementary volume analysis of digital rocks. CobWeb 1.0 is developed on the MATLAB® framework and can be used as MATLAB® plugin or as a standalone executable. It offers robust image segmentation schemes based on machine learning (ML) techniques (unsupervised and supervised), where the accuracy of the segmentation schemes can be determined and results can be compared.

Dedicated image processing filters such as the non-local means, anisotropic diffusion, averaging and the contrast enhancement functions help to reduce artefacts and increase the signal-to-noise ratio. The petrophysical and geometrical properties such as porosity, pore size distribution and volume fractions can be computed quickly on a single representative 2D slice or on a complete 3D stack. This had been validated using synchrotron datasets of the Berea sandstone (at a spatial resolution of 0.74 μm), a gas hydrate-bearing sediment (0.76 μm) and a high resolution lab-based cone-beam tomography dataset of the Grosmont Carbonate rock (2.02 μm). The gas hydrate dataset, despite its nanoscale resolution, was hampered with strong edge enhancement (ED) artefacts, which causes discrepancies in modelling and transport simulation. A combination of the dual filtering and dual clustering approach is proposed to completely eliminate the ED effect in the gas hydrate sediments, and the code is attached as an appendix. The REV studies performed on Berea sandstone, Grosmont carbonate rock and GH sediment using CobWeb1.0 shows relative porosity trends with very low linear regression values of 0.092, 0.1404, 0.0527 respectively. CobWeb1.0 ability to accurately segment data without compromise on the data quality at a reasonable speed makes it an favorable tool for REV analysis.

CobWeb1.0 is still somewhat limited regarding its volume rendering capabilities, which will be one of the features to improve in the next version. The volume rendering algorithms implemented in CobWeb 1.0 so far do not reach the capabilities offered by ParaView or DSI studio, which relies on the OpenGL marching cube scheme. At present, the densely nested loop structure appears to be the best choice for systematic processing. As an outlook, vectorization and indexing approaches (*bsxfun*, *repmat*) have to be checked in detail to improve on processing speed. MATLAB®—Java synchronization will be explored further to configure issues related to multi-threading and visualization (Java OpenGL). Furthermore, a module CrackNet (crack network) is planned to be implemented, which will explicitly tackle the segmentation of cracks and fissures in geomaterials using machine learning techniques and a mesh generation plugin (stl format) for 3D printing. Pore network extraction and skeletonization schemes such as modified maximum ball algorithm (Arand and Hesser, 2017) and medial axis transformation (Katz and Pizer, 2003) will be considered such that the data can be exported to open-source pore network modelling packages (Gostick et al., 2016).

25 **Code availability / Data availability**

With regards to the code availability, the MATLAB® code for removal of edge enhancement artifacts from the gas hydrate bearing sediment is attached as appendix. The CobWeb executable as well as the user manual and The gas-hydrate bearing sediment, XCT datasets are [available to public](http://dx.doi.org/10.5281/zenodo.2390943) on Zenodo repository <http://dx.doi.org/10.5281/zenodo.2390943>.

The CobWeb executable requires a MATLAB® runtime compiler R2017b (9.3), which can be downloaded and installed from <https://ch.mathworks.com/products/compiler/MATLAB®-runtime.html>. The XCT dataset of Berea Sandstone and Grosmount Carbonate Rock can be obtained from *GitHub* FTP server (<http://github.com/cageo/Krzikalla-2012>). The gas-hydrate XCT datasets are not publicly available.

Author contribution

Contributor Roles Taxonomy (CrediT) is used to specify author contribution. <https://casrai.org/credit/>.

Swarup Chauhan conceptualized, investigated and performed the study. Further, implemented the machine learning workflow and graphical user interface design. Additionally, Swarup Chauhan performed the formal analysis and developed a software code for the removal of the edge enhancement artefact using the dual clustering approach. Further contributions of Swarup Chauhan: data curation on of the CobWeb software, writing the software manual, figures and writing, reviewing and editing the manuscript.

10 Kathleen Sell conceptualized, investigated and performed a case study on gas hydrates. Further, performed a study on the removal of edge enhancement artefacts and phase segmentation of methane hydrate X-ray tomograms (XCT). Also, she did a formal analysis by implementing dual filtration approach to reduce the edge enhancement artefacts. Kathleen Sell participated in discussions to validate phase segmentation using the dual segmentation approach and was involved in writing, reviewing and editing the manuscript

15 Frieder Enzmann was involved in the investigation of the X-ray tomography experiments and in acquiring synchrotron radiation beam time (TOMCAT beamline) at Paul Scherrer Institute, Villigen, Switzerland. He performed the reconstruction of XCT tomograms. Provided data curation in the form of archiving and maintenance - initial and later re-use for the XCT data.

Wolfram Rühaak was involved in the project administration of the CobWeb activities, provided resources with respect to graphical user interface (GUI) and inputs on improving GUI functionalities.

20 Thorsten Wille was involved in funding acquisition and sponsoring the CobWeb project, under the framework of the SUGAR (Submarine Gashydrat Ressourcen) III project by the Germany Federal Ministry of Education and Research grant number: 03SX38IH. He was involved in project administration and provided feedback on GUI functionalities

Ingo Sass was involved in the concept and funding acquisition for the CobWeb project, under the framework of the SUGAR (Submarine Gashydrat Ressourcen) III project by the Germany Federal Ministry of Education and Research (grant number: 03SX38IH) . He also provided supervision, project administration, resources and periodic review to improve GUI functionalities.

25 Michael Kersten was involved in conceptualization, preliminary investigation, funding acquisition for the project "Poroelastic wave propagation modelling on gas hydrate-bearing sediment based on tomographic microstructure data" from Deutsche Forschungsgemeinschaft (DFG grant number Ke 508/20 and Ku 920/18). He provided supervision, resources and was involved in writing, review and editing of the manuscript.

30 **Acknowledgements**

We thank Heiko Andrä and his team at Fraunhofer ITWM, Kaiserslautern, Germany, for providing us with the synchrotron tomography benchmark dataset of the Berea sandstone. The acquisition of the gas hydrate synchrotron-data was funded by the German Science Foundation (DFG grant Ke 508/20 and Ku 920/18). This study was funded within the framework of the

SUGAR (Submarine Gashydrat Ressourcen) III project by the Germany Federal Ministry of Education and Research (BMBF grant 03SX38IH). The sole responsibility of the paper lies with the authors.

7. References

- 5 A. Buades, B. Coll, and J. M. Morel: A non-local algorithm for image denoising, in: 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05), 60-65 vol. 2, 2005.
- Al-Raoush, R. and Papadopoulos, A.: Representative elementary volume analysis of porous media using X-ray computed tomography, *Powder Technology*, 200, 69–77, doi:10.1016/j.powtec.2010.02.011, 2010.
- Altman, Y.: *Accelerating MATLAB Performance*, CRC Press, 2014.
- 10 Andrä, H., Combaret, N., Dvorkin, J., Glatt, E., Han, J., Kabel, M., Keehm, Y., Krzikalla, F., Lee, M., Madonna, C., Marsh, M., Mukerji, T., Saenger, E. H., Sain, R., Saxena, N., Ricker, S., Wiegmann, A., and Zhan, X.: Digital rock physics benchmarks—Part I: Imaging and segmentation, *Benchmark problems, datasets and methodologies for the computational geosciences*, 50, 25–32, doi:10.1016/j.cageo.2012.09.005, 2013a.
- Andrä, H., Combaret, N., Dvorkin, J., Glatt, E., Han, J., Kabel, M., Keehm, Y., Krzikalla, F., Lee, M., Madonna, C., Marsh, M., Mukerji, T., Saenger, E. H., Sain, R., Saxena, N., Ricker, S., Wiegmann, A., and Zhan, X.: Digital rock physics benchmarks—part II: Computing effective properties, *Benchmark problems, datasets and methodologies for the computational geosciences*, 50, 33–43, doi:10.1016/j.cageo.2012.09.008, 2013b.
- 15 Arand, F. and Hesser, J.: Accurate and efficient maximal ball algorithm for pore network extraction, *Comput. Geosci.*, 101, 28–37, doi:10.1016/j.cageo.2017.01.004, 2017.
- 20 Bishop, C. M.: *Pattern Recognition and Machine Learning (Information Science and Statistics)*, Springer-Verlag, Berlin, Heidelberg, 2006.
- Bradley, A. P.: The use of the area under the ROC curve in the evaluation of machine learning algorithms, *Pattern Recognition*, 30, 1145–1159, doi:10.1016/S0031-3203(96)00142-2, 1997.
- Breiman, L.: Bagging predictors, *Machine Learning*, 24, 123–140, doi:10.1007/BF00058655, 1996.
- 25 Buschkuehle, B. E., Hein, F. J., and Grobe, M.: An Overview of the Geology of the Upper Devonian Grosmont Carbonate Bitumen Deposit, Northern Alberta, Canada, *Natural Resources Research*, 16, 3–15, doi:10.1007/s11053-007-9032-y, 2007.
- Chauhan, S., Rühaak, W., Anbergen, H., Kabdenov, A., Freise, M., Wille, T., and Sass, I.: Phase segmentation of X-ray computer tomography rock images using machine learning techniques: an accuracy and performance study, *Solid Earth*, 30, 7, 1125–1139, doi:10.5194/se-7-1125-2016, 2016a.

- Chauhan, S., Rühaak, W., Khan, F., Enzmann, F., Mielke, P., Kersten, M., and Sass, I.: Processing of rock core microtomography images: Using seven different machine learning algorithms, *Computers & Geosciences*, 86, 120–128, doi:10.1016/j.cageo.2015.10.013, 2016b.
- 5 Cnudde, V. and Boone, M. N.: High-resolution X-ray computed tomography in geosciences: A review of the current technology and applications, *Earth-Science Reviews*, 123, 1–17, doi:10.1016/j.earscirev.2013.04.003, 2013.
- Costanza-Robinson, M. S., Estabrook, B. D., and Fouhey, D. F.: Representative elementary volume estimation for porosity, moisture saturation, and air-water interfacial areas in unsaturated porous media: Data quality implications, *Water Resour. Res.*, 47, doi:10.1029/2010WR009655, 2011.
- 10 Dietterich, T. G.: Approximate Statistical Tests for Comparing Supervised Classification Learning Algorithms, *Neural Computation*, 10, 1895–1923, doi:10.1162/089976698300017197, 1998.
- Dunn, J. C.: A Fuzzy Relative of the ISODATA Process and Its Use in Detecting Compact Well-Separated Clusters, *Journal of Cybernetics*, 3, 32–57, doi:10.1080/01969727308546046, 1973.
- Gitman, I. M., Gitman, M. B., and Askes, H.: Quantification of stochastically stable representative volumes for random heterogeneous materials, *Archive of Applied Mechanics*, 75, 79–92, doi:10.1007/s00419-005-0411-8, 2006.
- 15 Gostick, J., Aghighi, M., Hinebaugh, J., Tranter, T., Hoeh, A., Michael, Day, H., Spellacy, B., Sharqawy, H., Mostafa, Bazylak, A., Burns Alan, Lehnert, W., and Putz, A.: OpenPNM: A Pore Network Modeling Package, *Computing in Science & Engineering*, 18, 60–74, doi:10.1109/MCSE.2016.49, 2016.
- Haykin, S. S.: *Neural networks: A comprehensive foundation*, [Nachdr.], Macmillan, New York, NY, 696 pp., 1995.
- Heckbert, P. S.: Survey of Texture Mapping, *IEEE Computer Graphics and Applications*, 6, 56–67, doi:10.1109/MCG.1986.276672, 1986.
- 20 Hopfield, J. J.: Neural networks and physical systems with emergent collective computational abilities, *Proc Natl Acad Sci USA*, 79, 2554, doi:10.1073/pnas.79.8.2554, 1982.
- Iassonov, P., Gebrenigus, T., and Tuller, M.: Segmentation of X-ray computed tomography images of porous materials: A crucial step for characterization and quantitative analysis of pore structures, *Water Resour. Res.*, 45, doi:10.1029/2009WR008087, 2009.
- 25 Jain, A. K.: Data clustering: 50 years beyond K-means, *Pattern Recognition Letters*, 31, 651–666, doi:10.1016/j.patrec.2009.09.011, 2010.
- Jain, A. K., Murty, M. N., and Flynn, P. J.: Data Clustering: A Review, *ACM Comput. Surv.*, 31, 264–323, doi:10.1145/331499.331504, 1999.
- 30 Jovanović, Z., Khan, F., Enzmann, F., and Kersten, M.: Simultaneous segmentation and beam-hardening correction in computed microtomography of rock cores, *Computers and Geosciences*, 56, 142–150, doi:10.1016/j.cageo.2013.03.015, 2013.

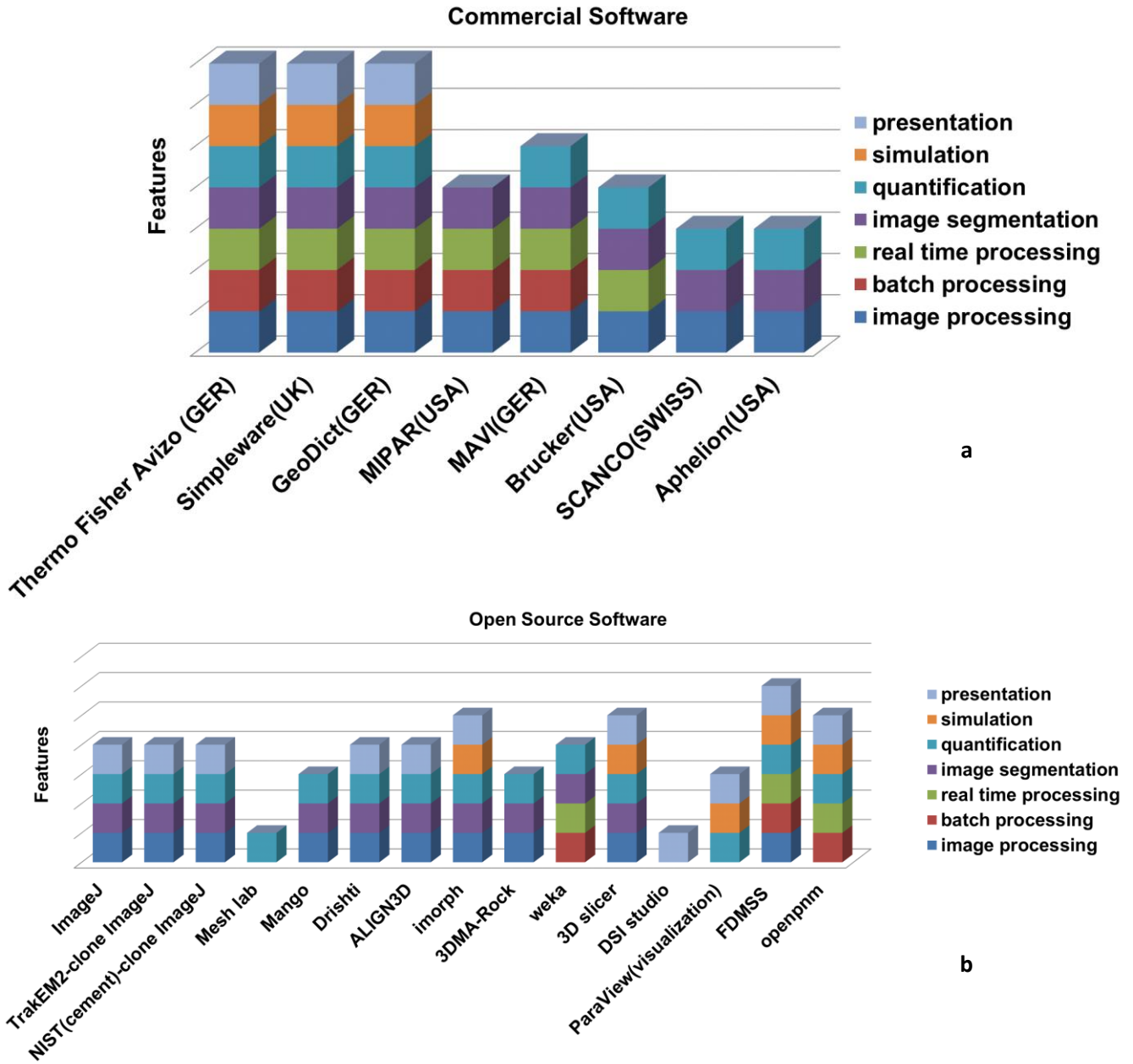
- Kaestner, A., Lehmann, E., and Stampanoni, M.: Imaging and image processing in porous media research, Quantitative links between porous media structures and flow behavior across scales, 31, 1174–1187, doi:10.1016/j.advwatres.2008.01.022, 2008.
- Katz, R. A. and Pizer, S. M.: Untangling the Blum Medial Axis Transform, International Journal of Computer Vision, 55, 139–153, doi:10.1023/A:1026183017197, 2003.
- Leu, L., Berg, S., Enzmann, F., Armstrong, R. T., and Kersten, M.: Fast X-ray Micro-Tomography of Multiphase Flow in Berea Sandstone: A Sensitivity Study on Image Processing, Transport in Porous Media, 105, 451–469, doi:10.1007/s11242-014-0378-4, 2014.
- Machel, H. G. and Hunter, I. G.: Facies models for middle to late devonian Shallow-Marine carbonates, with comparisons to modern reefs: a guide for facies analysis, Facies, 30, 155–176, doi:10.1007/BF02536895, 1994.
- MacQueen, J. (Ed.): Some methods for classification and analysis of multivariate observations, Fifth Berkeley Symposium on Mathematical Statistics and Probability, University of California Press, 281–297, 1967.
- Mjolsness, E. and DeCoste, D.: Machine Learning for Science: State of the Art and Future Prospects, Science, 293, 2051, doi:10.1126/science.293.5537.2051, 2001.
- Perona, P. and Malik, J.: Scale-space and edge detection using anisotropic diffusion, IEEE Transactions on Pattern Analysis and Machine Intelligence, 12, 629–639, doi:10.1109/34.56205, 1990.
- Parker, J. R.: Algorithms for Image Processing and Computer Vision, Wiley, 2010.
- Porter Mark L., Wildenschild Dorthe, Grant Gavin, and Gerhard Jason I.: Measurement and prediction of the relationship between capillary pressure, saturation, and interfacial area in a NAPL-water-glass bead system, Water Resour. Res., 46, doi:10.1029/2009WR007786, 2010.
- Rabbani, A., Jamshidi, S., and Salehi, S.: An automated simple algorithm for realistic pore network extraction from micro-tomography images, Neural network applications to reservoirs: Physics-based models and data models, 123, 164–171, doi:10.1016/j.petrol.2014.08.020, 2014.
- Razavi, M., Muhunthan, B., and Al Hattamleh, O.: Representative Elementary Volume Analysis of Sands Using X-Ray Computed Tomography, Geotechnical Testing Journal, 30, 212–219, doi:10.1520/GTJ100164, 2007.
- Schlüter, S., Sheppard, A., Brown, K., and Wildenschild, D.: Image processing of multiphase images obtained via X-ray microtomography: A review, Water Resour. Res., 50, 3615–3639, doi:10.1002/2014WR015256, 2014.
- Sell, K., Quintal, B., Kersten, M., and Saenger, E. H.: Squirt flow due to interfacial water films in hydrate bearing sediments, Solid Earth, 9, 699–711, doi:10.5194/se-9-699-2018, 2018.
- Sell, K., Saenger, E. H., Falenty, A., Chaouachi, M., Habberthür, D., Enzmann, F., Kuhs, W. F., and Kersten, M.: On the path to the digital rock physics of gas hydrate-bearing sediments – processing of in situ synchrotron-tomography data, Solid Earth, 7, 1243–1258, doi:10.5194/se-7-1243-2016, 2016.
- Shreyamsha Kumar, B. K.: Image denoising based on non-local means filter and its method noise thresholding, Signal, Image and Video Processing, 7, 1211–1227, doi:10.1007/s11760-012-0389-y, 2013.

- Smith, S. M. and Brady, J. M.: SUSAN—A New Approach to Low Level Image Processing, *International Journal of Computer Vision*, 23, 45–78, doi:10.1023/A:1007963824710, 1997.
- Suykens, J. A. K. and Vandewalle, J.: Least Squares Support Vector Machine Classifiers, *Neural Processing Letters*, 9, 293–300, doi:10.1023/A:1018628609742, 1999.
- 5 T. M. Cover: Geometrical and Statistical Properties of Systems of Linear Inequalities with Applications in Pattern Recognition, *IEEE Transactions on Electronic Computers*, EC-14, 326–334, doi:10.1109/PGEC.1965.264137, 1965.
- van Gestel, T., Suykens, J. A. K., Baesens, B., Viaene, S., Vanthienen, J., Dedene, G., Moor, B. de, and Vandewalle, J.: Benchmarking Least Squares Support Vector Machine Classifiers, *Mach. Learn.*, 54, 5–32, doi:10.1023/B:MACH.0000008082.80494.e0, 2004.
- 10 Zhang D, Zhang R, Chen S, and Soll, W. E.: Pore scale study of flow in porous media: Scale dependency, REV, and statistical REV, *Geophys. Res. Lett.*, 27, 1195–1198, doi:10.1029/1999GL011101, 2000.

Table 1: Class labels of different phases

Labels	Phases
0	Noise
1 and 3	Void (liquid)
2	Edge enhancement low intensities (EDL)
4	Quartz
5	Edge enhancement high intensities (EDL)
6 and 7	Gas hydrate

Figures



5 Figure 1: Market survey of the currently available commercial software (a) and open source software (b) assisting in digital rock physics analysis with features as indicated in legend

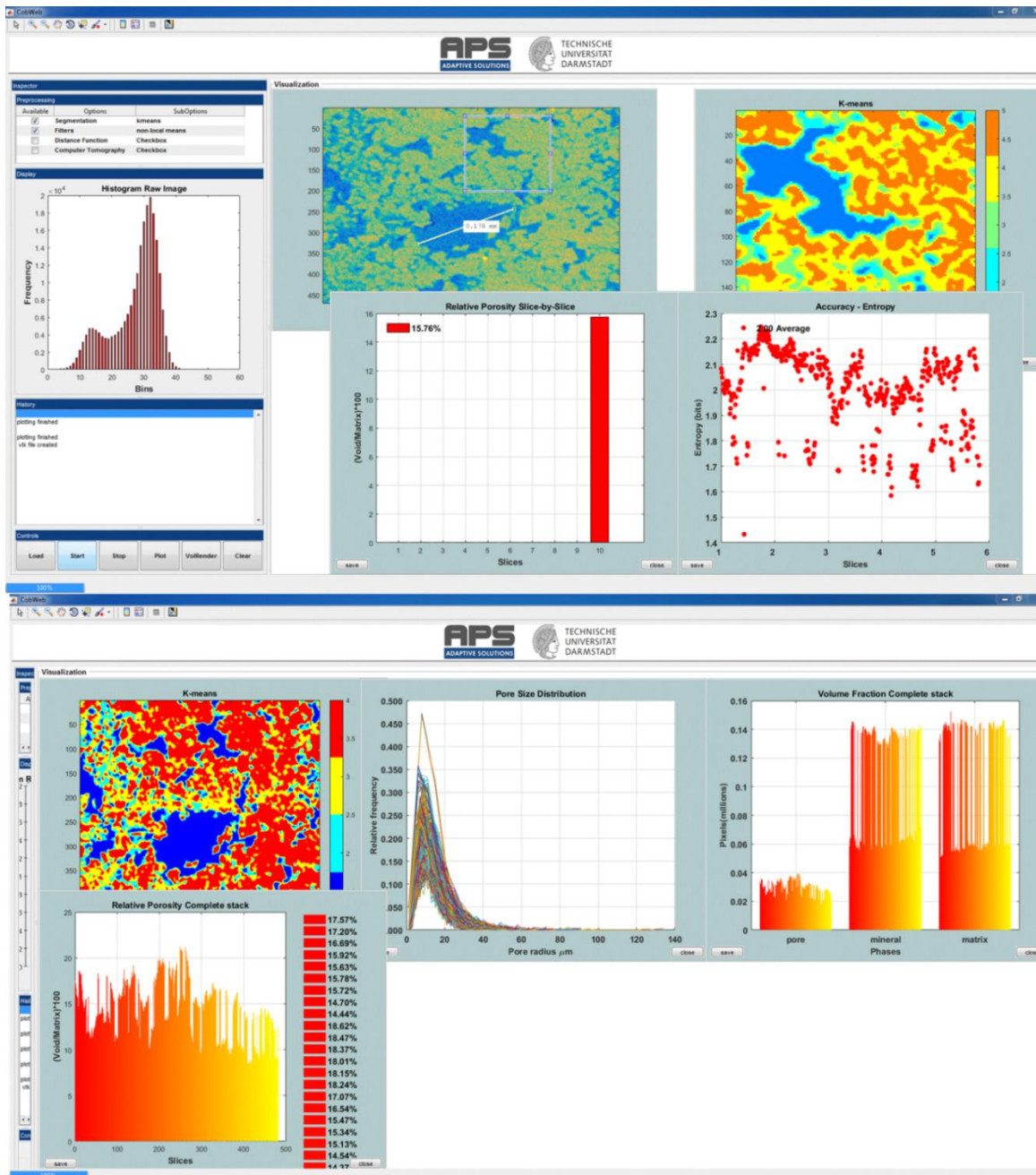


Figure 2: Snapshots of the CobWeb GUI. XCT stack of Grosmont Carbonate rock is shown as an example for representative elementary volume analysis. The top panel displays the XCT raw sample, the K-means segmented ROI, and the porosity of single slice No. 10. The bottom plot shows pore size distribution of the complete REV stack, the relative porosity and volume fraction, respectively

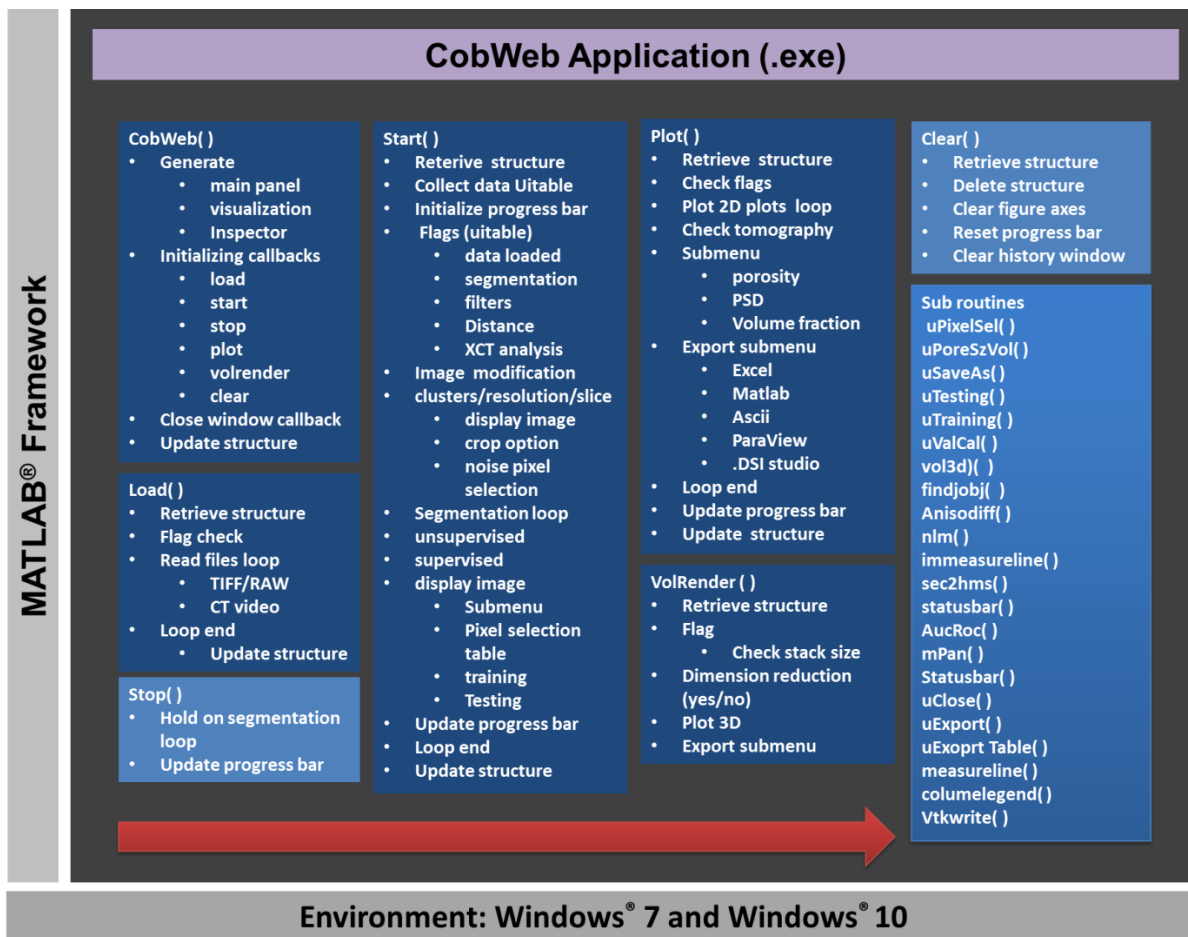
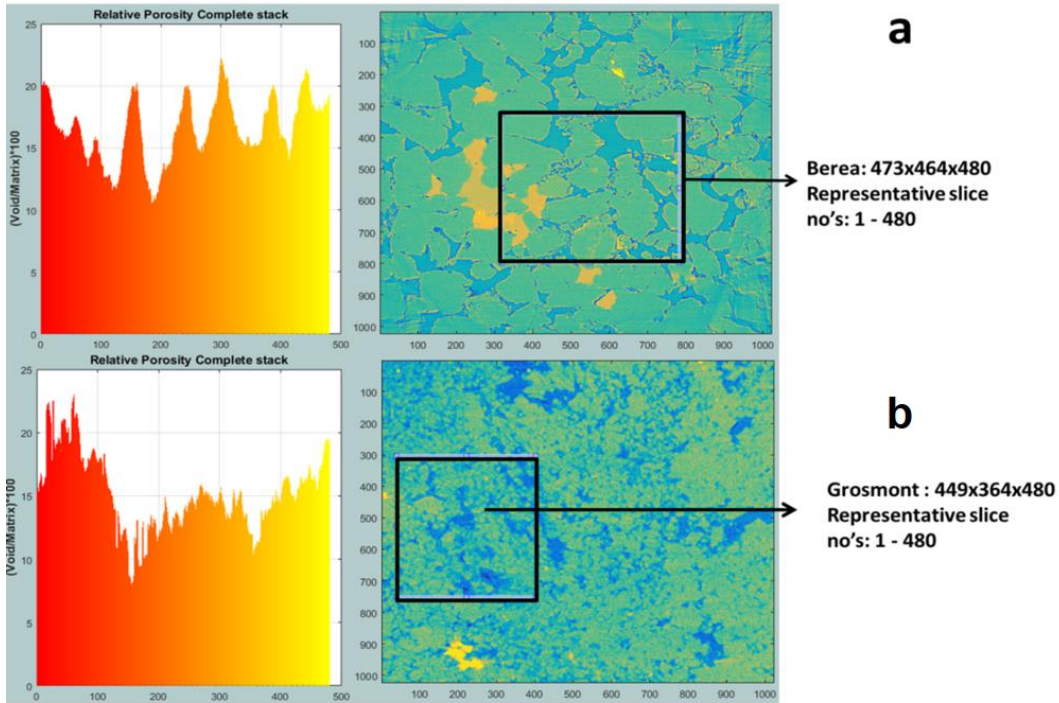


Figure 3: The general workflow of the CobWeb software tool, where the arrow denotes the series in which different modules are compiled and executed. A separate file script is used to generate .dll binaries and executables



Gas Hydrate

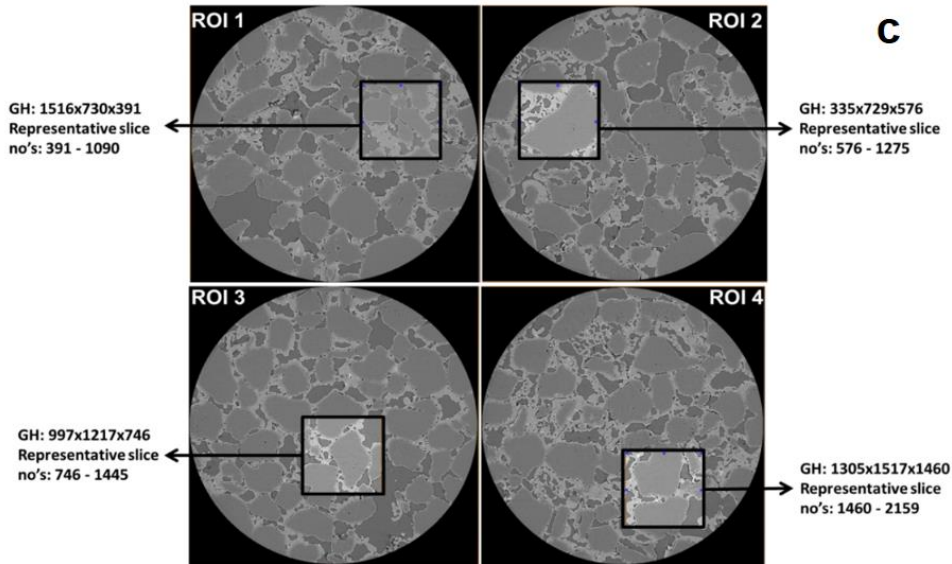


Figure 4: The most suitable ROIs and corresponding REV dimensions of Berea sandstone and Grosmont carbonate Gas Hydrate-bearing sediment is shown in the panel a b and c respectively

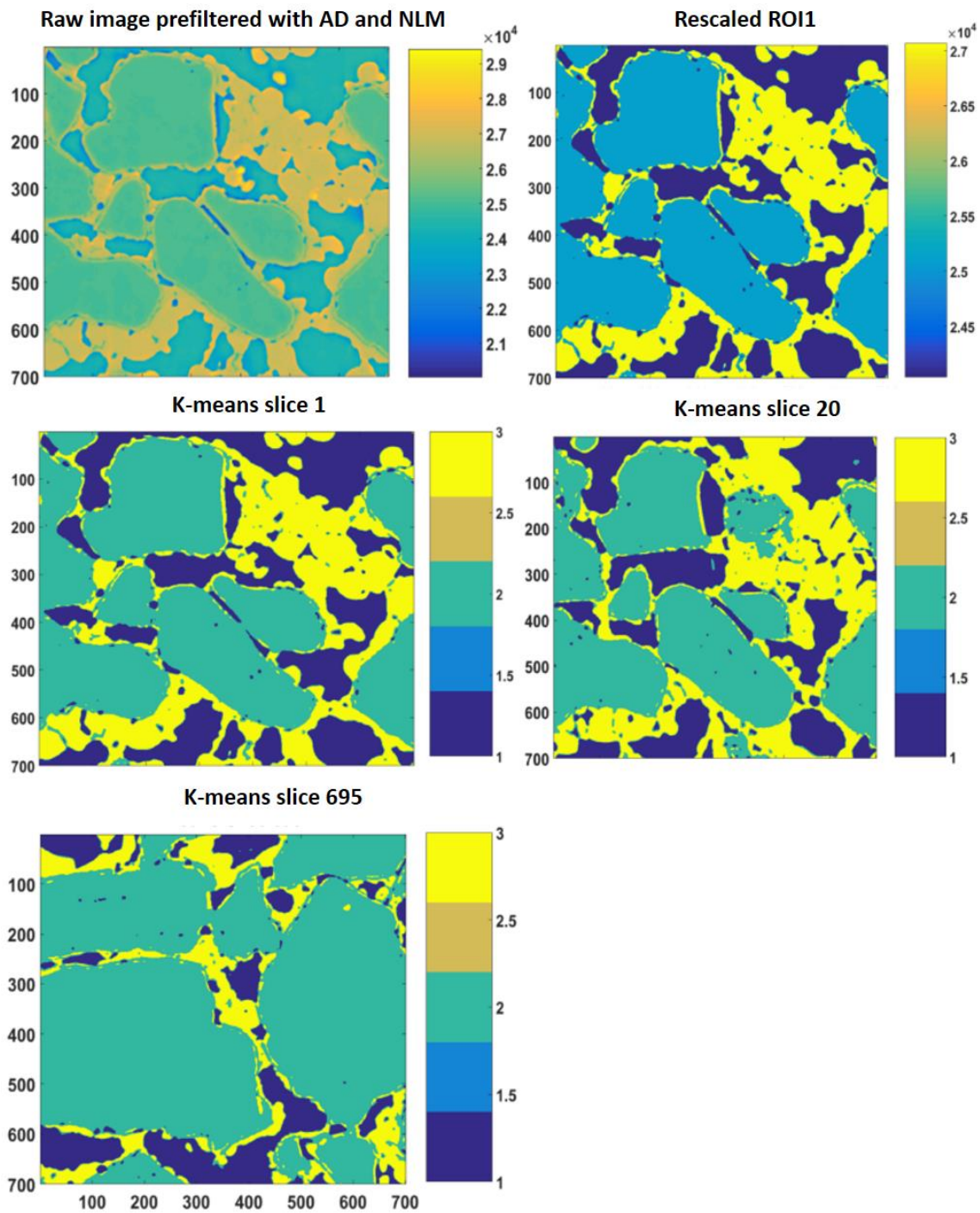
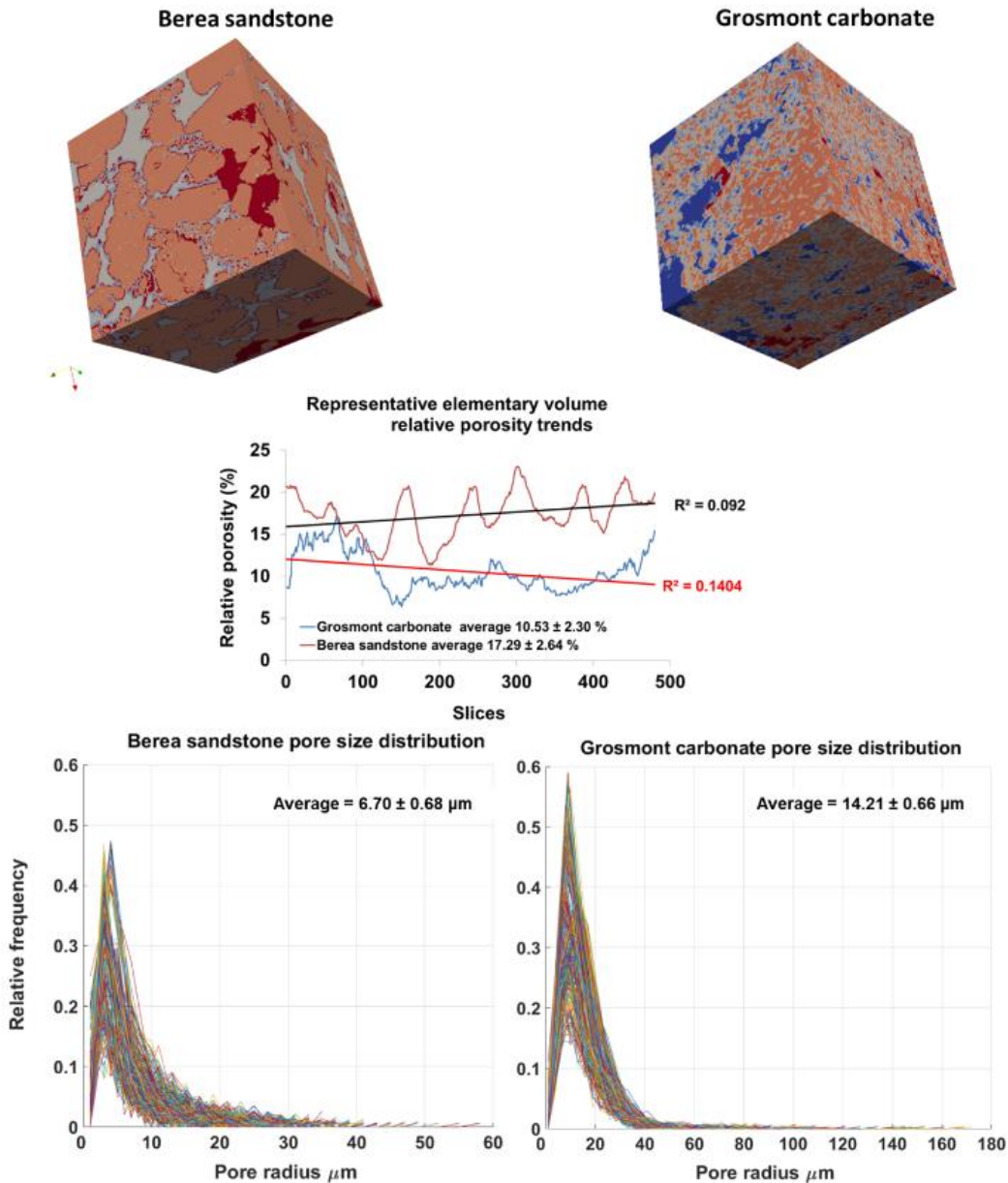


Figure 5: 2D filtered, rescaled, and segmented slices of gas hydrate REV1 dataset



5

Figure 6: Top panel shows surface plot of REVs Berea sandstone and Grosmont carbonate (size 471x478x480) using visualisation software ParaView. Middle plot shows the relative porosity (%) trend for Berea sandstone and Grosmont carbonate REVs samples. Bottom plot shows the pore size distribution of Berea sandstone and Grosmont carbonate. XCT images were segmented using K-means. In the case of Grosmont, a non-local means filter was used

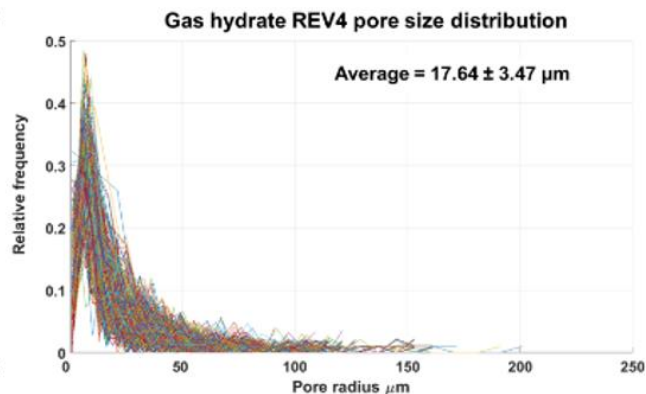
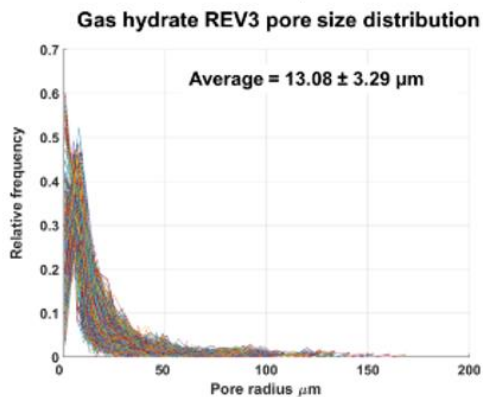
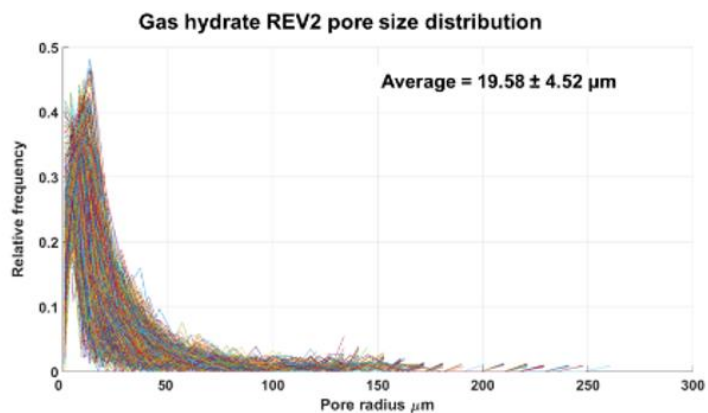
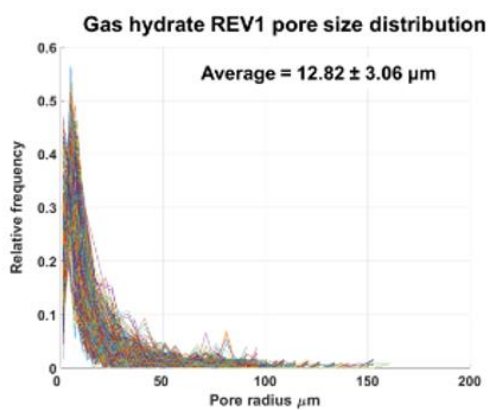
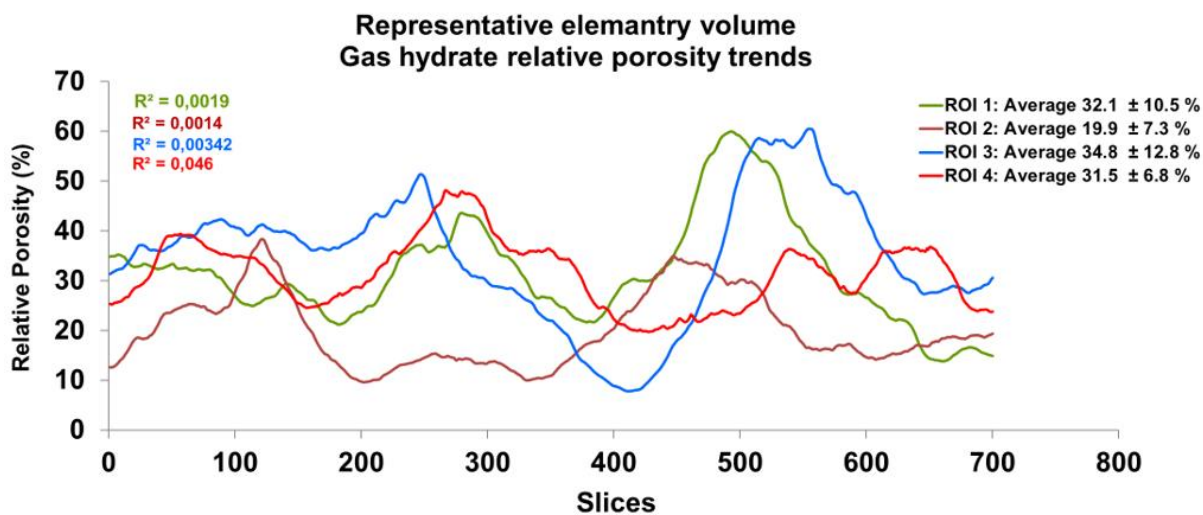


Figure 7: The top panel shows relative porosity trend analysis of gas hydrates, the middle and bottom panel show the geometrical pore size distribution of the respective REVs. The analysis was performed using CobWeb 1.0

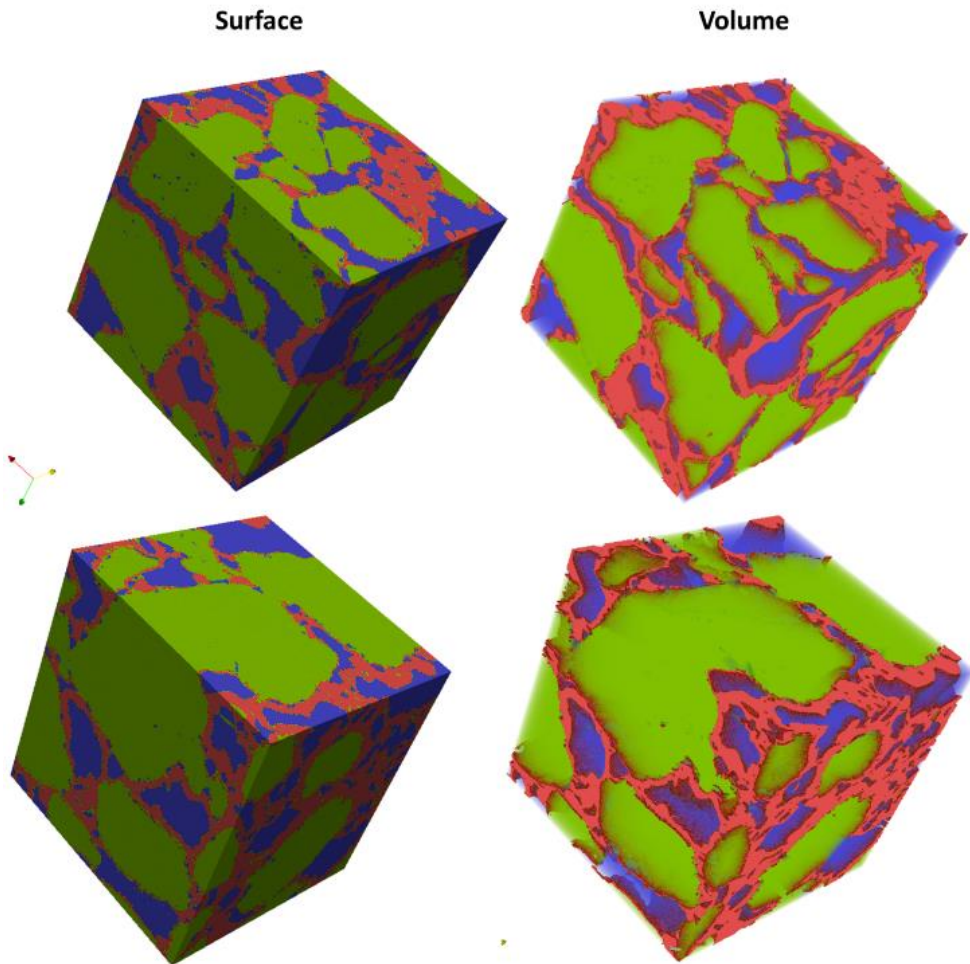


Figure 8: Segmented REVs of a gas hydrate sample displayed as surface and volume rendered. Analyzed using CobWeb 1.0 and exported to VTK format using CobWeb 1.0 ParaView plug-in. Quartz grain phase is represented in green color, gas hydrate in red, and in blue is the void space.

5