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.

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 REVs for all your samples. Are these REVs 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.

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; Costanzaâ˘ A˘RRobinson 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 analized and calculated.

Thanks for the suggestion, The section 3.3 and the respective figure 6 has been 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.

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.

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

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 classifies) 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.

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.

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.

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 prepossessing 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.

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.

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.

Moreover, you could compute permeabilities for your segmentations and compare then against available data, e.g., "Laboratory measurements of porosity and permeability reported by (Andrä et al., 2013b) are around 21 % (ϕ = 0.21) and κ = 150 mD âˇ T˘ A 470 mD, respectively".

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

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.